UNIVERSITY OF LIVERPOOL

Department of Civil Engineering

PROBABILISTIC ASSESSMENT OF THE SAFETY OF COASTAL STRUCTURES

Thesis submitted in accordance with the requirements of
the University of Liverpool for the Degree of
Doctor in Philosophy

by

Maria Teresa Leal Gonsalves Veloso dos Reis

January 1998

TO TERRY

For all those times you stood by me

For all the truth you made me see...

Seawall overtopping: Cascais, Portugal.

Seawall overtopping: north Wirral coast, UK.

Dune erosion: Formby Point, UK.

Dune erosion: east coast, USA.

ABSTRACT

TITLE: **Probabilistic Assessment Of The Safety Of Coastal Structures**
(Ph.D. thesis submitted to the University of Liverpool, 1998)

AUTHOR: Maria Teresa Leal Gonsalves Veloso dos Reis

This thesis is about the safety of coastal structures. To date, applications of probabilistic methods in the design of coastal engineering works have been very limited. The present study concentrates on the probabilistic assessment of single failure modes using First Order Reliability Methods (FORM). FORM have been implemented in a computer program called PARASODE which has been developed as part of this research. The program has been used to study the failure mechanisms of seawall overtopping by waves and of dune erosion during a storm surge.

A brief review of available equations for predicting wave overtopping is followed by the development of a new model and re-analysis of a large set of existing overtopping data for simple seawalls having uniform seaward slopes of 1:1, 1:2 and 1:4. Both the new model and an earlier formulation are used as input to PARASODE. It is suggested that, in both cases, regression coefficients contained within the overtopping equations should be established using a robust regression technique such as Least Absolute Deviations (LAD). It is shown that the two overtopping models are little different in their ability to represent the data, but the new model is inherently better suited to describing low overtopping rates.

An example is given of the application of the new and existing overtopping models in predicting the freeboards necessary to limit discharges to specified values. This example shows that, for the small allowable rates associated with normal design conditions, the new model predicts seawall crest elevations which may be several metres lower than the values from the earlier model. Such differences may have significant financial and environmental consequences and are worthy of further investigation. Calculations using PARASODE show that the choice of overtopping model is also very important in the probability assessment of the safety of seawalls. The FORM sensitivity parameters demonstrate that the main influence on the variability of the probability of failure is generally provided by the uncertainty in the sea state. The accuracy of the FORM reliability algorithms used in PARASODE is confirmed by comparing with results provided by the simulation method of Latin Hypercube Sampling (LHS).

Dutch experience with regard to the probabilistic design of dunes is also examined. The computational procedures currently used in The Netherlands are based on an equilibrium profile model. They are not directly applicable to conditions along coasts such as that in Sefton, UK, where there is a much

weaker correlation that in The Netherlands between wave heights and water levels. Consequently, only some features of the Dutch methods are incorporated in PARASODE. Examples illustrate how nourishment can be used to decrease a dune's failure probability caused by erosion during a storm surge. PARASODE is run in two modes. In mode 1, a nourishment width is chosen and the corresponding probability of failure is calculated. In mode 2, a probability of failure is input and a corresponding nourishment width is computed. These tests demonstrate the converse nature of modes 1 and 2 and show consistency between the results. The FORM sensitivity parameters show that the most important contributions to the resulting variance in the probability of failure are provided by uncertainties in the maximum water level during surge and the sea state.

ACKNOWLEDGEMENTS

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF NOTATION

Wherever possible, the notation used in the text of this thesis follows the recommendations of the International Association of Hydraulic Research (IAHR, 1989). The notation used in PARASODE is defined within the program.

NOTATION IN TEXT	DEFINITION
a	• Constant (Chapter 2) • Coefficient (Chapter 4)
A	• Regression coefficient (Chapters 3, 5 & 6; Appendix D) • Area between the surge level and the parabolic part of Vellinga's profile (Appendix C)
Ac	Accuracy of Vellinga's computational model
A_N^2	Anderson-Darling statistic
b	• Coefficient (Chapter 4) • Regression parameter considered, b_0 or b_1 (Appendix A)
b_0	Estimate of parameter β_0
b_1	Estimate of parameter β_1
B	• Regression coefficient (Chapters 3, 5 & 6; Appendix D) • Area between the surge level and the gradient 1:mt of Vellinga's profile (Appendix C)
BD	Area of a depression
BetaAcc	Relative accuracy of the reliability index
BH	Area of a hump
C	• Coefficient (Chapter 3) • Ratio of the maximum run-up to the significant height of the incident waves (=R_{max}/H_s) (Chapters 3 & 5) • Erosion quantity (m^3/m) above storm surge level (Chapter 4 & Appendix C5)
CL	Crest level above datum
C_d	Discharge coefficient
$C_{ov}[X]$	Covariance matrix of $X=(X_1,...,X_N)$
$C_{ov}[X,Y]$	Covariance of X and Y
d	Depth
dX	Infinitely small integration step on variable X
d_S	Still-water-depth at the toe of the seawall

Table 1: Notation.

NOTATION IN TEXT	DEFINITION
D	• Design point (Chapter 2) • Matrix of eigenvalues of $C_{ov}[X]$ (Chapter 2) • Area between the two 1:md gradients, the surge level and the nourished profile (Appendix C)
Depth	Depth of the most seaward point of the parabolic part of Vellinga's post-storm profile
DF	Degrees of freedom
DP	• Change in the initial profile (Chapter 4; Appendices B, D3 & D4) • Design Point (Appendix D2)
D_N	Kolmogorov-Smirnov statistic
D_{50}	Median grain size diameter (50% of the weight being finer)
e	Estimate of the error term ε; also called residual from the regression line
e_A	Parameter representing the degree of variability in regression coefficient A
e_B	Parameter representing the degree of variability in regression coefficient B
E	Area which lies between points (S9,T9), (S2,T2), the surge level and the nourished profile
E[...]	Expected value operator; it represents the expected value of its argument
Err	Error in the balance between erosion and accretion
Err1	Error in the balance between TSurch and area D
E_{nc}	Encounter probability (probability that the T_r-year return load will be exceeded at least once during the design life or reference period)
$f(X_1,...,X_N)$	A function of the N variables, X_i, i=1,...,N
f_X	Probability density function of variable X
$f_X(x)$	Probability density function of variable X evaluated at point X=x
$f_{X_1,...,X_N}$	Joint probability density function of variables $X_1,...,X_N$
F	F statistic
F(t)	A function of time (in the description of water surface elevation)
F_{Crit}	Value of the F statistic corresponding to the one-tailed F distribution with p and N-p-1 degrees of freedom, for a specific level of significance α
F_X	• Cumulative distribution function of variable X • Input data cumulative distribution function of variable X (Appendix A)
$F_X(x)$	Cumulative distribution function of variable X evaluated at point X=x
F_{X_i,r_i}	The cumulative distribution function of the maximum intensity of action X_i within the reference period, T_{ref}, which is subdivided into a number, r_i, of elementary time intervals for action X_i

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
F_X^F	Fitted cumulative distribution function of variable X
$F_X^F(x)$	Fitted cumulative distribution function of variable X evaluated at point X=x
$F_{X_i}^{r_i}$	Cumulative distribution function of the intensity of the action X_i raised to the power r_i
F_X^{-1}	Inverse of the cumulative distribution function of variable X
$F_X^{-1}(x)$	Inverse of the cumulative distribution function of variable X evaluated at point X=x
g	Acceleration due to gravity
G	Net loss of sand from a coastal profile owing to a gradient in the longshore transport rate
GB	Gust bumps
G_o	Reference value for G
h	Maximum water level during storm surge
H	Wave height
Ho	Null hypothesis
H_{rms}	Root mean square wave height
H_S	Significant wave height
$H_S\|h$	Value of H_S given a value for h
H_S_Inacc	Variable representing the inaccuracy in H_S given a value for h
$H_{wavemax}$	Upper limit on $\mu_{H_S\|h}$ to account for depth limitations
i, j	Integers
k	• An integer • Number of time-varying actions (Chapter 2 & Appendix C6) • Coefficient (in the description of water surface elevation) (Chapter 3) • Number of classes into which the data is grouped for use of the Chi-Square test (Appendix A)
K	Armour stability coefficient (in Iribarren's equation)
K_D	Armour stability coefficient (in Hudson's equation)
K_{X_R}	Coefficient defining the fractile which corresponds to the characteristic value of the resistance variable, X_R
K_{X_s}	Coefficient defining the fractile which corresponds to the characteristic value of the load variable, X_S
ℓ	An integer
Le	Length of the parabolic part of Vellinga's post-storm profile
L_m	Airy wavelength at the toe of the seawall calculated using the mean zero-crossing wave period
L_{op}	Airy wavelength in deep water calculated using the period of peak spectral density $(= gT_p^2 / 2\pi)$

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
L_{os}	Airy wavelength in deep water calculated using the significant wave period ($=gT_s^2/2\pi$)
L_p	Airy wavelength at the toe of the seawall calculated using the period of peak spectral density
L_S	Airy wavelength at the toe of the seawall calculated using the significant wave period
m	• Total number of time intervals per year (Chapter 2) • Number of parameters of a probability distribution estimated from a data set (Appendix A)
1:md	Gradient of the eroded dune face
1:mnour	Gradient of the nourished face
1:mt	Gradient of the toe of the post-storm profile
MaxIter	Maximum number of iterations in a FORM calculation
MAD_{Reg}	Mean absolute deviations regression
MAD_{Res}	Mean absolute deviations residual
MS_{Reg}	Mean square regression
MS_{Res}	Mean square residual
n	Ratio of the prototype value to the model value
nourtlev	Nourishment top level
nourwidt	Nourishment width at top level
n_d	Depth scale (for beach profile and hydraulic conditions)
n_l	Length scale for beach profile
n_t	Time scale
n_w	Scale for the fall velocity of the sand
n_H	Wave height scale
n_L	Wavelength scale
n_T	Wave period scale
N	• Number of variables (Chapter 2) • Number of run-up values (Chapter 3) • Number of observed data points in a sample (Appendix A)
NPch	Number of points to be changed in the initial profile
NPD	Initially, the number of points defining the initial profile; then, the number of points defining the nourished profile
NPV	Total number of points defining Vellinga's profile
NumDep	Number of depressions
NumHump	Number of humps
N_j	Number of data points in the jth class (in the Chi-Square test)
N_S	Stability number of armour stones
$N_{WO\%}$	Percentage of waves passing over the crest of a structure

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
N_X	Number of data points, X_i's, less than or equal to x, i=1,...,N
NR_{ji}	Power to which each basic distribution, F_{X_i}, should be raised, for each time-varying action X_i, i=1,...,k, and for each combination j, j=1,...,k
p	• Probability of occurrence of a time-varying action in each elementary time interval, τ (Chapter 2) • Number of independent variables in the model (Appendix A)
p_j	Expected proportion of the data points that would fall in the jth class if sampling was done from the fitted distribution (in the Chi-Square test)
P_f	Probability of failure
q	Instantaneous discharge of water over unit length of seawall
Q	• Mean overtopping discharge over unit length of seawall (Chapter 3) • Area between the surge level and the nourished profile below surge (Appendix C)
Q_p	Peak overtopping discharge
Q_{PRED}	Predicted mean overtopping discharge over unit length of seawall
Q_*	Dimensionless overtopping discharge
r	Effective roughness of the seawall's front slope
r_i	Number of elementary time intervals during the design life or reference period of a structure for the time-varying action X_i
R	• Reliability (Chapter 2) • Most landward position to which a dune profile has been eroded during a storm surge (Chapter 4)
RD	Retreat distance
ReqBetaAcc	Required relative accuracy of the reliability index
ReqZAcc	Required accuracy of the failure function
R_c	Seawall's freeboard (the height of the crest of the structure above the still-water-level)
R_{max}	Maximum run-up $(=CH_s)$
$(R_{max})_{p\%}$	p% confidence value of the estimated maximum run-up
R_S	Significant wave run-up
$R_{2\%}$	Run-up exceeded by only 2% of the incident waves
R^2	Coefficient of determination
R_a^2	Adjusted statistic which attempts to correct R^2 to more closely reflect the goodness of fit of the model in the population
R_*	Dimensionless freeboard

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
sup[...]	Smallest upper bound on members of the set [...]
s_i	Number of elementary time intervals during a period of time shorter than the reference period of a structure for the time-varying action X_i
s_m	Deep water wave steepness calculated using T_m
s_p	Deep water wave steepness calculated using T_p
S	Sample standard error of the estimate
SD	Surge duration
SDep	Cumulative area of the depressions starting from the seaward end of the profiles
SEb	Standard error of the parameter b considered, b_o or b_1, (or estimate of the standard error)
SEb_o	Standard error of b_o (or estimate of the standard error of b_o)
SEb_1	Standard error of b_1 (or estimate of the standard error of b_1)
$SEInd\hat{Y}$	Standard error of the individual prediction at a specific value Xo of X (or estimated standard error of the individual prediction at a specific value Xo of X)
$SE\hat{Y}$	Standard error of the predicted mean value of Y at a specific value Xo of X (or estimated standard error of the predicted mean value of Y at a specific value Xo of X)
SHump	Cumulative area of the humps starting from the seaward end of the profiles
Smooth	Smoothing coefficient for the iteration process
Surcharge	A coefficient for the surcharge on the erosion area above surge level
SurchEros	Surcharge on erosion area C above surge level to take into account the effects of the storm surge duration, of the gust bumps and of the accuracy of the computation
SurchLongT	Surcharge on erosion area C to take into account the effect of a gradient in the longshore transport rate
SurD	Total surcharge distance
S1	X-coordinate of the intersection point between the nourished profile and the surge level
S2	X-coordinate of the intersection point between the nourished profile and the gradient 1:mt of Vellinga's profile
S3	X-coordinate of the intersection point between the nourished profile and the gradient 1:md of Vellinga's profile
S4	X-coordinate of the intersection point between the nourished profile and the surcharge gradient, 1:md
S8	X-coordinate of the starting point of the parabolic part of Vellinga's post-storm profile
S9	X-coordinate of the point where the parabolic part of Vellinga's profile finishes

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
S10	X-coordinate of the point of intersection between the surge level and the gradient 1:md of the surcharge
SAD_{Reg}	Regression sum of absolute deviations
SAD_{Res}	Residual sum of absolute deviations
SS_{Reg}	Regression sum of squares
SS_{Res}	Residual sum of squares
S_X	Standard deviation of the X values
S_1	Parallel-series system
S_{11}, S_{12}, S_{13}	Sub-systems of system S_1
t	• Time (Chapters 2, 3 & 4) • t statistic (Appendix A)
$\tan\alpha$	• Tangent of the angle of seawall front slope measured from horizontal (Chapters 3 & 5; Appendix D2) • Tangent of the angle of the armour slope measured from horizontal (Chapter 3)
t_{Crit}	Value of the t statistic corresponding to the two-tailed Student's t distribution with N-p-1 degrees of freedom, for a specific level of significance α
T	Wave period
TL	Toe level above datum
TR	Target value for each FORM calculation
TSS	Total sum of squares
TSurch	Total surcharge on erosion area C which is the sum of the surcharges SurchEros and SurchLongT
T1	Y-coordinate of the intersection point between the nourished profile and the surge level
T2	Y-coordinate of the intersection point between the nourished profile and the gradient 1:mt of Vellinga's profile
T3	Y-coordinate of the intersection point between the nourished profile and the gradient 1:md of Vellinga's profile
T4	Y-coordinate of the intersection point between the nourished profile and the surcharge gradient, 1:md
T9	Y-coordinate of the point where the parabolic part of Vellinga's profile finishes
T_m	Mean zero-crossing wave period
T_p	Wave period corresponding to peak spectral density
T_r	Return period
T_{ref}	Design life or reference period of a structure
T_S	Significant wave period
u	Variable in the definition of the incomplete Beta function; $u=(X-x_1)/(x_2-x_1)$
U	Normalised variable

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
V	Matrix of eigenvectors of $C_{ov}[X]$
Var[X]	Variance of X (σ_X^2)
V_p	Peak volume of water in an individual wave
w	Sediment fall velocity for a given water temperature
W	Weight of armour stones
x	Particular value of the variable X
x_1, x_2	Respectively, the lower and upper limits on X for a Beta distribution
X	- Basic variable (Chapter 2 & Appendix D) - X-coordinate (Chapters 4, 5 & 6; Appendix C5) - New unsmoothed value of X (Chapter 5) - Independent variable or X-ccordinate of an observed data point in a sample (Appendix A) - Variable (Appendix C3)
XB	X-coordinate of the intersection point between the changed profile and the surge level
XDEnd	X-coordinate of the end point of a depression
XDStart	X-coordinate of the starting point of a depression
XHEnd	X-coordinate of the end point of a hump
XHStart	X-coordinate of the starting point of a hump
XM	X-coordinate of the most seaward point at the nourishment top level
XMax	Maximum value of variable X
XMin	Minimum value of variable X
XN	X-coordinate of the intersection point between the changed profile and the nourishment top level
Xo	- Point of truncation of a probability distribution, if the distribution is truncated (Chapter 5) - Particular value of X (Appendix A)
XP	Initially, the X-coordinate of the points defining the initial profile; then, the X-coordinate of the points defining the changed profile; finally, the X-coordinate of the points defining the nourished profile
XPV	X-coordinate of the points defining Vellinga's profile
XQ	X-coordinate of the intersection point between the nourishment slope 1:mnour and the changed profile
X_{max}	Seaward X-limit of Vellinga's parabolic profile
X_q	Quantile of the input data cumulative distribution function
X_{New}	New smoothed value of X
X_{Old}	Value of X in the previous iteration
X_R	- Resistance variable (Chapter 2) - Distance from the origin to the most landward position to which the dune profile has been eroded during a storm surge (Chapter 4)
$X_{R_{ch}}$	Characteristic value of the resistance variable, X_R
X_S	Load variable

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
$X_{S_{ch}}$	Characteristic value of the load variable, X_S
X_T	Target X-coordinate
X^*	Linearization point used in the Level II methods
$\overline{X}$	Mean of the X values of the data points in a sample
X^*_{New}	New linearization point
X^F_q	Quantile of a fitted cumulative distribution function
Y	• Non-correlated variable (Chapter 2 & Appendix D) • Y-coordinate (Chapters 4, 5 & 6; Appendix C5) • Dependent variable or Y-coordinate of an observed data point in a sample (Appendix A)
YB	Y-coordinate of the intersection point between the changed profile and the surge level
YDEnd	Y-coordinate of the end point of a depression
YDStart	Y-coordinate of the starting point of a depression
YHEnd	Y-coordinate of the end point of a hump
YHStart	Y-coordinate of the starting point of a hump
YM	Y-coordinate of the most seaward point at the nourishment top level
YN	Y-coordinate of the intersection point between the changed profile and the nourishment top level
YP	Initially, the Y-coordinate of the points defining the initial profile; then, the Y-coordinate of the points defining the changed profile; finally, the Y-coordinate of the points defining the nourished profile
YPT9	Y-coordinate of the point in the nourished profile which has X=S9
YPV	Y-coordinate of the points defining Vellinga's profile
YQ	Y-coordinate of the intersection point between the nourishment slope 1:mnour and the changed profile
Y_{max}	Seaward Y-limit of Vellinga's parabolic profile
Y_{med}	Median of the Y values of the data points in a sample
$\hat{Y}$	Predictive value of Y
$\overline{Y}$	Mean of the Y values of the data points in a sample
z	z statistic
z_{Crit}	Value of the z statistic corresponding to the two-tailed standard Normal distribution, for a specific level of significance α
Z	Failure function or limit state function
Z_i	Value of the fitted cumulative distribution function evaluated at x_i, $F^F_X(x_i)$
Z_{Resid}	Standardised residuals
Z^*	Value of Z evaluated at the point X^*

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
α	• Sensitivity factor (Chapters 2, 5 & 6; Appendices D2 & D4) • Angle of the seawall front slope measured from the horizontal (Chapters 3 & 5) • Armour slope (Chapter 3) • Significance level (Appendix A)
β	• Reliability index (Chapter 2) • Angle of wave approach measured from the normal to the seawall (Chapter 3) • Parameter of a model, β_0 or β_1 (Appendix A)
$\beta(\zeta, \lambda)$	Beta function; $\beta(\zeta, \lambda) = \Gamma(\zeta)\Gamma(\lambda)/\Gamma(\zeta + \lambda)$
$\beta_u(\zeta, \lambda) / \beta(\zeta, \lambda)$	Incomplete Beta function
β_{New}	Reliability index of the current iteration
β_{Old}	Reliability index of the previous iteration
β_0	Unknown parameter of a model (also called the intercept)
β_1	Unknown parameter of a model (also called the slope)
γ	• Partial coefficient (Chapter 2) • Reduction factor to account for influences of berms, roughness, shallow water and oblique wave attack on wave run-up and overtopping (Chapter 3)
$\Gamma(\zeta)$	Gamma function; if ζ is an integer value, $\Gamma(\zeta) = (\zeta - 1)!$
$\Gamma(\zeta, \lambda x) / \Gamma(\zeta)$	Incomplete Gamma function
Δh	A short-term increase in the water level due to gust bumps and squall oscillations
ΔC	Increase in the volume of erosion due to Δh
ΔSH	Increased amount of erosion due to a smoothed hydrograph with a maximum Δh higher
ΔX	Finite discrete integration step on X
ε	Random error term which takes into account the fact that a model does not exactly describe reality
ζ	Parameter of probability distributions
η	• Water surface elevation above still-water-level at the seawall (Chapter 3) • Parameter of probability distributions (Appendix C)
$\left(\dfrac{\partial Z}{\partial X}\right)^{*}$	Partial derivative of Z with respect to X, evaluated at the point X^{*}
λ	Parameter of probability distributions
μ	Mean value
μ_f	Coefficient of friction between the armour stones (in Iribarren's equation)
$\mu_{X_i}^{N}$	Mean value of the approximate Normal distribution according to the Rackwitz & Fiessler (1978) approximation

Table 1: Notation (continued).

NOTATION IN TEXT	DEFINITION
ξ_p	Surf similarity parameter calculated using the period of peak spectral density ($=\tan\alpha / \sqrt{H_S / L_{op}}$)
π	3.14159.......
ρ	• Correlation coefficient (Chapters 2 & 6; Appendices C6 & D) • Water density (Chapter 3)
ρ_S	Density of armour stones
σ	Standard deviation
$\sigma_{X_i}^N$	Standard deviation of the approximate Normal distribution according to the Rackwitz & Fiessler (1978) approximation
τ_i	Length of the elementary time interval for the time-varying action, X_i
φ	Probability density function for the standard Normal distribution
ϕ	Natural angle of repose of the armour slope material
Φ	Cumulative distribution function for the standard Normal distribution
Φ^{-1}	Inverse function of Φ
χ^2	Chi-square statistic
χ_{Crit}^2	Value of the χ^2 statistic corresponding to the one-tailed χ^2 distribution with N-p-1 degrees of freedom, for a specific level of significance α

Table 1: Notation (continued).

LIST OF ABBREVIATIONS

ABBREVIATIONS	DEFINITION
A-D	Anderson-Darling
AIME	American Institution Of Mechanical Engineers
ANOLAD	Analysis Of Least Absolute Deviations
ANOVA	Analysis Of Variance
ASCE	American Society Of Civil Engineers
ASME	American Society Of Mechanical Engineers
BS	British Standard
BSI	British Standards Institution
CDF	Cumulative Distribution Function
CERC	Coastal Engineering Research Center
CIAD	Association For Computer Applications In Applied Engineering
CIRIA	Construction Industry Research And Information Association
CUR	Centre For Civil Engineering Research And Codes / Centre For Civil Engineering Research Codes And Specifications
DUNE	Name of Dutch computer program
DUNEPROB	Name of Dutch computer program
FOMVA	First Order Mean Value Approach
FORM	First Order Reliability Method
FORTRAN	FORmula TRANslation
HR	Hydraulics Research
H&R	Hedges And Reis
IABSE	International Association For Bridge And Structural Engineering
IAHR	International Association Of Hydraulic Research
ICCE	International Conference On Coastal Engineering
ICE	Institution Of Civil Engineers
ICOSSAR	International Conference On Structural Safety And Reliability
ICTM	Institute For Marine Science And Technologies
IEEE	Institute Of Electrical And Electronic Engineers
IFIP	International Federation For Information Processing
IML	Interactive Matrix Language
IOS	Institute Of Oceanographic Sciences
IWEM	Institution Of Water And Environmental Management

Table 2: Abbreviations.

ABBREVIATIONS	DEFINITION
jpdf	joint probability density function
K-S	Kolmogorov-Smirnov
LAD	Least Absolute Deviations
LBD	Liverpool Bay Datum
LHS	Latin Hypercube Sampling
LNEC	National Laboratory Of Civil Engineering
LS	Least Squares
MAFF	Ministry Of Agriculture Fisheries And Food
MLEs	Maximum Likelihood Estimators
MSL	Mean-Sea-Level
NAG	Numerical Algorithms Group Limited
NAP	Standard Amsterdam Datum
NATO	North Atlantic Treaty Organisation
NERC	Natural Environment Research Council
NRA	National Rivers Authority
OD	Ordnance Datum
OTC	Offshore Technology Conference
PARASODE	Probabilistic Assessment Of Risks Associated With Seawall Overtopping And Dune Erosion
PDF	Probability Density Function
PIANC	Permanent International Association Of Navigation Congresses
POL	Proudman Oceanographic Laboratory
P-P	Probability-Probability
Q-Q	Quantile-Quantile
SAS	Statistical Analysis System
SI	Système International D'Unités
SORM	Second Order Reliability Method
SPE	Society Of Petroleum Engineers
SPSS	Statistical Program For Social Sciences
SSL	Storm Surge Level
SWL	Still-Water-Level
TACPI	Technical Advisory Committee On Protection Against Inundation
TAW	Technical Advisory Committee On Water Defences
2D	Two-Dimensional
3D	Three-Dimensional

Table 2: Abbreviations (continued).

1   INTRODUCTION

1.1   Importance Of The Research And Its Main Objectives

A significant proportion of the world's population lives in areas at risk from coastal erosion and inundation by the sea. For example, the coastline of England and Wales is approximately 4500km long (Birks, 1993; MAFF, 1993c). Yet, despite having significant lengths of high rocky cliffs in the west, there are large areas in the south and east below the highest sea levels. A survey (NRA, 1991) showed a total of nearly 1300km of coastal structures protecting low-lying areas. About a quarter of the total coast has been developed for housing, industry, or some other purpose (MAFF, 1993a). Over 5% of the population and of the nation's industry is in areas below the mean annual maximum water level (approximately the 5m Ordnance Survey contour) and many of these areas are protected by structures (Birks, 1993). Over 50% of agricultural land is also below this level and is dependent on drainage and/or flood defence in some way to maintain its productivity. A number of cities, including London, have significant defences against river or tidal flooding, and coastal towns such as Blackpool are defended against flooding by the sea (MAFF, 1993c). Without defences, urban areas with key infrastructure, businesses, homes, and agricultural and recreational land would all be vulnerable to flooding and coastal erosion. Also, a number of historic sites and buildings, some of which are protected by statute, are at risk, together with environmentally valuable areas such as Sites of Special Scientific Interest.

CIRIA (1986b) stated that the replacement value of existing coastal structures in the UK was about £4000 million, with an average cost of £2.5 million per km for the country as a whole. Many structures are more than 100 years old and in need of replacement. Maintenance costs vary from about £500 to £2400 per km (CIRIA, 1986a). Although these values relate to the 1980s, they still indicate the considerable investment involved in the provision of coastal structures.

In The Netherlands, the River Meuse floods of 1993 and 1995 caused economic damage of 250 and 400 million Dutch guilders, respectively (Dirkson, 1996). In 1993, 8000 inhabitants had to leave their homes because

of the high water levels. In 1995, the stability of the dikes along the River Rhine could no longer be guaranteed and 0.2 million people had to be evacuated. Fortunately, none of the dikes failed.

Climatic changes over the next few decades may cause a rise in sea levels and increased storminess (Doornkamp, 1990; Vrijling, 1990; Stive et al, 1991; Townend, 1994b; Samuels & Brampton, 1996). As pointed out by Hedges (1993), substantial erosion of beaches may be expected, together with an increased flow of water over coastal structures (wave overtopping) and damage to these structures. Climatic changes may also increase vulnerability to inland flooding if there are alterations to the frequency or intensity of rainfall patterns combined with greater difficulties in river discharge to the sea if the sea level has risen (MAFF, 1993c). It is important to ensure that coastal structures are designed and managed within a framework which accounts for likely future climate changes as these changes will occur within the lifetime of current coastal defence schemes (Sorensen, 1991; Naden et al, 1996; Simm et al, 1996).

Clearly, there is a need to minimise risk to life and protect natural and man-made assets by providing defences against inundation and erosion. Furthermore, it is important that coastal structures are planned in the most environmentally sensitive manner whilst also providing an appropriate level of protection. Complete safety against flooding or erosion is unattainable. A balance has to be struck between costs and benefits to a nation as a whole. It is important that coastal defence policy and practices contribute to wider social, economic and environmental objectives. Unfortunately, anyone whose work is related to coastal structures cannot fail to be aware of the severe damage which has been inflicted on some large structures during the last few decades (Harlow, 1980; Sorensen & Jensen, 1985; Burcharth, 1987). An example is the failure of the Sines breakwater, Portugal. Questions then arise concerning the effectiveness of existing structures and of the design methods which have been commonly applied. The traditional approach to design may be inadequate and unsatisfactory, not only from an engineering stand-point but also from an economic point of view (Mol et al, 1983). That is why the probabilistic methods used in other areas of engineering have been applied in assessing the levels of safety provided by existing and new coastal structures. These methods can help the designer achieve a balanced approach in which most effort is put into addressing those parameters that

make the greatest contribution to the total probability of failure. In other words, if limited finance is available for reducing the overall failure probability of a structure, it should be invested in those components which show the largest reduction in overall failure probability for the given amount of money (Mol et al, 1984).

The recent shift towards probabilistic design in coastal engineering (Thompson & Scheffner, 1996) has been pioneered by the Dutch. They have applied the methods to natural and man-made structures, such as breakwaters, dikes and dunes. The scale and importance of coastal defence in The Netherlands has encouraged this new approach. But practice in other countries is also moving in the same direction. In this connection, the main objective of the present research is to assess the safety of coastal structures by means of probabilistic methods, with particular reference to wave overtopping of seawalls and to dune erosion. A secondary objective is to increase interest within the maritime community in the use of probabilistic techniques. Note that there is already clear cooperation in this field, evident in such multi-national publications as PIANC (1992).

1.2   Structure Of The Thesis

The structure of the present research is shown in Figure 1.1. Chapter 2 contains a review of concepts and methods of probabilistic analysis and of the state of the art in their application to the design of coastal structures. At present, applications are limited and available knowledge on coastal structures is insufficient to enable the probabilistic analysis of a whole structure to be carried out in full. Therefore, it was decided to concentrate this study on methods for the probabilistic assessment of single failure modes. Special attention is given to the particular methods used in developing a computer program, PARASODE, and in validating the results of the program using a commercial software package, @Risk. PARASODE is used to study the failure mechanisms of wave overtopping of seawalls and dune erosion.

Chapter 3 is generally concerned with wave overtopping of seawalls. It starts with a brief review of the subject, including the models currently used in predicting overtopping and the permissible levels of overtopping. Then, an

alternative model is presented; this is the H&R model. This new model is conceived through theoretical considerations and special care has been taken to consider the appropriate physical boundary conditions. A re-analysis of Owen's data (Hydraulics Research Station, 1980; Owen, 1982a) for simple seawalls possessing uniform seaward slopes of 1:1, 1:2 and 1:4, is also presented. The H&R and Owen models are used in this re-analysis and represent part of the input to PARASODE. The implications of the new model for seawall freeboards are then discussed. Finally, the reliabilities of the two models are assessed.

Chapter 4 focuses upon the Dutch experience in the probabilistic assessment of dune erosion during a storm surge. Firstly, a review of some key references is presented. Secondly, the current application of Vellinga's and Steetzel's models is discussed. Since Vellinga's model is currently the one used in The Netherlands for probabilistic calculations, the two Dutch computer programs (DUNEPROB and DUNE) which rely on its use are then presented. Finally, the applicability of these programs in the British context is examined.

The outcome of the literature review on probabilistic methods (Chapter 2) and the formulation of the failure modes of wave overtopping of seawalls (Chapter 3) and dune erosion (Chapter 4) have been assembled in the form of a program, PARASODE, to undertake probabilistic calculations. The program, its input and output, are described in Chapter 5. For the failure mode of wave overtopping of seawalls, the results obtained from PARASODE have been validated using the software package @Risk which is also briefly introduced in Chapter 5.

Chapter 6 illustrates the application of PARASODE: the first set of examples relates to wave overtopping of seawalls; the second relates to dune erosion. The results obtained from PARASODE are discussed and, for overtopping, they are validated using @Risk.

The most important conclusions arising out of this research are presented in Chapter 7. This chapter also provides the recommendations for further research.
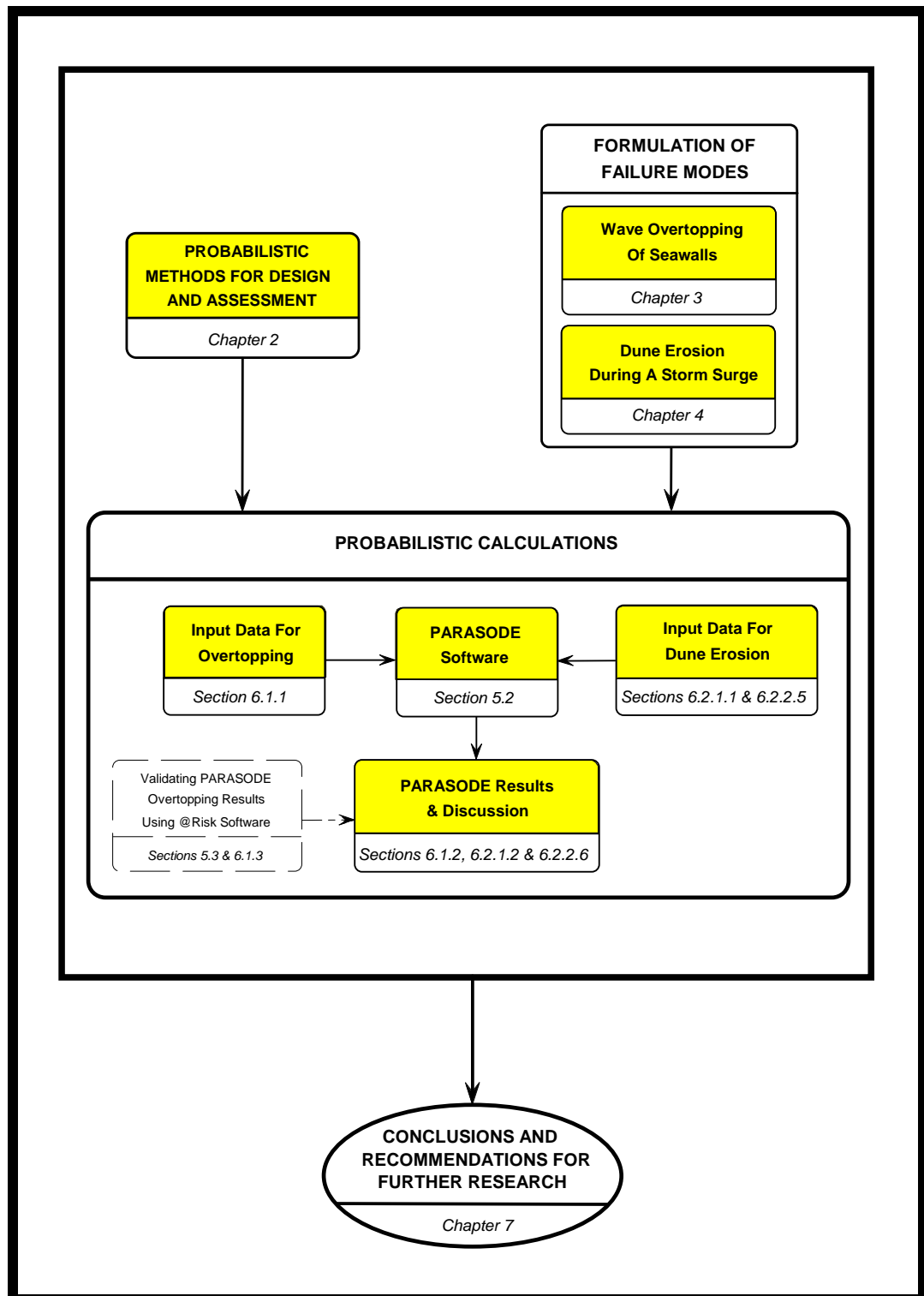
Figure 1.1: Flowchart of the structure of the thesis.

2 PROBABILISTIC METHODS FOR DESIGN AND ASSESSMENT

2.1 Introduction

Conventional practice for the design of coastal structures is essentially deterministic (Mol et al, 1983; Melchers, 1987; Burcharth, 1992): the resistance of a structure should exceed the load by an appropriate margin which is an indication of the level of safety (Townend, 1994a). This margin is required to counter lack of knowledge and uncertainties with respect to resistance, load and other factors (Lee & Mays, 1983). It is based mainly on experience rather than on quantification of the unknowns and uncertainties. As a result, some structures may be designed to unwarranted levels of conservatism but, conversely, other structures will be subject to unacceptably high risks of failure. Using conventional design practice, it is not possible to determine the extent of under-design or over-design relative to an acceptable level of risk (Meadowcroft et al, 1996).

The design load is usually defined on a probabilistic basis. For example, it might be the value x of the variable load, X, which is, on average, exceeded once during some specified period (e.g. 100 years). This period, T_r, is the return period of x. It is given by $T_r = 1/[m(1-F_X(x))]$ where $F_X(x)$ represents the probability of $X \leq x$ in any one time interval. Since this time interval need not be in years, but T_r is normally expressed in years in engineering applications, the total number of time intervals per year, m, needs to be calculated. Examples are wave data collected at 3-hourly intervals, resulting in m=365x24/3=2920 and high water levels collected at intervals of 12h25min which gives m=365x24/12.42=705. The return period for the extreme load may itself be chosen with regard to the value of the encounter probability, $E_{nc} = 1-[1-1/T_r]^{T_{ref}}$, which is the probability that the T_r-year return load will be exceeded at least once during the reference period, T_{ref} (Burcharth, 1987, 1990; Casciati & Natale, 1992; PIANC, 1992; Carvalho, 1992a). However, there is often little consideration given to the uncertainties involved in establishing the probabilities (Burcharth, 1992).

In most cases, the resistance is defined in terms of the load which causes a certain degree of damage to the structure and is not given as an ultimate

force or deformation. As mentioned by Burcharth (1992), this is because most of the available formulae only give the relationship between wave characteristics and structural response (e.g. in terms of run-up, overtopping, armour layer damage). Almost all such design formulae are semi-empirical relationships (Burcharth, 1987), being based mainly on experience, engineering skill and central fitting to model test results (using fitting methods like least squares regression). The test results do not all fall on the line represented by a particular formula; there is often considerable scatter around the line which is not incorporated into the design process in any systematic fashion (Burcharth, 1992). Consequently, the applied characteristic value of the resistance is the mean value and not a lower fractile as is usually the case in other engineering fields (e.g. in the manufacturing and aeronautical industries).

Recent experience of well-publicised severe damage to some large coastal structures (Harlow, 1980; Burcharth, 1987, 1990) has led to the conclusion that damage was caused by a combination of aspects and that the safety levels for these structures were far too small. In other words, taking into account all stochastic variables influencing load and resistance, the probability of failure was too high, resulting in a high encounter probability of a severe damage in the years after completion of the structures (Mol et al, 1983). The deterministic methods used, which do not explicitly consider the reliability of a proposed design through the incorporation of information on the uncertainties involved in the load and resistance variables, together with other sources of uncertainties, do not allow an accurate assessment of the degree of safety in terms of the probability of failure (CIRIA, 1984; Burcharth, 1987, 1990). These deterministic methods, which disregard the fundamental stochastic nature of the problem, are inadequate (Burcharth, 1985).

Instead of the above simplistic approach which requires relatively little input data, a probabilistic approach is preferred. The latter has a number of advantages (Van der Meer, 1987; CUR-TAW, 1990; CIRIA/CUR, 1991; Lamberti, 1992) including: i) the structure under study can be analysed and described as a whole; ii) the uncertainties are rationally incorporated in the assessment of the safety of the structure; iii) it is possible to obtain a better insight into the sensitivity of the structure's failure probability to the various uncertainties: this enables a more balanced design in which priority for

further research is given to those parameters that make the greatest contribution to the total probability of failure; and iv) it is possible to assess explicitly the cost of improving the structure and of damage or loss.

This chapter continues by reviewing general probabilistic concepts and their application in design and assessment of the safety of coastal structures (Section 2.2). Section 2.3 introduces the probabilistic methods which are particularly relevant to the present work. The chapter is not intended to be an exhaustive review of the subject; a large number of references has been published and it is difficult to select among them, particularly for a brief review. The reader is referred to Shinozuka (1983), Ferry Borges & Castanheta (1983), Ditlevsen & Bjerager (1986) and Bjerager (1991), for historic reviews of probabilistic methods. The chapter illustrates the types of concepts and methods which have been used in probabilistic design and assessment of coastal structures, together with their strengths and limitations. In addition to literature related to coastal structures, a range of literature on probabilistic methods applied in other areas of engineering has also proved useful in preparing this chapter, including: Alonso (1976), D'Andrea & Sangrey (1982) and Nguyen & Chowdhury (1985) - geotechnical engineering; Flint & Baker (1976), Schueller & Choi (1977), Fjeld (1977), Jensen et al (1990), Ronold (1990), Potts (1993) and Duggal & Niedzwecki (1994) - offshore structures; Ferry Borges & Castanheta (1983) and Casciati & Faravelli (1985) - structural engineering; Plate & Duckstein (1988) - levees on a river; Yen (1989) - culvert flooding; Helton & Breeding (1993) - nuclear power plants; Jang et al (1994) - ground water flow and contaminate transport; Lumbers & Cook (1993) - water supply systems; Cullen (1990), Lafitte (1993) and Kreuzer (1994) - dams; and Melchers & Stewart (1993) and Frangopol et al (1996) - general engineering.

2.2   General Concepts And Their Application To Coastal Structures

In order to judge whether a man-made or natural structure (e.g. a seawall or a dune) satisfies the requirements that users and society apply with regard to safety and economy, it is possible to use risk analysis methods (CUR-TAW, 1990). The term <u>risk</u> can be defined in different ways, but for the purposes of this research, risk is defined (BSI, 1991a; Royal Society, 1992) as the combination of the probability, during a reference period of time, of an

undesirable event (e.g. a storm, which is a combination of waves and water levels resulting in extreme loading on a structure) and the consequences of its occurrence (e.g. economic loss, casualties, impact on flora and fauna). The method of combination is generally to multiply the probability and consequences. Risk analysis may then be understood as the whole set of activities aimed at quantifying the probability of occurrence, during a reference period of time, of an undesirable event (probability of failure) and its consequences. Calculation of the probability of failure alone is useful; but it fits particularly well into an analysis in which consequences of failure are also considered. This is because it is not only the probability of failure that is important: an event which has a major impact will generally be accepted less readily (i.e. it should have a lower probability of happening) than one which has only minor consequences. For example, it is clear that an accident which does not involve loss of a single life is more acceptable than one in which a thousand lives are lost. In its most general sense, reliability, R, is the probability that the structure will fulfil its design purpose during the reference period (CIRIA, 1984; Thoft-Christensen & Murotsu, 1986). Note that risk analysis and, consequently, the associated concepts (such as probability of failure, reliability, etc.) should always be referred to an interval of time (Joint Committee on Structural Safety, 1978; Ferry Borges & Castanheta, 1983). This interval of time may be taken as the lifetime of the structure (the time of undisturbed functioning) or as a standard time adopted as reference. Even if, for simplicity, this fact is omitted in the following discourse, the reader should always keep it in mind.

The three main elements of a risk analysis (Van der Meer, 1987; CUR-TAW, 1990) are hazards, failure mechanisms and consequences (Figure 2.1).

```
┌─────────────────────────────────────────────┐
│                                               │
│        ┌─────────────────────────────┐       │
│        │  Preparation of an inventory │       │
│        │        of hazards           │       │
│        └─────────────────────────────┘       │
│                       │                       │
│                       ▼                       │
│        ┌─────────────────────────────┐       │
│        │  Formulation of the failure │       │
│        │         mechanisms          │       │
│        └─────────────────────────────┘       │
│                       │                       │
│                       ▼                       │
│        ┌─────────────────────────────┐       │
│        │     Calculation of the      │       │
│        │   probabilities of failure  │       │
│        └─────────────────────────────┘       │
│                       │                       │
│                       ▼                       │
│        ┌─────────────────────────────┐       │
│        │    Quantification of the    │       │
│        │   consequences of failure   │       │
│        └─────────────────────────────┘       │
│                       │                       │
│                       ▼                       │
│   ┌───────────────────────────────────────┐ │
│   │ RISK = Σ Probability • Consequences   │ │
│   └───────────────────────────────────────┘ │
│                                               │
└─────────────────────────────────────────────┘
```

$$RISK = \sum Probability \cdot Consequences$$
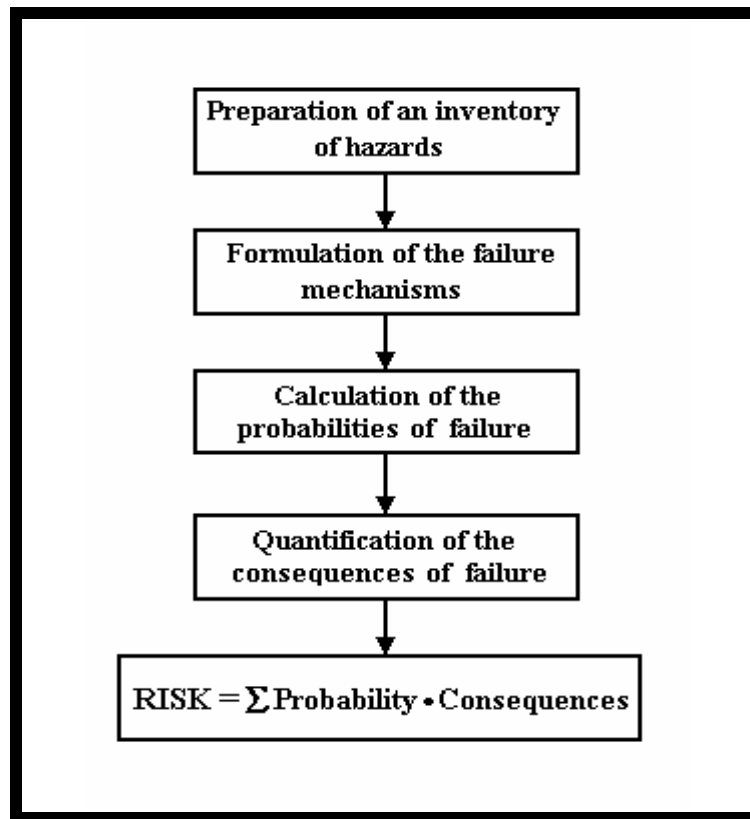
Figure 2.1: Elements of a risk analysis (modified after Van der Meer, 1987); the symbol "•" in the definition of risk means the product.

A risk analysis begins with the preparation of an inventory of the <u>hazards</u> which can be defined as anything which may occur during the lifetime of the structure that can potentially cause harm or loss (Godfrey, 1994). The ways in which the structure responds to hazards are called the <u>failure mechanisms</u> or <u>failure modes</u>. Note that in assessing the safety of coastal structures, it is very important to consider the structure as a whole system. This is known (Burcharth, 1992) as <u>probabilistic analysis of failure mode systems</u>. Some structures are rather complex and, for simplicity, they are considered as series systems, parallel systems, or a combination of both (CIAD, 1985; CIRIA/CUR, 1991). Techniques also exist to provide a logical description of the many hazards and mechanisms resulting in failure of a structure. These include the so-called event trees, which relate to consequences, and fault trees, which relate to causes (see, for example, Paté-Cornell, 1984; Melchers, 1987; Van der Meer, 1987; Yen, 1989; Andrews & Moss, 1993). Fault trees and event trees can be very complex in practice (e.g. Cullen,

1990). Only schematic examples are shown here (Figures 2.2 and 2.3). For real examples, the reader is referred to Mol et al (1984), CIAD (1985), CUR et al (1987); Vrijling (1990), CUR-TAW (1990), CIRIA/CUR (1991), Thomas & Hall (1992), Townend (1994a), Thomas (1994), Simm et al (1996) and others.

The underline event tree is a deductive logical diagram (Paté-Cornell, 1984). Starting from an undesirable initiating event leading to a consequence for the state of the structure, it gives all possible sequences of following events (both wanted and unwanted) and determines the outcome of each considered sequence (Figure 2.2). Typical examples of initiating events include intense wave action, high water levels, strong currents, earthquakes, ice damage, collision and vandalism. Each branch of the event tree is unique and represents a distinct series of events possibly leading to failure. The assessment of an event cannot be made in isolation; it must be considered as part of a sequence of events and changes to the structure that have lead to its occurrence. The event tree methodology is useful in the analysis of the consequences of an initiating event and provides a means of identifying top events for fault trees (CIAD, 1985). A top event is a particular failure mode. In the case of a seawall, for example, it might represent the fact that the crest of the wall is built too low.
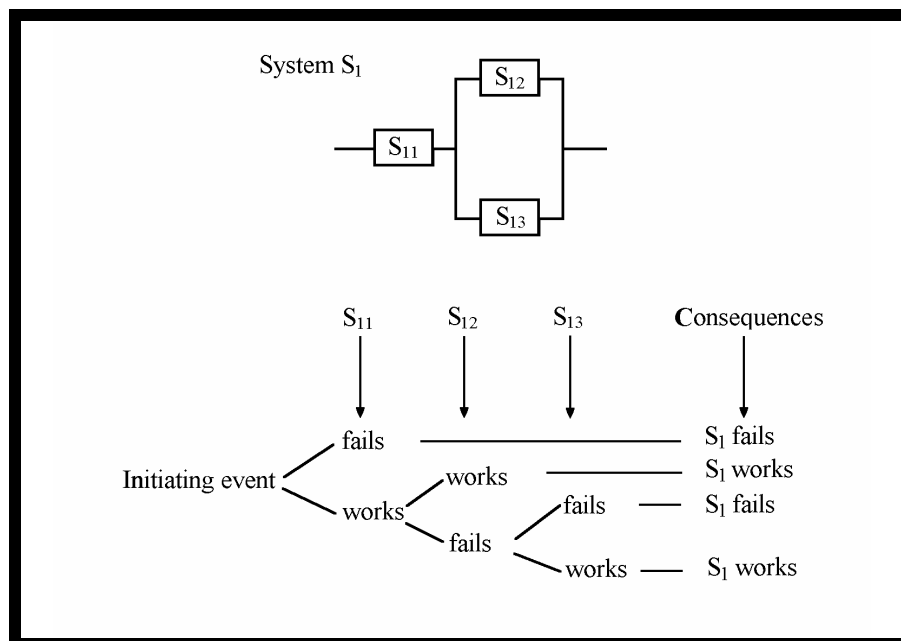
Figure 2.2: Event tree of parallel-series system S_1 (modified after CIAD, 1985).

A <u>fault tree</u> can be constructed starting at the top event and describing, by means of inductive logic, the possible initiating causes leading to the particular failure (Figure 2.3). It is related to component events and basic events by means of logical AND and/or OR gates. If one event alone can cause the top event, the occurrence in the fault tree is represented by an OR gate. If all related events are required to cause the top event, this occurrence is represented by an AND gate. The development stops if the related inputs arise from basic events only, which are generally independent of one another (Lafitte, 1993). When developing fault trees, it is important to keep in mind that each pathway up through the tree forms a unique sequence of events describing a failure, starting from a basic event, and it describes all the unwanted events leading to the unwanted top event. The quantitative analysis of the fault tree involves calculating the probability of a unique undesirable top event from the probabilities of occurrence of the basic events (Lafitte, 1993). So one proceeds systematically from the base towards the top of the tree, the probability obtained at one level being used for the calculation at the level immediately above. Each fault tree considers only one of the many possible system failure modes. Consequently, more than one fault tree may be constructed during the assessment of any system.
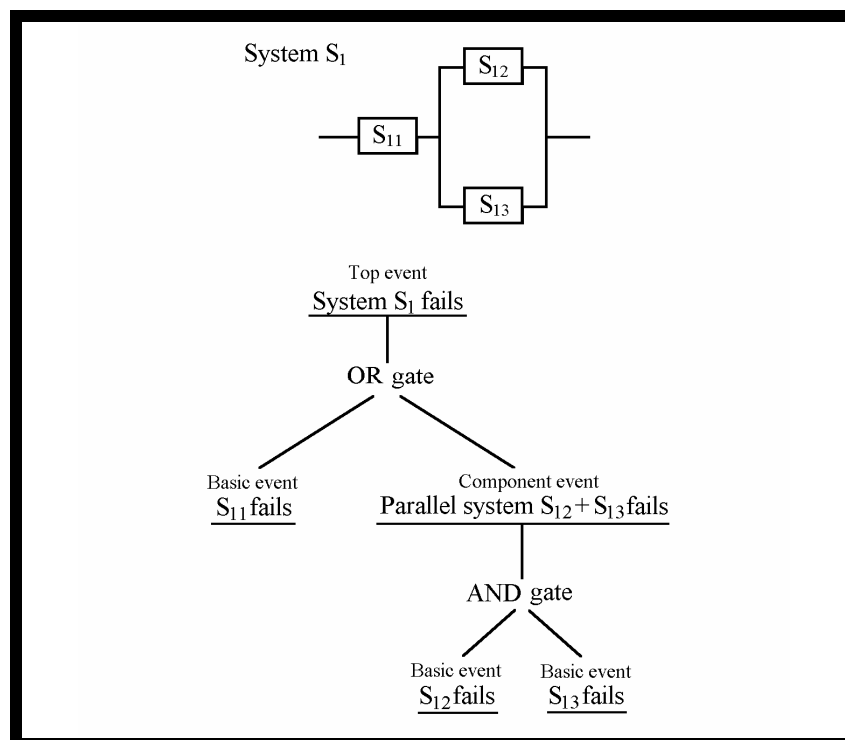
Figure 2.3: Fault tree of parallel-series system S_1 (modified after CIAD, 1985).

Those combinations of components which, if they all fail, cause system failure are called <u>cut sets</u> (Andrews & Moss, 1993). A <u>minimal cut set</u> is a cut set such that if any component is removed from the combination, the system no longer fails (Paté-Cornell, 1984). The calculation of the system probability of failure can only be performed after the minimal cut sets have been determined. For example, two minimal cut sets exist in Figure 2.3: one contains S_{11} and the other contains S_{12} and S_{13}.

The drawback of event trees and fault trees is that they are rather strictly regulated (CUR-TAW, 1990; Burcharth, 1992): in an event tree, it is not possible to combine branches, and no dividing of branches is possible in a fault tree. Furthermore, the system is essentially binary in character: an event occurs or it does not. In coastal engineering, problems of a more continuous character occur.

Event trees and fault trees have been constructed and presented for some coastal structures (CIAD, 1985; Van der Meer, 1987; CUR-TAW, 1990; CIRIA/CUR, 1991), but they have not been applied in full as logic diagrams, except for individual structures which are the subject of very detailed study. They have almost always served as schematic representations of failure modes rather than as strict logical representations of failures. Some authors (Meadowcroft et al, 1994) have found that fault trees and event trees, as used in the electronics and chemical industries, may be suitable for systems of binary components that either fail or do not fail, but they are not sufficient on their own to represent failure of seawalls and related structures which exhibit complex failure modes with interactions between them. For example, overtopping and geotechnical failure of the landward face of a seawall may not, individually, pose a high risk, but the damage due to the geotechnical failure will make erosion due to overtopping much more likely: the modes interact. Furthermore, the quantity of water overtopping is of great importance in determining the consequences: it is not possible to say that the system either "fails" or "works", since a spectrum of outcomes can result.

A useful alternative to event and fault trees is provided by the so-called <u>cause-consequence charts</u>. They overcome many of the drawbacks of event and fault trees outlined above. An example is given in Figure 2.4.
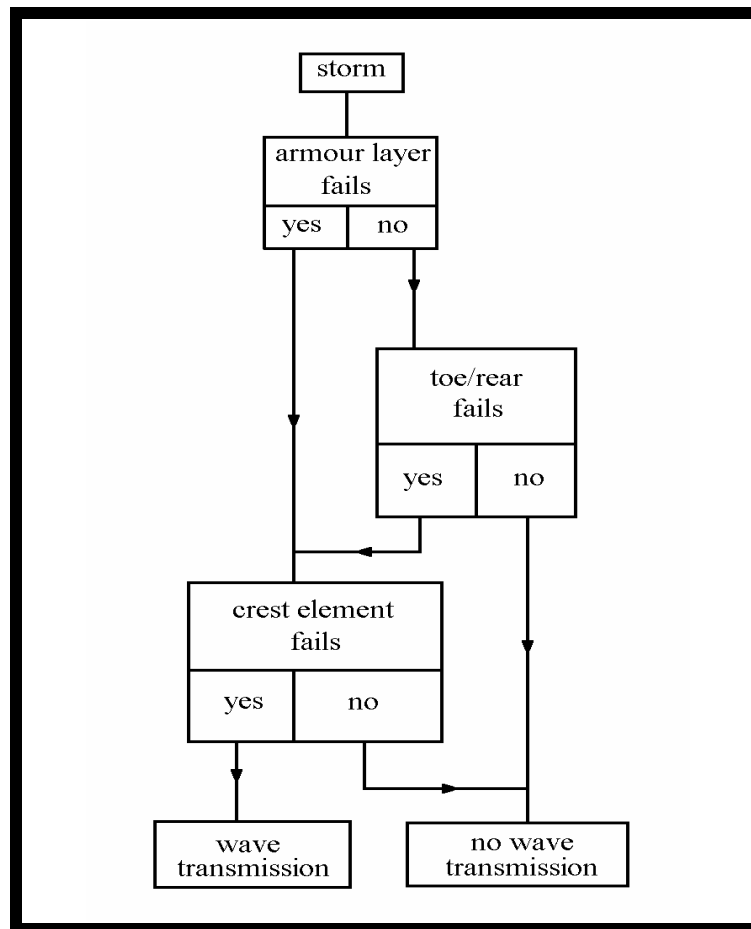
Figure 2.4: Example of a cause-consequence chart
(after Van der Meer, 1987).

The concepts outlined above concern static event trees: the corresponding probability of failure applies to the final state, assuming a rapid propagation of damage once a section fails. In general, the use of dynamic event trees, which cover time-dependent aspects, should be considered. It is possible to assess dynamic problems using model simulations which include time-dependent effects. However, this approach requires complex methods which are not further discussed here (see, for example, Cumo & Naviglio, 1987; Casciati & Faravelli, 1991; Andrews & Moss, 1993).

While preparing the inventory of hazards and identifying the possible failure modes of a structure, it is obvious that from all possible failure modes, only a few are of real importance (Van der Meer, 1987). Others have such a low probability of occurrence that they may be disregarded, provided that they are independent with respect to other failure modes. However, in principle,

all failure modes must be identified and studied as far as is necessary to establish what degree of risk they pose (Meadowcroft et al, 1994). Neglect of an important failure mode will bias the estimation of the safety of the structure (Burcharth, 1990, 1992; Lamberti, 1992). Furthermore, it is important to investigate the way failure modes combine or interact, as indicated, for example, in a cause-consequence chart. Consideration must be given to two particular factors (PIANC, 1992): physical correlation (e.g. the failure of one mode triggers the failure of another) and correlation through common parameters (e.g. the same parameter triggers two different failure modes). It is not generally known how to quantify the first of these factors, even if physical correlation can be identified. Consequently, only the second can be dealt with in a quantitative way.

After identifying the failure modes and their mutual relationships, assessment of the safety of a structure depends fundamentally on the description of the individual failure modes (Thoft-Christensen & Murotsu, 1986). For each failure mode, a theoretical model may exist. Failure modes for which no mathematical-physical description is available, or for which the model is rather poor, become apparent (CIAD, 1985; CIRIA/CUR, 1991). This situation arises mainly because the load and/or the structural behaviour is complex and is not fully understood. If no theoretical models are available for a failure mechanism, simple empirical formulae can be used to describe the physical process (CIAD, 1985). When no theoretical models or empirical formulae are available, it is necessary to work on the basis of engineering judgement (Van der Meer, 1987).

Theoretical models or empirical formulae may be applied to define what is called a failure function or limit state function, Z (Figure 2.5). This is a function of the basic variables, X_i i=1,...,N, of the problem (e.g. water level, wave conditions, structure dimensions, material properties), which are the fundamental quantities that the designer has to consider and which may influence the reliability of the structure with respect to a particular failure mode (CIRIA, 1984). Note that it may not be possible to express Z as an explicit function of the basic variables. Provided the function is continuous, this is of no consequence. However, if the function is discontinuous (e.g. the Van der Meer expressions for the stability of rock armour on breakwaters), it must be examined as a series of continuous functions. In any case, the simultaneous values of the variables X_i must stay within certain limits in order

that the structure behaves as it is intended to do for the failure mode under study. For a given structure and failure mode, these limits may be described in terms of the failure function which divides the N-dimensional space of the X-variables into two sets (Figure 2.5): $Z=f(X_1,...,X_N)$ denotes <u>safe states</u> if $Z>0$ and <u>failure states</u> if $Z<0$. $Z=0$ defines the <u>failure surface</u>. As noted by Madsen et al (1986), it is often convenient to include the failure surface in the failure states (i.e. $Z \leq 0$ defines failure) and it is this definition of failure states which is used in the present research. In other words, the <u>probability of failure</u>, P_f, is defined as $P_f=P(Z \leq 0)$, whilst the <u>reliability</u>, R, is defined as $R=1-P_f$.
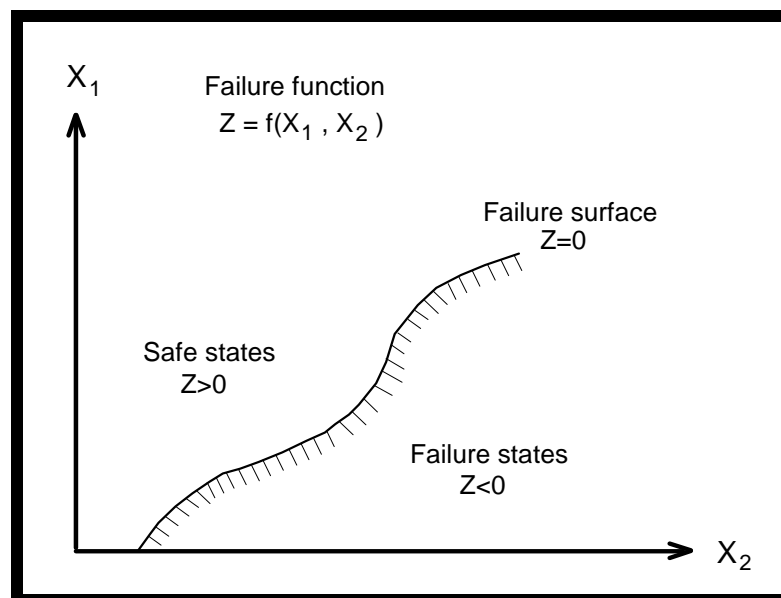
Figure 2.5: Definition of the failure surface Z=0 for the case of two basic variables, X_1 and X_2.

Note that Z defines what is generally called a <u>limit state</u>, which is a limiting condition beyond which a structure is assumed to become unfit for its purpose (CIRIA, 1984). A distinction has to be drawn between <u>ultimate limit states</u> and <u>serviceability limit states</u>. The former refer to conditions in which the structure is unable to fulfil its principal functions (e.g. the failure of a seawall in preventing extensive flooding or erosion of the backshore). Serviceability limit states, on the other hand, exist when damage of considerable magnitude to the structure has occurred but it remains possible to rely on the structure for its main function. Under these conditions,

some disturbance related to normal use and durability of the structure is expected (BSI, 1991a).

After describing the relevant failure modes, calculation of the probability of failure associated with each of them can be performed using a single failure mode probability analysis (Burcharth, 1992), as described in Section 2.3. These probabilities are then used to determine upper and lower bounds for the probability of failure of the whole structure (Ditlevsen, 1979b; CUR-TAW, 1990). For more detailed discussions of this subject, see Thoft-Christensen & Baker (1982), Hohenbichler & Rackwitz (1983), Ang & Tang (1984), CIAD (1985), Madsen et al (1986), Thoft-Christensen & Murotsu (1986), Ditlevsen & Bjerager (1986), Melchers (1987), CUR-TAW (1990) and Bjerager (1991).

Finally, the consequences of failure must be considered. Whilst there are many events for which the consequences are obvious, there are others for which the outcomes are less easy to predict (Lumbers & Cook, 1993). An example of consequences which are obvious might be the overtopping of a seawall by a relatively small volume of water causing inconvenience and/or injury to pedestrians, but not affecting the safety of the structure. The outcome would be less easy to predict where the volume of water causes significant flooding. In this case, the consequences would depend on a number of factors such as the time of day during which flooding occurs, the storm warning service, the efficiency of people's evacuation of the expected flooded area, etc.

The probability of failure multiplied by the consequences constitutes the risk (Figure 2.1). Risk has the units of the consequences (Meadowcroft et al, 1995; Simm et al, 1996): for risk to an individual, the units of expression may be in terms of fatalities per hour or per year of the individual's activity; for risk to society, it may be expressed in units of the expected number of people to be affected to a specified degree per year; economic risk expresses the expected loss in monetary terms.

In civil engineering, there is no such concept as total safety, but there are higher or lower risks of failure (Pate, 1981; CIRIA, 1984; Vasco Costa, 1990; Pita, 1992; Vrijling, 1993). The risk of failure can only be minimised or, more realistically, the safety of the structures can be optimised to a degree

consistent with available information and with justified socio-economic investment. The acceptable risk depends on the structural characteristics and on the consequences of failure. For a breakwater, the acceptable risk during the expected lifetime of the structure can vary from a large value (e.g. 10^{-1}) if the consequences are insignificant, to a very small number (e.g. 10^{-4}) if the failure of the breakwater would result in significant damage (Burcharth, 1991b). For offshore structures, values in the range of 10^{-6} to 10^{-8} are recommended (Potts, 1993). Optimal design is, essentially, weighing the risks against the costs of providing higher levels of safety (BSI, 1991a; MAFF, 1993c). If there are no intangible damages (damages which cannot be expressed or evaluated in monetary terms) then the design probability of failure or acceptable level of risk may be chosen by a process of optimisation using methods like cost-benefit analysis (CIAD, 1985; CIRIA/CUR, 1991; PIANC, 1992; Lafitte, 1993; Parker, 1993). Otherwise, the total damage (tangible and intangible) must be quantified in some way; this is a complex problem which gives rise to many discussions and ethical objections (Pate, 1981; Madsen et al, 1986; Green & Penning-Rowsell, 1989; CUR-TAW, 1990; Penning-Rowsell et al, 1992). Discussion of design optimisation is beyond the scope of the present research. Reference is made to Dover & Bea (1979), Brennan & Stickland (1981), Pate (1981), Allen (1981), Nielsen & Burcharth (1983), CIRIA (1984), CIAD (1985), Bruun (1985), Casciati & Faravelli (1985), CIRIA (1986a), Smith (1987), Parker et al (1987), CUR-TAW (1990), Vrijling (1990), De Haan (1991), Ryu et al (1992), Thomas & Hall (1992), Barber (1993), MAFF (1993b), and Ruiz & Quirós (1994).

In risk analysis, it is important to have a good overview both of the uncertainties involved and of the related consequences. Without such knowledge, it is impossible to evaluate the safety of a structure, a situation that is unacceptable for a professional engineer (Burcharth, 1985). To investigate the influence of uncertainty on safety evaluation, it is first necessary to identify and define what is not known (Kreuzer, 1994). Engineering systems inevitably involve many uncertainties in their planning, design and operation (Yen, 1989). It is important to acknowledge these uncertainties, even though they are obviously difficult to quantify (Burcharth, 1985; Melchers, 1987). It is a huge step forward simply to identify them and quantify them approximately.

Probabilistic methods are used to evaluate engineering safety. At the same time, they must account for the uncertainties in the various contributing factors and evaluate their implications for engineering design. Each basic variable, X_i, in the failure function, Z, is a potential contributor to these uncertainties. Moreover, a random variable, X_i, might not be a directly measurable physical quantity; it can itself represent the uncertainty in a specific factor. It can be an error term included as a variable in the failure function (Manners, 1990; Der Kiureghian, 1990).

Uncertainties in the study of a single failure mode may include (Yen, 1989; Burcharth, 1992):

- **Uncertainties related to failure mode formulae**

    Whether the formula used to describe the "real" behaviour of the structure is based on theoretical considerations or physical model tests, simplifications and idealisations are made during its development which give rise to uncertainties. In certain cases, the uncertainties associated with a failure mode formula may be much more significant than the uncertainties associated with the basic variables in the problem (Ang & Tang, 1975; Thoft-Christensen & Murotsu, 1986; Burcharth, 1992). This is clearly seen from the many diagrams presenting a formula as a smooth curve shrouded in a widely scattered cloud of data points (usually from physical model tests) which are the basis for the curve fitting. Coefficients of variation of 15-20% or even larger are quite normal. The range of validity and the related coefficient of variation should always be considered when using a formula.

- **Uncertainties related to environmental parameters**

    The specification of environmental criteria is one of the crucial steps in maritime engineering. In various coastal areas, the engineer is confronted with waves, currents, winds, storm surges, etc. The ideal situation, where both short-term and long-term wave statistics can be established from on-site measurements, almost never exists (Burcharth, 1985). According to PIANC (1992), uncertainties related to environmental parameters arise, mainly, due to: i) errors in instrument response or visual observation; ii) variability and errors due to different and imperfect calculation methods; iii) statistical uncertainties related both to short-term randomness of the variables and to extrapolation from small sets of data to events of low probability of occurrence; and iv) choice of theoretical distributions as representatives of the unknown long-term distributions.

- **Uncertainties related to structural parameters**

  The uncertainties related to material parameters (e.g. density), to geometrical parameters (e.g. slope angle, size of structural elements) and fracture strength (e.g. of concrete blocks) are generally much smaller than the uncertainties related to the environmental parameters and to the design formulae.

Note that in a considerable number of failures, human factors have been the predominant overall component (Blockley, 1981; Madsen et al, 1986; Thoft-Christensen & Murotsu, 1986; Melchers, 1987; Burcharth, 1987; Manners, 1990; Melchers, 1993). They may be mistakes in design, analysis, construction, maintenance, or use of the structure (Townend, 1994a). Therefore, an estimate of its reliability is incomplete without considering human factors. Ways of reducing accidents caused by human errors include quality assurance techniques (see for example, PIANC, 1988; BSI, 1990, 1991c; CIRIA/CUR, 1991; Lafitte, 1993) which have developed considerably in recent years.

It is beyond the scope of this contribution to discuss in any more detail the many uncertainties related to the study of a single failure mode. However, further information may be found in Ang & Tang (1975, 1984), Blockley (1981), Thoft-Christensen & Baker (1982), Burcharth (1985), Ditlevsen & Bjerager (1986), Melchers (1987), Manners (1990), Der Kiureghian (1990), Burcharth (1992) and HR Wallingford/Sir William Halcrow & Partners Ltd (1993).

Some existing studies on coastal structures have aimed at being very broad, covering various failure modes and consequences (CIRIA, 1984; CUR/TAW, 1990; HR Wallingford/Sir William Halcrow & Partners Ltd, 1993, 1995). Others have focused mainly on the study of a specific failure mode (CIAD, 1985; PIANC, 1992; Allsop & Meadowcroft, 1995). This demonstrates the difficulty in achieving a satisfactory compromise between engineering and mathematical accuracy, and developing a procedure which can be applied to a wide range of coastal structures.

This research does not consider a full risk assessment procedure for coastal structures. It has been obvious during the current review of literature that, at the moment, there is insufficient knowledge to enable such an analysis to be carried out (Burcharth, 1990). This study concentrates on the probabilistic

assessment of individual failure modes, particularly seawall wave overtopping and dune erosion. For these failure modes, it concentrates on calculating the probability of failure during a specified reference period.

2.3　Single Failure Mode Probability Analysis

2.3.1　Introduction

This section discusses techniques to quantify the probability of occurrence of a particular failure mode represented by the failure function $Z=f(X_1,...,X_N)$ where X_i are the basic variables of the problem. For most practical applications, each basic variable, X_i, is random with a probability density function f_{X_i}. The failure function, Z, is generally a non-linear function of the basic variables.

Now, assume that Z is a function of only two random variables, X_1 and X_2, i.e. $Z=f(X_1,X_2)$. Given f_{X_1,X_2} as the joint probability density function (jpdf) of X_1 and X_2, then the probability of failure, P_f, during a specified reference period, can be expressed as:

$$P_f \; = \; P(Z \le 0) = \int\!\!\int_{Z \le 0} f_{X_1,X_2} \, dX_1 \, dX_2 \qquad (2.1)$$

If, and only if, the variables can be assumed statistically independent, the jpdf is determined from the product of the probability density functions f_{X_1} and f_{X_2}. The above equation may then be replaced by:

$$P_f = \int\!\!\int_{Z \le 0} f_{X_1} f_{X_2} \, dX_1 dX_2 \qquad (2.2)$$

In this case, the integral is equivalent to the volume enclosed between the horizontal plane $f_{X_1,X_2} = 0$ and the jpdf f_{X_1,X_2}, in the space where the condition $Z \le 0$ is fulfilled. With two variables only, the jpdf can be shown as a surface represented by contours and the failure boundary can be drawn as a line (Figure 2.6). Figure 2.6 also shows the so-called design point which is the point on the failure surface where the jpdf attains its maximum value, i.e. the most probable point of failure.
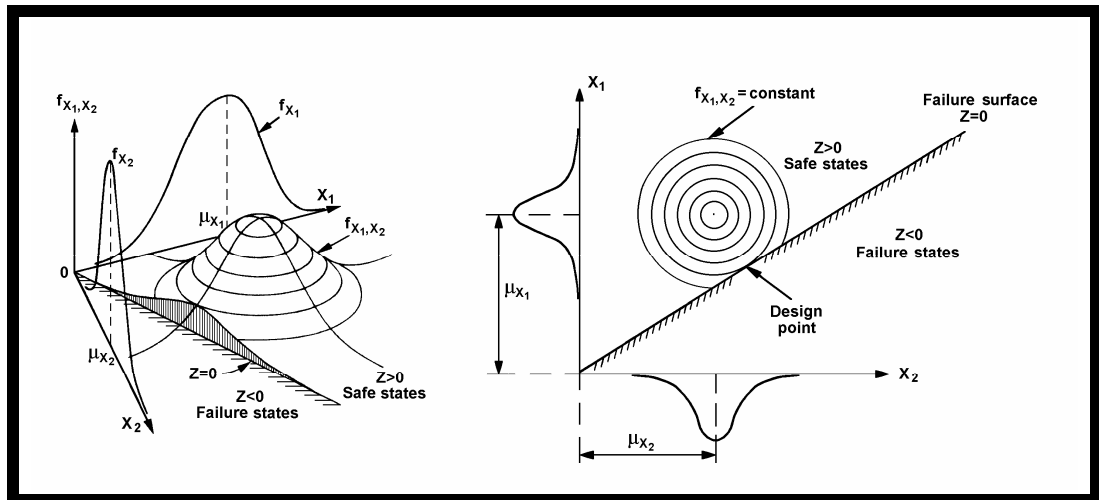
Figure 2.6: Two-dimensional joint probability density function for variables X_1 and X_2 (modified after Melchers, 1987).

Generally, Z is a function of more than two random variables. In this case, it is not possible to describe the jpdf as a surface but it requires an imaginary multi-dimensional space (CIRIA/CUR, 1991; Burcharth, 1992). The probability of failure is then written as follows:

$$P_f = P(Z \leq 0) = \int \int_{Z \leq 0} \ldots \int f_{X_1,\ldots,X_N} dX_1 \ldots dX_N \qquad (2.3)$$

where, again, if $X_1,\ldots,X_N$ are statistically independent:

$$P_f = \int \int_{Z \leq 0} \ldots \int f_{X_1} \ldots f_{X_N} dX_1 \ldots dX_N$$

$$= \int \int_{Z \leq 0} \ldots \int \prod_{i=1}^{N} f_{X_i} dX_i \qquad (2.4)$$

These equations form the mathematical basis for probabilistic analysis. Except for some special cases, the above integrations cannot be performed analytically and have to be approximated in some way (Ferry Borges & Castanheta, 1983; Melchers, 1987; Hohenbichler et al, 1987). This is the aim of the various probabilistic methods. They are classified on the basis of the types of calculations performed and of the approximations made. In essence, the designer must choose to what degree of sophistication he wants to formulate failure. In general, three common levels are distinguished

in the literature (Ferry Borges & Castanheta, 1983; Mol et al; 1983; CIAD, 1985; CUR-TAW, 1990; Burcharth, 1992; PIANC, 1992). They are listed in order of decreasing accuracy and complexity as follows:

- **Level III** - **Full distribution approach**

   This method provides an "exact" probabilistic analysis for whole structural systems, or structural elements, using full joint probability density functions including the correlations among the variables. The probability content of the entire failure region is evaluated (as opposed to Level II methods which comprise a check at only a single point on the failure surface).

- **Level II** - **Semi-probabilistic approach**

   Approximation methods are applied in which the generally correlated and non-Normal variables are transformed into uncorrelated and Normal variables. Reliability indices are used as measures of the structure reliability. Non-linear failure functions are approximated using a tangent hyperplane at some point (First Order Methods), using a quadratic approximation (Second Order Methods) or even higher order approximations[1]. Since the second and higher order methods complicate considerably the computations and, in many cases, the First Order Methods give very good approximations (Ditlevsen & Bjerager, 1986), only the First Order Methods are described in this thesis. In these methods, the failure function is linearized at a specific point in order to determine the actual probability of failure. If linearization is performed about the expected mean values of the variables involved, the method is known as the First Order Mean Value Approach, FOMVA. If the failure function is linearized about the point in the failure surface having the highest joint probability density (design-point) then the method is called a First Order Reliability Method, FORM (Burcharth, 1990, 1992). This approach requires an iterative procedure in the case of non-linear failure functions.

- **Level I** - **Limit state approach**

   This level comprises calculations based on characteristic values and partial load and resistance factors. The factors represent, for example, the ratio of load at failure to permissible working load. This creates a desired margin between the characteristic values of

---

[1]   For further details of the second and the higher order methods, the reader is referred to Fiessler et al (1979), Tvedt (1983), Breitung (1983, 1984), Madsen et al (1986), Der Kiureghian et al (1987), Naess (1987), Tvedt (1988, 1990), Der Kiureghian & De Stefano (1990), Hohenbichler & Rackwitz (1990), Casciati & Faravelli (1991), Bjerager (1991), Jang et al (1994), Koyluoglu & Nielsen (1994), Cai & Elishakoff (1994), Wang & Grandhi (1995) and Grandhi & Wang (1996).

resistance and working load. In this approach, a characteristic load is established (for example, a wave height with a certain return period) for which hardly any damage should occur. Strictly speaking, a calculation at Level I does not allow the determination of the reliability (or the failure probability) of the design. Consequently, it is neither possible to optimise nor to avoid over-design of a structure. It does, however, provide a method of checking whether a defined level of safety is satisfied.

A less common fourth level of probabilistic approach has also been advanced (Madsen et al, 1986; Plate & Duckstein, 1988; Casciati & Natale, 1992). This level accounts for the principles of engineering economic analysis under uncertainty, considering costs and benefits of construction, maintenance, repair, consequences of failure, etc.

The above classification of reliability methods is not exhaustive, but it has proved to be very useful in practical discourse on reliability methods. For example, this classification does not refer to any combination of the above methods using the advantages of each (see, for example, Super-Software, 1994). In the next few sections, special attention is paid to the FORM method since it is the basis of a FORTRAN program, PARASODE, developed as part of this research. However, for a fuller treatment of the method, the reader is referred to Thoft-Christensen & Baker (1982) and Madsen et al (1986). Numerical methods which rely on sampling are also briefly outlined since they are the methods applied by the software package @RISK which has been used in this research to validate the results obtained from PARASODE. Finally, Level I methods are introduced for completeness (for more detailed information, see CIRIA, 1984).

2.3.2 Level III Methods

2.3.2.1 Numerical Integration

The N-fold integral in eq. (2.3) is solved by full numerical integration, i.e. substitution of the infinitely small integration steps dX_i by finite discrete integration steps ΔX_i , and substitution of the integrals by summations. Each summation must be done using a discrete number of steps and the integration area for each random variable must be evaluated between a starting value and an end value instead of running from $-\infty$ to $+\infty$ (see, for example, Melchers, 1987). The method would be exact if an unlimited number of integration steps were used. Of course this is not possible. International research seems to focus on Level II methods and traditional sampling variations, but rarely on numerical integration, although this is the most correct method (Super-Software, 1994). Little literature on this topic is available. No general formulae have been found in the literature for solving a multi-dimensional integral.

2.3.2.2 Numerical Methods Which Rely On Sampling

Introduction

As the complexity of an engineering system increases, the required analytical model may become extremely difficult to formulate mathematically unless gross idealization and simplifications are invoked; moreover, in some cases even if a formulation is possible, the required solution may be analytically intractable. In these instances, a probabilistic solution may be obtained through sampling which is the process by which values are randomly drawn from the input probability distributions. In this case, the N-fold integral in eq. (2.3) is solved by letting the computer generate values for the limit state function, using a random number generator. Basically, this means that the computer needs an algorithm which generates random numbers between zero and one from a uniform distribution. Several techniques for drawing random samples are available (Ang & Tang, 1984). For work that requires the use of digital computers, it is convenient to compute a sequence of numbers by a systematic procedure. Such procedures are devised so that reasonable statistical tests do not detect any

significant deviation from randomness. Such a sequence of numbers can be duplicated exactly (e.g. for checking purposes) and therefore, strictly speaking, are really not random; for this reason, they are called "pseudo" random numbers (Hammersley & Handscomb, 1964; Haugen, 1968; Rubinstein, 1981; Cope et al, 1982; Ang & Tang, 1984; Melchers, 1987). For most practical purposes, a sequence of numbers generated by a suitable "pseudo" random number generator is indistinguishable from a sequence of strictly true random numbers (Rubinstein, 1981). The generated "pseudo" random numbers are cyclic, that is, they are repeated with a given period. To insure reasonable randomness, the period should be as long as possible. When evaluating the techniques for drawing samples, the most important factor to consider is the number of iterations required to accurately recreate an input distribution. Choosing a sampling method affects both the quality of the results and the length of time necessary for simulation. The two sampling methods briefly described in this section are those used by the software package @RISK which has been applied in this research to validate the results obtained from the Level II calculations: traditional sampling (often called Monte Carlo Sampling) and Latin Hypercube Sampling (LHS).

Once the standard uniformly distributed numbers have been obtained, random numbers with a prescribed distribution may be generated through (Rubinstein, 1981; Ang & Tang, 1984; Law & Kelton, 1991): i) direct methods (e.g. the inverse transform method, the composition method, and the convolution method); or ii) indirect methods (e.g. acceptance-rejection method). The particular method used depends on the distribution from which one wishes to generate. The reader is referred to the literature for more details on these methods.

Sampling is done repetitively, with one sample being drawn every iteration from each input probability distribution. With enough iterations, the sampled values become distributed in a manner which approximates the known input probability distributions.

Clearly, the probability distributions of the governing parameters must first be specified. An equation is then used to link the parameters to the outcome. The output process is sampled by choosing a random value for each input parameter. The outcome of the event is then recorded. The procedure is

repeated for a sufficient number of events to accurately determine the probability distribution of the outcomes.

Traditional Sampling

The term "Monte Carlo" was introduced by von Neumann and Ulam during World War II, as a code word for the secret work at Los Alamos; it was suggested by the gambling casinos in the city of Monte Carlo in Monaco (Rubinstein, 1981).

Monte Carlo Sampling is the traditional technique for using random numbers to draw samples from a probability distribution. Any given sample may fall anywhere within the range of the input distribution (Figure 2.7). Samples are, of course, more likely to be drawn from areas of the distribution associated with the higher probabilities of occurrence.
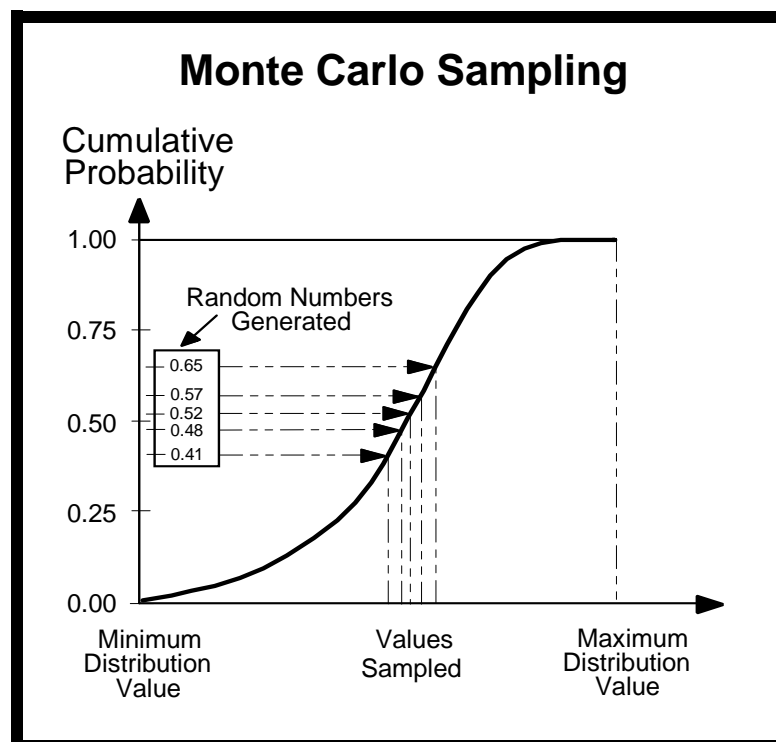
Figure 2.7: Five iterations of Monte Carlo Sampling with clustering (modified after Palisade Corporation, 1994).

When a small number of iterations is performed, a problem of clustering may arise. This causes particular difficulties when a distribution includes low probability outcomes which could have a major impact on the results (Haugen, 1968; Ang & Tang, 1984; Startzman & Wattenbarger, 1985; Ditlevsen & Bjerager, 1986; Wen & Chen, 1987). These outcomes have to be sampled. But if their probability is very low, a small number of iterations may not provide sufficient of these outcomes to accurately represent their probability. However, an increase in the number of iterations is not always computationally convenient. This problem has led to the development of techniques which reduce the sampling error without increasing the sample size. These techniques are known as variance reduction techniques and include (Hammersley & Handscomb, 1964; Halton, 1970; Rubinstein, 1981; Ang & Tang, 1984; Morgan, 1984; Smith & Buckee, 1985; Melchers, 1987; Bjerager, 1991; Law & Kelton, 1991): i) Control-Variate Sampling or Correlated Sampling; ii) Antithetic Variate Sampling; iii) Importance Sampling; iv) Stratified Sampling or Systematic Sampling (e.g. LHS); v) Implicit Multicorrelated Sampling or the E-Z-H method after Ermakov, Zolotukhin and Handscomb; and vi) Conditional Expectation Sampling. Note that comparison of earlier references indicates some differences in nomenclature for similar techniques.

In general, the reason why people specify the traditional method is because of the length of time that it has been around. Other sampling techniques such as LHS have not been as widely implemented.

Latin Hypercube Sampling

The LHS method is designed to accurately recreate the input distribution, preserving the randomness of the traditional method while using fewer samples. Typically, LHS requires about one third of the traditional method's iterations to get equal or better results (Palisade Corporation, 1994; Murtha, 1995). The key to this process is stratification of the input probability distribution (McKay et al, 1979; Startzman & Wattenbarger, 1985). Firstly, the cumulative distribution is divided using equal intervals on the probability scale of 0 to 1 (Figure 2.8). Then a sample is randomly taken from each part of the input distribution. Thus, sampling is forced to represent values in each

of the divisions, thus avoiding clustering of values and more accurately reflecting the input probability distribution.
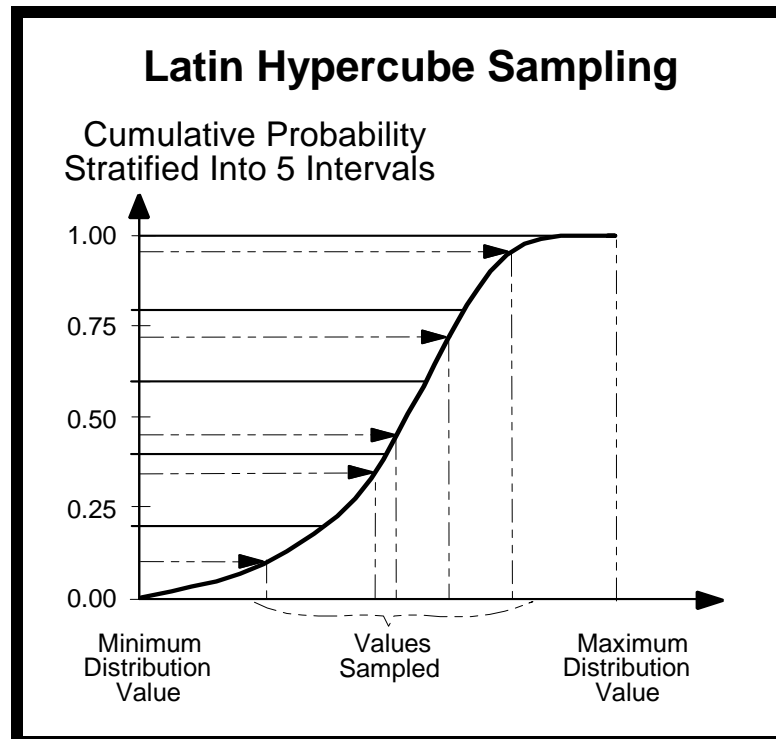
Figure 2.8: Five iterations of Latin Hypercube Sampling (modified after Palisade Corporation, 1994).

The technique being used during LHS is so-called sampling without replacement: the number of divisions of the cumulative distribution is equal to the number of iterations performed. Hence, a sample is taken randomly from each division and once a sample is taken, the division is not sampled again. Note that if a LHS simulation is stopped prior to the execution of the specified number of iterations, the results are still valid. However, they do not reflect all the benefit of the stratified sampling since not all the input strata have been filled. Since the strata which have been sampled from have been randomly selected from across a distribution, the results are at least as good as the equivalent results produced from the same number of Monte Carlo iterations, but not as good as a complete LHS simulation of the same number of samples.

2.3.3   Level II Methods

2.3.3.1 Basic Features Of The First Order Reliability Method

Suppose that the failure function, Z, can be expressed as follows:

$$Z = a_0 + a_1 X_1 + a_2 X_2 + ... + a_N X_N = a_0 + \sum_{i=1}^{N} a_i X_i \qquad (2.5)$$

where $a = (a_0,...,a_N)$ are constants and $X = (X_1,...,X_N)$ are mutually independent Normal basic variables with known means $\mu_X = (\mu_{X_1},...,\mu_{X_N})$ and standard deviations $\sigma_X = (\sigma_{X_1},...,\sigma_{X_N})$. It can be shown (Smith, 1986) that provided the variables, X, follow the Normal distribution and are mutually independent, Z will also follow a Normal distribution having mean, μ_Z, and variance, σ_Z^2 :

$$\mu_Z = a_0 + a_1 \mu_{X_1} + ... + a_N \mu_{X_N} = a_0 + \sum_{i=1}^{N} a_i \mu_{X_i} \qquad (2.6)$$

$$\sigma_Z^2 = a_1^2 \sigma_{X_1}^2 + ... + a_N^2 \sigma_{X_N}^2 = \sum_{i=1}^{N} (a_i \sigma_{X_i})^2 \qquad (2.7)$$

The probability of Z being less than or equal to zero follows from the Normal distribution, with known mean and standard deviation:

$$P_f = P(Z \leq 0) = \int_{Z=-\infty}^{0} f_Z dZ = \Phi(-\beta) \qquad (2.8)$$

where f_Z is the probability density function of Z, Φ is the cumulative distribution function of the standard Normal distribution (tabulated in statistical books such as Abramowitz & Stegun, 1964) and β is the reliability index:

$$\beta = \frac{\mu_Z}{\sigma_Z} \qquad (2.9)$$

Note that β is the inverse of the coefficient of variation of Z and is the distance from the mean value of Z, μ_Z, to the failure surface, Z=0, in terms of the number of standard deviations (Figure 2.9).
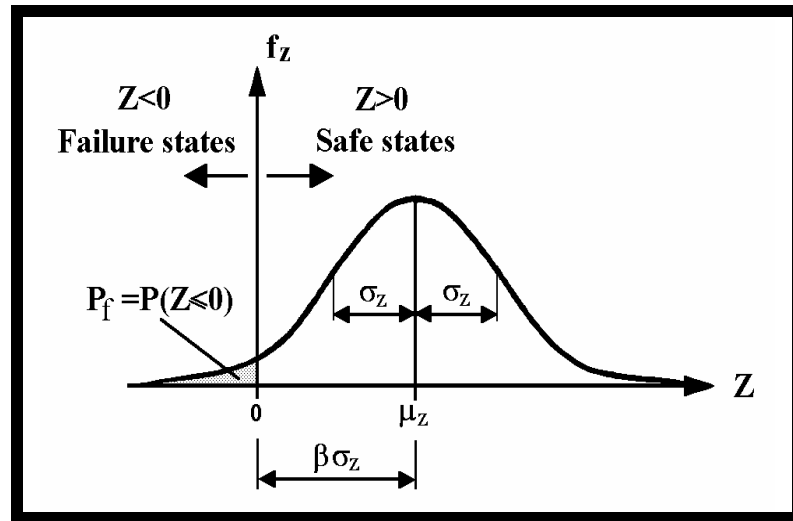
Figure 2.9: Illustration of the reliability index (modified after Burcharth, 1992).

If the basic variables are Normal but correlated then the expression for μ_Z still holds but σ_Z^2 is given by:

$$\sigma_Z^2 \;=\; a_1^2 \sigma_{X_1}^2 + ... + a_N^2 \sigma_{X_N}^2 + \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} \rho_{X_i X_j} \, a_i \, a_j \, \sigma_{X_i} \, \sigma_{X_j} \qquad (2.10)$$

where the last term accounts for correlation between any pair of basic variables and $\rho_{X_i X_j}$ denotes the correlation coefficients:

$$\rho_{X_i X_j} = \frac{C_{ov}[X_i, X_j]}{\sigma_{X_i} \sigma_{X_j}} = \frac{E[(X_i - \mu_{X_i})(X_j - \mu_{X_j})]}{\sigma_{X_i} \sigma_{X_j}} = \frac{E[X_i X_j] - \mu_{X_i} \mu_{X_j}}{\sigma_{X_i} \sigma_{X_j}} \qquad (2.11)$$

E[...] is the expected value operator and represents the expected value of its argument. $C_{ov}[X_i, X_j]$ is the covariance of X_i and X_j; X_i and X_j are said to be uncorrelated if $C_{ov}[X_i, X_j] = 0$, i.e. $\rho_{X_i X_j} = 0$.

Note that eq. (2.8) gives an "exact" probability of failure only if the failure function is linear in X and if all the basic variables are normally distributed.

2.3.3.2 Non-Linear Failure Functions

Normal Random Variables

If the failure function Z is non-linear then approximate values of μ_Z and σ_Z can be obtained by using a linearized failure function (Ferry Borges & Castanheta, 1983; CIRIA, 1984; Melchers, 1987; Burcharth, 1992). Linearization is generally performed by a truncated Taylor series expansion around some point, X^*, retaining only the linear terms. This results in the following approximation for Z:

$$Z \approx Z^* + \sum_{i=1}^{N}\left(X_i - X_i^*\right)\left(\frac{\partial Z}{\partial X_i}\right)^* \tag{2.12}$$

where Z^* is the value of the function Z at the point X^* under consideration, $\left(\partial Z / \partial X_i\right)^*$ is the partial derivative with respect to X_i, likewise evaluated at the point X^*. The mean value and the variance of Z are, respectively:

$$\mu_Z = Z^* + \sum_{i=1}^{N}(\mu_{X_i} - X_i^*)\left(\frac{\partial Z}{\partial X_i}\right)^* \tag{2.13}$$

$$\sigma_Z^2 = \sum_{i=1}^{N}\left[\left(\frac{\partial Z}{\partial X_i}\right)^*\sigma_{X_i}\right]^2 \tag{2.14}$$

The probability of failure and the reliability index are again expressed by eq. (2.8) and eq. (2.9), respectively.

When linearization is performed around the expected mean values, i.e. $X^* = (\mu_{X_1},...,\mu_{X_N})$, the method is often called a First Order Mean Value Approach, FOMVA (Burcharth, 1992). For non-linear failure functions, the errors incurred by neglecting second-order and higher terms in the Taylor expansion increase with increasing distance from the linearization point. Since the mean point $X = (\mu_{X_1},...,\mu_{X_N})$ is likely to be well within the safe region and not on the failure surface, there are likely to be considerable errors in approximating the failure surface if FOMVA is used (CIRIA, 1984).

Another problem associated with FOMVA is that the values of μ_Z and σ_Z, and thereby also the value of β, depend on the choice of the linearization point (Thoft-Christensen & Murotsu, 1986). Moreover, the value of β will change when different but equivalent non-linear failure functions are used (Melchers, 1987; Burcharth, 1992). However, there is no arbitrariness due to the choice of failure function if only information about the failure surface is used, i.e. if the linearization point is selected as a point on the failure surface (Madsen et al, 1986).

The first step to obtain invariability of β is to apply the transformation proposed by Hasofer & Lind (1974) in which the basic variables, X_i, are transformed into a new set of normalised variables, U_i. For uncorrelated normally-distributed basic variables, X_i, the transformation is:

$$U_i = \frac{X_i - \mu_{X_i}}{\sigma_{X_i}}$$ (2.15)

in which case $\mu_{U_i} = 0$ and $\sigma_{U_i} = 1$. Using this linear transformation, the failure surface Z=0 in the X-coordinate system is mapped into a failure surface in the U-coordinate system which also divides the space into a safe region and a failure region (Figure 2.10). Due to the zero mean and the unit standard deviation, the new U-coordinate system has an important characteristic, namely a rotational symmetry with respect to the standard deviations (Ditlevsen, 1979a; Melchers, 1987). Note that the origin of the normalised U-coordinate system corresponds to the mean value of the initial variables and will usually be within the safe region.

Figure 2.10 introduces the Hasofer and Lind reliability index (often called the first order second-moment reliability index) which is defined as the distance from the origin to the nearest point, D, of the failure surface in the U-coordinate system (Hasofer & Lind, 1974). D is called the design point and is the point on the failure surface at which the linear approximation to the failure surface is made. For normally-distributed variables, the coordinates of the design point in the original X-coordinate system are the most probable values of the variables at failure i.e. the design point is that point on the failure surface where the probability density attains a maximum (see CIRIA, 1984, for proof). These coordinates are given as follows:

$$X_i^* = \mu_{X_i} + \alpha_i \beta \sigma_{X_i} \qquad (2.16)$$

where α_i are the so-called sensitivity factors defined as follows:

$$\alpha_i = \left(-\frac{\sigma_{X_i}}{\sigma_Z}\right)\left(\frac{\partial Z}{\partial X_i}\right)^* \qquad (2.17)$$
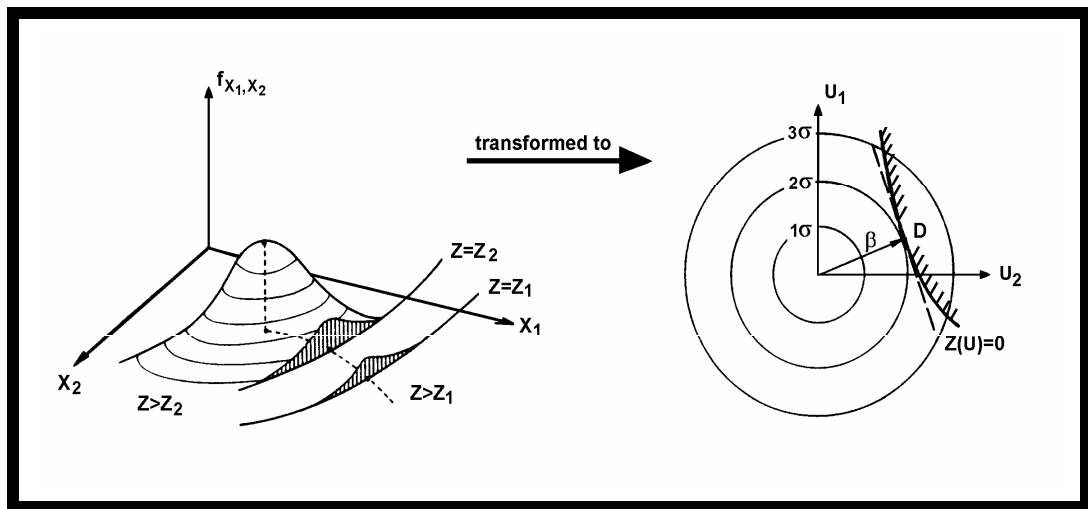
Figure 2.10: Joint probability density function for two random variables and for two reduced variables.

The special feature of the Hasofer and Lind reliability index is that it is related to the failure surface Z(U)=0 which is invariant to the failure function because equivalent failure functions result in the same failure surface (Thoft-Christensen & Murotsu, 1986). The two reliability indices coincide when the failure surfaces are linear (Ditlevsen, 1979a; Madsen et al, 1986). Obviously, this is also the case if non-linear failure functions are linearized by Taylor series expansion around the design point.

Note that linearization about the mean or the design point leads to different results, depending on the shape of the failure function Z. In general, linearization around the design point is very much to be preferred (Melchers, 1987; CUR-TAW, 1990) because the design point is the most probable point of failure. Linearization around mean values can lead to quite erroneous results, but due to the simplicity of the method it might be used to get a first estimate of the failure probability (Burcharth, 1992).

How well a linear function approximates a non-linear function in terms of the resulting probability of failure, P_f, obviously depends on the shape of the non-linear function (Casciati & Faravelli, 1991). If it is concave towards the origin, P_f is under-estimated by the hyperplane approximation. Similarly, a convex function towards the origin implies over-estimation, as in Figure 2.10. Note that all failure surfaces that are tangential to each other at the design point have the same reliability index (Ditlevsen & Bjerager, 1986); for example, if the curved and flat surfaces of Figure 2.10 are considered as failure surfaces of two different structures, the reliability indices for the two structures are the same suggesting equal reliability, whereas the structure with the curved surface is clearly more reliable than the one with the flat surface. It would be useful to have a measure of comparativeness of the reliability indices with respect to the implied reliability or probability content (Melchers, 1987; Casciati & Faravelli, 1991). For this purpose, Ditlevsen (1979a) introduced a reliability index which is known as the generalised reliability index. However, the lack of comparativeness is not critical for single failure mode probability analysis (Ditlevsen & Bjerager, 1986) and this is why the simple Hasofer and Lind reliability index is usually used for this purpose. Hence, details of the generalised reliability index and methods consistent with its definition (e.g. Der Kiureghian & Liu, 1986) are beyond the scope of this work.

The method in which linearization is performed around the design point is often called a First Order Reliability Method, FORM. Since the design point is not known a priori, and in most cases cannot be determined directly (except if Z is linear), finding the shortest distance, β, in U-space, subject to Z(U)=0 is strictly a minimisation problem (Rackwitz, 1976; Flint et al, 1981; Shinozuka, 1983; Casciati & Faravelli, 1991; Liu & Der Kiureghian, 1991). There are several ways in which a solution may be found (Melchers, 1987) such as by direct minimisation using a Lagrangian multiplier (Schittkowski, 1985; Burcharth, 1990), by a numerical approach (Melchers, 1987) or by an iterative procedure. In this study, the latter is used; other methods are beyond the scope of this research.

Several iteration schemes exist (Thoft-Christensen & Baker, 1982; CIRIA, 1984; Smith, 1986; CUR-TAW, 1990; Ahammed & Melchers, 1993). In the following, the simple scheme suggested, for example, in CUR-TAW (1990)

and Ahammed & Melchers (1993), is introduced. It is used in the Level II program, PARASODE, developed as part of this research (see Chapter 5):

1) Set the initial design values of the basic variables (e.g. $X_i^* = \mu_{Xi}$)

2) Compute Z and the partial derivatives $\partial Z / \partial X_i$ at the point X_i^*

3) Compute μ_Z and σ_Z

4) Compute $\beta = \mu_Z / \sigma_Z$

5) Compute α_i

6) Compute new X_i^*

7) Repeat steps 2 to 6 until convergence is achieved within specified limits

8) Check that $Z^* = 0$, within specified limits

9) Compute the probability of failure from $P_f = \Phi(-\beta)$

The way this iteration procedure is set-up allows the safety of a structure to be assessed, i.e. it allows a calculation of the probability of failure for a given design parameter (e.g. the crest level of a seawall). However, this iteration procedure, slightly adjusted, and Level II methods in general, may also be used for design i.e. a probability of failure is first specified and one design parameter is modified until the target reliability is achieved (see Chapter 5 and Appendices C and D for further details).

Note that the iteration procedure can fail in certain circumstances (Rackwitz, 1976; Madsen et al, 1986; Melchers, 1987; CUR-TAW, 1990; Super-Software, 1994). One case is for a highly non-linear function for which it is possible to alternate between successive approximation points i and i+1 (Fiessler, 1979). This difficulty can be overcome by starting the new iteration using a point between i and i+1 (see Section 5.2.2.5). A second breakdown case is when the trial initial design point, X^*, lies close to a stationary point which is not a minimum; this is because the iteration procedure can only search for local stationary points and cannot distinguish between maxima, minima or saddle points (Ditlevsen & Madsen, 1980). The problem can only be overcome by selecting different starting points (see Section 5.2.2.1) and common sense appraisal of results.

A useful by-product of FORM is its ability to quantify the sensitivity of the reliability index to inaccuracies in the value of X_i at the design point, i.e. to

determine the contribution to the spread of Z made by each random variable. Eq. (2.14) can be rewritten as:

$$\sum_{i=1}^{N} \left[\frac{\left(\frac{\partial Z}{\partial X_i}\right)^* \sigma_{X_i}}{\sigma_Z} \right]^2 = \sum_{i=1}^{N} \alpha_i^2 = 1 \qquad (2.18)$$

Thus, α_i^2 represents the contribution to σ_Z^2 due to $\sigma_{X_i}^2$. If α_i^2 is small, X_i might be modelled as a deterministic quantity equal to its mean value. Typically, the acceptable values of the probability of failure are very small (CIRIA, 1984; CUR-TAW, 1990). This fact makes the reliability evaluations quite sensitive to the choice of the distributions of some variables and, in particular, to the choice of the tails of the distributions (which, of course, are the most difficult parts to verify by data). Thus α_i^2 gives a powerful means of examining which variables are most important and which make a negligible contribution to the variance of Z. Knowing α_i^2, one might focus attention only on the most important variables (Ditlevsen & Bjerager, 1986).

If basic variables are not independent, three possible courses of action are open (Joint Committee on Structural Safety, 1978; CIRIA, 1984):

- if the variables are strongly correlated (say, correlation coefficients greater than 0.8), the variables may be conservatively assumed to be exactly dependent, in which case the effective number of basic random variables is reduced;

- if the variables are weakly correlated (say, correlation coefficients less than 0.2), the variables may be assumed to be independent;

- for all other cases it is necessary to use a method which deals with correlated variables.

In these latter cases, correlation between variables can be dealt with using different methods, with different levels of performance and complexity. Which method should be used depends upon the problem under study, the accuracy required for the answer, and the data available.

To transform non-Normal correlated variables to independent Normal variables the Rosenblatt transformation (Rosenblatt, 1952) is usually

recommended (Ang & Tang, 1984; Madsen et al, 1986; Thoft-Christensen & Murotsu, 1986; Melchers, 1987; Casciati & Natale, 1992). However, there are other simpler approaches. For example, correlation can be accounted for by allowing the distribution of one random variable to be expressed as a function of another random variable (Van de Graaff, 1986; Burcharth, 1992; Townend, 1994a; Super-Software, 1994). Another alternative is outlined below. It transforms a set of correlated variables, X_i, into a set of non-correlated variables, Y_i, where Y_i are linear functions of X_i, i=1,...,N (Thoft-Christensen & Baker, 1982; Ang & Tang, 1984; CIRIA, 1984; Smith, 1986; Melchers, 1987; Burcharth, 1992):

- Compute the covariance matrix $C_{ov}[X]$ from

$$C_{ov}[X] = \begin{bmatrix} Var[X_1] & C_{ov}[X_1,X_2] & ... & C_{ov}[X_1,X_N] \\ C_{ov}[X_2,X_1] & Var[X_2] & & \vdots \\ \vdots & & \vdots & \vdots \\ C_{ov}[X_N,X_1] & & & Var[X_N] \end{bmatrix} \qquad (2.19)$$

- Compute the matrix of eigenvectors, V, and the vector of eigenvalues, D, of $C_{ov}[X]$

- Compute $\mu_Y = V^T \mu_X$ where V^T is the transpose matrix of V

- Compute $\sigma_Y^2 = V^T C_{ov}[X] V = D$

- Compute $Y = V^T X$

Note that $C_{ov}[Y]$ is a diagonal matrix as follows:

$$C_{ov}[Y] = \begin{bmatrix} Var[Y_1] & ... & ... & 0 \\ \vdots & Var[Y_2] & & \vdots \\ \vdots & & \vdots & \\ 0 & & & Var[Y_N] \end{bmatrix} = \begin{bmatrix} \sigma_{Y_1}^2 & & & 0 \\ \vdots & \sigma_{Y_2}^2 & & \vdots \\ \vdots & & \vdots & \\ 0 & & & \sigma_{Y_N}^2 \end{bmatrix} \qquad (2.20)$$

So, no correlation between any pair of random variables Y exists, as expected.

Non-Normal Random Variables

It is not always reasonable to consider random variables to be normally-distributed. For example, significant wave height, H_S, is likely to follow other distributions (e.g. Gumbel and Weibull) quite different from the Normal distribution, and it cannot be described solely by its mean value and standard deviation. For such cases of non-Normal variables, the Rosenblatt transformation (mentioned above) could be used. On the other hand, it is also still possible to use the reliability index concept but an extra transformation of the non-Normal basic variables into Normal basic variables must be performed. A commonly-used transformation (Thoft-Christensen & Baker, 1982; Ferry Borges & Castanheta, 1983; Ang & Tang, 1984; CIRIA, 1984; CUR-TAW, 1990) is that of Rackwitz & Fiessler (1978). It is based on substitution of the non-Normal distribution of the variable X_i by a Normal distribution in such a way that the original probability density and cumulative distribution functions (f_{X_i} and F_{X_i}, respectively) are equal to the corresponding values of the probability density and the cumulative distribution functions for a normally-distributed variable (φ and Φ, respectively) at the design point X^* (Figure 2.11):
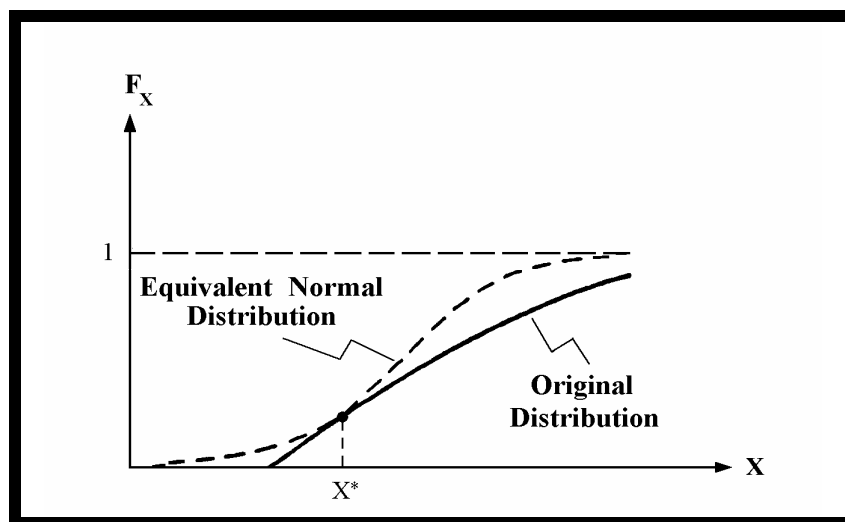
Figure 2.11: Rackwitz and Fiessler approximation for non-Normal variables (modified after Van der Meer, 1987).

$$F_{X_i}(X_i^*) = \Phi\left(\frac{X_i^* - \mu_{X_i}^N}{\sigma_{X_i}^N}\right)$$

$$f_{X_i}(X_i^*) = \frac{1}{\sigma_{X_i}^N}\ \varphi\left(\frac{X_i^* - \mu_{X_i}^N}{\sigma_{X_i}^N}\right)$$

(2.21)

where $\mu_{X_i}^N$ and $\sigma_{X_i}^N$ are the mean and standard deviation of the approximate (fitted) Normal distribution. Solving the above equations for $\mu_{X_i}^N$ and $\sigma_{X_i}^N$:

$$\mu_{X_i}^N = X_i^* - \Phi^{-1}[F_{X_i}(X_i^*)]\sigma_{X_i}^N$$

$$\sigma_{X_i}^N = \frac{\varphi[\Phi^{-1}[F_{X_i}(X_i^*)]]}{f_{X_i}(X_i^*)}$$

(2.22)

The iterative method presented earlier can still be used if the values of $\mu_{X_i}^N$ and $\sigma_{X_i}^N$ are calculated during each iterative loop. Eq. (2.16) for calculation of the design point is applied using the values of $\mu_{X_i}^N$ and $\sigma_{X_i}^N$. In this case of non-Normal variables, the point determined by iteration does not correspond exactly to the point of maximum probability of failure density (see CIRIA, 1984, for further details). In general terms, the point may be considered to be a close approximation to the set of values of the basic variables most likely to cause failure.

2.3.3.3 Combinations Of Actions

In nature, many actions (or loads) vary with time[1]. If a structure is subject to only one significant time-varying action, it is necessary to consider only the distribution of the maximum action during the anticipated life of the structure or the reference period, T_{ref}, for which the risk of failure is being assessed. However, if the structure is subjected to the effects of more than one time-varying action (e.g. waves and surges), then it is extremely unlikely that all of the actions will reach their peak lifetime values at the same moment (Turkstra, 1970; Der Kiureghian, 1980; Thoft-Christensen & Baker, 1982; CIRIA, 1984; Smith, 1986). Some benefit can be gained, in terms of reduced structural capacity, if this fact is taken into account, i.e. a structure can be

---

[1] Note that resistance also changes with time (e.g. Vasco Costa, 1987) and it can also be dealt with in a probabilistic manner (e.g. Nielsen & Burcharth, 1983). This aspect is beyond the scope of the present research.

designed for a total action less than the sum of the peak actions. This fact has long been recognised (Turkstra, 1970; Ferry Borges & Castanheta, 1971, 1983).

Although complex stochastic models may be used with Level III methods (CIRIA, 1984), some simplifications are required at Level II. A popular model for treating combinations of actions at Level II is that due to Ferry Borges & Castanheta (1971, 1983), which is a development of an earlier proposal by Turkstra (1970). Several meetings have been held between Mr. Castanheta, a Research Engineer at the National Laboratory of Civil Engineering (LNEC), Lisbon, Portugal, and the author. They have stimulated very useful discussions about the implementation of combinations of actions at Level II.

Ferry Borges and Castanheta's model has generally been very well accepted in the field (Turkstra & Madsen, 1980; Ditlevsen & Madsen, 1981; Bjerager & Skov, 1982; Thoft-Christensen & Baker, 1982; Melchers, 1987). Applications of the method are given in Ferry Borges & Castanheta (1971), CIRIA (1984) and Allsop & Meadowcroft (1995). It is described here and subsequently applied in the development of PARASODE. Note that some approximations in its basic concepts have been proposed by authors such as Paloheimo (1975), Ditlevsen (1976) and Rackwitz & Fiessler (1978), and other models have also been developed (Der Kiureghian, 1980; Madsen & Tvedt, 1990).

In the combinations of actions with which engineers are normally concerned, some actions change in intensity very much more rapidly than others (see Figure 2.12). If one action reaches its extreme value at some time during the design life, the combination of this action with the simultaneous values of the other actions may give the worst loading case (Turkstra, 1970).
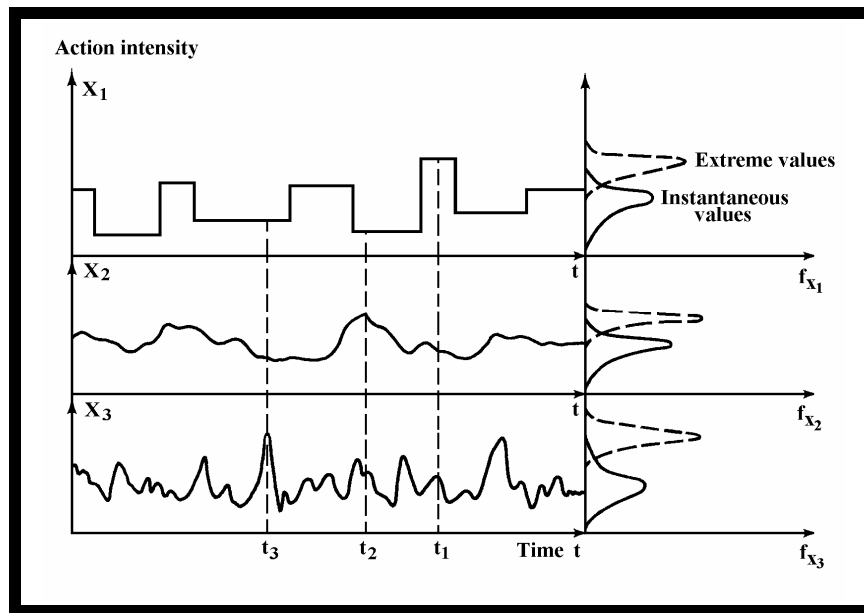
Figure 2.12: Representation of three time-varying actions (modified after CIRIA, 1984).

In the Ferry Borges and Castanheta model, it is first assumed that for each time-varying action, X_i, the design life or reference period, T_{ref}, is sub-divided into a number, r_i, of elementary time intervals of equal length, τ_i, such that $r_i = T_{ref} / \tau_i$ and $\tau_i = \ell_i \tau_{i+1}$ in which ℓ_i is an integer. The three following requirements should also be satisfied (CIRIA, 1984):

- for each interval, the occurrence or non-occurrence of the action corresponds to repeated independent trials with a probability p_i of occurrence;

- for the duration of the time interval, the load X_i remains at a constant intensity (or zero);

- the intensities of the action in different time intervals are independent.

This model is represented in Figure 2.13 for three time-varying actions.
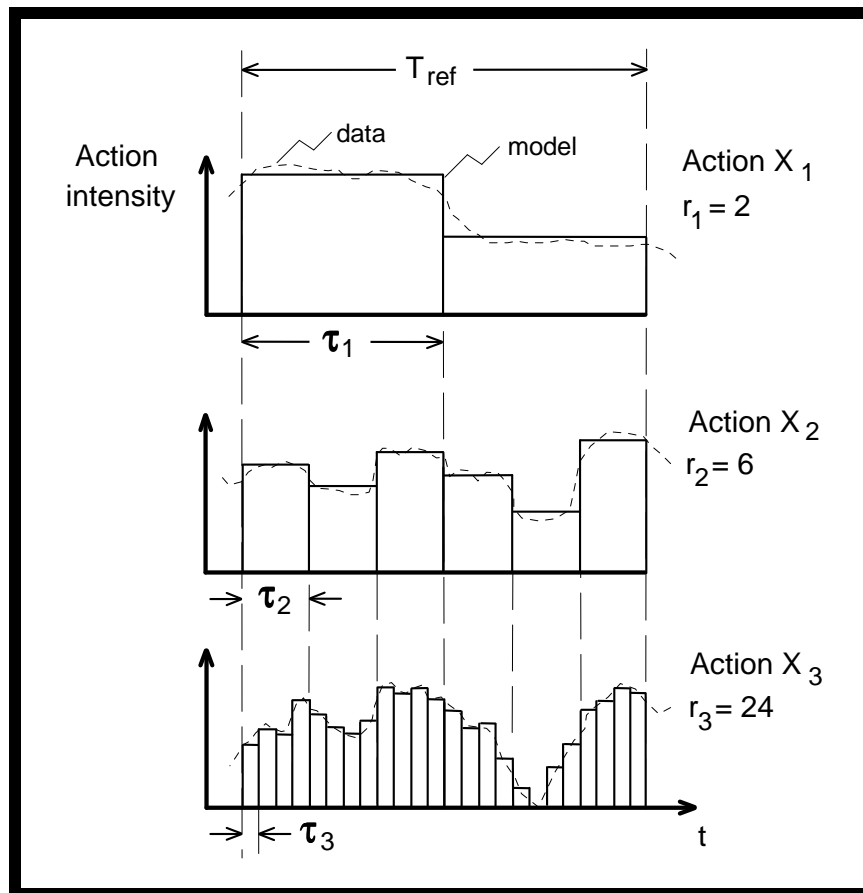
Figure 2.13: Idealised model of three time-varying actions
(modified after CIRIA, 1984).

For each time-varying action, X_i, it is then necessary to define the cumulative distribution function of the intensity of the action, F_{X_i}, corresponding to the basic time interval, τ_i. The cumulative distribution function of the maximum intensity of action X_i within the reference period, F_{X_i,r_i}, may then, in most cases, be approximated by (CIRIA, 1984):

$$F_{X_i,r_i} \approx F_{X_i}^{m_i} \tag{2.23}$$

where $m_i=p_i r_i$. For any other shorter period $t<T_{ref}$, corresponding to, say, s_i intervals of duration τ_i, the cumulative distribution function of the maximum intensity becomes:

$$F_{X_i,s_i} \approx F_{X_i}^{p_i s_i} \tag{2.24}$$

The model then requires the ranking of the actions in increasing order of the individual number of possible repetitions r_i during the reference period such that $r_1 \leq r_2 \leq ... \leq r_k$. As Ferry Borges & Castanheta (1983) demonstrate, an approximation to the maximum combined action may then be obtained by considering 2^{k-1} combinations of actions. For example, for k=3 time-varying actions:

$$
\begin{array}{llll}
\text{Comb. 1} & F_{X_1}^{r_1} & F_{X_2}^{r_2/r_1} & F_{X_3}^{r_3/r_2} \\
\text{Comb. 2} & F_{X_1} & F_{X_2}^{r_2} & F_{X_3}^{r_3/r_2} \\
\text{Comb. 3} & F_{X_1} & F_{X_2} & F_{X_3}^{r_3} \\
\text{Comb. 4} & F_{X_1}^{r_1} & F_{X_2} & F_{X_3}^{r_3/r_1}
\end{array}
$$

In practice, and consequently in certain literature (Ferry Borges & Castanheta, 1974; CIRIA, 1984; Allsop & Meadowcroft, 1995), some of the 2^{k-1} combinations of actions are neglected and only k combinations are considered. In this case (i.e. when the number of combinations taken into account is equal to the number of time-varying actions), the combination rule can easily be generalised as follows: when k time-varying actions are combined in k combinations of actions, each combination includes only one extreme distribution in the design life $F_{X_i}^{r_i}$; actions having a number of repetitions less than r_i are idealised by their basic distributions F_{X_i} ; actions having a number of repetitions exceeding r_i are idealised by the reduced distributions $F_{X_i}^{r_i/r_{i-1}}$ (see Table 2.1).

Referring to the example of the three time-varying actions shown in Figure 2.13, application of the combination rule (or Table 2.1) would require consideration of the three first combinations listed above, i.e.:

- combination of the distribution of the maximum value of the action X_1 during the reference period T_{ref} ($NR_{11}=2$; NR_{ji} is the power to which each basic distribution, F_{X_i} , should be raised, for each time-varying action X_i, i=1,...,k, and for each combination j, j=1,...,k) with the distribution of the maximum value of X_2 during a period τ_1 ($NR_{12}=6/2=3$), and with the distribution of the maximum value of X_3 during a period τ_2 ($NR_{13}=24/6=4$);

- combination of the distribution of the action X_1 based on its elementary time interval τ_1 ($NR_{21}=1$) with the distribution of the maximum value of X_2 during the reference period T_{ref} ($NR_{22}=6$), and with the distribution of the maximum value of X_3 during a period τ_2 ($NR_{23}=24/6=4$);

- • combination of the distribution of the action X_1 based on its elementary time interval τ_1 ($NR_{31}=1$) with the distribution of the action X_2 based on its elementary time interval τ_2 ($NR_{32}=1$), and with the distribution of the maximum value of X_3 during the reference period T_{ref} ($NR_{33}=24$).

Combination j	Action, X_i	X_1	X_2	X_3	...	X_k
	Number of time intervals in T_{ref}, r_i	r_1	r_2	r_3	...	r_k
1	Power to which	$NR_{11}=r_1$	$NR_{12}=r_2/r_1$	$NR_{13}=r_3/r_2$	...	$NR_{1k}=r_k/r_{k-1}$
2	each basic	$NR_{21}=1$	$NR_{22}=r_2$	$NR_{23}=r_3/r_2$	...	$NR_{2k}=r_k/r_{k-1}$
3	distribution, F_{X_i} ,	$NR_{31}=1$	$NR_{32}=1$	$NR_{33}=r_3$	...	$NR_{3k}=r_k/r_{k-1}$
⋮	should be raised,	⋮	⋮	⋮	⋮	⋮
k	NR_{ji}	$NR_{k1}=1$	$NR_{k2}=1$	$NR_{k3}=1$	...	$NR_{kk}=r_k$

Table 2.1: Values of NR_{ji} (modified after Rackwitz, 1976).

The distributions used in the combinations are referred to here as the modified distributions (note that, depending on the combination considered, these modified distributions may include basic, extreme and reduced distributions). To apply this model, either the modified distributions are known or they can be obtained if the basic distributions are available together with the number of repetitions of each action in the reference period. In the latter case, care should be taken in determining the number of repetitions to be adopted. As an example, Ferry Borges & Castanheta (1971, 1983) have shown that, for a reference period of fifty years, probability distributions of mean wind velocities with respect to elementary time intervals of one hour are transformed into distributions of yearly maxima by considering a fictitious number of repetitions of r=50x1000=50000, considerably less than the total number of elementary intervals r=24x365x50=438000. In this case, the difference is due to the strong correlation between the successive hourly mean velocities, i.e. hourly mean values are not independent. When there are insufficient observations to determine the number of repetitions, the values adopted should be based on experience (Joint Committee on Structural Safety, 1978).

Finally, to determine the probability of failure of a structure associated with a specific failure mode and subjected to more than one time-varying action, the

Level II methods described in Section 2.3.3.2 are applied for each of the k combinations using the appropriate modified distributions of the time-varying actions. This yields k values of the reliability index, β, which may then be combined to estimate the total failure probability using the relationship:

$$P_f \approx \sum_{j=1}^{k} \Phi(-\beta_j) \tag{2.25}$$

2.3.4 Level I Methods

As noted earlier, each failure mode must be described by a formula. Level I methods involve introducing partial coefficients (or partial safety factors) into the formula. The coefficients are chosen to give an acceptable margin between the characteristic resistance and the design load. By this means, the probability of failure is kept to a suitably low level. Safety factors were traditionally selected largely on the basis of intuition and experience (Committee of the Institution of Structural Engineers, 1955). However, the availability of Level II probability methods has made it possible to relate probabilistic measurements of safety, such as P_f or β, to the partial coefficients in Level I methods. Hence, the safety factors can now be determined on a more scientific and rational basis (Yen, 1989).

The partial coefficients, γ, to be applied to a failure mode formula are usually larger than or equal to one. Consequently, if one splits the formula into either load variables $X_{S,i}$ or resistance variables $X_{R,i}$ then the related partial coefficients should be applied as follows to obtain the corresponding design values, $X_{S,i}^*$ and $X_{R,i}^*$:

$$X_{R,i}^* = \frac{X_{R,i_{ch}}}{\gamma_{X_R,i}}$$
$$X_{S,i}^* = \gamma_{X_S,i} \cdot X_{S,i_{ch}} \tag{2.26}$$

$X_{R,i_{ch}}$ and $X_{S,i_{ch}}$ are characteristic values of the resistance and load variables, respectively. These equations represent the key to the relationship between Level I and Level II methods: the so-called design point (Melchers, 1987; Van der Meer, 1987).

The term characteristic value was introduced in the late 1950s at the time when probabilistic concepts were first being introduced into structural codes of practice, and when it was recognised that few basic variables have clearly-defined upper and lower limits that can sensibly be used in design. For example, in civil engineering codes of practice, the characteristic values of resistance parameters, $X_{R,i_{ch}}$, might be chosen as values below which not more than 5% of test results may be expected to lie (the 5% fractile). Similarly, characteristic values of loads, $X_{S,i_{ch}}$, might be defined as the loads with a 5% probability of being exceeded (the 95% fractile) during the lifetime of the structure. Alternatively, they may be chosen to be the mean values. Other definitions may also be used (Madsen et al, 1986). Note, however, that the values of the partial coefficients are uniquely related to the chosen definitions of the characteristic values.

The magnitude of γ_i reflects both the uncertainty in evaluating the related parameter X_i and the relative importance of X_i in the failure function (Burcharth, 1990). It is to be stressed that the magnitude of γ_i is not, in a mathematical sense, a rigorous measure of the sensitivity of the failure probability to the parameter X_i (PIANC, 1992).

When the partial coefficients are applied to the characteristic values of the parameters in a failure function, a design equation is developed. For the basic case of one resistance variable, X_R, and one load variable, X_S, the minimum requirement applied to a structure at Level I is that the following condition is satisfied:

$$Z = \frac{X_{R_{ch}}}{\gamma_{X_R}} - \gamma_{X_S} \; X_{S_{ch}} \; > \; 0 \tag{2.27}$$

γ_{X_R} and γ_{X_S} are partial safety factors of the resistance and load, respectively. Assuming a Normal distribution for X_R and X_S, as adopted in the Level II analysis, the characteristic values are defined as:

$$\begin{aligned} X_{R_{ch}} &= \mu_{X_R} - K_{X_R}\sigma_{X_R} \\ X_{S_{ch}} &= \mu_{X_S} + K_{X_S}\sigma_{X_S} \end{aligned} \tag{2.28}$$

where K_{X_R} and K_{X_S} are coefficients defining the fractile which corresponds to the characteristic value (Madsen et al, 1986). This is illustrated in Figure 2.14 for the special case where $K_{X_R} = K_{X_S} = 1.96$.
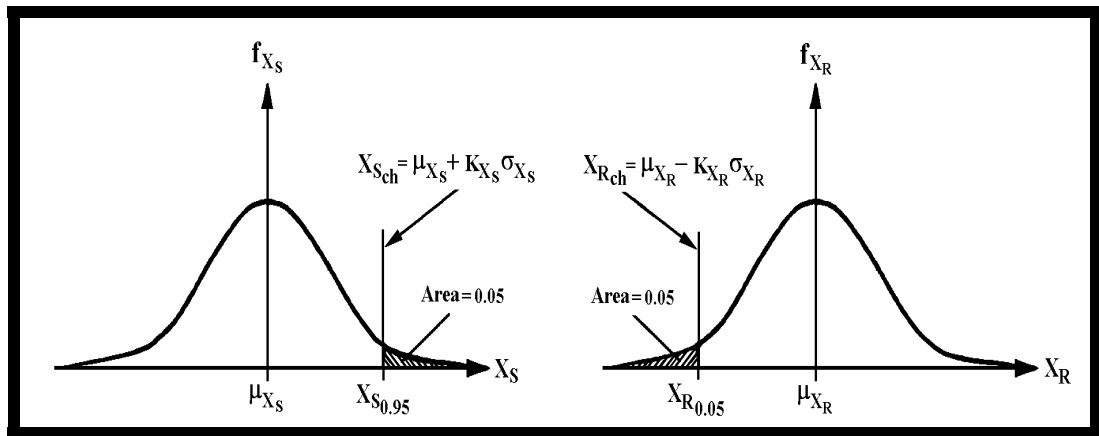
Figure 2.14: Definition of characteristic resistance, $X_{R_{ch}}$, and characteristic load, $X_{S_{ch}}$, when X_R and X_S are both normally distributed (modified after Melchers, 1987).

Considering the general multi-variate basic variable problem and comparing eqs. (2.28) with eq. (2.16) for the design point, X_i^*, a relationship between the partial coefficients in eqs. (2.26), the reliability index, β, and the sensitivity factors, α_i, can be obtained (Ferry Borges & Castanheta, 1983; PIANC, 1992):

$$\gamma_{X_R,i} = \frac{1 - K_{X_R,i}\dfrac{\sigma_{X_R,i}}{\mu_{X_R,i}}}{1 + \alpha_i\beta\dfrac{\sigma_{X_R,i}}{\mu_{X_R,i}}} \qquad \gamma_{X_S,i} = \frac{1 + \alpha_i\beta\dfrac{\sigma_{X_S,i}}{\mu_{X_S,i}}}{1 + K_{X_S,i}\dfrac{\sigma_{X_S,i}}{\mu_{X_S,i}}} \qquad (2.29)$$

The partial coefficients can be related either to each parameter or to combinations of the parameters (as overall coefficients). They can also be tuned to ensure equal contributions from the various failure modes to the failure probability of the structure (Burcharth, 1990). Clearly, it is desirable to have a system which is as simple as possible, i.e. with as few partial coefficients as possible, but without invalidating the accuracy of the design equation beyond acceptable limits. Fortunately, it is very often possible to use overall coefficients without losing significant accuracy within the realistic

range of combinations of parameter values. This is the case for the system of partial safety factors developed for rubble-mound breakwaters proposed by PIANC (1992) where only two partial coefficients are used in each design formula. For full details, the reader should consult references such as Burcharth (1991a) and PIANC (1992).

The accuracy of the Level I methods depends on several factors. The most important have been summarised by Allsop & Meadowcroft (1995) as follows:

- the range of applicability of the design procedure (the wider the area of application the less accurate are the safety factors);

- the complexity of the partial safety factor system (a system with many safety factors will easily match the target reliability over the area of application, but will be more complex to develop and apply);

- the underlying probabilistic methods (the reliability of sample designs is assessed using Level III or Level II methods; hence, the partial factors developed depend on the accuracy of those methods);

- the underlying failure functions (the reliability with which the failure functions represent the failure modes affects the accuracy of the safety factors);

- the source data (if the data used are not very reliable, the safety factors cannot be relied upon either).

Level I safety-checking methods are the basis for codes of practice, although the safety factors used are assessed by Level II and/or Level III methods (CIRIA, 1984). Codes of practice exist for offshore structures (see, for example, Fjeld, 1977) and nearly all types of land-based structures (Burcharth, 1987). The question of whether coastal structures need such codes of practice is often raised (CUR-TAW, 1990) but has not yet been fully answered.

2.3.5 Comparison Of The Methods

The last few sections illustrate the features of different probabilistic methods. All of them have their advantages and disadvantages (Yen, 1989). There are

various reasons why one method might be used instead of another. The issues are discussed below.

In the numerical integration method, the calculation of an N-fold integral may be extremely time-consuming and it usually requires a considerable computational effort. Even with modern computer facilities, an enormous number of calculations is involved if the number of variables exceeds 5 or 6 and if the failure function is a complex one (Hohenbichler et al, 1987; Bjerager, 1991).

Traditional sampling is an acceptable alternative when dealing with simple failure functions and failure probabilities which are not very low. However, it suffers from the fact that if an "accurate" answer is desired for extreme conditions associated with relatively low probabilities of failure (approximately $10^{-4} < P_f < 10^{-8}$, according to Bjerager, 1991), many simulations are required. This is a drawback that recent methods, like Latin Hypercube sampling, may address to some extent by reducing the required number of simulations. In other cases, difficulties can be overcome by using Level II methods like FORM. The main practical advantages of this approach are that it is less time-consuming than Level III methods, the computational effort is independent of the probability level, it provides a rational basis for evaluating partial safety factors and it also provides an automatic procedure for determining the sensitivity of the computed failure probability to each of the basic design variables. This latter characteristic allows the designer to focus his attention on the parameters which are of greatest significance and shows where effort to reduce uncertainty should be concentrated. Due to their simplicity, these methods have become very popular, particularly in calibration work for codes of practice (Melchers, 1987). Note that unlike Level III methods which can be used only for reliability analysis (safety checking), Level II procedures may be used also for design (i.e. design for a specified reliability level). However, these procedures also have their limitations. Amongst others, the main reason for discrepancy between a Level II and a Level III method is that the failure function is usually non-linear. The stronger the non-linearity, the greater is the chance that the Level II results will differ considerably from the "exact" answer. However, the FORM results can be improved through a second or higher order approximation (Ang & Tang, 1984; Madsen et al, 1986; Jang et al, 1994), as mentioned briefly in Section 2.3.1. But computational complications are

increased considerably. Therefore, at present, these methods are seldom used. It is more common to use the Level III methods, especially simulation, to validate the Level II results. Although a FORM method can provide an answer to a problem, it is never known how accurate the answer is unless a check is done using numerical integration or simulation techniques. Nevertheless, the FORM method is one of the most important tools in probabilistic design because one can rarely afford to make a million Level III calculations during preliminary design.

Besides the calculations at Level III and Level II, there are those at Level I. Level I calculations are particularly suitable for everyday design (where a large body of previous experience of similar systems is available), although the determination of the partial coefficients must be based upon higher level results. Level I calculations are the basis of codes of practice.

If probabilistic methods are used with foresight and understanding, they are powerful and can provide reliable results. For example, comparison of design alternatives using these methods is a promising way in which to apply them (Dover & Bea, 1979; Townend & Fleming, 1991; Melchers, 1993). One can use these methods to decide what is the difference in probability of failure of a structure designed with strategy A compared with one designed with strategy B and what are the associated projected consequences. Meaningful decisions can be based on such comparisons. However, application of probabilistic methods leads to the question of whether or not the calculated probability of failure corresponds to reality. It is often argued that a probabilistic analysis is meaningful only if there is a complete understanding of the physics and if the analysis is based on accurate computational models and on sufficient statistical data (Burcharth, 1983, 1985; CUR-TAW, 1990). However, in practice, these requirements are seldom fulfilled. Analysis has to address an idealised system founded upon assumptions, simplifications and the collective judgement of a number of individuals, so that the real system becomes manageable. As CIRIA (1984) emphasise, probabilistic methods should be viewed as an aid to the application of humanity, experience and judgement, not as their replacement. In this connection, the calculated probability of failure is related to the idealised system (and not directly to the real system) and should be interpreted as a measure of the confidence in a particular design. It is a notional probability of failure, instead of the frequency of cases of failure (Veneziano, 1976; CIRIA, 1984; Van der Meer,

1987, CUR-TAW, 1990). In other words, the numerical and graphical results produced by the probabilistic methods should be taken as illustrating orders of magnitude and trends, rather than describing reality (Fjeld, 1977; Burcharth, 1985). Hence, the purpose of a reliability analysis is not so much to calculate the exact failure frequency as to produce as good and balanced a design as possible with the available information. In fact, the engineer is expected to develop a model of the phenomenon under study which embodies its salient features and which can be used to make optimal decisions using the data available (Turkstra, 1970; Burcharth, 1985; Ditlevsen & Bjerager, 1986; Melchers, 1993). He is not expected to produce a perfect image of reality which is an impossible task (Lamberti, 1992). In this pragmatic sense, even simplistic models, approximate probabilities, notional reliabilities, etc., may be satisfactory (Veneziano, 1976). It is also worthwhile noting that the less one knows, the more important it is to try to assess the reliability of a structure (Burcharth, 1983, 1990). The probabilistic approach is the only one which gives information on the risk of failure with due consideration to the uncertainties of the various parameters involved. As Harlow (1985) stated:

> *The art and science of civil engineering deals with applying the materials of nature to the use of mankind, but it does not presuppose complete understanding of all facts. Civil engineers have always worked with incomplete information and probably they always will.*

The fact that there is neither a complete understanding of the physics nor accurate computational models and sufficient statistical data to make the best use of probabilistic methods, does not mean that they should be discarded (Yen, 1989). On the contrary, effort should be devoted to describing the physical processes and establishing the appropriate models and data sets required for their full implementation. Thus, much more work is required if complete and objective risk assessments are to be reached. For instance, there is a great need for detailed monitoring of existing coastal structures, including recording the wave conditions to which they are subjected (Ouellet, 1974; MAFF, 1993c). Furthermore, it is important to incorporate as much experience as possible from failures. Although unacceptable and costly, failures test the limits of our knowledge and, in some way, are the price of progress (Eberhardt, 1979; Sorensen & Jensen, 1985). Failures are of such great importance to the engineering community that their details should be widely published. For example, a team from HR Wallingford/Sir William Halcrow & Partners Ltd created, as part of

research carried out for the National Rivers Authority, a database of flooding instances, including information such as details of the location and type of structure, a description of the failure mode, and an assessment of consequences (Meadowcroft et al, 1996). This sort of initiative should be encouraged.

2.4 Closing Remarks

Risk analysis provides a powerful framework for the design of coastal structures, accounting for the probability and consequences of failure as well as coping, to some degree, with variability and uncertainty. However, when assessing structural safety using probabilistic methods, it must be stressed that the process involves knowledge about the individual structure. Therefore, confidence in the calculated value of the probability of failure must change with the amount and quality of the information used for its calculation. With this philosophy in mind, risk analysis may be seen simply as a design tool based on scientific methods which can facilitate good engineering decisions, but not a process which will necessarily provide a precise assessment of safety.

In recent years, much has been learned by coastal engineers about risk analysis, but progress in formulating methods and gaining confidence in new design procedures is inevitably slow. At present, there is insufficient knowledge about coastal structures to enable a probability analysis of failure mode systems to be carried out in full. However, instead of abandoning this "new" approach to design, efforts should be made to better identify the specific physical processes with which coastal engineers must deal, to better communicate their data requirements to researchers, to subsequently collect the required data sets, and to establish the appropriate models necessary for the complete implementation of the methods.

Some existing studies have aimed at being very broad, covering failure modes, consequences and costs. Others have focused on particular failure modes. This demonstrates the difficulty in achieving a satisfactory compromise between theory and practice, and developing an approach applicable to a wide range of coastal structures.

Diagrams like event trees, fault trees and cause-consequence charts have been presented for some coastal structures. However, such techniques have still almost always served essentially as schematic representations or research tools rather than as strict logical analyses of failure. Information on failures tend to concentrate on the consequences rather than on the causes of failure.

Assessment of the safety of coastal structures depends fundamentally on assessment of individual failure modes. The single failure mode probabilistic methods relevant to this research have been presented in Section 2.3. All of them have their advantages and disadvantages (see Section 2.3.5). The important issue is to be aware of the characteristics of the methods, their applicability and their limitations, otherwise wrong conclusions can be drawn, incorrect decisions can be made and unsound action may be taken.

3   WAVE OVERTOPPING OF SEAWALLS

3.1   Introduction

Seawalls are expensive, and fixing a seawall freeboard at too large a value has both a financial penalty and is unnecessarily damaging to the natural environment owing to the increased impact of the structure on its surroundings. On the other hand, if the crest of a seawall is set too low, then there are problems with structural safety and potential social problems associated with flooding from wave overtopping. Hence, it is important to strike the correct balance between satisfying the structural and functional requirements of the project, avoiding unnecessary expense, and having undesirable impacts on the surrounding environment.

Recent damage caused by excessive wave overtopping (Jensen, 1984), concerns over global warming, the allied rise in mean-sea-level and increased storminess have all drawn attention to the fitness of existing coastal structures. There is a need to assess their capability in withstanding both higher water levels and increased wave activity. Effective evaluation depends upon having an adequate theoretical framework for predicting wave overtopping and suitable data for validating the theory, for evaluating the associated empirical coefficients and for undertaking risk analyses.

Overtopping may occur for relatively few waves under the design event, and low overtopping rates may often be accepted without severe consequences for the seawall or the area protected by it. On the other hand, some structures are designed to have quite severe overtopping under design conditions. Other structures, such as breakwaters, may be so low that they are overtopped daily. The acceptable overtopping discharge depends upon the activities normally performed in the lee of the structure, the need to prevent erosion of its rear face, and the socio-economic consequences of flooding.

Under random waves, the overtopping discharge varies greatly from wave to wave. There are very few data sets available to quantify this variation. It may be described by (Allsop, 1994):

- the percentage of waves passing over the crest, $N_{WO\%}$;

- the mean overtopping discharge per unit length of structure, Q ($m^3/s/m$);

- the peak volume in an individual wave, V_p.

Each of these responses depends on wave and structure parameters, including: the seawall freeboard, the crest geometry, the seaward slope, the significant wave height, and the mean or peak wave period, the angle of wave attack measured from the normal to the structure, the water depth at the toe of the seawall, and the seabed slope. In many cases, for example for determining the required drainage capacity behind seawalls and the depth of flooding in the hinterland, it is sufficient to use the mean discharge, Q (Jensen & Sorensen, 1979; Owen, 1982b; Jensen & Juhl, 1987; Kobayashi & Wurjanto, 1989; CIRIA/CUR, 1991; Van der Meer, 1993). The calculation of the mean overtopping discharge for a particular structure geometry, water level and wave condition is mainly based on empirical equations fitted to hydraulic model test results (Ward et al, 1994). A well-known data set applies to flat-topped embankments fronted in some cases by a flat berm (Hydraulics Research Station, 1980; Owen, 1982a). These tests were aimed at establishing the impact on overtopping discharge of the wave climate (including the angle of wave attack and the wave steepness), the seawall slope, the crest and berm elevations, and the berm width.

In the remainder of this chapter, a brief review of wave overtopping of seawalls is first given, including the models currently used for prediction and the permissible limits of overtopping. This brief review has the objectives of highlighting the deficiencies in current knowledge which are relevant to the work developed as part of this research, and to direct the reader towards sources of more detailed information. Then, a new model is presented: this is the H&R model. This model is conceived from theoretical considerations: the purpose has been to construct a model which accounts for the physical boundary conditions. The H&R model and the Owen model are used in a re-analysis of Owen's data for simple seawalls possessing uniform seaward slopes of 1:1, 1:2 and 1:4, subjected to random waves approaching normal to the slope. Owen's data were chosen for re-analysis because of their extensive nature and ready availability from HR Wallingford Ltd. There were two main difficulties in processing these data: first, the database contained a considerable number of errors; and, second, some information was missing.

The problems were overcome by consulting the original data sheets at HR Wallingford. The implications of the H&R model for seawall freeboards are also discussed and an illustrative example is given. Finally, the reliability of the two models is assessed by considering the randomness of possible model parameters. The software package BestFit is used to support the choice of probability distributions to describe the model parameters.

3.2   Literature Review

3.2.1   Historical Perspective

Until about 1980, the crest levels of UK seawalls were generally set in relation to an extreme wave run-up level (Allsop et al, 1985a; Allsop, 1986). In The Netherlands, the level for coastal embankments was an extreme water level with an allowance for the 2% run-up, $R_{2\%}$ (TACPI, 1974; CUR-TAW, 1990). This fact implied acceptance that about 2% of waves might overtop under the design event. Random wave run-up levels are described in the CIRIA/CUR (1991) manual. Methods from the USA and Japan predicted wave run-up levels based on regular waves only (U.S. Army Corps of Engineers, Coastal Engineering Research Center, 1984; Goda, 1985; Douglass, 1986).

Since the late 1970s, there have been major advances in the prediction of wave overtopping, particularly from laboratory work in the UK, The Netherlands and Italy (Allsop, 1994). These studies concentrated on the prediction of mean overtopping discharge over unit length of seawall. In the late 1970s, studies using random waves were analysed by Owen. He developed a design equation relating the mean overtopping discharge to incident wave conditions, described by the significant wave height and the mean zero-crossing wave period, and the seawall's freeboard (Hydraulics Research Station, 1980; Owen, 1982a). The results covered simple and bermed seawalls, but were then extended to include slopes with crown or recurved walls and rough or armoured slopes (Bradbury & Allsop, 1988; Bradbury et al, 1988; Aminti & Franco, 1988; Owen & Steele, 1991; Pedersen & Burcharth, 1992; Allsop & Franco, 1992; De Waal & Van der Meer, 1992; Besley et al, 1993; Herbert et al, 1994; Monso et al, 1996). Methods to predict overtopping of vertical seawalls were derived by

Goda (1971), Goda et al (1975) and Goda (1985), and were extended by random wave tests in the UK and The Netherlands (Herbert, 1993; De Waal, 1993; Herbert et al, 1994).

Recent studies in Italy, The Netherlands and the UK have quantified not only the mean overtopping discharge but also the distribution of overtopping volumes per wave and, particularly, the maximum individual overtopping volume (Smith et al, 1994; Franco et al, 1994; Van der Meer & Janssen, 1995). A relationship may then be used to relate overtopping volume and mean overtopping discharge, but this relationship varies with the structure geometry and wave conditions.

3.2.2   Prediction Methods

In general, the mean overtopping discharge per unit length of seawall, Q, depends upon the wave motion, the seawall profile, the foreshore characteristics, and the water properties:

$$Q = f(H_s, T_m, \beta, R_c, \alpha, d_s, g, ...)$$

(3.1)

H_s is the significant height of the incident waves; T_m is the mean zero-crossing wave period; β is the angle of wave approach measured from the normal to the seawall; R_c is the seawall's freeboard (the height of the crest of the structure above the still-water-level); α is the angle of the seawall front slope measured from the horizontal; d_s is the still-water-depth at the toe of the structure; and g is the acceleration due to gravity.
Figure 3.1 shows this notation. In the figure, T_p is the wave period corresponding to the peak spectral density, CL denotes crest level, TL denotes toe level and SWL is the still-water-level above datum.
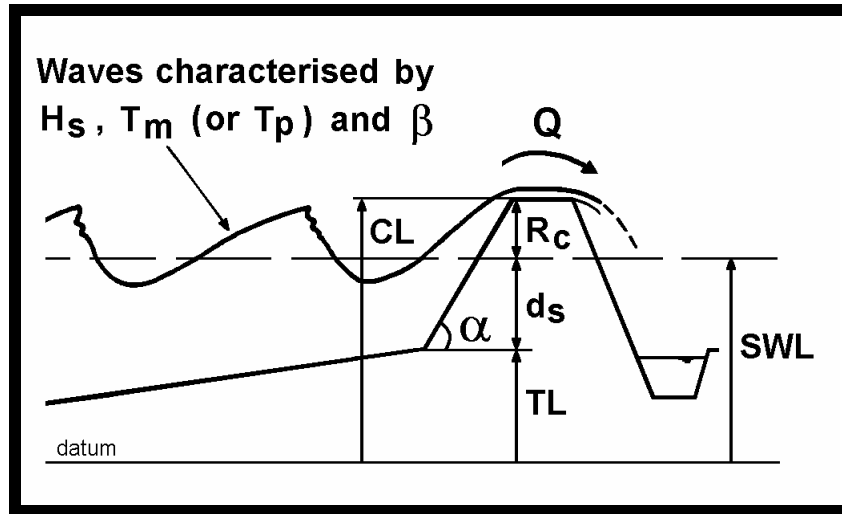
Figure 3.1: Notation for seawall overtopping.

Eq. (3.1) may be rewritten in the form of dimensionless groups:

$$\frac{Q}{\sqrt{gH_s^3}} = f\left(\frac{R_c}{H_s}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, ...\right) \tag{3.2}$$

or

$$\frac{Q}{g^2 T_m^3} = f\left(\frac{R_c}{H_s}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, ...\right) \tag{3.3}$$

H_s/gT_m^2 is a measure of the incident wave steepness. Owen combined this group both with $Q/\sqrt{gH_s^3}$ (or $Q/g^2 T_m^3$) and with R_c/H_s, to write:

$$\frac{Q}{T_m gH_s} = f\left(\frac{R_c}{T_m\sqrt{gH_s}}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, ...\right) \tag{3.4}$$

He then suggested that:

$$\frac{Q}{T_m gH_s} = A\exp\left(-B\frac{R_c}{T_m\sqrt{gH_s}}\right) \tag{3.5}$$

A and B are best-fit coefficients determined from experimental data. However, other arrangements are possible (Aminti & Franco, 1988; Ahrens &

Bender, 1991), including use of the wave period of peak spectral density, T_p, rather than the mean zero-crossing period, T_m. Table 3.1 summarises some of the options for dimensionless discharge, Q_*, and dimensionless freeboard, R_*, used in studies of the overtopping of seawalls and other structures.

Authors	Dimensionless Discharge, Q_*	Dimensionless Freeboard, R_*	Overtopping Model
Hydraulics Research Station (1980); Owen (1982a)	$\dfrac{Q}{T_m g H_s} = Q\sqrt{\dfrac{s_m}{2\pi g H_s^3}}$	$\dfrac{R_c}{T_m\sqrt{gH_s}} = \dfrac{R_c}{H_s}\sqrt{\dfrac{s_m}{2\pi}}$	$Q_* = A\exp(-BR_*)$
Bradbury & Allsop (1988)	$\dfrac{Q}{T_m g H_s}$	$\dfrac{R_c^2}{T_m\sqrt{gH_s^3}}$	$Q_* = A(R_*)^{-B}$
Ahrens & Heimbaugh (1988)	$\dfrac{Q}{\sqrt{gH_s^3}}$	$\dfrac{R_c}{(H_s^2 L_p)^{1/3}}$	$Q_* = A\exp(-BR_*)$
Sawaragi et al (1988)	$\dfrac{Q}{\sqrt{gL_s H_s^2}}$	$\dfrac{R_c}{H_s}$	-----------------
Aminti & Franco (1988)	$\dfrac{Q}{T_m g H_s}$	$\dfrac{R_c}{H_s}$	$Q_* = A(R_*)^{-B}$
Ward (1992)	$\dfrac{Q}{\sqrt{gH_s^3}}$	$\dfrac{R_c}{(H_s^2 L_{op})^{1/3}}$	$Q_* = A\exp(-BR_*)\exp(C\cot\alpha)$
Pedersen & Burcharth (1992)	$\dfrac{Q T_m}{L_m^2}$	$\dfrac{R_c}{H_s}$	$Q_* = A(R_*)^{-B}$
De Waal & Van der Meer (1992)	$\dfrac{Q}{\sqrt{gH_s^3}}$	$\dfrac{R_c - R_{2\%}}{H_s}$	$Q_* = A\exp(-BR_*)$
Van der Meer (1993); Smith et al (1994); Van der Meer & Janssen (1995)	$\dfrac{Q}{\sqrt{gH_s^3}}\dfrac{\sqrt{\tan\alpha}}{\xi_p}$ for $\xi_p < 2$ $\dfrac{Q}{\sqrt{gH_s^3}}$ for $\xi_p > 2$	$\dfrac{R_c}{H_s}\dfrac{1}{\xi_p}\dfrac{1}{\gamma}$ for $\xi_p < 2$ $\dfrac{R_c}{H_s}\dfrac{1}{\gamma}$ for $\xi_p > 2$	$Q_* = A\exp(-BR_*)$
Franco et al (1994)	$\dfrac{Q}{\sqrt{gH_s^3}}$	$\dfrac{R_c}{H_s}$	$Q_* = A\exp(-BR_*)$

Table 3.1: Some options for dimensionless discharge, dimensionless freeboard and overtopping model.

Note that H_s has been used to denote the significant wave height calculated either as the mean height of the highest one third of the waves in a record or estimated from the zeroth moment of the surface elevation spectrum (IAHR, 1989). L_s is the Airy wavelength calculated using the water depth at the toe of the structure and the significant wave period, T_s. L_m and L_p are the

corresponding wavelengths calculated using T_m and T_p. $s_m = 2\pi H_s / g T_m^2$ is the deep water wave steepness calculated using T_m. $R_{2\%}$ is the run-up exceeded by only 2% of the incident waves. γ is a reduction factor to account for influences of berms, roughness, shallow water and oblique wave attack on wave run-up and overtopping. ξ_p is the surf similarity parameter calculated using the wave period of peak spectral density ($\xi_p = \tan\alpha / \sqrt{H_s / L_{op}} = \tan\alpha / \sqrt{s_p}$ in which $L_{op} = g T_p^2 / 2\pi$ and $s_p = 2\pi H_s / g T_p^2$). C, like A and B, is a coefficient. Finally, note that R_* for De Waal & Van der Meer (1992) is not strictly a dimensionless freeboard but the dimensionless excess of the crest level above the 2% run-up level.

Table 3.1 shows that the two most common overtopping models relating the dimensionless groups are:

$$Q_* = A \, \exp(-B R_*) \qquad\qquad (3.6)$$

$$Q_* = A (R_*)^{-B} \qquad\qquad (3.7)$$

where, again, A and B are best-fit coefficients determined from the experimental data. Clearly, coefficients A and B must account for all influences on Q_* other than R_*. Note that the γ reduction factor used by some authors (Van der Meer, 1993; Van der Meer & Janssen, 1995) to account for influences of berms, roughness, shallow water and oblique wave attack on wave run-up and overtopping can similarly be incorporated in any other empirical model.

Dimensional analysis provides no means for determining which sets of dimensionless groups may be especially informative or helpful in dealing with a particular data set. A possible problem in using many of the pairings in Table 3.1 is the potential for spurious correlation. A spurious correlation may arise when dimensionless groups plotted against one another contain a common variable (Massey, 1971). There is nothing wrong with the presence of a common variable, but care must be taken in interpreting such plots. Scatter in the data may be suppressed simply by the presence of this variable.

3.2.3 Numerical And Physical Modelling

Much of the existing literature on overtopping reports the results of physical model tests. Note, in particular, that:

- It is not yet possible to predict wave overtopping from an entirely theoretical basis, even for regular waves.

- The empirical methods (Table 3.1) were derived by means of measurements in random wave physical models. Such models can be constructed in any well-equipped wave basin or flume. Test procedures used by the major laboratories are well-established, and overtopping measurements made using these procedures are generally comparable. However, measurements can be subject to scale effects and this fact has to be taken into consideration (Thomas & Hall, 1992).

- The empirical methods (Table 3.1) are valid only for the ranges of structure configurations and of wave conditions tested, but many practical situations fall outside of these ranges. Uncertainties in a simple empirical method may not be acceptable for particular design problems. In such instances, a short series of physical model tests may provide the most accurate and economic design tool (CIRIA/CUR, 1991; Allsop, 1994).

- The physical model results may be used to calibrate numerical models.

Numerical modelling of wave overtopping is still under development. The description of wave hydrodynamics at structures is very complex which makes the task very difficult (Allsop, 1994). However, knowing the wave hydrodynamics (such as the water velocity and depth of the water overtopping the crest of the structure) would enable, for example, a more sensible prediction of the severity of the damage caused by wave overtopping (Lording & Scott, 1971; Jensen, 1983; Kobayashi & Raichle, 1994). In addition, it would help in filling the gap between empirical formulae (valid only for the ranges of structure configurations tested) and the many practical situations which fall outside of the conditions tested (Kobayashi & Wurjanto, 1989). Two types of numerical models are relevant for overtopping (Van der Meer, 1994): i) the simplest one uses a depth-averaged wave formulation, running a bore-like wave up a (shallow) slope; ii) an alternative method uses the "Volume of Fluid" method to describe the full fluid motions in two or, possibly, three dimensions; this method can calculate fluid motions

beyond the point of wave breaking, but requires super-computer power for even relatively few waves.

Prototype measurements of wave overtopping quantities are scarce. To the author's knowledge, only two sets of measurements are available from literature. Fukuda et al (1974) made direct observations of the rate of wave overtopping at the wave absorbing revetment of Niigata East Port, facing the Sea of Japan, in the winter of 1971 and 1972. The Danish Hydraulic Institute made field measurements on a small Danish breakwater in the Port of Hundested in 1977 (Jensen & Juhl, 1987).

3.2.4   Permissible Limits Of Overtopping

The definition of tolerable limits for overtopping is still an open question, given the high irregularity of the phenomenon and the difficulty of measuring it and its consequences (Franco et al, 1994). Remarkably little information is available on the effects of wave overtopping on the defence structure
itself, or on activities behind the structure (Allsop, 1994). Obviously the overtopping criteria for design depend upon the structure's function and degree of protection required, and upon the associated risks, taking into account the joint probability of wave heights and water levels. In fact, relatively large overtopping might be allowed during extreme storms if pedestrians and vehicle movements on the structure are prohibited.

Until very recently, the permissible limits on the mean overtopping discharge were set by three criteria (Allsop, 1994):

- storage volume available behind the structure for overtopping during high water;
- potential damage to the crest or rear slope of the structure;
- danger or inconvenience to people or vehicle traffic, or damage to buildings.

The limits set by storage volume are specific to the individual site; they are not amenable to general guidance. An analysis of wave overtopping of seawalls in Japan suggested limiting values of the mean overtopping discharge to ensure the safety of vehicles and people in the immediate

vicinity of the seawall, and to prevent damage to buildings in its lee and to the seawall itself. Based on the impressions of experts observing prototype overtopping, the values can be seen in Figure 3.2. They are included in design manuals/standards such as the CIRIA/CUR (1991) manual and BS 6349 (BSI, 1991b). A further design manual is being planned which will describe in detail the overtopping performance of seawalls and the standards to which they should be designed (Herbert et al, 1994). Although anticipated to be available in draft form in late 1995, it has not been released to date.
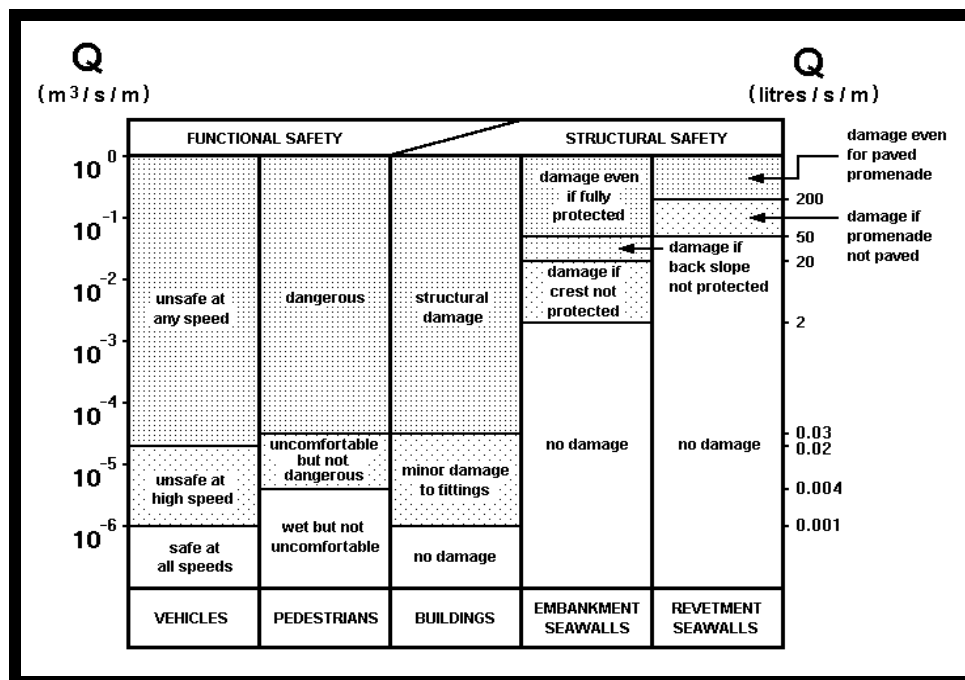
Figure 3.2: Critical mean overtopping discharges for use in design (modified after CIRIA/CUR, 1991).

Figure 3.2 shows that full-scale discharges greater than about $0.001 \times 10^{-3} m^3/s/m$ will be unsafe for vehicles at high speed and may cause minor damage to the fittings of buildings. Conditions become dangerous for pedestrians when the discharge exceeds $0.03 \times 10^{-3} m^3/s/m$. Discharges greater than about $2 \times 10^{-3} m^3/s/m$ may damage embankment seawalls, whilst $50 \times 10^{-3} m^3/s/m$ is approximately the critical discharge for seawalls without back slopes. Research in Italy using model cars and people (De Gerloni et al, 1991; Franco et al, 1994) suggests that higher limits than those shown in Figure 3.2 might be appropriate in some circumstances. However, these limits take no account of the psychological effects of sudden wetting, shock, and related factors (Allsop, 1994).

The main point to note from Figure 3.2 studies is that the range of critical mean discharges runs from as little as 10^{-6}m³/s/m to about 2×10^{-1}m³/s/m. Higher overtopping rates are of little interest in the design of seawalls, though they may be of concern to the designers of breakwaters. These apparently low figures account for the fact that danger levels are actually determined by the single largest overtopping wave which, due to the high irregularity of the physical phenomenon, can produce peak intensities much greater than the average intensity (Aminti & Franco, 1988).

According to Allsop (1994), continuing research in Italy and in the UK suggests that the volume in the largest individual overtopping wave may be about 6 to 10 times the average volume in an overtopping wave. Assuming that this quantity is discharged over 1/4 of a wave period, it may be shown that the peak overtopping discharge, Q_p, is then given by:

$$Q_p \approx 4000 \frac{Q}{N_{WO\%}} \qquad\qquad (3.8)$$

Now, consider a low level of overtopping, say $N_{WO\%}$=0.5%, and Q=0.03x10^{-3}m³/s/m. Under this condition, the worst wave could well project about 50 litres at significant speed, equivalent to an instantaneous discharge of 240x10^{-3}m³/s/m, probably with relatively little warning (Allsop, 1994). The impact of such a volume of cold water could cause anyone walking or running to fall. In this connection, another factor which is worth considering is the intensity of water falling as a function of the horizontal distance behind the structure. Work has already begun on this subject related to breakwaters (Jensen & Sorensen, 1979; Jensen & Juhl, 1987). Similar research for seawalls would also be beneficial to designers.

Whether considering mean overtopping discharges or volumes of overtopping, attention must be paid to the influence which the wind has both on overtopping quantity and distance of travel (TACPI, 1974; Jensen & Sorensen, 1979; Gadd et al, 1984; Allsop, 1986; Thomas & Hall, 1992). The most severe incident wave conditions are likely to be associated with strong onshore winds which increase the overtopping rate. As a guide, the overtopping discharges calculated using the empirical models (Table 3.1) may be multiplied by the wind correction factor quoted in the Shore Protection Manual (U.S. Army Corps of Engineers, Coastal Engineering

Research Center, 1984). The factor ranges from a value of unity for a very flat slope with a low freeboard or when there is no wind, to over three for a vertical wall with a high freeboard and with very strong onshore winds. However, Jensen & Juhl (1987) show physical model results which suggest that the effects of the wind may be even greater if the seawall surface throws water into the air, such as may occur when the front slope is armoured with natural stone or concrete units with large voids. Unfortunately, spray is not correctly simulated in hydraulic model tests owing to the influence of surface tension. Research on the effects of wind on wave run-up and overtopping is currently being conducted (Ward et al, 1994, 1996; De Waal et al, 1996), including laboratory and prototype measurements. Laboratory results from research undertaken by Ward et al (1996), which have not yet been published but have been made available to the author, suggest that model wind speeds of up to 6.5m/s have little effect on run-up and overtopping of smooth and rough slopes of 1:1.5, 1:3 and 1:5. Only strong winds of 12m/s and 16m/s have greatly increased both run-up and overtopping. The difficulty remains of scaling up the effects of wind in the model to prototype values. More information from the above studies will be very useful in improving understanding of the phenomenon, with a view to further development of the models used to predict overtopping. The studies can also help in defining the content of further research on the effects of wind.

3.3   Regression Analysis

Once experimental data have been collected, they may be used to confirm the validity of some theory or, where no satisfactory theory exists, they may be used to construct regression models. However, it is always useful to have some theoretical basis for choosing amongst the possible models. Furthermore, there are many techniques available for fitting regression models (Gunst & Mason, 1980). Which ones are appropriate for a particular study depend upon its objectives. For example, it may be possible to develop a model which is good at predicting values of the response variable but which, nevertheless, is incorrectly specified (i.e. the model does not include all relevant predictor variables or it has an incorrect functional form). In describing a regression model, care should be taken to emphasise the range of conditions over which there are data to support its use. Unfortunately, it is sometimes impossible to collect data on the dependent or response variable

(in this instance, overtopping) over the entire range of interest of the independent or predictor variables (wave height, structure profile, etc).

Consider the following example. The ability of armour stones to remain in place on a slope under the action of waves may be characterised in terms of their stability number, N_s:

$$N_s = \frac{H}{\left(\dfrac{\rho_s}{\rho} - 1\right)\left(\dfrac{W}{\rho_s g}\right)^{1/3}} \tag{3.9}$$

In this expression, H is the height of regular waves for which armour stones of weight W and density ρ_s are just stable; ρ is water density and g is gravitational acceleration. The influence of the armour slope, α, on the value of N_s has been investigated in many studies including those of Iribarren (1938) and Hudson (1959). They proposed the following relationships:

$$\text{Iribarren:} \quad N_s = \frac{(\mu_f \cos \alpha - \sin \alpha)}{K^{1/3}} \tag{3.10}$$

and

$$\text{Hudson:} \quad N_s = (K_D \cos \alpha)^{1/3} \tag{3.11}$$

K and K_D are armour stability coefficients and μ_f is the coefficient of friction between the armour stones. Note that Iribarren's equation correctly predicts that N_s will be zero when the armour slope is at its natural angle of repose, ϕ. In this state, $\mu_f = \tan\phi = \tan\alpha$. Under these conditions, there is a balance between the shear resistance along potential failure planes within the mass of stones and the downslope force induced by gravity (Hedges, 1983, 1985). Any small wave height would disturb this balance and, consequently, N_s must be zero. However, Hudson's relationship gives N_s as zero only when $\alpha = 90°$ (assuming that K_D is not zero). Note, also, that Hudson's formula suggests that the armour will be infinitely stable when $\alpha = 0°$ (suggesting that even particles of sand would not move on a horizontal seabed), whilst Iribarren's formula predicts a finite level of stability. Figure 3.3 shows the relationships between N_s and α provided by the two empirical expressions.
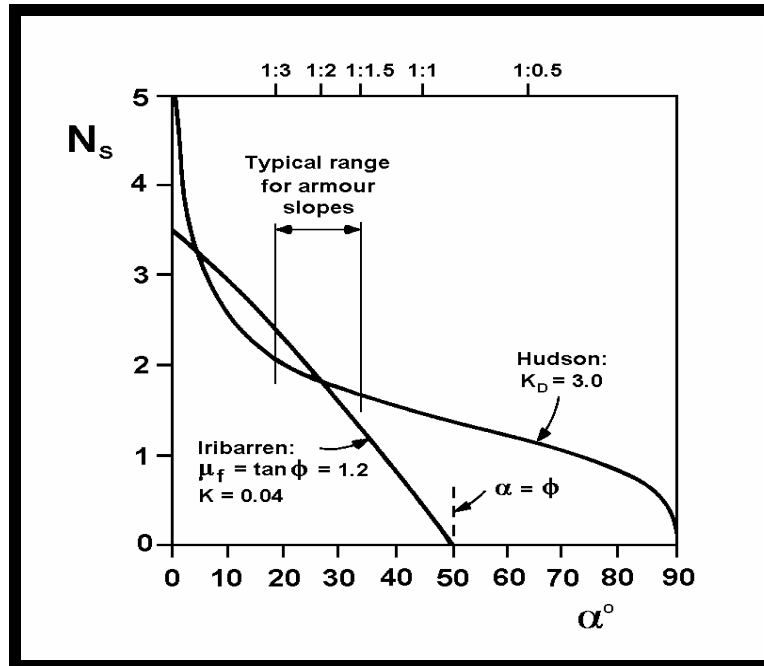
Figure 3.3: Relationships between N_s and α provided by Iribarren's and Hudson's empirical expressions (Hedges, 1995).

Both Iribarren's and Hudson's formulae may be used to fit experimental data over the normal range of slopes upon which armour is placed, for the types of armouring and wave conditions for which they have been developed, and so on. In other words, they may be good for predicting responses within the limitations imposed by the range of conditions for which they have been tested. However, there could be considerable errors if these formulae were used for other purposes (for example, to assess the relative importance of individual predictor variables to armour failure). Note that neither formula explicitly includes wave period as a predictor variable (though its influence could be contained within the values of the coefficients K and K_D).

The above example suggests that whilst Hudson's formula may fit data within the normal range of slopes, it does not fit the obvious physical boundary conditions which require that $N_s=0$ when $\alpha = \phi$ and N_s remains finite when $\alpha = 0°$. If this formula was to be applied near the boundaries, it would result in considerable errors. This experience can be used to provide a more physically reasonable regression model for wave overtopping data rather than merely providing an empirical fit to the data. As well as ensuring that all

relevant predictor variables are identified, it seems important to address the physical boundary conditions which should be satisfied. As a start, consider the physical boundary conditions to be met in addressing wave overtopping:

i) when the embankment has a large freeboard (i.e. when its crest elevation is well above the level of wave uprush), the predicted overtopping discharge should be zero (assuming that the effects of wind-blown spray are ignored);

ii) when the embankment has zero freeboard (i.e. when still-water-level is at the crest level of the embankment) then the predicted overtopping discharge may be large but should still remain finite.

As mentioned in Section 3.2.2, eqs. (3.6) and (3.7) represent two of the more common functions used to predict wave overtopping. However, when R_* is large, both expressions suggest that the discharge will be finite rather than zero (though it is small provided that A is not very large and provided also that B>1). When R_* is zero, the first of these expressions gives $Q_*=A$, a finite quantity, whilst the second expression gives $Q_* = \infty$. Thus neither expression satisfies both boundary conditions, with the second of them satisfying neither. Since most seawalls are designed to permit only relatively small overtopping discharges (see Section 3.2.4), it is especially important to satisfy the first of the two boundary conditions.

In addition to considering the boundary conditions, there is also the need to establish the line of "best fit" to the observed data. There are many criteria for defining the best fit. One possibility is to minimise the sum of the squared deviations of the observations from the values predicted from the expression. But real data usually do not completely satisfy the classical assumptions for a least-squares (LS) fitting (Rousseeuw & Leroy, 1987). For example, the deviations may not follow a Normal distribution. Reliable inferences may be drawn from regression models fitted by the LS method only if the assumptions are valid (Draper & Smith, 1981; Rousseeuw & Leroy, 1987). Furthermore, an LS fitting has the disadvantage that the result is not "robust": it is sensitive to outlying data points. Whilst "outliers" could be removed, such a procedure should only be considered if there is reason to doubt their validity. Such data must not be removed merely because they do not support the regression model: it may be the model which is wrong.

Performing a least-absolute-deviations (LAD) fitting, involves minimising the sum of the absolute deviations rather than the sum of the squared deviations. It does not rely upon the Normal assumption and allows outliers to be retained but prevents these points from exerting a disproportionate influence on the values of the regression coefficients. If the deviations are assumed to follow a Double Exponential distribution, which has thicker tails than the Normal distribution, then the parameter values are maximum likelihood estimates. In this research, it was decided to fit the regression lines using both the LS and LAD methods. The results of the two approaches could then be compared.

3.4   A New Regression Model

3.4.1   A Simple Overtopping Theory For Regular Waves

Stepping back from the complications of random waves, consider the simpler case of regular waves of height H approaching normal to a seawall. It will be assumed that the instantaneous discharge of water over unit length of the seawall, q, is given by the weir formula (Streeter & Wylie, 1979):

$$q = \frac{2}{3} C_d \sqrt{2g} (\eta - R_c)^{3/2} \quad \text{for } \eta > R_c \tag{3.12}$$

in which η is the water surface elevation above still-water-level at the seawall (a periodic function of time); C_d is a discharge coefficient. Obviously, overtopping occurs only when the water surface is above the structure's crest.

Assume that:

$$\eta = kHF(t) \tag{3.13}$$

F(t) denotes a function of time, t. For simple, sinusoidal, progressive waves, k=0.5 and $F(t) = \cos(2\pi t / T)$, where T is the wave period. However, following Kikkawa et al (1968), an even simpler form for F(t) will be adopted (see Figure 3.4); k remains a coefficient determined by the particular wave and wall details.

The mean discharge, Q, is determined as follows:

$$Q = \frac{2}{3} C_d \sqrt{2g} \frac{1}{T} \int_{t_1}^{t_2} \{kHF(t) - R_c\}^{3/2} dt \tag{3.14}$$

in which $t_1 < t < t_2$ corresponds to the interval during each wave period for which $kHF(t) > R_c$. Using the form for F(t) given in Figure 3.4 then yields:

$$\frac{Q}{\sqrt{g(kH)^3}} = \frac{2\sqrt{2}}{15} C_d \left\{1 - \frac{R_c}{kH}\right\}^{5/2} \quad \text{for } 0 < R_c < kH$$
$$= 0 \qquad\qquad\qquad \text{for } R_c \geq kH \tag{3.15}$$

Note that overtopping occurs only when $R_c < kH$. In other words, kH represents the run-up on the face of the seawall. Since wave run-up is a function of the incident wave height and steepness, and of the seawall slope, the overtopping discharge can be expected also to depend upon these parameters.
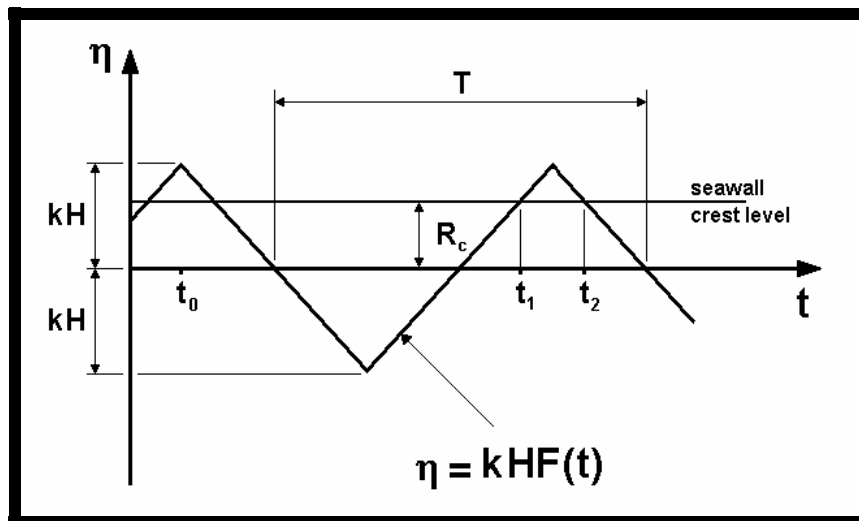
Figure 3.4: Form of function F(t).

3.4.2 The Hedges & Reis Overtopping Model

The above theory suggests a regression equation for the random overtopping data of the following form:

$$
\begin{aligned}
Q_* &= A(1-R_*)^B \quad &\text{for } 0 \le R_* < 1\\
&= 0 &\text{for } R_* \ge 1
\end{aligned}
\tag{3.16}
$$

in which

$$
Q_* = \frac{Q}{\sqrt{gR_{max}^3}} = \frac{Q}{\sqrt{g(CH_s)^3}}
\tag{3.17}
$$

and

$$
R_* = \frac{R_c}{R_{max}} = \frac{R_c}{CH_s}
\tag{3.18}
$$

Eq. (3.16) is the Hedges & Reis overtopping model (H&R model). Coefficient k in the expression for regular waves has been replaced by C in this regression model for random waves characterised by H_s. Note that CH_s represents R_{max}, the maximum run-up induced by the random waves, not the run-up induced by a wave of height H_s. Consequently, C will depend upon the duration of the incident wave conditions unless the wave heights in front of the wall are limited by the available water depth. Unless the maximum run-up, R_{max}, exceeds the freeboard, R_c, there is no overtopping (apart from wind-blown spray). It is also clear that coefficient B is related, in the case of regular waves, to the shape of the function F(t) which describes the water surface variation on the seaward face of the wall. There will be a similar dependence on the detailed behaviour of the water surface at the face of the wall in the case of random waves. Finally, coefficient A represents the dimensionless discharge over the seawall when the freeboard is zero. All three coefficients will be influenced by the seaward profile of the structure.

The above model for overtopping has the advantage that $Q_*=0$ when $R_* \ge 1$ and that $Q_*=A$ when $R_*=0$, in accordance with the required boundary conditions. Figure 3.5 shows the influences of coefficients A, B and C in the new overtopping model.
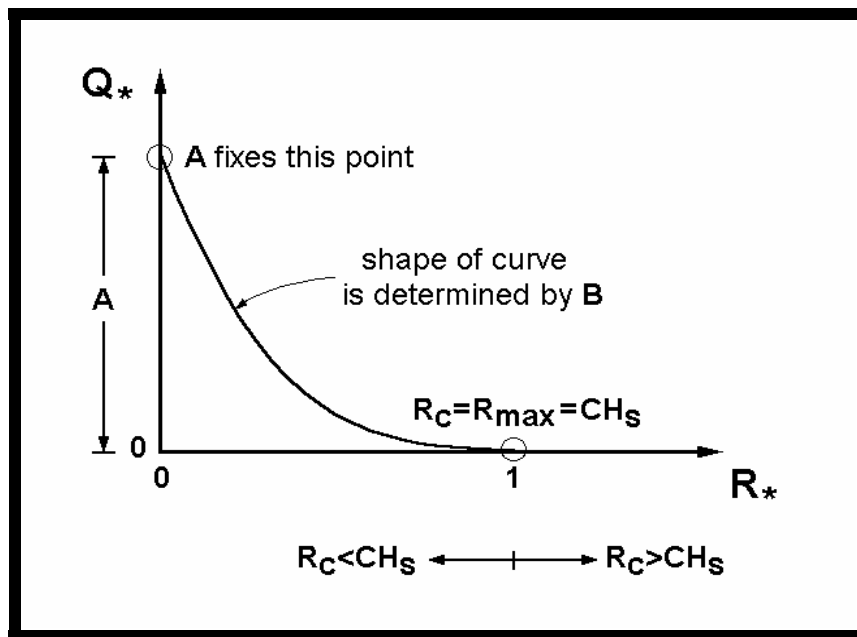
Figure 3.5: Influences of coefficients A, B and C in the new overtopping model.

The value of C ($=R_{max}/H_s$) to be adopted would best be determined from experimental data. Unfortunately, Owen's data set (and others) do not provide an adequate number of cases involving zero or very small discharges. Consequently, its value has been estimated from run-up measurements for random waves acting on slopes for which there is no overtopping. Although not ideal for the determination of C, these additional data on run-up complement Owen's overtopping results, allowing the new model to be applied outside the range of his experimental data. This option has been adopted rather than including C alongside A and B as a regression coefficient.

A number of equations describing random wave run-up are available (CIRIA/CUR, 1991; Van der Meer & Janssen, 1995). For example, the CIRIA/CUR (1991) manual gives two equations for evaluating the significant wave run-up, R_s, on smooth slopes without overtopping. It notes that these equations, based upon Ahrens' data (Ahrens, 1981), are probably conservative and that data from Allsop et al (1985b) give values 20 to 30% lower. Rewritten in the present notation and allowing for a printing error, the expressions are:

$$\frac{R_s}{H_s} = 1.35\,\xi_p \qquad \text{for}\;\; 0 < \xi_p < 2$$

$$\frac{R_s}{H_s} = 3.00 - 0.15\,\xi_p \qquad \text{for}\;\; 2 < \xi_p < 20 \tag{3.19}$$

Here, ξ_p is the surf similarity parameter calculated using T_p which was estimated for Owen's data using the relationships between H_s, T_m and T_p provided by Isherwood (1987).

Adopting the common assumption that run-up may be described by a Rayleigh distribution (Battjes, 1974; Ahrens, 1977; Bruun, 1985; CIRIA/CUR, 1991; Kobayashi & Raichle, 1994; Van der Meer & Janssen, 1995), then the p% confidence value of maximum run-up (defining a level below which p% of the cases should lie) is related to the significant wave run-up by (Hogben, 1990):

$$(R_{max})_{p\%} = \left[\frac{1}{2}\left(Ln\,N - Ln\left(-Ln\frac{p}{100}\right)\right)\right]^{1/2} R_s \tag{3.20}$$

N is the number of run-up values, here taken conservatively to be equal to the number of incident waves.

Owen recorded his overtopping discharges during tests involving sets of five different runs, each of 100 waves, characterised by the same significant wave height. The most probable maximum run-up during each run (the value not exceeded in 37% of the cases for a Rayleigh distribution of run-ups) is then:

$$(R_{max})_{37\%} = \sqrt{(Ln\;100)/2}\; R_s = 1.52R_s \tag{3.21}$$

In none of Owen's cases were there overtoppings for freeboards greater than $(R_{max})_{37\%}$ if R_s was evaluated using eqs. (3.19). In fact, all nine reported cases of zero overtopping were for freeboards of less than this value. Hence, setting $C=(R_{max})_{37\%}/H_S$ is conservative in this instance and the following expressions for C then arise from eqs. (3.19) and (3.21):

$$C = 1.52\,(1.35\,\xi_p) \qquad \text{for}\;\; 0 < \xi_p < 2$$

$$C = 1.52\,(3.00 - 0.15\,\xi_p) \qquad \text{for}\;\; 2 < \xi_p < 20 \tag{3.22}$$

The fact that these expressions for C are conservative may be a result either of the conservative nature of eqs. (3.19) or of deficiencies in the assumptions relating to the distribution of run-ups. However, setting $C=(R_{max})_{37\%}/H_S$ may not always be appropriate. Note that the value of C to be adopted in the regression model depends both upon the level of confidence associated with the prediction of R_{max} and upon the duration of the incident wave conditions. If C is changed then there will be corresponding changes in the values of A and B. The implications for seawall freeboards of adopting different levels of confidence in R_{max} are considered later.

3.5   Results Of Regression Analysis

Appendices A1 to A3 describe in detail how Owen's data have been used in carrying out regression analyses for the three embankment slopes of 1:1, 1:2 and 1:4. The H&R model (employing both $(R_{max})_{37\%}$ and $(R_{max})_{99\%}$ in defining the value of C) and Owen's model have each been considered. Regression analysis started by applying the LS method. The presence of both outlier data points and violation of the Normal error LS assumption lead to the subsequent use of the LAD technique (Gentle, 1977; Narula & Wellington, 1977; Sposito et al, 1977; Rousseeuw & Leroy, 1987). The LAD results appear more reliable than those of the LS method. Consequently, the LAD estimates of the regression coefficients are recommended for further use in the two models.

Figure 3.6 shows an example of the overtopping data collected by Owen: the results for a simple seawall with a uniform front slope of 1:2. The data are plotted in the formats required for fitting regression equations using both the H&R and the Owen models. Figure 3.6(a) shows the best-fit lines established using LS and LAD procedures for the H&R model. Comparison of the regression coefficients shows the influence which the outlying data points have on the LS values. For example, the magnitude of B obtained from the LAD fitting is about 92% of the LS result. Similar comments may be made about the regression lines obtained for Owen's model. Note, that the values of A and B reported in Figure 3.6(b) for Owen's model are not those which Owen himself recommended.
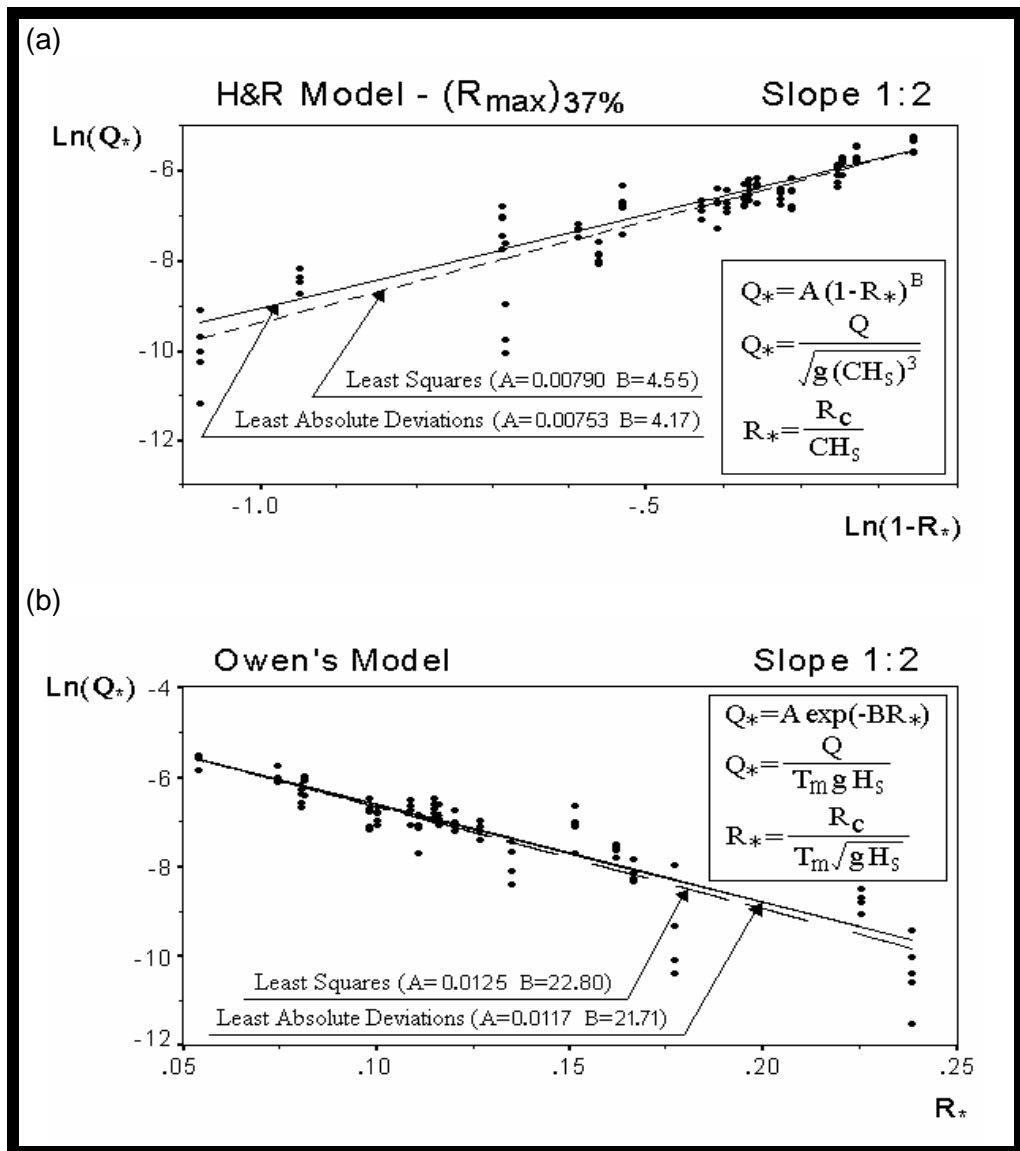
Figure 3.6: Wave overtopping data for slope 1:2 plotted in the formats required for fitting regression equations, (a) using the H&R model, $(R_{max})_{37\%}$, and (b) using Owen's model.

Table 3.2 gives the regression coefficients for all three slopes which have been obtained for the H&R model and for Owen's model, using both LS and LAD fitting. Also included for reference are Owen's recommended values.

		H&R MODEL (C given by $(R_{max})_{37\%}$)		H&R MODEL (C given by $(R_{max})_{99\%}$)		OWEN'S MODEL		
		LAD	LS	LAD	LS	LAD	LS	Rec.
Slope 1:1	**A**	0.00703	0.00581	0.00515	0.00474	0.00777	0.00758	0.00794
	B	3.42	3.22	6.06	6.04	20.44	21.27	20.12
Slope 1:2	**A**	0.00753	0.00790	0.00542	0.00614	0.0117	0.0125	0.0125
	B	4.17	4.55	7.16	7.98	21.71	22.80	22.06
Slope 1:4	**A**	0.0104	0.00792	0.00922	0.00870	0.0134	0.0164	0.0192
	B	6.27	5.94	10.96	11.12	42.92	46.12	46.96

Table 3.2: Regression coefficients obtained for use in the H&R model and Owen's model. Also included for reference are Owen's recommended (Rec.) values.

Owen restricted his analysis to a particular set of conditions whilst, in this analysis, all available data were included apart from eleven of the 110 results for the 1:4 slope. Of these eleven, nine had Q=0. Figure 3.5 shows that there will be many values of R_* for which $Q_*=0$ and data points with $Q_*=0$ must be excluded from a regression analysis, otherwise a regression line (if it could be fitted) would pass through these data rather than defining their lower limit. The other two excluded values (for which Q was not zero) were from a set of five runs with the same dimensionless freeboard, three of which had zero overtopping discharge. Including only two of these five data points would have severely biased the positions of the regression lines. Furthermore, the validity of these two data points is doubtful since a full set of five runs at a smaller dimensionless freeboard all had Q recorded as zero. Although removed for the purposes of regression analysis, the eleven points were reinstated for inclusion in Figure 3.7 (see later). The full data set fell within the following ranges:

$$0 < \frac{Q}{\sqrt{g(CH_s)^3}} < 0.0056 \qquad 0.14 < \frac{R_c}{CH_s} < 0.90$$

$$0 < \frac{Q}{T_m g H_s} < 0.0039 \qquad 0.053 < \frac{R_c}{T_m\sqrt{gH_s}} < 0.239$$

$$0.0053 < \frac{H_s}{gT_m^2} < 0.0095$$

$$1.65 < \frac{d_s}{H_s} < 5.20$$

(3.23)

Earlier, the problem of spurious correlation was mentioned. Along with most other overtopping models (see Table 3.1), the H&R model employs a

dimensionless discharge and a dimensionless freeboard which contain a common variable (R_{max} or CH_s). The presence of this common variable may reduce the apparent scatter in the data. Consequently, Figure 3.7 shows directly the level of agreement between Owen's measured values of Q (converted by Owen to full-scale discharges for a seawall in 4m water depth) and the predicted values, Q_{PRED}. Of course, the scatter in the relationship between Q and Q_{PRED} could have been disguised by plotting against logarithmic scales (Massey, 1971). But such an attempt is both misleading and unnecessary. Under random wave conditions, overtopping will be dominated by the few waves with large run-ups: most waves will contribute no overtopping if the seawall has a substantial freeboard (Jensen & Juhl, 1987; Aminti & Franco, 1988). Thus, particularly for short runs of random waves, as in Owen's tests, some variability in the measured values of Q can be expected. Indeed, one of the purposes of Owen's tests was to show this inherent variability.

In Figure 3.7, most data points lie within a range for Q/Q_{PRED} of 3/4 to 4/3, whichever model is adopted. It is not obvious from the figure which model best fits the data, nor is it obvious from the plots for simple seawalls with 1:1 and 1:4 front slopes. Consequently, the data points for discharges in the ranges of practical interest (see Figure 3.2) were looked at in more detail. For these purposes, the H&R model appeared generally better than Owen's model owing to its ability to predict zero overtopping at finite values of freeboard. Furthermore, the next section shows that it tends to give lower required crest levels than Owen's model for small permissible discharges, offering lower environmental impact and potential cost savings.
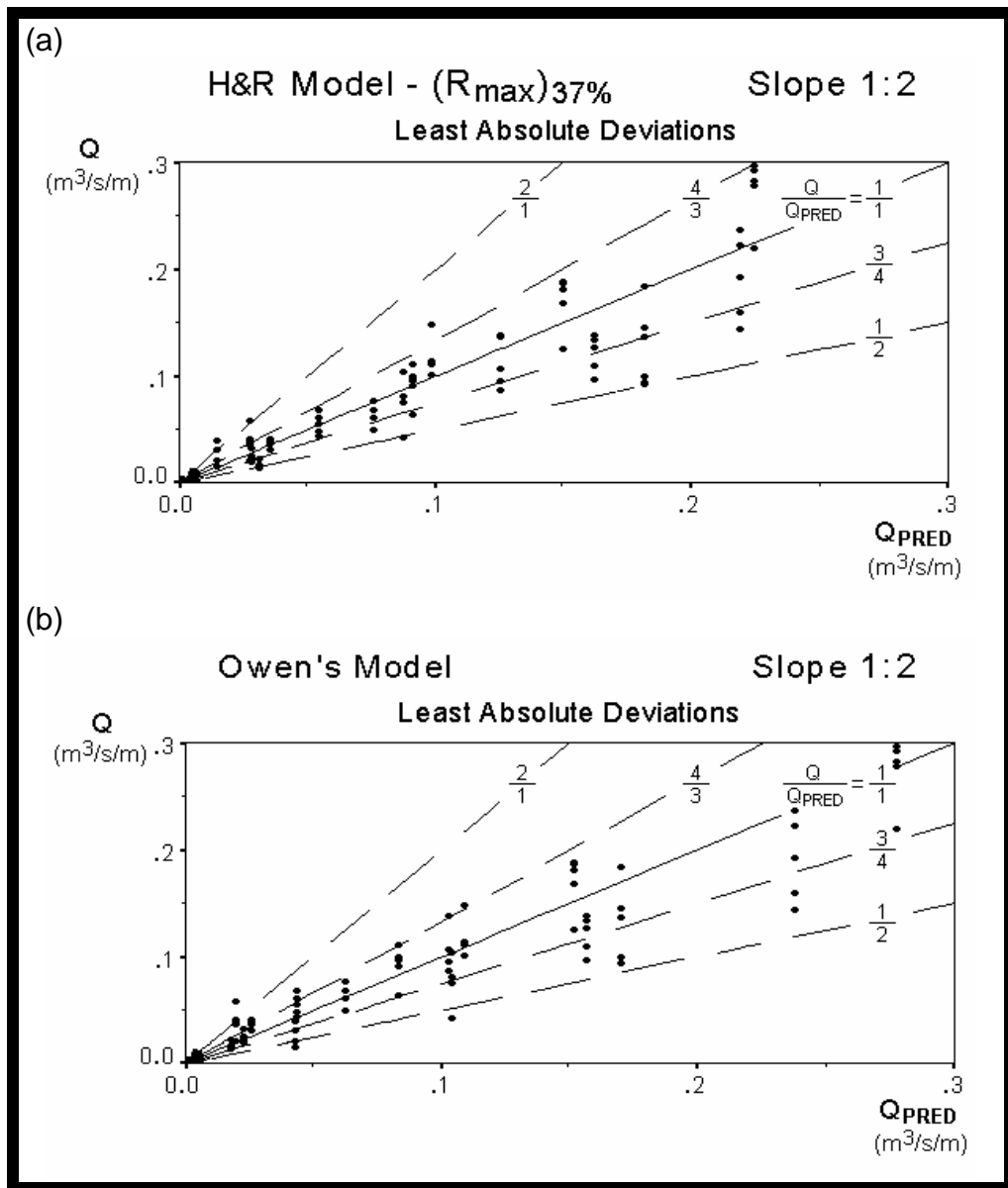
Figure 3.7: Wave overtopping data for slope 1:2, showing the level of agreement between Q and Q_{PRED}, (a) using the H&R model, $(R_{max})_{37\%}$, and (b) using Owen's model.

3.6   Some Implications Of The New Model For Seawall Freeboards

According to Owen's model, the freeboard, R_c, necessary to limit overtopping to a specified value, Q, is given by:

$$R_c = \frac{T_m\sqrt{gH_s}}{B} \, Ln\left\{\frac{AT_mgH_s}{Q}\right\}$$

(3.24)

The H&R model gives:

$$R_c = CH_s \left\{ 1 - \left[\frac{Q}{A\sqrt{g(CH_s)^3}} \right]^{1/B} \right\} \qquad (3.25)$$

Note that eq. (3.24) incorporates the mean zero-crossing wave period, T_m, whilst eq. (3.25) involves coefficient C which has been described in terms of the period of peak spectral density, T_p. In order to compare the output from the two expressions, it has been assumed that the incident waves conform to the Pierson-Moskowitz spectrum. In this case (for H_s in metres, with T_m and T_p in seconds):

$$\begin{aligned} T_m &= 3.55\sqrt{H_s} \\ T_p &= 5.00\sqrt{H_s} \end{aligned} \qquad (3.26)$$

Figure 3.8 provides a comparison between the freeboards predicted using eqs. (3.24) and (3.25) for embankments with uniform front slopes of 1:2 subject to random waves with a significant height of 2m. Similar figures could be prepared for embankments with seaward slopes of 1:1 and 1:4, for additional incident significant wave heights and for different values of the confidence level associated with R_{max}.

Owen stated (Hydraulics Research Station, 1980) that his empirical coefficients A and B were determined only for particular ranges of the dimensionless groupings given in eq. (3.4). The conditions included the following: $10^{-6} < Q/T_m g H_s < 10^{-2}$ and $0.05 < R_c/T_m\sqrt{gH_s} < 0.30$. Many of the discharges shown in Figure 3.8 have $Q/T_m g H_s < 10^{-6}$. For the conditions of Figure 3.8, this limit is approximately equivalent to Q<10⁻⁴m³/s/m. Nevertheless, Owen suggested that it was possible to use his equation to extrapolate results when the dimensionless freeboard was such that the dimensionless discharge fell below 10⁻⁶. Thus, for a typical seawall in 4m water depth, it is possible to compare the minimum necessary freeboards predicted by the H&R model with those predicted by Owen's expression if overtopping is to be limited to specified values.
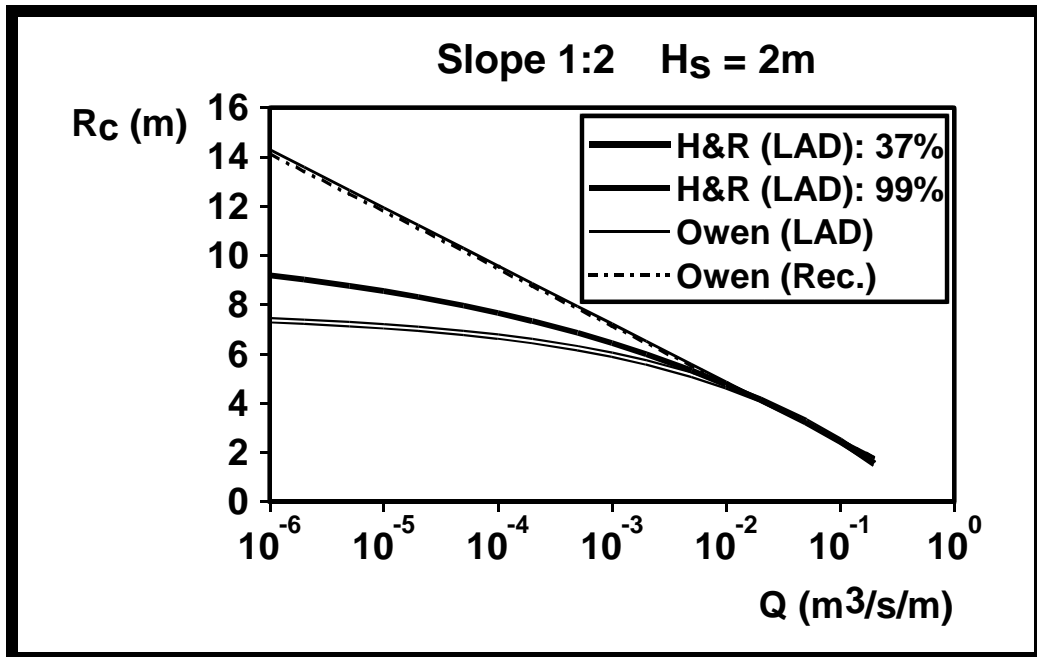
Figure 3.8: Comparison of predicted freeboards for slope 1:2
(LAD indicates that the values of the regression coefficients have been established using the method of least absolute deviations; Rec. refers to the coefficient values recommended by Owen for use in eq. (3.5); 37% and 99% refer to the confidence level associated with R_{max}).

Two points are worth noting:

- There is reasonable agreement between the H&R model and Owen's model for overtopping discharges in the range of 10^{-2}m^3/s/m to 2×10^{-1}m^3/s/m. This is irrespective of the confidence level (37% or 99%) assigned in the evaluation of R_{max}. However, it is in the range where there are significant differences between the two models that most seawalls are designed (see Figure 3.2).

- As the confidence level in R_{max} is increased, the freeboards predicted by the H&R model approach those values obtained from Owen's model. Nevertheless, even using $(R_{max})_{99\%}$ there remain significant differences. This observation has important implications for seawall design. For example, for an expected overtopping discharge of 10^{-4}m^3/s/m, the difference amounts to about 1.9m. It is even greater both for the lower expected overtopping rates associated with functional safety requirements (Figure 3.2) and for higher H_s values. Owen's model suggests that the freeboard must continue to increase in order to reduce the overtopping rate even when the crest of the seawall is clearly above any possible run-up level induced by the random waves.

3.7   Model Reliability

Neither the H&R model nor Owen's model provide a perfect description of the overtopping data. There is some scatter about the line of perfect agreement between predicted and measured values (see Figure 3.7). This scatter can be described by interpreting the coefficient A in the models as a random variable for a given coefficient B (Allsop & Meadowcroft, 1995) or by interpreting the coefficient B as a random variable for a given value of A (Van der Meer, 1993; Franco et al, 1994; Van der Meer & Janssen, 1995). Alternatively, the degrees of variability in A and B may be represented by parameters e_A and e_B, respectively:

$$\text{H\&R Model:} \quad Q_* = e_A\, A\left(1 - R_*\right)^B \quad \text{or} \quad Q_* = A\left(1 - R_*\right)^{e_B B}$$

leading to   (3.27)

$$e_A = \frac{Q_*}{A\left(1 - R_*\right)^B} \quad \text{and} \quad e_B = \frac{LnQ_* - LnA}{BLn\left(1 - R_*\right)}$$

$$\text{Owen's Model:} \quad Q_* = e_A\, A\exp\left(-BR_*\right) \quad \text{or} \quad Q_* = A\exp\left(-e_B BR_*\right)$$

leading to   (3.28)

$$e_A = \frac{Q_*}{A\exp\left(-BR_*\right)} \quad \text{and} \quad e_B = \frac{LnA - LnQ_*}{BR_*}$$

If the models were perfect representations of reality, e_A and e_B would both be equal to one. Otherwise they are random variables which may be described by a probability distribution.

The variability of e_A and e_B as functions of R_* have been investigated for the three front slopes of 1:1, 1:2 and 1:4. The H&R model (employing both $(R_{max})_{37\%}$ and $(R_{max})_{99\%}$ in defining the value of C) and Owen's model have each been considered. Similar results were obtained in all cases.
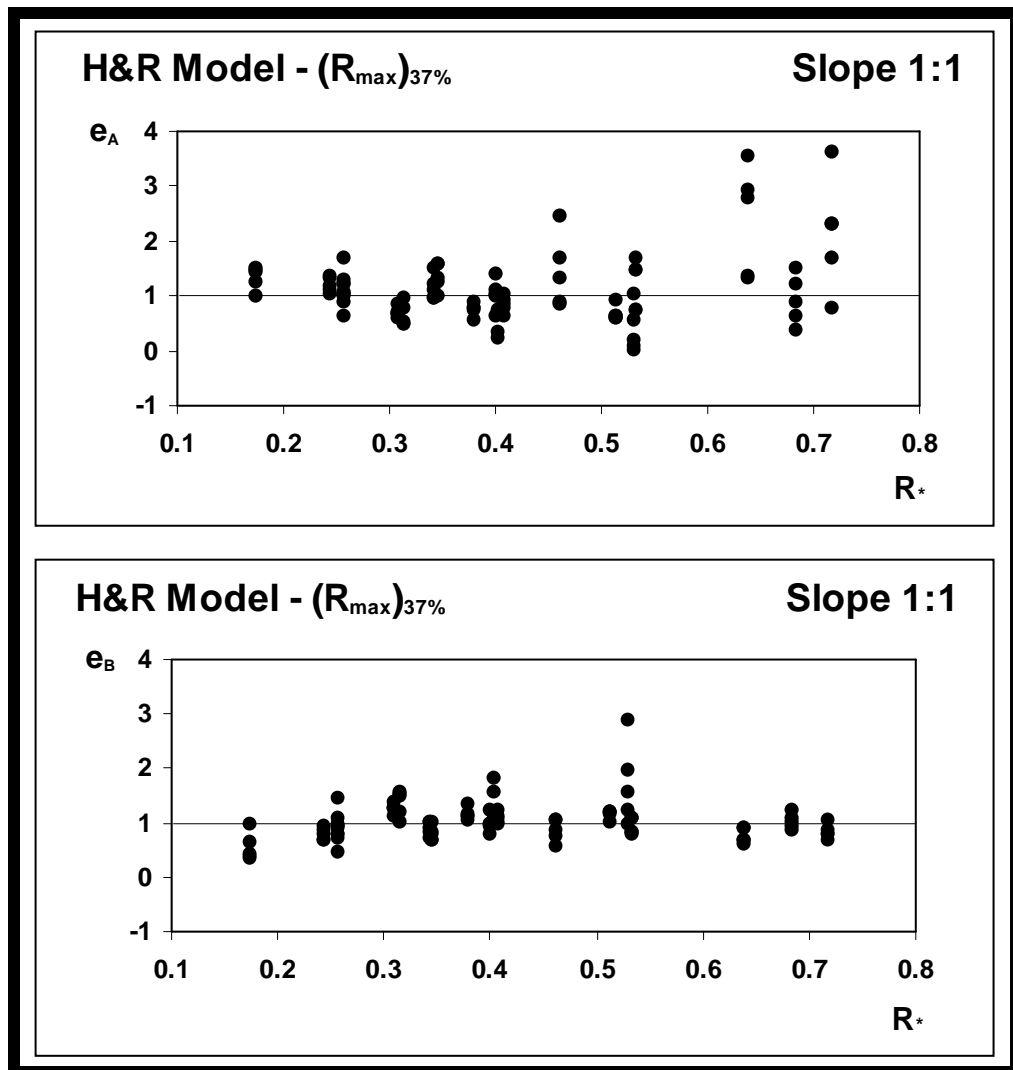
Figure 3.9: Variability of e_A and e_B as functions of R_*.

Figure 3.9 shows an example of the results for slope 1:1 using the H&R model with $(R_{max})_{37\%}$. The variability of e_A as a function of R_* is not constant: the figure suggests an increasing variability in e_A as R_* increases. The degree of variability in e_B as a function of R_* appears more constant. This, in addition to the smaller spread in the data for e_B, suggests that e_B is more appropriate than e_A for representing the reliability of the models. Hence the probability distribution of e_B for each model has been further investigated using the software package BestFit (see Appendix A4). Details of the analyses are contained in Appendix A5.

First, the e_B values were assessed using the summary statistics provided by BestFit and by plotting histograms of the data. General guidance on the

choice of distribution was obtained from these sources. Then, all the statistical distributions available both in PARASODE and BestFit were considered and full optimisation was adopted for calculation of their parameters. Finally, the adequacy of these distributions was determined using the three goodness-of-fit tests available in BestFit, together with histograms, P-P and Q-Q plots. Based on the numerical and graphical results, the Log-Normal, Maxima Type I (Gumbel) and Gamma distributions were chosen as possible candidates for describing the randomness of e_B. Thus, these three are used in the probability calculations performed with PARASODE and @Risk. Table 3.3 shows the means and the standard deviations of the input data for e_B. Table 3.4 shows the corresponding means and standard deviations of the fitted distributions.

INPUT DATA	H&R MODEL $(R_{max})37\%$						H&R MODEL $(R_{max})99\%$						OWEN'S MODEL					
	Slope 1:1		Slope 1:2		Slope 1:4		Slope 1:1		Slope 1:2		Slope 1:4		Slope 1:1		Slope 1:2		Slope 1:4	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
e_b	1.030	0.346	1.049	0.236	1.020	0.183	1.023	0.293	1.044	0.204	1.026	0.157	1.044	0.236	1.027	0.156	1.030	0.134

Table 3.3: Mean, μ, and standard deviation, σ, of the input data for e_b.

DISTRIBUTION	H&R MODEL $(R_{max})37\%$						H&R MODEL $(R_{max})99\%$						OWEN'S MODEL					
	Slope 1:1		Slope 1:2		Slope 1:4		Slope 1:1		Slope 1:2		Slope 1:4		Slope 1:1		Slope 1:2		Slope 1:4	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
LOG-NORMAL	1.031	0.337	1.049	0.241	1.019	0.167	1.022	0.264	1.044	0.200	1.025	0.142	1.043	0.214	1.027	0.150	1.029	0.122
MAXIMA TYPE I (GUMBEL)	1.044	0.346	1.062	0.248	1.020	0.183	1.039	0.294	1.048	0.211	1.034	0.155	1.044	0.236	1.031	0.162	1.030	0.134
GAMMA	1.030	0.324	1.049	0.235	1.020	0.144	1.023	0.265	1.044	0.198	1.026	0.145	1.044	0.217	1.017	0.150	1.030	0.125

Table 3.4: Mean, μ, and standard deviation, σ, of the three distributions chosen for e_b.

3.8   Summary

Wave overtopping of seawalls has been the subject of many studies. Nevertheless, field measurements are scarce and numerical modelling of wave overtopping is not yet well developed. The calculation of overtopping discharge is based mainly on equations which have been obtained from empirical fitting to hydraulic model test results. These equations have not been based upon any overtopping theory and no account has generally been taken of the physical boundary conditions.

As part of this research, a new regression model has been presented for describing wave overtopping data. Part of the motivation in deriving this new model was to improve the methods available to the designers of seawalls by developing a model closely related to the physics of wave overtopping. The important features of the model are as follows:

- Unlike existing expressions, it satisfies the relevant physical boundary conditions, a feature which is especially important when the model is used near these boundaries.

- It explicitly recognises (through its foundations in a simple theoretical model for regular waves) that regression coefficient A depends upon the shape of the structure since the shape, particularly at its crest, affects the discharge coefficient; coefficient A represents the dimensionless discharge when the dimensionless freeboard is zero.

- Coefficient B depends upon the detailed behaviour of the water surface on the seaward face of the structure; it increases as front slopes become flatter.

- Coefficient C relates the maximum run-up to the significant height of the incident waves and may be chosen to allow for the influences of the seawall slope, the surface roughness and porosity, and the incident wave steepness. Coefficient C can also account for storm duration in influencing R_{max} (though the regression coefficients in the present study have been established only for short sequences of 100 random waves). Finally, it may be chosen so that there is a specified confidence level associated with R_{max}. In this thesis, the most probable value of R_{max}, $(R_{max})_{37\%}$, has been adopted in order to establish the expressions for C (eqs. (3.22)) since they were conservative in this instance, but the use of $(R_{max})_{99\%}$ has also been illustrated. Note, that if the confidence level associated with R_{max} changes, the expressions for C change and there are corresponding changes in the values of A and B (see Table 3.2).

It is suggested that the regression coefficients contained within the model should be established using a robust regression technique. Examples are given of the differences between the LS and the LAD fitting methods in analysing overtopping data collected by Owen (Hydraulics Research Station, 1980; Owen, 1982a). The LAD regression coefficients are recommended for use both in the H&R and Owen models.

For the present test results, the H&R model is little different from Owen's model in its ability to represent the data, except for small discharges for which the H&R model is better suited. An example is given of the application of the H&R and Owen models in predicting the freeboards necessary to limit overtopping to specified values. This example shows that, for the small allowable discharges associated with normal design conditions, the H&R model predicts seawall crest elevations which may be several metres lower than values from Owen's model. Such differences may have very significant financial and environmental consequences and are worthy of further investigation.

The reliabilities of the H&R model and of Owen's model have been assessed by introducing a multiplying parameter to coefficient B of the models. In order to determine possible distributions for this model parameter, a software package called BestFit has been applied. The Log-Normal, Maxima Type I (Gumbel) and Gamma distributions are seen to be acceptable for use in the probabilistic calculations performed in this study.

Whilst it is possible to use Owen's data to show the validity of the approach adopted in developing the new wave overtopping model, the data are far from ideal for evaluating the empirical coefficients A, B and C. Owen collected his data for short runs of waves and the bulk of his data are for typical full-scale conditions which produce overtopping discharges well in excess of the allowable values shown in Figure 3.2. This second deficiency also applies to more recent data sets (Van der Meer & Janssen, 1995). Thus, the data available at present are not good for evaluating coefficient C which fixes the lowest value of R_* for which Q_* is zero. Neither are they good for evaluating coefficient A, the value of Q_* when R_* is zero. Owen performed no tests with zero freeboard. In the absence of information to locate the extremes of the curve shown in Figure 3.5, the values of coefficient B also remain in doubt. However, the author is aware that overtopping and run-up

tests now being carried out at the Technical University of Braunschweig, Germany (Schuttrumpf, 1997), include measurements when there is zero freeboard and when the freeboard is sufficiently great to prevent any overtopping. The latter condition will allow the direct measurement of R_{max}, obviating the need to estimate it.

In addition to considering the deficiencies of existing data for the above purposes, it is worth emphasising that approaches to coastal engineering design are shifting towards probabilistic rather than deterministic procedures. The variability in overtopping discharge must then be considered: it is necessary not only to predict the expected mean value of Q, but also the probability distribution of Q about this value. As a consequence, much larger data sets are needed. Furthermore, some attention must be paid to the horizontal distribution of the total overtopping volume and the influence which the wind has on overtopping discharges (De Waal et al, 1996; Ward et al, 1996).

4 Dune Erosion During A Storm Surge

4.1 Introduction

Dunes erode in two main ways: a gradual erosion (structural or long-term erosion) and a fast, sudden erosion/recession during a storm surge[1] (short-term erosion). Long-term erosion of beaches and dunes can be very inconvenient but future losses can be foreseen and, in most cases, can be predicted quantitatively. In contrast, high water levels and high waves during a storm surge erode huge quantities of dune material in a short time. Thus, short-term erosion is a condition which must be analysed very carefully. This review considers only the short-term erosion due to a severe storm surge.

Dunes occur naturally in many parts of the world. In their most natural state, they are associated with exposed dry sand being transported by the wind. In this state, they can migrate with the wind, sometimes invading and disrupting the works of man (e.g. to cover highways and railroads and to destroy productive agricultural land). On the other hand, dunes act as the primary protection against the sea in some regions of the world (e.g. in parts of the United Kingdom and The Netherlands). In these cases, effort must be put into preserving or enhancing the dunes (Thomas & Hall, 1992; Simm et al, 1996). In some regions, a high investment has been made in property which is very close to the edge of the sea. Figure 4.1 shows two schematic examples: (a) a low-lying area behind a narrow line of dunes; and (b) buildings close to the sea. Apart from a possible gradual overall erosion of the coast, both cases are safe under "normal" conditions. During a storm surge, however, the mean sea level rises considerably above normal high water level, higher waves than usual approach the shore and offshore transport occurs, especially of material from the dunes (Van de Graaff, 1986; Van de Graaff & Bijker, 1988). Figure 4.1 shows schematically what happens: sand eroded from the dunes is transported towards deeper water and settles there. The new beach profile develops at a more elevated level and the overall slope becomes less steep than the original. Consequently, the rate of erosion of the dunes slows down with time. After the storm, the water level and the wave heights go to "normal" conditions again, the dune

---

[1] Surge is the difference between the measured water level and the predicted (astronomical) tide level (Pugh, 1987).

erosion process is stopped, and a retreat distance, RD, can be observed. For safety reasons, the likely position of point R after the surge has to be known. In case (a), erosion of the dunes will cause flooding and damage to property with possible loss of life. In case (b), no serious flooding occurs but there may be destruction of buildings. Hence, the problem of managing a dune system is not so much to prevent erosion but to know in detail the rate of erosion that can be expected in order to judge dunes as safe or unsafe.
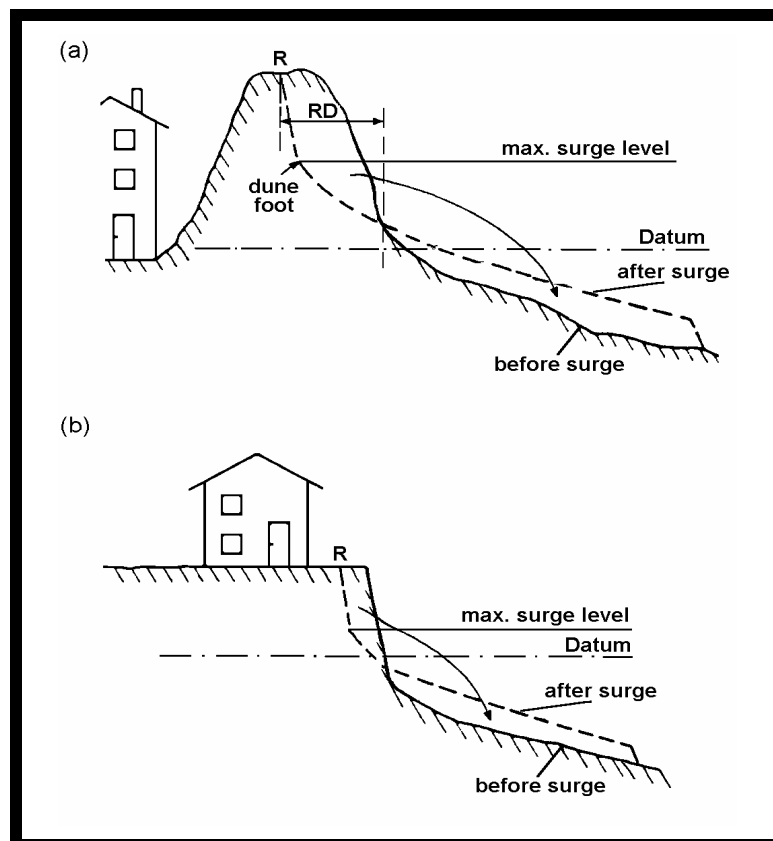
Figure 4.1: Schematic dune erosion situations (modified after Van de Graaff, 1986).

If an existing dune/beach system is unsafe, then consideration may be given to the possibility of providing nourishment. Although not appropriate for all locations, nourishment has proved to be a cost-effective, flexible and environmentally sensitive "soft" engineering strategy (when compared to other methods such as the use of groins, offshore breakwaters, etc.). It has been used in many places including the United Kingdom (Fleming, 1990; Townend & Fleming, 1991; Motyka & Brampton, 1994; HR Wallingford, 1994), The Netherlands (Vellinga, 1986; Van de Graaff & Bijker, 1988; Van

Raalte & Loman, 1993), and the USA (Dean, 1976; Housley, 1996). Nourishment usually does not create detrimental side effects in adjacent coastal areas (Van de Graaff et al, 1991). Consequently, there is a clear interest in nourishment as a strategy for controlling dune/beach erosion evident in such multi-national publications as Technical University of Braunschweig/SOGREAH Ingenierie/Centro de Estudios y Experimentacion de Obras Publicas (1997).

It must be stressed that a long-term commitment to nourishment is required to ensure that the benefits anticipated in the design will actually occur (Housley, 1996). Nourishment may seem expensive and the need for repetition may discourage coastal managers. However, repetition should be seen as regular maintenance, as conservation of valuable investments, just as for other structures (Van de Graaff et al, 1991). For example, in The Netherlands, dune/beach replenishment is generally expected to provide a buffer for more than five years (Van de Graaff et al, 1991; Van Raalte & Loman, 1993) and in the UK, "lives" of ten years or more are the norm (Motyka & Brampton, 1994). Careful consideration of capital and maintenance costs frequently proves that nourishment is the optimal solution. An added advantage is that the recreational function of the dune/beach system is preserved.

The type and volume of sediment required, possible borrow areas (e.g. inland sources or material dredged from navigation channels), the transportation system, the precise nourishment location and ease of placement, and socio-economic aspects have all to be considered. Possible nourishment areas include the land side of the dune, the seaward dune face, and the foreshore and inshore zones (CUR et al, 1987; Van de Graaff & Koster, 1990; Van de Graaff et al, 1991; D'Angremond, 1992; Van Raalte & Loman, 1993; Liverpool/Thessaloniki Network, 1996). The Manual on Artificial Beach Nourishment (CUR et al, 1987), the Beach Management Manual (Simm et al, 1996) and Beach Recharge Material - Demand and Resources (CIRIA, 1996) all provide useful further information.

The main approaches to modelling dune erosion may be categorised according to the type of integration (Steetzel, 1993):

- Space integration (a fixed shape for the cross-shore profile is assumed).

- Time integration (the net effect of the complete storm surge is accounted for).

Using these two types of integration, three main categories of dune erosion models can be distinguished (Steetzel, 1993):

- **Space and time integrated concept** - This approach results in a prediction of the erosion profile which is supposed to be present after a specific storm event. It can be characterised by so-called *equilibrium or erosion profile models* (e.g. Bruun´s model - Bruun, 1954, 1962; Dean´s model - Dean, 1977, 1982, 1987, 1991; the DUROS model - Vellinga, 1986).

- **Space integrated, instantaneous concept** - The development of the shape of the cross-shore profile is described during the storm event using a time-dependent shape function. Typically, this approach results in a negative exponential development of the profile (for constant hydraulic conditions). The term *quasi-equilibrium models* can be used for this category (e.g. Swart, 1974; the SBEACH model - Larson, 1988, Larson & Kraus, 1989; the DUIN model - Roelvink & Stive, 1989).

- **Local and instantaneous concept** - The local transport rate must be known at every position on the cross-shore profile in order to compute its development during a storm surge. An expression for the local transport rate has to be derived. The expressions available differ, mainly, as a result of the approach used in assessing the local transport rate (e.g. Kriebel & Dean, 1985; Kriebel, 1990; the DUROSTA model - Steetzel, 1990, 1993; Watanabe et al, 1994).

Obviously, the equilibrium and the quasi-equilibrium models are not suitable for the assessment of the effects of arbitrary hydraulic conditions on a cross-shore profile. However, they have the advantage of simplicity. Applications of Dean´s equilibrium model have been presented recently (see, for example, Kriebel, 1990; Dean, 1991). Vellinga´s model is still used in The Netherlands to check the safety of the Dutch dunes.

Coastal engineers are still becoming acquainted with the use of probabilistic methods for the design and assessment of coastal structures. A simple equilibrium model is a good starting point for combining probabilistic methods and dune erosion (Van de Graaff, 1986; Dong & Riddell, 1996). The use of simple erosion models allows the engineer to concentrate on understanding

the application of the probabilistic methods whilst also saving the considerable computational time needed in running the more sophisticated alternatives. Implementation of these sophisticated models in a probabilistic approach is possible but represents such a complex task that, as far the author knows, it has not yet been put into practice, even in pioneering countries like The Netherlands. Furthermore, some of the simpler models, in particular Vellinga´s approach, have proved to be generally conservative when compared to Steetzel's more sophisticated formulation (Van de Graaff, 1995). This may cause problems in relation to the public perception of the safety in applying Steetzel's model.

In the following sections, especial attention is given to the research carried out in The Netherlands. Vellinga's cross-shore erosion model (Vellinga, 1986) is studied in detail, not only because it has been the basis for the calculation of the safety of the dunes throughout The Netherlands but also because probabilistic methods have been applied in conjunction with it. Steetzel's model (Steetzel, 1993) is referred to only because it is the latest Dutch model; however, no probabilistic calculations have been undertaken with it so far. Next, the computer programs which are currently used in The Netherlands are presented briefly. Finally, the applicability of these programs in the British context is examined.

4.2   Vellinga's Model

4.2.1   Prior Research

Edelman (1968) was the first to present a method for the prediction of dune erosion in The Netherlands. His method was based on the assumption that during a storm surge a normal beach slope develops but at a higher level than before. Edelman used a straight beach slope of 1:50 in his computations. His method was improved by Van de Graaff (1977) who employed a realistic concave-upward erosion profile based on field observations. This profile is used in what is known as the provisional computational model (Vellinga, 1983, 1986).

Van der Meulen & Gourlay (1968) were the first in The Netherlands to investigate the process of dune erosion in small scale movable bed models.

The tests were mainly carried out in a basin with monochromatic waves. The tests provided qualitative answers to the question of how dune erosion is influenced by dune height, initial beach profile, wave height, wave period, sea level, and grain size characteristics. However, Hulsbergen (1974) found that such tests suffered from the effects of secondary waves.

In 1972, a research project started on the erosion of coastal dunes during storm surges. The aims of the research were (Vellinga, 1983, 1986): i) to increase insight into the phenomenon, and to develop a general model for the computation of the erosion quantity as a function of the hydraulic conditions and the coastal profile; and ii) to use this model to check the safety of existing beaches and dunes as primary coast protection and to determine the required reinforcement. Engineers in charge of coast protection were conscious that there was insufficient theoretical background and that a firmer basis for decisions was urgently needed. Hence, it was decided that the research would consist mainly of extensive experimental tests on small and large scale models, together with prototype measurements involving waves up to 2.0m significant height. Table 4.1 summarises how this research evolved. Vellinga (1982, 1983, 1986) provides details about the various stages of the project.

PRIOR RESEARCH		
1972	Provisional computational model - provisional guideline for the computation of dune erosion during a storm surge (TAW, 1972)	

1974-1975	2D scale series	Idealised coastal profile
		4 depth scales (n_d=26, 47, 84, 150)
		2 sand grain sizes (D_{50}=150, 225μm)
		Design Storm Surge Conditions:
		  - water level=5m MSL
		  - H_s=7.6m; T_p=12s

1976-1978		Idealised coastal profile
		3 depth scales (n_d=26, 47, 84)
		4 sand grain sizes (D_{50}=95, 130, 150, 225μm)
		Design Storm Surge Conditions:
		  - water level=5m MSL
		  - H_s=7.6m; T_p=12s

	Analysis and evaluation of scale relationships[2]: i) Froude scale for hydraulic conditions: $n_H=n_L=n_d=n_T^2$; ii) morphological time scale: $n_t=n_d^{0.5}$; iii) model distortion: $n_l/n_d=(n_d/n_w^2)^{0.28}$	
	Analysis of field data for dune erosion caused by the storm surge of January 1976. Evaluation of the Provisional Model (Delft Hydraulics Laboratory, 1978).	
1979-1980	Verification of the 2D approach by means of 3D movable bed small scale tests: these tests confirmed that a 2D approach was fully acceptable for relatively straight beaches.	
1981-1982	Verification of scale relationships by means of five large scale tests: n_d=1, 3.27 or 5; n_l=1, 4.56 or 7.85; D_{50}=225μm; Design storm surge conditions: i) constant or varying water level; ii) water depth in front of wave generator d=4.20 or 5m; iii) wave height at depth d: H_s=1.5, 1.8 or 2m; wave period at depth d: T=5, 5.4 or 7.6s. Erosion quantities confirmed the scale relationships from small scale tests. Erosion profiles also confirmed the scale relationships if finer sands than in the prototype were used in the small scale tests. Agreement between the erosion (quantities and profiles) obtained in the tests and in the field was satisfactory.	
	Parametric small scale model investigations to define the effect of dune erosion parameters on the rate of erosion: i) water level during storm surge, wave height and particle diameter were determining parameters: an increase in the water level during storm surge produced an increase in the dune recession and erosion quantity; the wave height had the same effect but to a much lesser extent (the angle of wave incidence did not have a significant effect on the erosion quantity); the finer the dune material the less erosion quantity and distance expected; ii) wave period and shape of the energy spectrum of the incoming waves did not have a significant effect; iii) steeper dune fronts had the highest amounts of erosion and the lowest erosion distances; iv) as the dune height increased the erosion quantity also increased while the dune recession decreased.	

Table 4.1: Research prior to the development of Vellinga's model.

4.2.2 Formulation

Based on the above research project, a dune erosion model was developed by Vellinga. It is known as the DUROS model (see Figure 4.2):

---

[2] n - ratio of the prototype value to the model value; n_H - wave height scale; n_L - wavelength scale; n_d - depth scale (for beach profile and hydraulic conditions); n_T - wave period scale; n_t - time scale; n_l - length scale for beach profile; n_w - scale for the fall velocity of the sand.
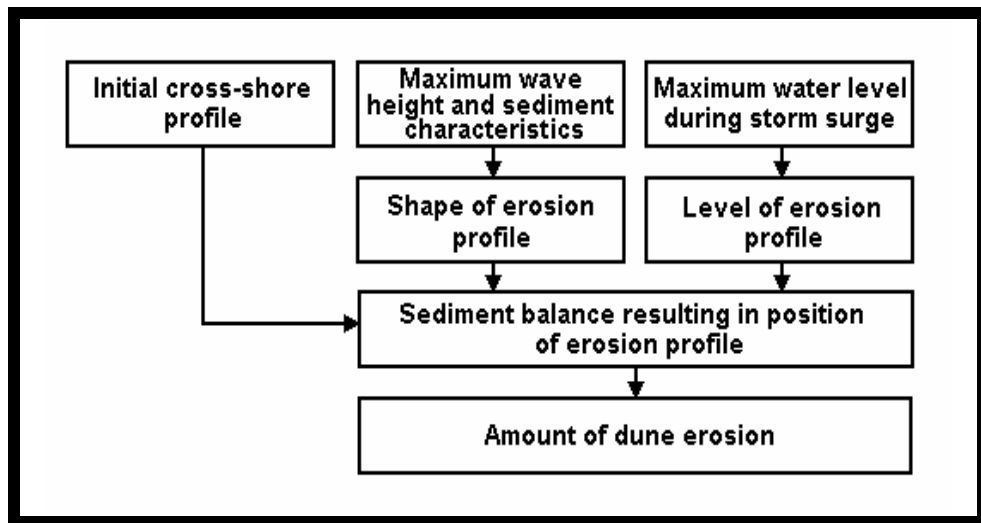
Figure 4.2: Outline of erosion prediction according to Vellinga (modified after Steetzel, 1993).

- The input parameters required by the model are (Vellinga, 1983): i) coordinates of the initial profile (X_i, Y_i); ii) significant deep water wave height, H_S (significant wave height at depth $d > 0.5L_{OS}$, where L_{OS} is the significant deep water wavelength); iii) median grain size diameter of dune sand, D_{50} (50% of the weight being finer) and its corresponding fall velocity, w, for a given water temperature; iv) maximum water level during storm surge, h.

- During a storm surge, an erosion profile develops (Figure 4.3). The level of the profile is determined by the maximum water level during the storm surge. Its shape, perpendicular to the coast, is determined by the wave height and the fall velocity of the bed material and can be described by the following equation (Vellinga, 1986):

$$\left(\frac{7.6}{H_S}\right) Y = 0.47 \left[\left(\frac{7.6}{H_S}\right)^{1.28} \left(\frac{w}{0.0268}\right)^{0.56} X + 18 \right]^{0.5} - 2.00 \qquad (4.1)$$

X is the distance (in metres, positive seawards) from the new dune foot and Y is the depth (in metres) below maximum water level during storm surge.
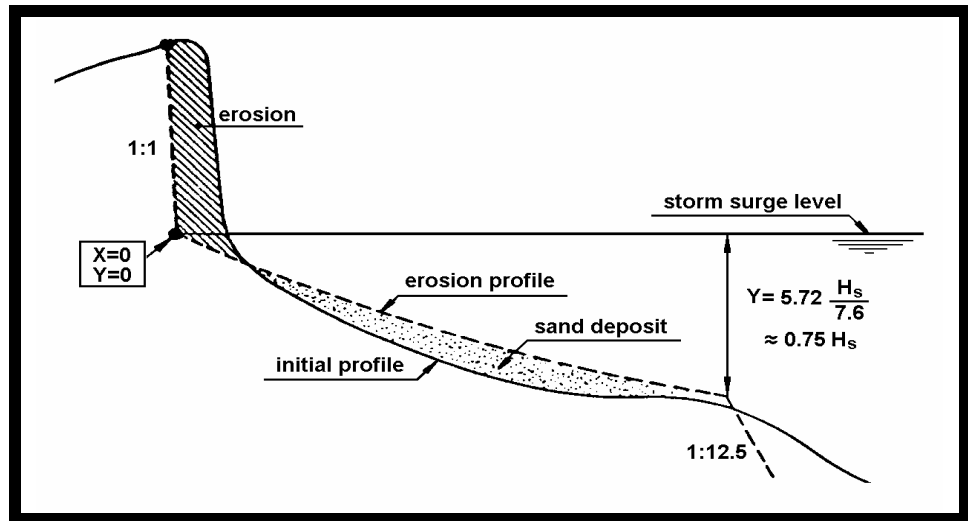
Figure 4.3: Representation of Vellinga's profile after a storm surge (modified after Van de Graaff & Koster, 1990).

- Eq. (4.1) is valid up to a seaward limit (X_{max}, Y_{max}) given by:

$$X_{max} = 250 \left(\frac{H_S}{7.6}\right)^{1.28} \left(\frac{0.0268}{w}\right)^{0.56}$$

$$Y_{max} = 5.72 \left(\frac{H_S}{7.6}\right) \approx 0.75 H_S$$

(4.2)

  Seaward from this point, the profile continues as a straight line with a gradient of 1:12.5 (fixed in agreement with model tests) until it intersects the initial profile. The slope of the dune face ($X<0$) is at 1:1 (consistent with field observations and large scale tests).

- After determining the shape of the erosion profile for a given set of parameters, this profile must be shifted in relation to the initial profile (the profile before the storm) in such a way that erosion is in balance with accretion. Transport of sand is in the seaward direction and there is no provision for handling longshore gains or losses to the profile.

- The erosion quantity is determined by the difference between the initial profile and Vellinga´s erosion profile.

- The outputs from the model are (Vellinga, 1983): i) the recession of the dune front; ii) the erosion quantity above storm surge level; and iii) the beach profile after the storm surge.

Vellinga's model accounts for the following factors in predicting dune erosion (Van de Graaff, 1983, 1986):

- maximum water level during storm surge;

- significant deep water wave height during the maximum of the surge;

- particle diameter of dune material;

- shape of initial profile (including dune height);

- storm surge duration;

- gust bumps and squall oscillations;

- accuracy of the computation method.

The model disregards the influences of:

- temperature of the sea water (which affects w in eq. (4.1));

- irregularities in cross-sections over small distances alongshore;

- storm surge direction;

- groins;

- dune vegetation.

4.2.3 Verification From Laboratory And Field Data

The computational model was verified using both large scale tests and tests with a depth scale n_d=30. The effects of storm surge level, significant wave height, significant wave period, shape of the spectrum, dune height and initial profile were all checked. It was concluded that (Vellinga, 1983):

- The computational model accounts adequately for the impacts of storm surge level, wave height and profile shape.

- The model is valid for typical North Sea storm surge conditions (Figure 4.4) involving waves with steepness $H_s / L_{os} \geq 0.02$. Erosion during such an event is equivalent to the erosion which takes place during a period of about 5 hours when the water level is held constant at the maximum surge level (Van de Graaff, 1983, 1986; Vellinga, 1986). Storm surges with different hydrographs may have different equivalent durations. A correction factor for a

different hydraulic regime has been determined on the basis of the model tests with constant water level. The erosion quantity above storm surge level, C (see Figure 4.10 later), should be increased by 5% to 10% for each additional hour of storm duration (maximum addition not to exceed 50% of C). Storm duration is defined as the amount of time in hours that the surge level is within 1m of the maximum level (5 hours for the North Sea hydrograph). For example, if the storm duration is 9 hours and assuming a conservative 10% additional erosion per additional hour, then a total of 10(9-5)=40% additional erosion should be considered. Since this is less than the maximum recommended addition of 50%, the value of C adjusted to account for the duration of the storm can be computed as 1.4C (Sargent & Birkemeier, 1985).
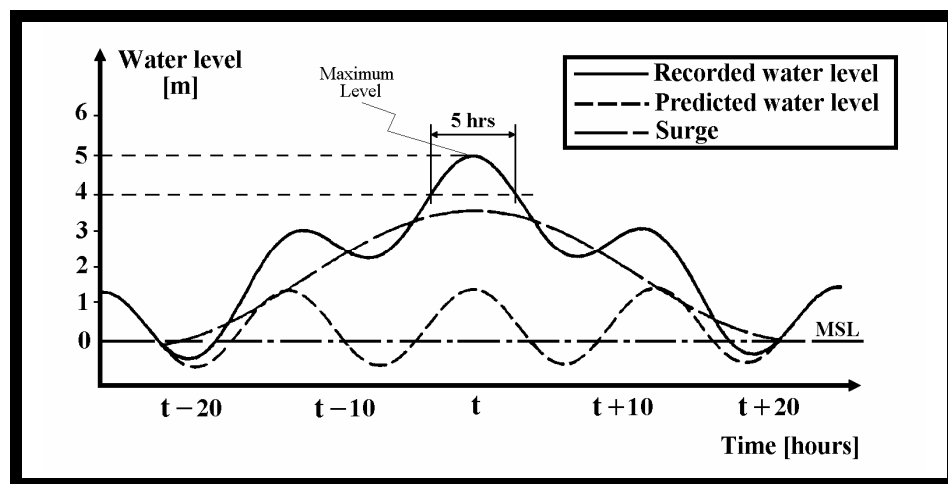
Figure 4.4: Example of the standard North Sea storm surge hydrograph (modified after Vellinga, 1986).

- The prediction model is somewhat conservative for beaches with large bars and troughs, but it should be considered reasonably accurate for a large range of hydraulic conditions and initial profiles normally found in the field (Vellinga, 1986).

The computational model has also been verified using field measurements from the 1953 and 1976 storm surges in The Netherlands and from Hurricane Eloise in Walton County, Florida in 1975 (Hughes & Chiu, 1981). In addition, Sargent & Birkemeier (1985) verified the model for a number of storm surges along the USA East coast and the Gulf coast.

4.2.4   Accuracy

Vellinga's model is a relatively simple way of schematising a complicated natural process. Its accuracy was established in the following way (Vellinga, 1983). First, all measurements were considered 100% true. Next, the distribution of the differences between the measured and computed quantities was evaluated. It was concluded that the accuracy of the computational model could be described by a Normal distribution with a mean of zero and a standard deviation $\sigma = 0.10\,C + 20$ m³/m, in which C is the computed erosion above storm surge level in m³/m and σ is the standard deviation of the differences between the computed and measured dune erosion quantities (Figure 4.5). This relationship describes the accuracy of the prediction model for given input parameters (such as initial profile, maximum surge level, wave height and particle size diameter). The inaccuracy of these input parameters is an additional source of errors. The effects of gust bumps, squall oscillations and the duration of the storm are not included in the model, although Vellinga identified these factors as important (see Section 4.2.2) and addressed the problem to some degree (see Sections 4.5.2 and 4.5.3). Gradients in the longshore transport are also ignored in the model.
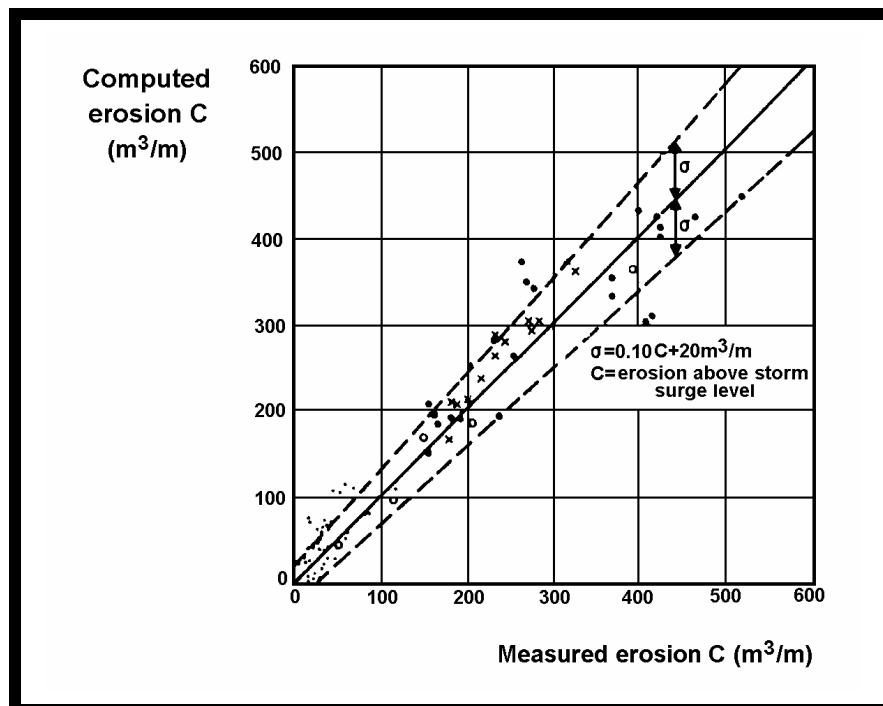
Figure 4.5: Comparison between measurements and computations (modified after Vellinga, 1983).

4.2.5   Application And Limitations

Vellinga´s model has been applied to check the safety of existing dunes in The Netherlands and to determine the required nourishment. However, the model is based upon certain assumptions and has limitations of which the user should be aware. The assumptions have been described in Section 4.2.2. The model's limitations are described here.

According to Vellinga (1986), the model is only valid formally for waves with $H_S/L_{OS}=0.034$ and for conditions with a constant water level for 5 hours. However, in practice, the model has been applied to conditions with $0.02 \leq H_S/L_{OS} \leq 0.04$ and for a realistic North Sea storm surge hydrograph as shown in Figure 4.4. This hydrograph is characterised by its large height and short duration. However, with minor adjustments for the effect of storm surge duration, the model can also be used for the prediction of beach and dune erosion in other parts of the world (Vellinga, 1983).

The model also has some limitations related to cross-shore profiles. The model results have not always proved to be reliable (Steetzel, 1993) for an initial profile with a nearshore bank or a very steep bottom slope (owing, perhaps, to the presence of a tidal gully). Furthermore, Vellinga's erosion profile does not deal with beaches which have a wide range of grain sizes such as occur when there is a mixture of sand and shingle. Such cases exist, for example, along parts of the UK coastline (Simm et al, 1996). One should also note that the model is only applicable in situations with relatively straight homogeneous coastlines, i.e. where a two-dimensional idealisation of the dune erosion process is possible (Vellinga, 1986). Extra dune erosion should be expected where the shoreline is curved, to compensate for possible losses of sand due to gradients in the longshore transport rate. Furthermore, some coastal problems cannot adequately be solved using this kind of dune erosion model (Table 4.2).

TYPE OF PROBLEM	RELATED EXAMPLES
non-standard coastal profiles or hydraulic conditions	- nearshore banks - tidal gullies - non-standard surge conditions
interference by structures	- dune revetments - offshore breakwaters
effect of longshore transport gradients	- shoreline curvature - tidal gradients

Table 4.2: Limitations of Vellinga's model (modified after Vellinga, 1986).

4.3   Steetzel's Time-Dependent Model

Although Vellinga's model provides a fair estimate of the amount of dune erosion (Steetzel, 1993), a number of problems remain (see Section 4.2.5). To overcome some of the limitations, the Technical Advisory Committee on Water Defences (TAW) had a more sophisticated model developed. It is based on the vertical water velocity and sediment concentration profiles. Erosion is then calculated as a function of time. The main goal of the new model was to quantify the amount of dune erosion for arbitrary profiles (e.g. with bars and tidal gullies) and arbitrary hydraulic conditions. This model, developed by Steetzel, is known as the DUROSTA model (Delft Hydraulics, 1991). Existing data on storm-induced profile changes were re-analysed and some additional investigations were carried out. The overall performance of the model was good (Steetzel, 1993).

4.4   Current Application Of Vellinga's And Steetzel's Models

Currently, the safety of the Dutch dunes is determined by the use of a design method (seen as Level I) which is based on Vellinga´s equilibrium profile model (CUR-TAW, 1989). Probabilistic calculations at higher levels are only made on a few occasions when there is some particular need for them.

Steetzel´s time-dependent model is more sophisticated. At present, it is applied by specialists only. Nevertheless, it is expected eventually to replace

the current procedures. But, owing to its complexity, much still remains to be done before it can be successfully combined with probabilistic calculations. Such a task is beyond the scope of the present study.

4.5   Two Computer Programs Which Use Vellinga's Model

4.5.1   Introduction

Many parameters are important in the design of dune nourishment. Furthermore, the precise values of these parameters, both before and during a surge, are uncertain and can vary in time and space. Thus, a probabilistic design approach seems appropriate.

The occurrence of future storm surges accompanied by dune erosion is a highly stochastic process (Van de Graaff, 1986). At best, one is able to predict the probability of occurrence of a certain set of extreme conditions during a certain storm season or in a certain year. Such predictions are based on long-term observations and/or simulations of the process. Using probabilistic methods and taking into account the transfer function between the surge conditions and the amount of dune erosion, one is eventually able to determine the probability of exceedance per year of particular dune retreats (Figure 4.6).
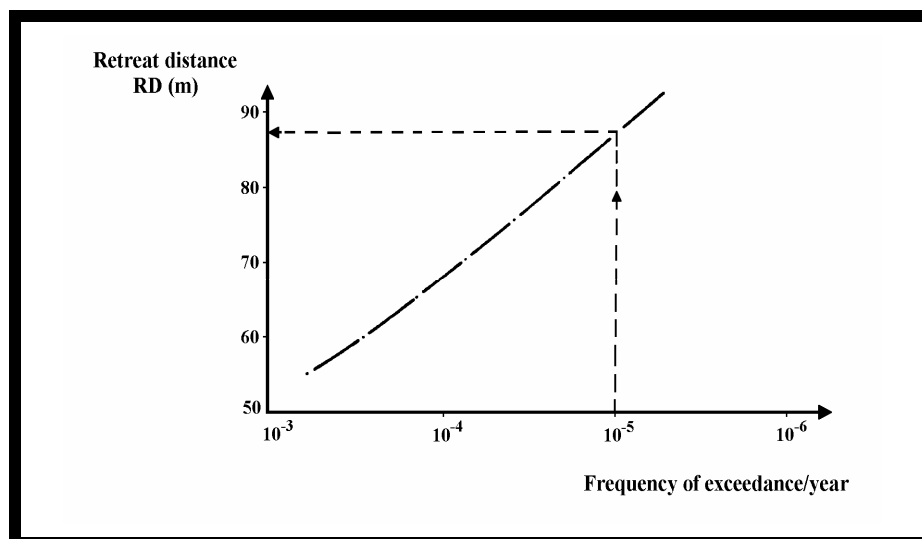
Figure 4.6: Erosion as a function of frequency of exceedance (modified after Van de Graaff, 1986).

An acceptable risk of failure of the dunes has to be established. The level of acceptability depends mainly on the likely consequences. For example, a very low chance of failure is demanded in The Netherlands, due to the high importance associated with the low-lying hinterland. There, a probability of failure of no more than 10^{-5}/year (a return period of 100000 years) is expected (Van de Graaff, 1983, 1986). That is, a dune system is assumed safe in The Netherlands only if it is wide enough to withstand erosion with a chance of exceedance of no more than 10^{-5}/year. However, a situation like that shown in Figure 4.1(b) might permit the allowable probability of a retreat beyond point R to be far larger (10^{-2}/year or 10^{-3}/year) due to the relatively minor importance of the threatened area.

Predicted dune erosion during a surge depends on the values of the seven determining parameters in Vellinga's model (see Section 4.2.2). Knowing the density functions of the parameters, the probability of occurrence of a set of particular values can be computed using a Level III approach. A probability of exceedance curve of properties such as erosion distance, RD, can then be found by integration (Van de Graaff, 1986). However, the number of computations is enormous when seven parameters are involved. Due to the mass balancing procedure required in calculating the erosion volume, the computation time for even one calculation is very long. Thus, a large number of computations is not an attractive proposition. In The Netherlands, a Level III method has been applied only for test purposes. In these tests, large integration steps were adopted.

Another problem with Level III methods is that it is impossible to gain insight into the relative importance of the parameters involved. This shortcoming, together with the number of computations needed, is the main reason why Level III methods have not been used frequently. Furthermore, the results obtained hardly differ from those of a Level II approach (Van de Graaff, 1983). Probabilistic calculations on a personal computer then become possible.

In The Netherlands, Vellinga's model has been used as the basis for the development of two main computer programs:

- a Level II probabilistic program (**DUNEPROB**);
- a Level I simplified calculation program (**DUNE**).

The two programs are introduced here. Appendix B contains an example of their use and a comparison of results.

4.5.2   DUNEPROB

DUNEPROB was developed by Koster Engineering. It calculates the probability that erosion will exceed a certain distance, assuming that the eroded sand is transported only in a seaward direction. For this purpose, the probabilistic FORM is used. In other words, given a certain target X-coordinate, X_T (e.g. the location of a building close to the sea), the program calculates the probability that erosion occurs such that X_R in Figure 4.7 is landward of X_T.
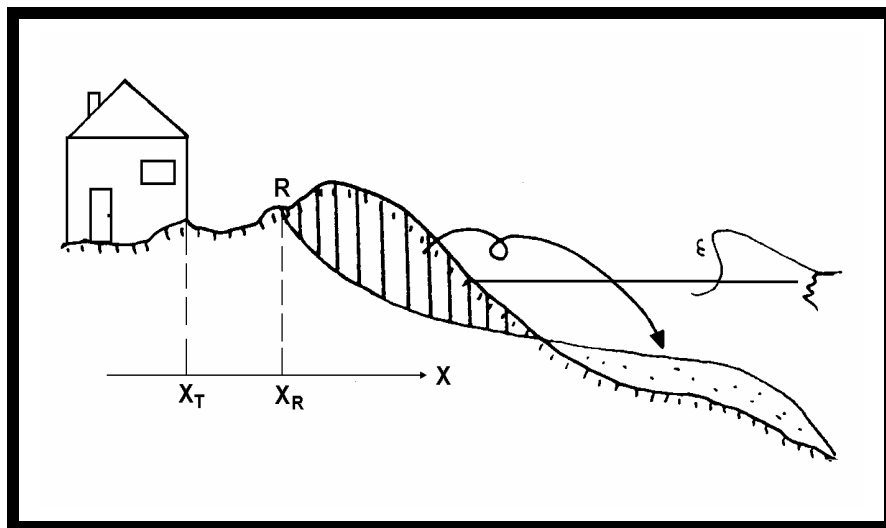
Figure 4.7: Schematic representation of the dune erosion problem (modified after Van de Graaff & Koster, 1990).

As noted earlier, a reliability function Z is defined in FORM calculations such that $Z \leq 0$ represents failure of the system. In this case, the following reliability function meets this requirement:

$$Z = X_R - X_T \qquad\qquad (4.3)$$

X_R is the X-coordinate of point R and depends on the seven parameters affecting the erosion profile according to Vellinga. The main characteristics of these seven parameters, as used in DUNEPROB, are described next.

4.5.2.1 Maximum Water Level During Storm Surge

The total water level reached during a storm depends mainly upon (Van de Graaff, 1983, 1986; Pugh, 1987; Thomas & Hall, 1992): (i) the astronomical tide; and (ii) the wind and wave setup and low atmospheric pressure associated with meteorological forces (see Figure 4.8). Accounting for these phenomena, it is possible to derive the resulting frequency of exceedance curve of an arbitrary maximum water level during storm surge (Vrijling & Bruinsma, 1980). The Delta Committee (1960) presented similar frequency of exceedance curves, based largely on extrapolation of historical data, for locations along the Dutch coast. As the Vrijling & Bruinsma curves do not differ essentially from the Delta Committee curves, the latter are used here.

The probability of exceedance curve for the maximum water level during storm surge in metres above datum (NAP under Dutch conditions) can be described by an Exponential distribution as follows (Van de Graaff & Koster, 1990):

$$1 - F_h = \exp\left(-\frac{h - \zeta}{\lambda}\right) \qquad (4.4)$$

ζ and λ are parameters depending on site location along the Dutch coast. DUNEPROB assumes this distribution for h and requires ζ and λ as input parameters.

4.5.2.2 Significant Wave Height During The Storm Surge

Water levels and wave heights along a coast are related. Figure 4.8 shows why. Wind blowing over water exerts a shear stress which may pile water against the coast. Waves induced by the wind add a further setup in the water surface as they break. The effects are enhanced by low atmospheric

pressure causing a general rise in the sea surface. Water levels and wave heights may be strongly or weakly related depending upon the location. Wind speeds, directions and durations are all important determining factors, as is the tidal range.
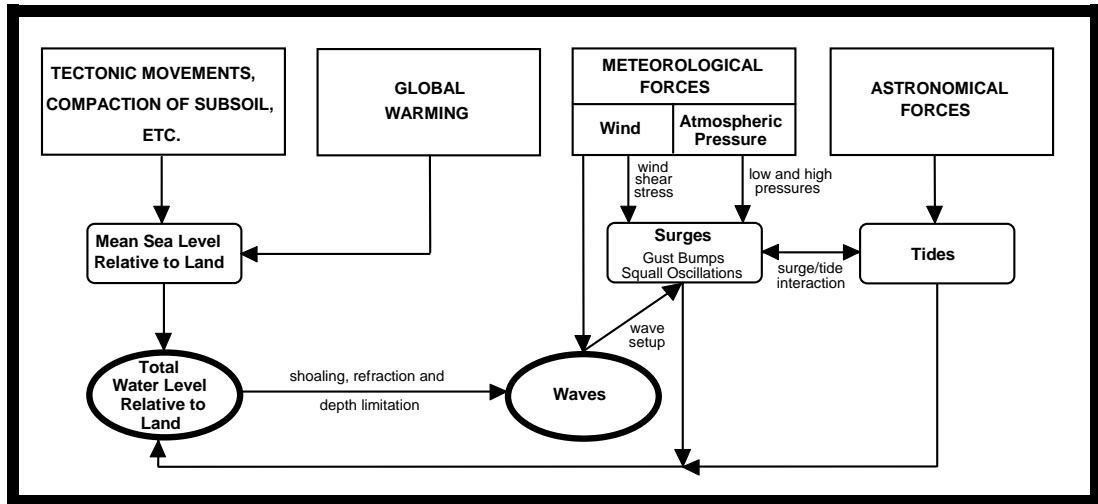
Figure 4.8: Major relationships between waves and total water level.

Vrijling & Bruinsma (1980) studied the joint distribution of water levels and wave heights in establishing the boundary conditions for the Oosterschelde storm surge barrier in The Netherlands. Van Aalst (1983) derived the maximum water level during storm surge versus significant wave height relationships for various locations along the Dutch coast (Figure 4.9). The given significant wave height represents the mean value, μ. For each location, a standard deviation, σ, is also shown. DUNEPROB uses the following expressions to describe the statistical distribution of the deep water significant wave height as a function of the water level during storm surge:

$$\mu_{H_S|h} = ah^b \qquad (4.5)$$

where $\mu_{H_S|h}$ is the mean value of H_S given a value for h; a and b are coefficients that depend on the location and

$$H_S|h = \mu_{H_S|h} + H_{S\_}\,Inacc \qquad (4.6)$$

H_S_Inacc is a Normal random variable having a zero mean and a standard deviation, $\sigma_{H_S\_Inacc}$, which is usually set to 0.6 or 0.75 for The Netherlands, depending on the site location (Van de Graaff, 1986).
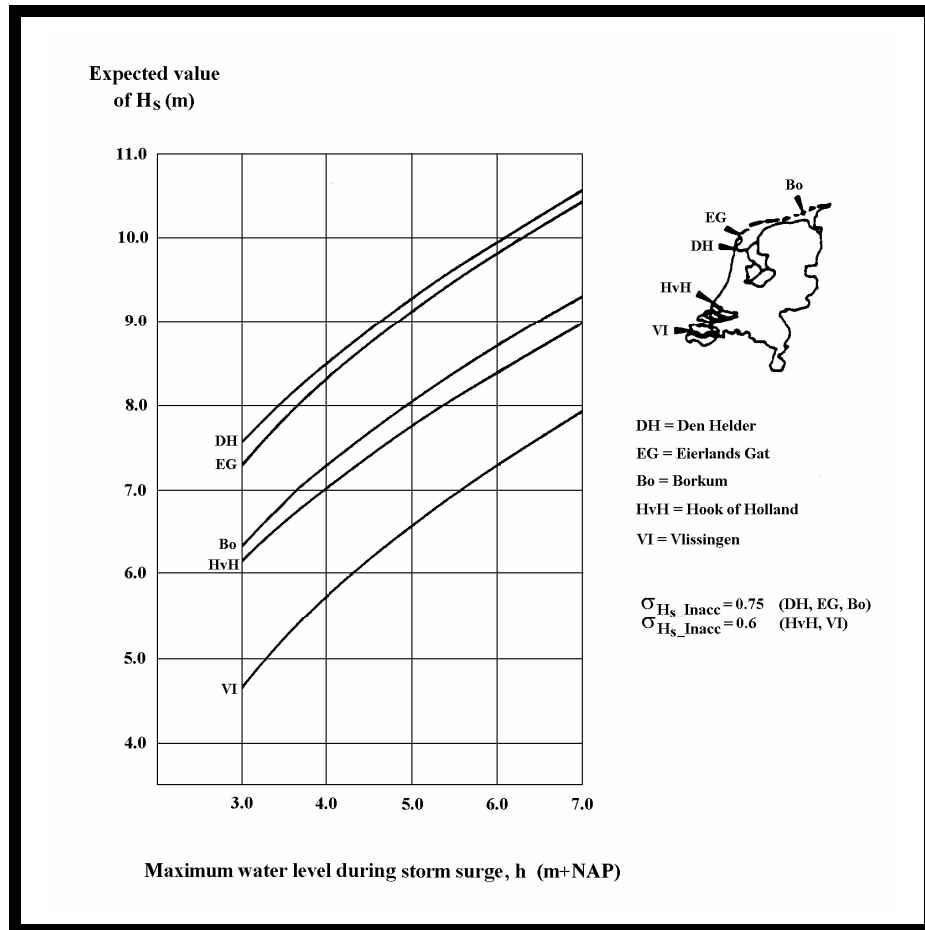
Figure 4.9: Expected value of H_S as a function of h at locations along the Dutch coast (modified after Van de Graaff, 1986).

There are situations where the expected value for H_S, according to eq. (4.5), might be too high if the wave conditions are depth limited. For this reason, an upper limit, $H_{wavemax}$, can be introduced in the program.

4.5.2.3 Particle Diameter Of Dune Material

As noted earlier, the amount of dune erosion depends on the grain size of the sand (via its fall velocity w). According to CUR-TAW (1989), the fall velocity for the period of the year during which storm surges can be expected

in The Netherlands, should be calculated for a salt water temperature of 5°C. It can be approximated by:

$$\log\left(\frac{1}{w}\right) = 0.476\ (\log D_{50})^2\ +\ 2.18\ (\log D_{50})\ +\ 3.226 \qquad (4.7)$$

where D_{50} and w are expressed in SI units.

Kohsiek (1984) analysed samples taken from different locations along the coast and determined for each location a mean value and a standard deviation for D_{50} (Normal distribution). The ratio $\sigma_{D_{50}}/\mu_{D_{50}}$ varied from location to location with values from 0.01 to 0.15.

4.5.2.4 Shape Of Initial Profile

Two different cases have to be considered when analysing the influence of the shape of the initial beach profile on the resulting dune erosion (Van de Graaff, 1983): (i) there is a stable profile; and (ii) there is an unstable (long-term eroding) profile. Even a stable profile varies from day to day and from season to season. One single profile measurement represents only one sample from some distribution. Assuming a Normal distribution, the mean profile, μ_{DP}, and the standard deviation of the profile, σ_{DP}, can be calculated when a series of measurements is available. Note that μ_{DP} and σ_{DP} are expressed in terms of units of m³/m above a datum and not in metres as is DP in PARASODE.

4.5.2.5 Storm Surge Duration

Storm surge duration is not a parameter in eq. (4.1). In reality, dune erosion is time-dependent and the time during which the water level is near the maximum surge level is one of the main factors determining the amount of erosion (Van de Graaff, 1983, 1986). Vellinga carried out two sets of tests to investigate the matter: a) tests with a constant maximum surge level; and b) tests with an actual hydrograph (see Figure 4.4). Note that the shape of this hydrograph was only one possibility out of a number of alternatives, all with the same maximum water level (Van de Graaff, 1986). From the tests, it was concluded that the normal result of an erosion computation (volume C

above surge level) can be assumed valid for a surge duration of 5 hours. Variations in the storm surge duration around this value can be represented as an additional erosion above storm surge level having a Normal distribution and the following characterising parameters: $\mu = 0$ and $\sigma = 0.1\,C$ (Van de Graaff, 1983, 1984, 1986).

4.5.2.6 Gust Bumps And Squall Oscillations

Squall oscillations are periodic, local variations in the water level during a storm surge with an amplitude of the order of 0.20m and a period of the order of 45min (Van de Graaff, 1986). Gust bumps are short-term increases in water level caused by the passage of a front or a heavy shower. In contrast to squall oscillations, gust bumps can be traced over a large area. Their amplitude is about 0.40m and they have a period of about 60min (Van de Graaff, 1986). The occurrence of both phenomena is highly variable. In recording storm surges, the effects of these irregularities, which mostly increase maximum water level, are neglected in Dutch practice. Actual hydrographs are smoothed and the maximum values are stored. Equations such as eq. (4.4) represent the smoothed curves. However, the amount of dune erosion is highly dependent on the maximum water level and any rise, even for a short time, results in an increase in the volume of erosion (Van de Graaff, 1983).

A short-term increase, Δh, in water level above the smoothed peak will lead to an increase, ΔC, in the volume of erosion. Van de Graaff (1984) argued that ΔC is approximately $0.5\,\Delta SH$, ΔSH being the increased amount of erosion due to a smoothed hydrograph with a maximum Δh higher:

$$\Delta C = 0.05\,C\left[\frac{\Delta h}{0.40}\right] \tag{4.8}$$

where C is the eroded volume above the maximum water level which is calculated ignoring gust bumps and squall oscillations. Generally, the effects of these phenomena can be represented as an additional erosion above storm surge level having a Normal distribution and the following parameters: $\mu = 0.05\,C$ and $\sigma = 0.0125\,C$ (Van de Graaff, 1984).

4.5.2.7 Accuracy Of The Computation Method

The accuracy of the computation method is taken into account as explained in Section 4.2.4 (assuming a Normal distribution with a zero mean and a standard deviation, $\sigma = 0.10\,C + 20\,m^3/m$).

4.5.2.8 Surcharge

Variations in the last three parameters listed above can be combined to form one new single variable, Surcharge, having a Normal distribution with the following characteristics: $\mu_{\text{Surcharge}} = 0.05\,C$, $\sigma_{\text{Surcharge}} = \sqrt{(0.1C)^2 + (0.0125\,C)^2 + (0.10\,C + 20)^2}$. The effect of this surcharge is expressed in an additional recession of the dune front corresponding to an amount of erosion referred to as SurchEros (Figure 4.10).
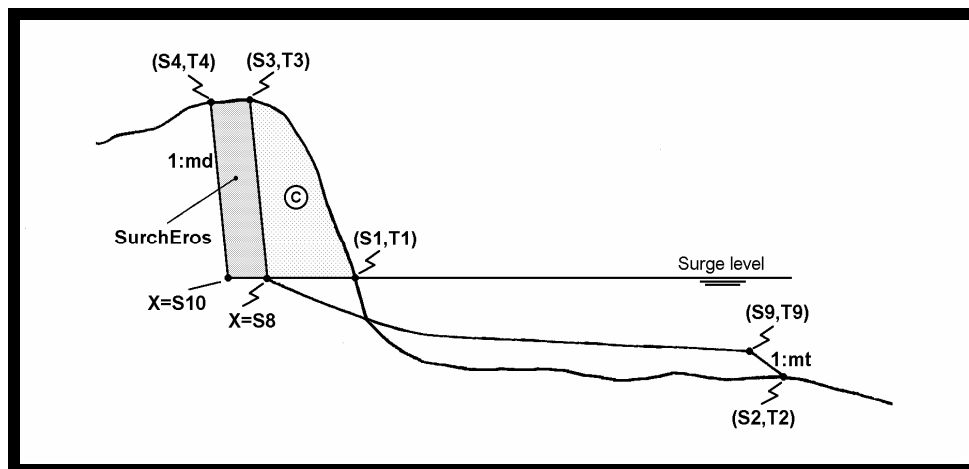
Figure 4.10: Surcharge on erosion area C above surge level.

Note that in DUNEPROB, the variable Surcharge has a slightly different meaning from that described here: it is a surcharge coefficient. This coefficient follows a Normal distribution and its mean and standard deviation are often approximated by $\mu = 0$ and $\sigma = 1$, respectively.

4.5.3 DUNE

At present, the probabilistic approach to dune erosion computations is still thought of as rather complicated for "everyday" computations by coastal managers. That is the reason why a simplified calculation method has been introduced in The Netherlands, based on the results of extended probabilistic calculations. This method can be seen as a Level I approach. It is presented in the "Guide to the Assessment of the Safety of Dunes as a Sea Defence", CUR-TAW (1989). In the Level I approach, it is necessary to account only for possible variations in D_{50} and in the initial profile for a specified surge level and significant wave height. Calculations are based on the assumption that the eroded sand is transported only in a seaward direction (CUR-TAW, 1989). Table 4.3 gives the set of characteristic values which yield, within 2% accuracy, the retreat distance expected with a 10^{-5}/year frequency of exceedance. If one needs to consider a higher chance of failure (e.g. 10^{-4}/year or 10^{-3}/year), adopting a maximum water level during surge, occurring with a probability of exceedance which is a factor 2.15 times the acceptable chance of failure, and maintaining the other parameters according to Table 4.3, a rather accurate approximation is found (Van de Graaff, 1986).

PARAMETER	CHARACTERISTIC VALUE
Maximum water level during surge	Value with a frequency of exceedance of 2.15×10^{-5}/year
Significant deep water wave height	Expected value for the given water level (Figure 4.9)
Diameter	$D_{50} = \mu_{D_{50}} \left(1 - 5\sigma_{D_{50}}^2 / \mu_{D_{50}}^2\right)$. Holds at least for $0\% < \sigma 100\% / \mu < 12.5\%$
Change in the initial profile	Profile containing $\sigma_{DP}^2 / 275\, \text{m}^3/\text{m}$ "less sand" than the average profile. Holds at least for $0 < \sigma_{DP} < 150\, \text{m}^3/\text{m}$ (σ_{DP} is the standard deviation of the volume about the mean initial profile)
Surge duration	$0.1C\, \text{m}^3/\text{m}$ addition (C being the volume of sand eroded above the maximum water level during surge, using the parameters defined as above)
Gust bumps and squall oscillations	$0.05C\, \text{m}^3/\text{m}$ addition
Accuracy of computation method	$(0.1C+20)\, \text{m}^3/\text{m}$ addition

Table 4.3: Characteristic parameter values (modified after Van de Graaff, 1986).

The above simplified calculation method has been implemented in a computer program called DUNE. This program has also been developed by Koster Engineering. Appendix B includes an example of its use.

Note that wave period is not an explicit parameter in the above table or in Vellinga's model. However, DUNE uses the peak period of the wave spectrum to calculate a limiting profile defined by the minimum necessary dune crest level and the minimum width at this level, together with an acceptable maximum backslope. The erosion profile must remain seaward of this limiting profile (CUR-TAW, 1989).

DUNE may also account for situations in which there is net loss of sand from the profile owing to a gradient in the longshore transport rate. The loss, G (m^3/m), for coastal sections of low or moderate curvature may be calculated as follows (CUR-TAW, 1989):

$$G = \frac{(C + \text{SurchEros})}{300} \left(\frac{H_S}{7.6}\right)^{0.72} \left(\frac{w}{0.0268}\right)^{0.56} G_O \qquad (4.9)$$

where G_o is a reference value for G (m^3/m) and is tabulated below:

Curvature (degrees/1000 m)	G_O (m^3/m)
0 - 6	0
6 - 12	50
12 - 18	75
18 - 24	100
> 24	further investigation

Table 4.4: G_o values (modified after CUR-TAW, 1989).

The result of calculating G may be expressed as an additional recession of the dune front above maximum water level.

4.5.4 Applicability Of The Programs In The British Context

The programs DUNE and DUNEPROB are based on the dune erosion profile determined by Vellinga, combined with probabilistic calculations. The program DUNE calculates the erosion profile after a design storm corresponding to a probability of exceedance of approximately 10^{-5}/year. This extremely low probability is justified for Dutch conditions (see Section 4.5.1) but it is not always appropriate. This drawback can be overcome as described in Section 4.5.3 or by using DUNEPROB.

Both programs relate the significant wave height to the maximum water level during storm surge (see Figure 4.9). If the expected values of H_S as a function of h do not follow a similar trend, then the direct application of DUNE or DUNEPROB will produce erroneous results. This is the case on the Sefton coast, UK (see Section 6.2.2), where there is little correlation between extreme wave heights and extreme water levels (Hawkes & Hague, 1994). For this reason, it was decided to introduce Vellinga's model and some of the features of DUNEPROB and DUNE into the Level II computer program, PARASODE, developed as part of this research (Chapter 5). Probabilistic calculations can be performed which allow for appropriate combinations of wave heights and storm surge levels. Some examples of the application of PARASODE to dune erosion are presented in Section 6.2 of this thesis.

4.6 Summary

Dutch experience with regard to the probabilistic design of dunes has been examined. The computational model currently used throughout The Netherlands is based on Vellinga's equilibrium profile model. The more sophisticated time-dependent model developed by Steetzel is not yet used as the basis for probabilistic calculations.

The Dutch programs are not directly applicable to conditions along coasts such as that in Sefton, UK, where there is a much weaker correlation than in The Netherlands between wave heights and water levels. Consequently, it was decided to introduce Vellinga's model and some features of the Dutch programs into PARASODE, and to carry out new probabilistic calculations.

5 SOFTWARE APPLIED IN THIS RESEARCH FOR UNDERTAKING PROBABILISTIC CALCULATIONS

5.1 Introduction

This chapter describes the software used in this research to perform probabilistic calculations on wave overtopping of seawalls and dune erosion. The main element of software, called PARASODE, has been developed as part of this work. It is based upon the Level II probabilistic methods reviewed in Chapter 2 and the formulation of the failure modes provided in Chapters 3 and 4.

Section 5.2.1 gives a general description of PARASODE. Section 5.2.2 explains the parameters in the program which control each FORM calculation. Section 5.2.3 illustrates how truncation of probability distributions has been implemented in PARASODE. Sections 5.2.4 and 5.2.5 describe, respectively, the incorporation of wave overtopping of seawalls and dune erosion into the program. Finally, Sections 5.2.6 and 5.2.7 refer to the input and the output of PARASODE, respectively.

To validate the results from PARASODE, various Level III calculations have been performed using the commercial software package @Risk, acquired from PALISADE Corporation. This program is briefly introduced in Section 5.3. For further details about the program, the reader is referred to its manual (Palisade Corporation, 1994).

5.2 PARASODE

5.2.1 General Description

PARASODE (Probabilistic Assessment of Risks Associated with Seawall Overtopping and Dune Erosion) has been developed for assessing the safety of coastal structures. In particular, as the name suggests, it concentrates on the potential failure mechanisms associated with wave overtopping of seawalls and dune erosion. The amount of wave overtopping is calculated by both the H&R equation and Owen´s formula. Dune erosion is calculated

using Vellinga´s model. Note that incorporation of dune erosion into PARASODE presented a more complex task than including the overtopping models. Additional problems arise because dune erosion cannot be described by an explicit failure function of the basic variables. Details of dune erosion calculations are given in Appendix C5.

Although the program incorporates two specific failure mechanisms, the majority of the code is generic and can be applied with minor adjustments to other types of failure.

PARASODE operates in two ways (see Figure. 5.1):

- MODE 1, the analysis mode, in which the failure probability is calculated for a given value of the design parameter, e.g. the crest level of a seawall.

- MODE 2, the design mode, in which the value of a specific design parameter is calculated for a target probability of failure.
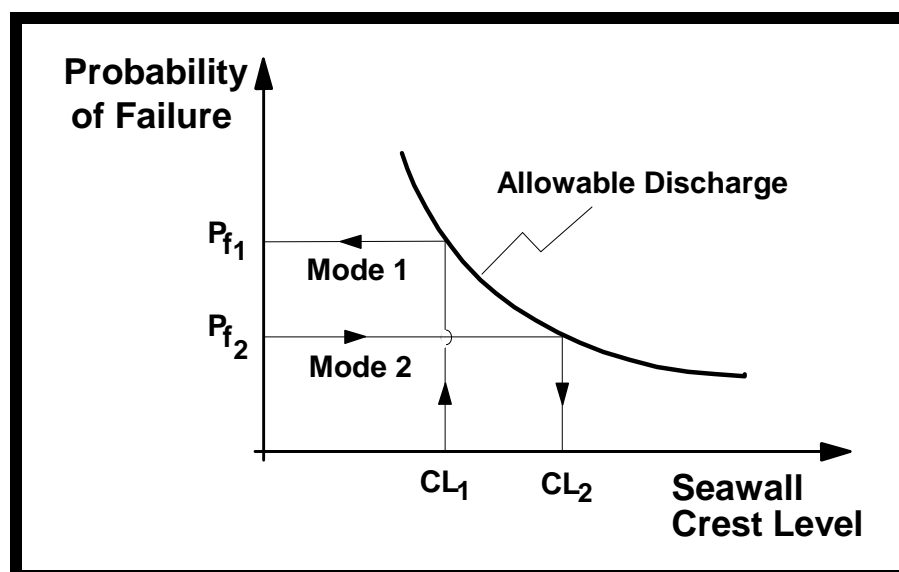
Figure 5.1: Illustration of Mode 1 and Mode 2 for failures resulting from overtopping.

Mode 1 allows for combinations of time-varying actions using the method of Ferry Borges & Castanheta (1983). This method could also have been implemented in Mode 2. However, implementation is complex and computational time would be considerable. Instead, Mode 2 results involving

combinations of actions are best provided by running the program in Mode 1 for several different values of the design parameter, producing a design curve as shown in Figure 5.1. The answers required from Mode 2 calculations can then be obtained from the curve. Appendix C1 shows simplified flowcharts of the program for the analysis and design modes, respectively.

The program uses the Level II First Order Reliability Method - FORM (see Sections 2.3.3 and 5.2.2). It incorporates routines for transforming the correlated variables to a set of non-correlated variables and for mapping non-Normal distributions to equivalent Normal distributions. There are ten continuous pre-defined statistical distributions programmed in PARASODE. Each distribution may be truncated either on the left or on the right side (see Section 5.2.3). Details of these distributions (Law & Kelton, 1991; Evans et al, 1993) are tabulated in Appendix C3. In PARASODE, the user can also add his or her own distributions (see Appendices C4 and C6 for more details). This facility is of particular help when the distribution of a variable, X, is the result of measurements which are not easily fitted by a pre-defined distribution. At present, PARASODE has three user-defined distributions: i) observed water levels at Liverpool; ii) observed extreme water levels at Liverpool; and iii) predicted tide levels at Liverpool (see Appendix C4). They are used in the examples in Chapter 6.

PARASODE has been written in FORTRAN 77. General references on the FORTRAN language are Koffman & Friedman (1987), Davis & Hoffman (1988) and Etter (1992). Pre-defined subroutines have been used in some cases. These subroutines have been extracted from Press et al (1992) and NAG (1993).

Appendix C2 shows the various subroutines used in PARASODE. The program listing is provided in Appendix C7. It contains a description of the subroutines, the variables used in each of them are listed and described, and there are guiding comments throughout the code. SI units are used within the program, except if otherwise specified.

Appendix C6 provides a detailed list of the input files required and their contents. Examples of input and output files are provided in Appendices D1 and D3.

Before applying PARASODE to wave overtopping of seawalls and dune erosion, it was tested using examples reported in literature such as Thoft-Christensen & Baker (1982), Ang & Tang (1984), Thoft-Christensen & Murotsu (1986), Madsen et al (1986), Smith (1986), Van der Meer (1987), CUR-TAW (1990), Pilarczyk (1990), and Burcharth (1992). Examples run with other computer programs like PROBA2 (Delft Hydraulics, undated) and Super-Risk (Super-Software, 1994) were also reproduced. The results from PARASODE were always highly satisfactory.

5.2.2 FORM In PARASODE

The parameters controlling each FORM calculation have to be specified in the input file *form.dad.* These parameters are described below.

5.2.2.1 Starting Point

The iteration process needs starting values for the design point. It is common to choose the mean value of each variable: in other words, iteration starts by using the mean value approach. Sometimes, there are reasons to start computations at another point. For example, a solution might be found only by specifying another starting point. Also, in a case where there is more than one solution to the problem, one might find other solutions by trying other starting points (see, for example, Wen & Chen, 1987).

When no combinations of actions are involved, or combinations of actions are considered and the modified distributions are provided (see Section 2.3.3.3), PARASODE uses the mean value starting point, unless otherwise specified by the user. If combinations of actions are considered and the basic distributions are given, then the starting values for the design point are obtained as follows:

- If the power, NR, to which the distribution of the variable is raised is 1 then the starting value is the mean value of the variable.

- If the power, NR, to which the distribution of the variable is raised is not 1, then the starting value corresponds to an extreme cumulative distribution function value of 0.5 ($X = F_X^{-1}[0.5^{1/NR}]$ where F_X^{-1} is the inverse of F_X evaluated at $0.5^{1/NR}$).

5.2.2.2 Minimum And Maximum Values

Variables which have a physical meaning may be restricted within particular limits during the iteration process. Also, the form of the failure function may prevent variables taking certain values (e.g. a variable which is raised to the power 0.5 cannot take negative values). In PARASODE the user can either accept the default minimum and maximum values (XMin=-1E25; XMax=1E25) or specify required values. However, if the user's limits exceed the boundaries defined previously by the variable's distribution, the program adopts the more limiting boundaries. So, for example, if a variable follows a Log-Normal distribution, then X>0; if the user inputs XMin=-10, the program neglects this latter value and adopts XMin=1E-25.

During the iteration process, if a variable X lies outside the boundaries (XMin, XMax), then the program continues calculations using a new value of X between the last computed value and the boundary which was exceeded. This procedure gives final results where otherwise the program would fail.

5.2.2.3 Number Of Iterations

In a FORM calculation, the design point can only be found by iteration. The number of iterations required depends on the failure function (the more linear the function is, the faster the iteration procedure converges), on the point used to start the iteration process, on the required relative accuracy of the reliability index, on the iteration smoothing process, and on the required accuracy of Z being zero.

The maximum number of iterations in a FORM calculation is designated in PARASODE by MaxIter. It can be set to any positive value less than 200. If no solution is found within 200 iterations then it is likely that some error has occurred (e.g. the calculation may be in a loop). After MaxIter iterations, the program stops its present calculation and either continues with the next one (specified in the input file *form.dad*) or ends (if no more calculations are required).

5.2.2.4 Accuracy Of The Reliability Index

Accuracy of the reliability index, β, is essential to insure a corresponding accuracy of the calculated probability of failure, P_f. It is important that the difference in the reliability index between the last two iterations does not correspond to a significant difference in the probabilities of failure. Suppose that the value of β in the last iteration resulted in $P_f = 10^{-1}$. This answer would be unreliable if the next iteration gave $P_f = 10^{-5}$.

One way of controlling the relative accuracy of the solution is by calculating a parameter, BetaAcc

$$BetaAcc = 100 \left| \frac{\beta_{New} - \beta_{Old}}{\beta_{New}} \right| \qquad (5.1)$$

such that the program does not stop iterating while BetaAcc is greater than the required accuracy, ReqBetaAcc. Of course, the required number of iterations increases for decreasing values of ReqBetaAcc. The default value in PARASODE for ReqBetaAcc is 1 which guarantees a relative accuracy of within 1%, which is usually sufficient. The program requires a value for ReqBetaAcc within the bounds 0 and 1.

5.2.2.5 Smoothing Of The Iteration Process

There are cases where the iteration process does not converge owing to instability: the new calculated design point differs considerably from the calculated design point of the preceding iteration. This difficulty can lead either to divergence of the process, or to values of the random variables which cause problems in the failure function. In such cases, "smoothing" of the iteration process may help. Smoothing is applied in the following manner:

$$X_{New}(i) = (1 - Smooth) \cdot X(i) + Smooth \cdot X_{Old}(i) \qquad (5.2)$$

where $X_{New}(i)$ is the new smoothed value of X_i, $X(i)$ is the new unsmoothed value, $X_{Old}(i)$ is the preceding value and Smooth is the smoothing coefficient for the iteration process which has a value between 0 and 1. Setting Smooth=0 means that no smoothing of the iteration process is performed.

Setting Smooth=0.5 provides an average between the old and the new value of X_i.

5.2.2.6 Accuracy Of The Failure Function

Just as a relative accuracy for β has been defined, it is possible to specify an accuracy for the failure function, Z, being zero. A dimensionless measure of accuracy is used in PARASODE (after Super-Software, 1994). The basis of this formulation is that if the standard deviation of the failure function, σ_Z, is small, the accuracy of Z being zero is more important than in the case where σ_Z is large. Hence, the program does not stop iterating while

$$\left|Z\right| > \frac{\mathrm{ReqZAcc}}{100}\sigma_Z \qquad (5.3)$$

where ReqZAcc is the required accuracy of Z. ReqZAcc has to be defined within the bounds of 0 and 1. If ReqZAcc=1, then the value of Z at the design point is less than 1% of the calculated standard deviation for Z away from zero. The default value of 1 is sufficient for most cases. Setting ReqZAcc to lower values means a higher accuracy for the answer but requires more iterations.

5.2.3   Truncation In PARASODE

Sometimes it is necessary to truncate a theoretical distribution of a random variable in order that it conforms to measurements or to known physical constraints. The truncation is said to be to the right of X=Xo if all values of X above Xo are discarded, and is said to be to the left of X=Xo if all values of X below Xo are discarded. Since the area beneath the probability density function must remain 1, it is necessary to scale the original non-truncated values of the probability density function over the truncated range.

Scaling can be performed in a number of ways, depending on factors such as the physical meaning of the variable. In this study, truncation has been performed as follows (Beaumont, 1986):

	Truncation For X Below Xo		Truncation For X Above Xo	
	X<Xo	X ≥ Xo	X ≤ Xo	X>Xo
Truncated PDF	0	$\dfrac{f_X(X)}{1-F_X(Xo)}$	$\dfrac{f_X(X)}{F_X(Xo)}$	0
Truncated CDF	0	$\dfrac{F_X(X)-F_X(Xo)}{1-F_X(Xo)}$	$\dfrac{F_X(X)}{F_X(Xo)}$	1

Table 5.1: Truncation method adopted in this study.
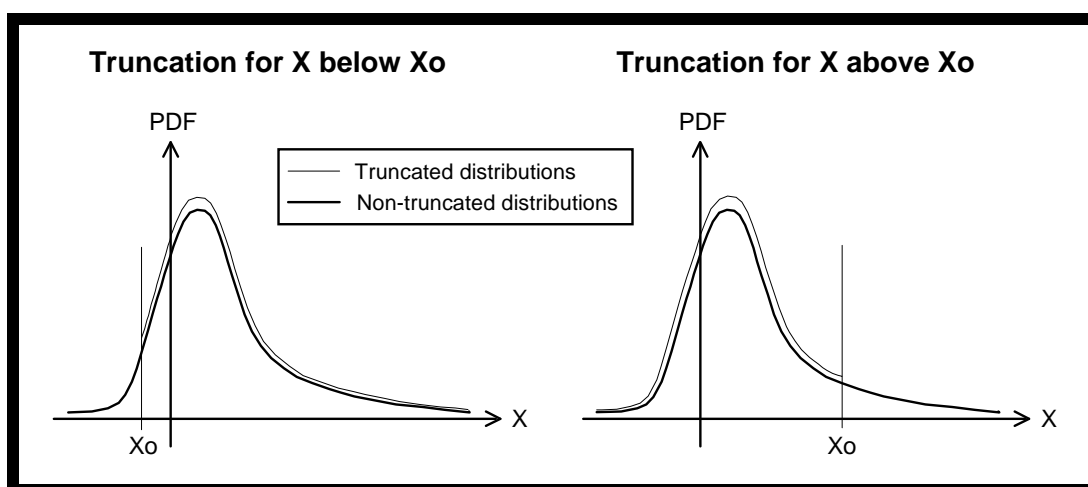
Figure 5.2: Truncation method adopted in this study.

The probability density function for significant wave height, H_S, provides an example of the need for truncation. H_S may be limited by the available water depth. In this case, several approaches for truncation to the right of Xo are possible, depending on the cut-off technique and the definition of the point of truncation, Xo (Thornton & Guza, 1983; Allsop & Meadowcroft, 1995). In this study, for the failure mode of overtopping, truncation of the distribution describing H_S has been performed according to Table 5.1 and Figure 5.2. To determine the point of truncation, Xo, it is assumed that the heights of individual waves (broken and unbroken) are described by the Rayleigh distribution and that $H_{rms} = 0.42\,d_S$ (Thornton & Guza, 1983). Furthermore, taking into account that for a Rayleigh distribution $H_S \approx 1.42 H_{rms}$, it follows that a sensible approximation for Xo is:

$$Xo = 0.6\,d_S = 0.6\,(SWL - TL) \tag{5.4}$$

where the variables d_S, SWL and TL are defined as in Chapter 3 (see Figure 3.1). Hence, the point of truncation is considered simply proportional to the local water depth. For simplicity, factors such as the bottom slope are not considered to influence Xo. Note that in PARASODE, SWL is forced to be always greater than TL.

This approach to truncating the distribution of H_S means that H_S is assumed to have some value smaller than Xo after wave breaking has been initiated and that the depth-limited values are redistributed across the range of significant heights in proportion to the unlimited values at each significant height.

5.2.4   Overtopping In PARASODE

The failure mode of overtopping is implemented in PARASODE using both the H&R model and Owen's model. The basic variables in these models are:

H&R Model

- Peak wave period, T_p
- Significant wave height, H_S
- H&R parameter, A
- H&R parameter, B
- Still-water-level, SWL
- Tangent of the seawall slope, $\tan\alpha$
- Roughness, r
- Model parameter, e_B

Owen's Model

- Mean wave period, T_m
- Significant wave height, H_S
- Owen parameter, A
- Owen parameter, B
- Still-water-level, SWL
- Roughness, r
- Model parameter, e_B

The program allows SWL to be specified either as a variable in its own right or as the sum of two variables: i) tide level (Tide); and ii) surge (Surge).

Model parameter r, describing the roughness of the seawall front slope, is the ratio of the run-up on a rough slope to that on the corresponding smooth slope. Values range from about 0.5 to 1.0 (CIRIA/CUR, 1991). It is readily incorporated in both the H&R model and Owen's model.

The two failure functions may then be written as follows:

H&R Model: $\quad Z = TR - A\sqrt{g(CH_S)^3}\left[1 - \dfrac{(CL - SWL)}{r\,CH_S}\right]^{e_B B} \quad$ for $0 \le R_* < 1$

$\qquad\qquad\quad Z = TR \qquad\qquad\qquad\qquad\qquad\qquad$ for $R_* \ge 1$

$$(5.5)$$

Owen's Model: $Z = TR - A\sqrt{\dfrac{gH_S^3}{\dfrac{s_m}{2\pi}}}\,\exp\left[-e_B B\,\dfrac{\sqrt{s_m}\,(CL - SWL)}{r H_S\,\sqrt{2\pi}}\right]$

where TR is the discharge allowed for a specific FORM calculation (the target value).

The program can run for as many as ten different values of TR. Plots can then be produced of the probability of failure as a function of the design parameter, seawall crest level, for different allowable discharges (see Figure 5.1). These plots are a valuable tool in the preliminary design of seawalls using probabilistic analysis. They can be used to make a cost optimisation for the structure during the reference period or design life (Van der Meer & Pilarczyk, 1987).

For the H&R model, the user has to choose if the coefficient C is calculated using a 37% or 99% confidence level for the maximum run-up. Depending on the choice made by the user, the program uses a constant of 1.52 or 2.15, respectively, in eq. (3.22). The value 2.15 arises from substitution of N=100 and p=99 in eq. (3.20). At present only these two alternatives are available. This is due to the fact that a change in C implies corresponding changes in the values of A and B.

For the failure mode of overtopping, the first partial derivatives of the failure function required to perform the FORM calculations can be calculated either by using their expressions, provided in the code, or by using the subroutine EO4XAF (NAG, 1993), called by PARASODE, which computes finite-difference approximations to the first derivatives for a given failure function.

5.2.5   Dune Erosion In PARASODE

The failure mode of dune erosion is implemented in PARASODE using Vellinga's model. In Mode 1, the program calculates the probability of failure associated with a prescribed value of the nourishment width. In Mode 2, the nourishment width is calculated for a target probability of failure.

The basic variables in the program are:

- Significant wave height, H_S
- Median sediment size, D_{50}
- Change in the initial profile, DP
- Surge duration, SD
- Gust bumps, GB
- Accuracy of the computation, Ac
- Maximum water level during surge, h

The program allows the maximum water level during surge, h, to be specified either as a variable in its own right or as the sum of two variables: i) tide level (Tide); and ii) surge (Surge).

Dune erosion is not an explicit function of the basic variables. Consequently, it is not possible to express the failure function as a simple equation. Thus, the first partial derivatives of the failure function required to perform the FORM calculations must be evaluated using the subroutine EO4XAF (NAG, 1993), called by PARASODE.

In PARASODE, the eroded sand can either be assumed to be transported only seaward (as in DUNEPROB and DUNE) or, alternatively, it can be assumed to move both landward and seaward during a storm surge. The common assumption, that during the short period of a storm surge the cross-shore sediment transport will principally be in an offshore direction, is conservative. Allowing the user to choose between movements only seaward or also in the landward direction, provides two answers which give an idea of the range of erosion to be expected.
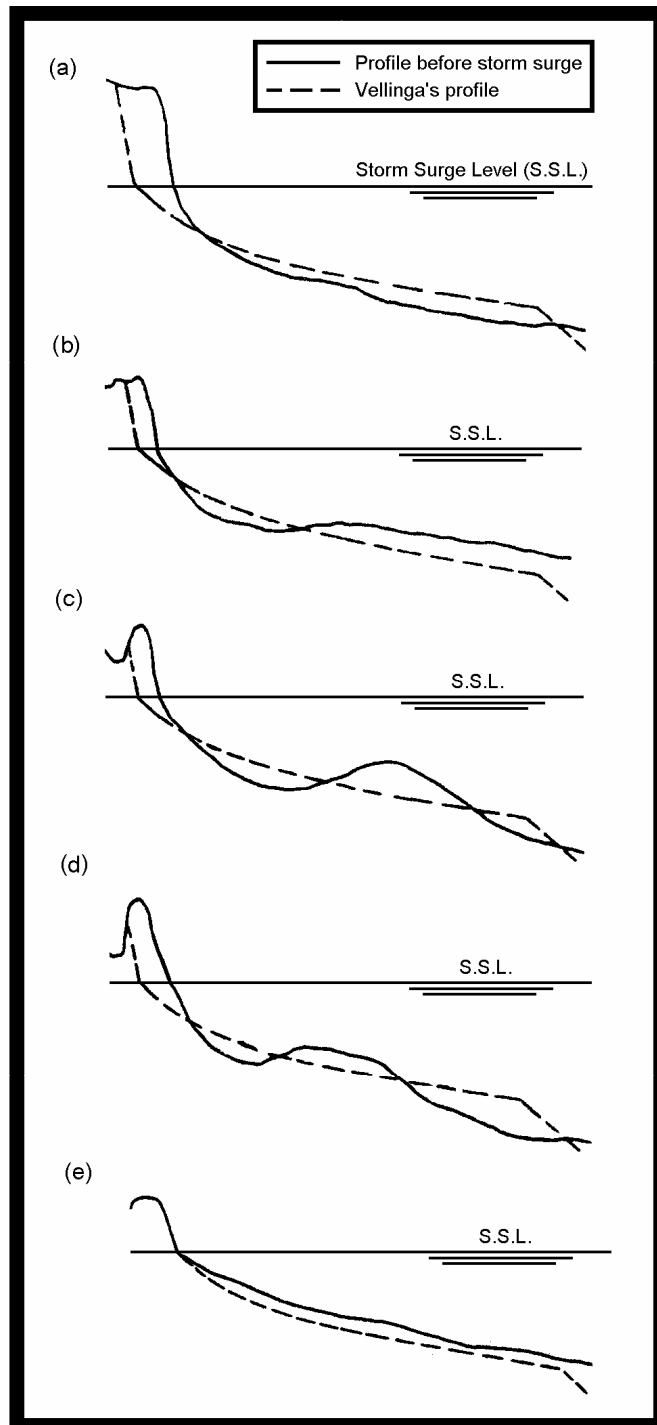
Figure 5.3: Illustration of the main erosion situations which can be studied using PARASODE (modified after CUR-TAW, 1989)

Figure 5.3 illustrates the main erosion situations which can be studied using PARASODE. Case (a) is the situation which normally occurs during high storm surges. Case (b) may occur for coastal profiles with flat slopes; after the storm surge, Vellinga's profile will be partly below the original profile. If

movements of sand are allowed only seaward, the original profile is raised by sand from the dune only. If movements of sand are allowed both seaward and landward, then the depression in the foreshore can be filled both from the dune and from the seaward part of the bed which lies above Vellinga's profile, resulting in a smaller amount of dune erosion. Case (c) is similar to case (b). If movements of sand are allowed only seaward, the sand movements on the seaward side of the bank are of no importance to the recession of the dune (since the sand between the bank and Vellinga's profile is sufficient to raise the seaward part of the original profile). If movements of sand are allowed both seaward and landward, then the depression can be filled both from the dune and from the bank, resulting again in a smaller amount of dune erosion. In case (d), the bank is fully eroded to raise the original bed towards Vellinga's profile, and the amount of sand further required for the development of Vellinga's profile is eroded from the dune. Finally, in case (e), Vellinga's profile is entirely below the original profile. This situation occurs frequently during storm surges at low tide levels. According to PARASODE, no dune erosion will take place (CUR-TAW, 1989). In practice, however, a minor amount of dune erosion may be expected owing to wave run-up, particularly if the dune face is steep.

The general calculation procedure is described here. For further details, the reader is referred to Appendices C5 and C7. Figures 5.4 to 5.7, presented at the end of this section, illustrate the procedure used and the notation applied in the FORTRAN code. Some of the notation in these figures is not mentioned in the main text but can be found in the program listing.

The calculation procedure differs depending on the direction chosen for the sand movements. In any case, for a given initial profile with NPD points, the program starts by establishing a changed profile. The latter is obtained by changing the Y-coordinate, YP, of some points in the initial profile (see Figure 5.4(a)). The number of points changed is NPch and the change is DP. The purpose of making these changes is to represent the possible error in the initial profile immediately before the storm surge. These errors arise as a consequence of measurement inaccuracies and changes in the profile between the time of measurement and occurrence of the storm surge.

Next, a new profile is defined based on the nourishment characteristics (nourishment top level, nourtlev, and gradient of the nourished face,

1:mnour) and on the value of the design parameter (nourishment width at top level, nourwidt). The new profile is referred to as the nourished profile (see Figure 5.4(b)). The point of intersection of the nourished profile and the surge level, h, is (S1,T1). Note that if the parameter nourwidt is set to zero, PARASODE assumes that there is no nourishment. Consequently, if nourishment is to be provided without a berm, nourwidt must be set at a small positive value. Note also that the gradient of the nourished face depends not only upon the method chosen for placing material[1] but also upon other factors such as the grain size (CUR et al, 1987). Table 5.2 provides some guidance on expected gradients.

MATERIAL	GRAIN SIZE (μm)	DRY FILL ABOVE WATER (usually for dune nourishment)	HYDRAULIC FILL (usually for beach nourishment)		
			Above Water	Below Water	
				Smooth Sea	Rough Sea
Fine Sand	60-200	up to the	1:50 - 1:100	1:6 - 1:8	1:15 - 1:30
Medium Sand	200-600	natural angle	1:25 - 1:50	1:5 - 1:8	1:10 - 1:15
Coarse Sand	600-2000	of repose	1:10 - 1:25	1:3 - 1:4	1:4 - 1:10
Gravel	>2000	(1:1 - 1:2)	1:5 - 1:10	1:2	1:3 - 1:6

Table 5.2: Expected gradients of nourished dune/beach face (modified after CUR et al, 1987).

In its present version, PARASODE allows consideration of a simplified form of nourishment to the seaward face of the dune and/or to the beach (see Figure 5.4(b)). However, it would be relatively straightforward to modify PARASODE to deal with more complicated nourishment profiles. Note that it is assumed that once nourishment has taken place, the material stays where it has been deposited until a storm surge occurs. The material used for nourishment is also assumed, for simplicity, to be of the same type and size as the native sediment; usually, the preferred grain size of borrow material is equal to or larger than that of the native sediment (Hedges, 1977; CUR et al, 1987; Simm et al, 1996).

After the nourished profile is defined, the shape of Vellinga's parabolic

---

[1] In broad outline, the methods can be distinguished as follows (CUR et al, 1987):
 i) dry fill - transportation of dry sand to the site by trucks, etc;
 ii) hydraulic fill - transportation of a sand-water mixture via a pipeline.

post-storm profile is calculated according to eq. (4.1). As a first approximation, it is assumed that the parabolic part of the profile starts at point (S1,T1) (see Figures 5.5(a) and 5.6(a)). The X-coordinate of this starting point is always designated as S8. The position of the offshore point, (S9,T9), where the parabolic part of the profile terminates is also calculated: the length of the profile is Le and the depth is Depth. If point (S9,T9) is located above the nourished profile (Figure 5.5), Vellinga's profile continues seaward as a straight line with a gradient of 1:mt until it intersects the seabed. Otherwise (Figure 5.6), a vertical line is drawn until intersection with the nourished profile occurs. The point of intersection is (S2,T2). Landward of X=S8, the gradient of the eroded dune face is 1:md. The point of intersection of the eroded dune face and the nourished profile is (S3,T3). Note that in PARASODE, 1:md is not a constant of 1:1 as in Vellinga's original model. The user is free to define this slope. Likewise, the gradient of the toe of the post-storm profile, 1:mt, need not be taken as 1:12.5.

After Vellinga's profile is defined, it has to be located in such a way with regard to the nourished profile that the total area of eroded sand is equal to the area of accretion. In order to achieve this required final position, Vellinga's profile is moved along the X-axis, the corresponding areas of erosion and accretion are calculated and the balance tested. The methods used to calculate the areas of erosion and accretion and to test the required balance between these areas depend on the direction of the sand movements. Details are provided in Appendix C5.

Finally, the failure function is calculated as follows (Figure 5.7):

$$Z = TR + S4$$

(5.6)

TR (the target value) is the allowable erosion distance measured from the reference line X=0 and S4 is the X-coordinate of the most landward point to which the profile has been eroded. Note that S4 is the estimate provided by PARASODE of the position R, in Figure 4.1. The program can be run for as many as ten different values of TR. Plots can then be produced of the probability of failure as a function of the design parameter, nourishment width, for the different allowable erosion distances. Such plots are a valuable tool in the preliminary design of dune nourishment using probabilistic analysis.
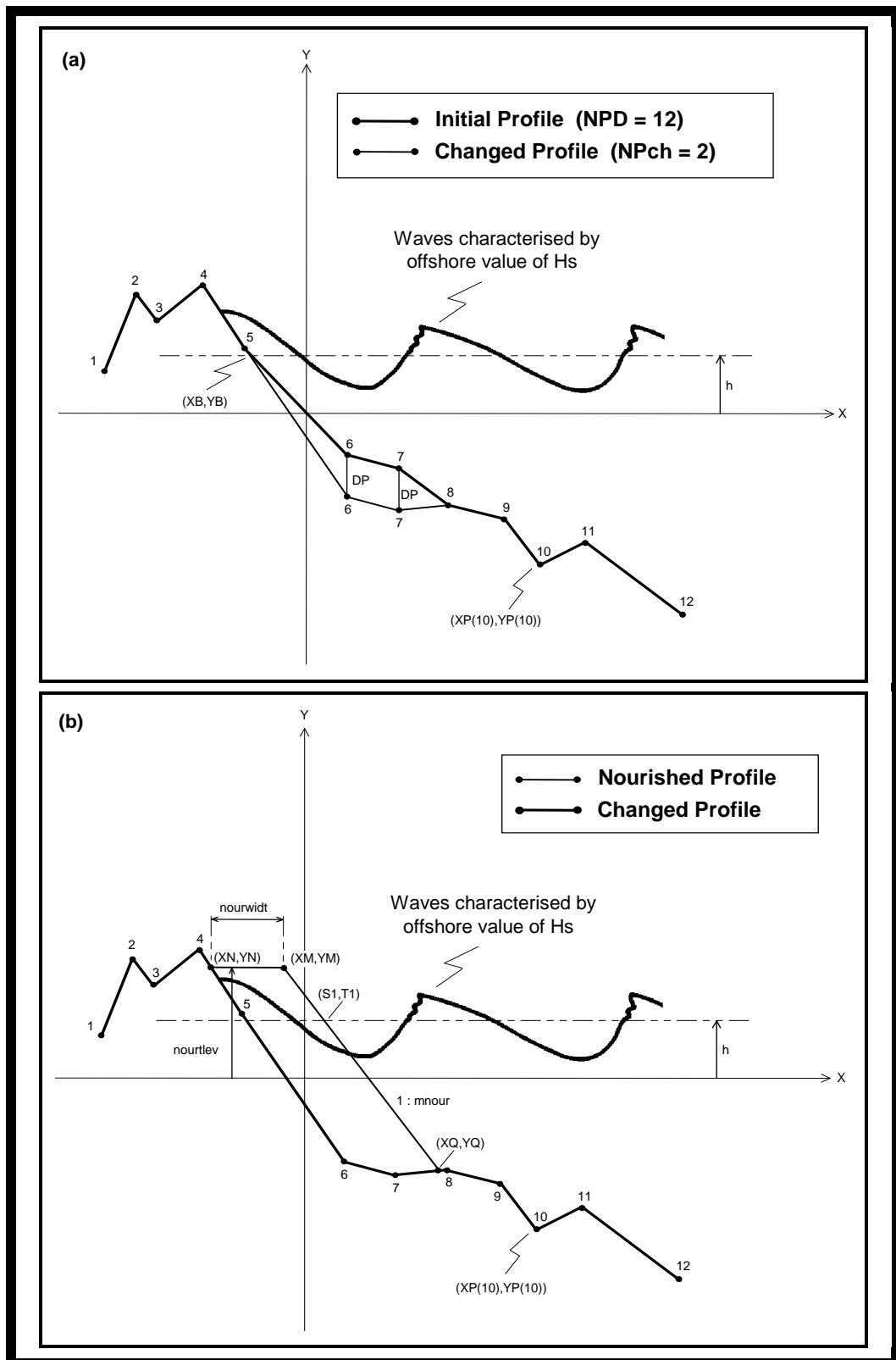
Figure 5.4: Definition of initial, changed and nourished profiles.
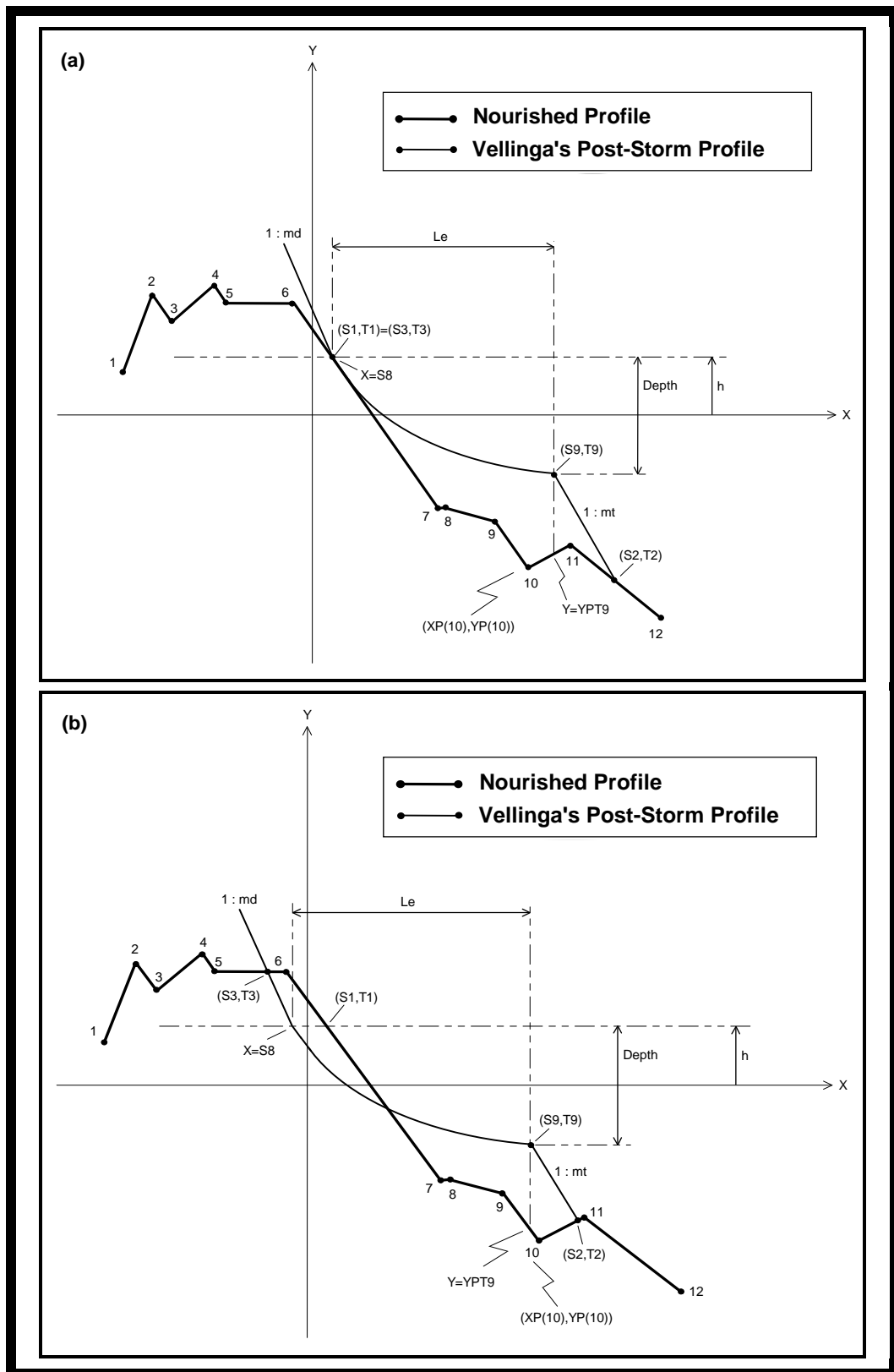
Figure 5.5: Definition of Vellinga's post-storm profile - example 1.
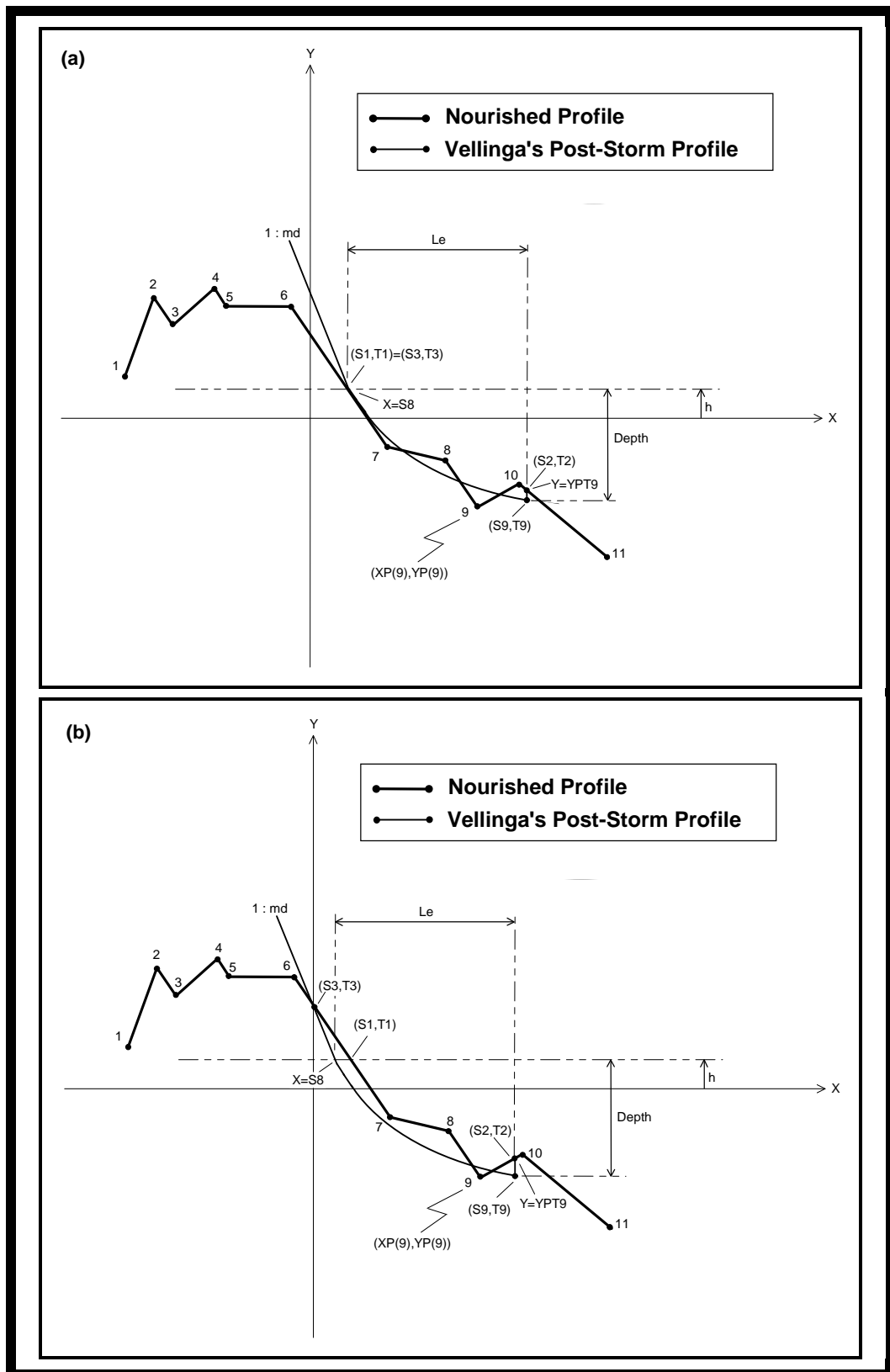
Figure 5.6: Definition of Vellinga's post-storm profile - example 2.
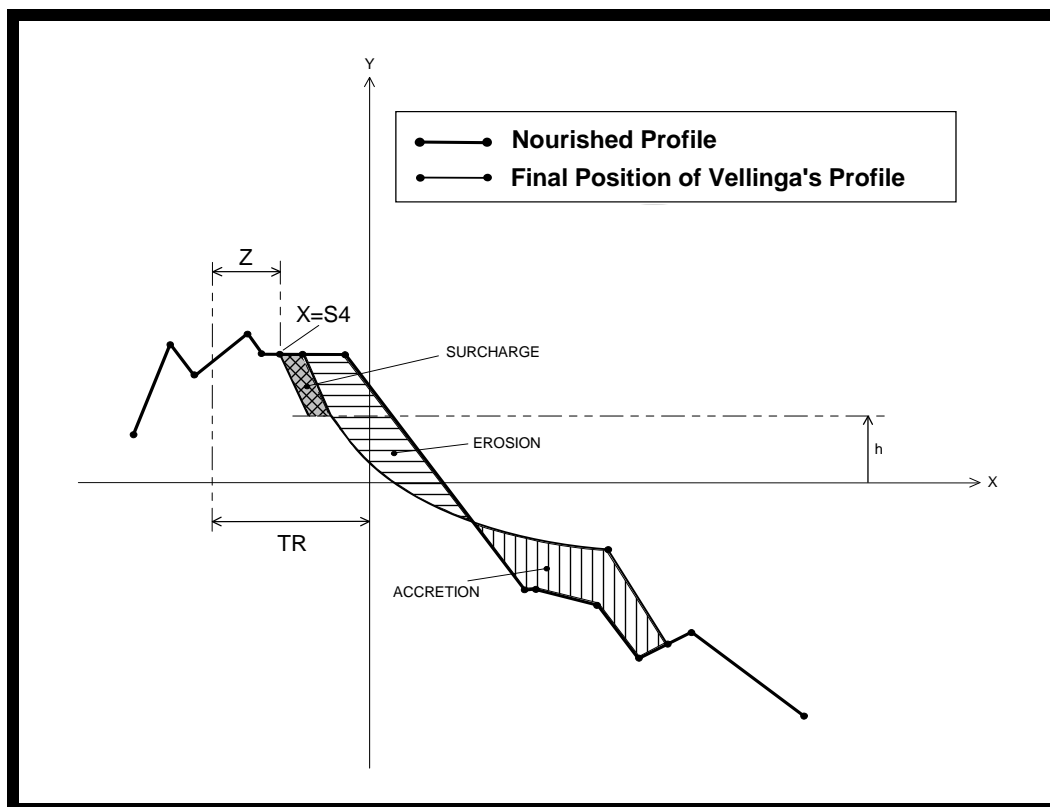
Figure 5.7: Definition of the failure function for dune erosion.

5.2.6 Input

The program runs simply by executing the command PARASODE. The input data can be read either from the computer screen or from input data files. If the data are to be read from the screen, the user only has to answer the questions asked and choose between alternatives. If the input is provided by data files, then the following four files have to be prepared by the user, no matter which failure mode is studied:

- *general.dad*
- *form.dad*
- *meandev.dad*
- *coefcor.dad*

A fifth data file is required if the failure mode under study is dune erosion. This file is called *perfil.dad* .

Description of the input files is given in Appendix C6 and examples are provided in Appendices D1 and D3.

5.2.7   Output

Two output files are created when running PARASODE:

- *summary.dat* - A file which contains the input data and the most important final numerical results only.
- *results.dat*   - A text file which contains the input data and the most important numerical results for all iterations.

Examples of the output file *summary.dat* are given in Appendices D1 and D3.

5.3   Validating PARASODE Using @RISK

To validate the results of a Level II program like PARASODE, Level III calculations have to be carried out (Ang & Tang, 1984; Van der Meer, 1987).

The normal way to implement Level III methods is to write a computer program. Such a program would consist of random number generation (normally a built-in function), solving the appropriate inverse distribution functions for the parameters which have been defined, calculating the result, and repeating for another set of random numbers. After the required number of samples, the results are summarised in terms of a probability distribution, or simply the proportion of results corresponding to failure.

A simpler way to carry out a Level III analysis is to use existing software packages. A search was made for software suitable for this task. Programs such as PREDICT (Risk Decisions Ltd., Oxfordshire, UK), @RISK (Palisade, New York), STRUREL (Reliability Consulting Programs, Munich, Germany) and SUPER-RISK (Super-Software, Heemstede, The Netherlands) were considered. Some programs included probabilistic methods other than Level III; however these packages were expensive. Since the main objective was to obtain a package solely to carry out Level III calculations, @RISK was chosen.

@RISK is sold as an add-in for Microsoft Excel or Lotus 123. It uses simulation to combine all the uncertainties identified in the modelling. The options available for controlling and executing a simulation in @RISK are quite powerful. They include: i) Traditional and Latin Hypercube Sampling (see section 2.3.2.2); ii) unlimited number of iterations per simulation; iii) multiple simulations in a single analysis; iv) continuing a simulation after viewing results and performing more iterations if necessary; and v) seeding the random number generator.

The random number generator used in @Risk is a portable random number generator based on a subtractive method, not linear congruential (for more details see, for example, Law & Kelton, 1991). The cycle time is long enough that it has no effect on the simulations (Palisade Corporation, 1994). The period of the generator is effectively infinite. The seed or starting value, if not set manually, is clock dependent, not machine dependent. The results of a simulation are reproducible from run to run. If the seed is set to zero it means that the sequence of random numbers will start at a random value. The result will differ each time a run is made (using the same input). If however the seed is set to any positive number, it means the random generated numbers will start at a specific place in the sequence. This allows @Risk to give reproducible results of a simulation from run to run, because each time a run is made the same sequence of random numbers will be used.

The way the program works is appealing because it conforms to the way that many engineers now carry out calculations: formulae are entered into the spreadsheet as usual, but any data item in a cell (or range of cells) can be specified as a probability distribution instead of as a single value. The software provides a library of about 30 different distributions, including the distributions available in PARASODE. The user issues the command to carry out a simulation and the software automatically carries out the task using the prescribed probability distributions, recording each interim result. Simulation results generated by @Risk include statistics and data reports for both input and output variables. The probability distribution of the results for each output cell is then displayed graphically. @ RISK graphs include: i) relative frequency distributions and cumulative probability curves; ii) summary graphs for multiple distributions across cell ranges; iii) statistical reports on generated distributions; and iv) probability of occurrence of target values in a

distribution. @Risk results and graphs can be placed directly in the Microsoft Excel or Lotus 123 spreadsheet for reporting purposes.

The software also has modelling techniques to deal with dependencies. This is very important because in many practical engineering analyses, random variables are often statistically and physically dependent. Furthermore, actual distribution types for the random variables involved can be a mixture of different theoretical distributions. To properly replicate such systems, simulation should be able to preserve the correlation relationship among the stochastic parameters and their distributions (Iman & Conover, 1980, 1982; Chang et al, 1994). In @Risk, to allow for correlation, one can build a correlation matrix for the input variables. This matrix forms the basis for the correlated sampling of the input variables during simulation. The facility is especially useful when pre-existing correlation coefficients are available and one wants sampling to be governed by those coefficients.

The main advantage of this software is its flexibility and ease of use for anyone familiar with spreadsheets. However, because of its user-friendly characteristics, there are dangers in the use of @Risk (and similar computer programs) unless the user is fully aware of issues such as the importance of formulating the correct relationships between input and output variables, the selection of the probability distributions, and the choice of sample size as it affects the stability of estimates of the output variables, including their extreme values.

5.4   Summary

A FORTRAN Level II program, PARASODE, has been developed. In particular, the program concentrates on the failure modes of random wave overtopping of simple seawalls and dune erosion. The quantity of wave overtopping is calculated using both the H&R formula and Owen's formula. Dune erosion is calculated using Vellinga's model. However, much of the program is generic and can be adapted to other failure modes without undue difficulty. A Level III software package, @Risk, has been used to validate the output from PARASODE with regard to wave overtopping.

6 APPLICATION OF PARASODE

6.1 Examples - Wave Overtopping Of Seawalls

6.1.1 Input To PARASODE

Section 6.1 illustrates the use of PARASODE and the differences between the H&R overtopping model and Owen's formulation. Note that only some features of PARASODE are shown here. For example, due to lack of data, Ferry Borges & Castanheta's method of combinations of time-varying actions (see Section 2.3.3.3) is not illustrated, although PARASODE is fully developed to allow its use.

The geometry of the simple seawall used in the case studies (see Figure 3.1) is as follows:

- impermeable slope of 1:2 with a relatively smooth surface;

- toe level of 0m OD (OD denotes Ordnance Datum);

- crest level of between 8m OD and 16m OD.

Such a seawall is typical of potential developments around Liverpool Bay in the south-eastern corner of the Irish Sea. The allowable discharges considered in the examples lie in the range 10^{-1} to 10^{-6} m^3/s/m (see Figure 3.2).

The main statistical characteristics of the basic variables in the H&R and Owen models (see Section 5.2.4) are described next. For each basic random variable, it is necessary to define a mean value, the corresponding standard deviation and to postulate a type of distribution. Depending on the distribution type, other statistical values may also have to be provided (e.g. the lower limit of a Weibull distributed variable or the lower and upper limits of a Beta distribution).

Examples of input files are provided in Appendix D1 as well as the corresponding *summary.dat* output files.

6.1.1.1 Distributions Of The Sea State Parameters

The scatter diagram describing the long-term distributions of wave heights and periods at the Mersey Bar in Liverpool Bay can be found in Salih (1989). Using the method of moments for one year's data (from September 1965 to September 1966) recorded at three-hourly intervals (Draper & Blakey, 1969), Salih fitted three-parameter Weibull distributions to the significant wave heights and mean zero-crossing wave periods. The following approximate statistical parameters are derived from his results:

H_s (m)			T_m (s)		
μ	σ	Lower Limit	μ	σ	Lower Limit
1.2	0.7	0.45	5	0.9	3.3

Table 6.1: Means, standard deviations and lower limits for the Weibull distributions of H_s and T_m.

Since the H&R model uses T_p instead of T_m in its formulation, a mean JONSWAP relationship between T_m and T_p was assumed as follows (Hogben, 1990): $T_p=1.28T_m$. Hence, a three-parameter Weibull distribution was adopted for T_p with $\mu = 6.4s$; $\sigma = 1.152s$ and lower limit $= 4.224s$.

Salih also reported a linear correlation coefficient between H_S and T_m of approximately $\rho = 0.6$. This same correlation has been assumed between H_S and T_p.

6.1.1.2 Distributions Of Water Levels

In this study, two main situations have been evaluated, bearing in mind that the design conditions which are critical for one type of problem might not be critical for others (CIRIA/CUR, 1991; Havno et al, 1996):

1) **Performance under normal conditions** - The performance of the structure for any possible value of total water level (tide plus surge) is evaluated.

2) <u>**Performance under extreme conditions**</u> - The ability of the structure to survive extreme total water levels is checked.

The first condition is relevant, for example, to design of retention and drainage systems and for checking the safety of people and vehicles. The second provides peak values which are important for structural safety.

PARASODE allows the water level to be specified either as a variable in its own right or as the sum of the tide and surge components (see Chapter 5). All the information on water levels at Liverpool made available to the author was provided by the Proudman Oceanographic Laboratory (POL), UK. However, only data on total water levels, both for normal and extreme conditions, were initially available. These data were applied to specify user-defined distributions for Liverpool (see Appendix C4). Calculations with PARASODE were then performed using this information. Later, data on the separate tide and surge components for normal conditions were also acquired. With the new data, a user-defined distribution for the predicted tide levels at Liverpool was established (see Appendix C4), and a Gumbel distribution ($\mu=0.019$m; $\sigma=0.192$m) was fitted to represent the surge component by applying the software package BestFit. Calculations using PARASODE were then repeated using these distributions. As expected, the results obtained were identical. Hence, for normal conditions, only the results of the calculations which used the separate tide and surge components are reported in this thesis in Appendix D2. In this same appendix, the results for extreme conditions using total water levels are also tabulated. The tide and surge components have been assumed independent, based on information also provided by POL. Note that tide-surge interaction may be important in very shallow water regions (see Alcock & Carter, 1985).

6.1.1.3 Interrelationship Between Sea State And Water Level

Overtopping of a seawall does not depend solely on the individual sea state or the individual water level but on their combination (HR Wallingford, 1989; Thomas & Hall; 1992). Even a very severe storm, leading to massive waves, may pass virtually unnoticed if water levels are low. At such times, because of the generally shallow beach slope below mean water level, waves break rather harmlessly by spilling rather than plunging, and there is considerable loss of energy due to friction over the beach. These effects, combined with

the higher freeboard, mean that no overtopping or damage is likely to occur. In contrast, at very high water levels, even quite modest waves can cause problems. The increased water depth allows waves to break by plunging on the seawall, and also reduces the effect of friction. With the reduction in freeboard caused by the high water level, waves can overtop the structure. Hence, in design of coastal structures, combinations of large waves and high water levels are of particular interest (Simm et al, 1996). In most cases, correlation between large waves and high water levels should be considered. However, the scope of each correlation assessment should be decided on its own merits, in terms of the input data available, the intended end use, and the potential benefits to be derived (Hawkes & Hague, 1994). It is beyond the scope of the present study to carry out a detailed analysis of the correlation between waves and total water levels but the interested reader can refer to work such as Hague (1992) and Hawkes & Hague (1994).

The main problem when trying to account for the correlation between sea state and total water levels is the fact that it is necessary either to assign a correlation coefficient directly between the sea state and the total water level or, alternatively, between the sea state and surge and between surge and tide level (see Figure 4.8). Unfortunately, information necessary to accurately determine the correlation coefficients is often unavailable. Suppose some correlation between waves and total water levels is expected, but the correlation coefficient is not known. Confidence in accepting a particular seawall configuration can be reinforced by examining a pessimistic view of the suspected correlation. If the results are still acceptable, then confidence in the seawall has been justified. Conversely, if a seawall configuration appears unacceptable, one may be reassured that rejection of the design is justified by examining an optimistic view of the effect of the suspected correlation.

In the Liverpool Bay area of the Irish Sea, the tidal range is around 10m. This very big tidal range masks the correlation between waves and surge (Hawkes & Hague, 1994). As a consequence, sea states and total water levels are not completely independent, but the correlation is very weak. The question remained of what degree of correlation to consider in the present study. Hawkes & Hague (1994) suggested a positive though weak correlation between waves and water levels for North Wales whereas, according to Alcock (1984), Hydraulics Research Station assumed that SWL and wave

heights were independent for North Wales and for Fleetwood and Cleveleys. Since Liverpool Bay falls between these locations (Figure 6.1), a similar assumption might reasonably be made in the absence of further data.

In this study, no correlation has been assumed either between waves and total water level or between waves and surge. Note that PARASODE allows the user to consider correlation between any two variables by providing a non-zero correlation coefficient between them. Likewise, independence between two variables can be ensured by adopting a value of zero for the correlation coefficient.
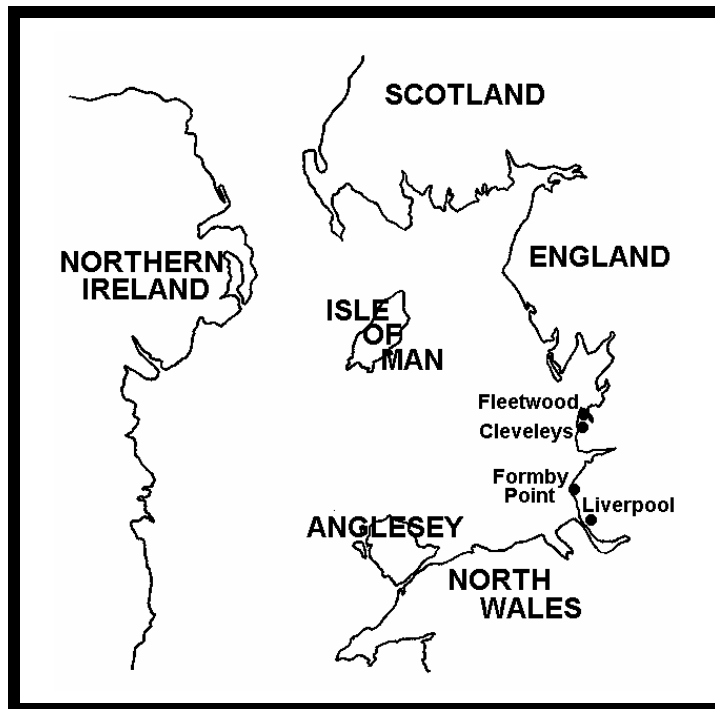
Figure 6.1: Location of Liverpool Bay in relation to North Wales, Fleetwood and Cleveleys.

6.1.1.4 Distribution Of The Tangent Of The Seawall Slope

The angle at which the seawall front slope is constructed will never be exactly as specified in its design. Therefore, this parameter has been introduced as a random variable having a Normal distribution with a mean $\mu=0.5$ and a standard deviation of 10% of the mean value, i.e. $\sigma=0.05$. One would expect the angle of the seawall slope to be formed with the same

tolerance above and below the design angle. Hence in a narrow band about the mean, the Normal distribution is expected to fit well. Obviously, the tailing off of the distribution to infinity is not representative of the slope angle. However, the behaviour of the tails is not important where there is a small standard deviation associated with a high mean as is the case in this example.

6.1.1.5 Distribution Of The Roughness Of The Seawall Slope

The slope roughness, r, for different types of cover layer can be found, for example, in CIRIA/CUR (1991) or Van der Meer & Janssen (1995). As for the angle of the seawall slope, the roughness of the relatively smooth impermeable slope has been considered as a random variable. A Beta distribution has been chosen, with μ=0.95, σ=0.01 and lower and upper limits x_1=0.9 and x_2=1. The use of the Beta distribution addressed the fact that r can never be greater than 1 and it has a specific range of values depending on the type of cover layer. A Rectangular distribution might have been used instead, particularly since it is simpler than the Beta distribution.

6.1.1.6 Distributions Of The Parameters Of The Models

In the example, the values of A and B for both the H&R and Owen models have been set to fixed values according to Table 3.2 of Chapter 3:

- H&R model, $(R_{max})_{37\%}$: A=0.00753 and B=4.17
- H&R model, $(R_{max})_{99\%}$: A=0.00542 and B=7.16
- Owen's model: A=0.0117 and B=21.71

Following the recommendations of Chapter 3 (Section 3.7), parameter e_B has been considered as a Log-Normal distributed variable, with mean and standard deviation as shown in Table 3.4 of that chapter:

- H&R model, $(R_{max})_{37\%}$: μ=1.049 and σ=0.241
- H&R model, $(R_{max})_{99\%}$: μ=1.044 and σ=0.200
- Owen's model: μ=1.027 and σ=0.150

Note that e_B could equally have been chosen as Gumbel or Gamma distributed (see Section 3.7).

6.1.2 PARASODE Results And Discussion

6.1.2.1 Normal Conditions

The results produced by PARASODE for wave overtopping of seawalls under normal conditions are presented both in tabular form in Appendix D2 and graphically in this section.

Figure 6.2 shows the probabilities of failure, P_f (%/year), versus the crest level, CL (m OD), for different values of the allowable discharge, for the H&R model and for Owen's model.

Figures 6.3 and 6.4 display the sensitivity of the probability of failure to inaccuracies in the values of the H&R model basic variables at the design point, as a function of the allowable discharge and the seawall crest level, respectively. Figures 6.5 and 6.6 give the same results for Owen's model.

Figures 6.7 to 6.14 show parameter values at the design point as a function of the seawall crest level and the allowable discharge, for the H&R model and for Owen's model. Only the values of the variables which were found to have a major contribution to the probability of failure have been plotted, i.e. sea state parameters (wave height and period), tide and model parameter e_B.

Note that the results for the H&R model, $(R_{max})_{99\%}$, are tabulated in Appendix D2 but are not plotted here since the observations which could be made are essentially identical to those for $(R_{max})_{37\%}$.

From the tables in Appendix D2 and the figures mentioned above, the following observations may be made:

- Figure 6.2 - As expected, P_f (%/year) decreases as the crest level of the seawall increases. Likewise, for the same value of the crest level, P_f decreases as the allowable discharge increases. For the

same values of the crest level and the allowable discharge, Owen's model predicts higher probabilities of failure than the H&R model. For the H&R model the probabilities of failure are higher for $(R_{max})_{99\%}$ than for $(R_{max})_{37\%}$. Note that these observations are consistent with the comments made in Section 3.6. Acceptable probabilities of failure for coastal structures are given in Section 4.5.1 as generally between 10^{-2} and 10^{-5} (i.e. 1 to 0.001%/year). Consequently, for the input conditions considered, a crest level greater than about 10m would be required to satisfy structural safety. A level of at least 12m would be required to satisfy functional safety according to the H&R model whilst Owen's model would demand a crest level greater than 16m.
As will be shown later in Section 6.2.1.2, the probability of failure over the lifetime of the structure may be determined quite simply from knowledge of the probability of failure in a year, provided that statistical independence of each year is assumed.

- Figures 6.3 and 6.4 - For the H&R model, the sensitivity parameters, $\alpha^2 (\%)$, show that the main influence on the probability of failure is generally provided by the uncertainty in the sea state, i.e. H_S and T_p (up to 58%). The tide also has a major contribution (up to 47%), although it is never as large as the contribution of the sea state. For the biggest allowable discharges, the model parameter, e_B, occasionally plays the strongest role (up to 49%). The effect of the surge is much less important (up to about 10%), and the angle of the seawall front slope and its roughness have only minor influences on the resulting variance. For each value of the seawall crest level, the importance of the sea state and the tide tends to increase as the allowable discharge decreases, while the effect of e_B decreases. For each allowable discharge considered, the contributions of the sea state and the tide decrease as the crest level increases, while the contribution of e_B increases.

- Figures 6.5 and 6.6 - As with the H&R model, the sensitivity parameters for Owen's model show that the most important factor is generally the sea state, i.e. H_S and T_m (up to 66%). The tide also has a major contribution (up to 72%), and in some cases, it is even more important than the contribution of the sea state. Model parameter e_B is, in some instances, more significant than the tide (up to 34%); as with the H&R model, the effect of e_B is greatest for the largest allowable discharges and highest crest levels. For Owen's model, the sensitivity to variability in the surge is even less important (up to only 4%) than for the H&R model, and again the roughness of the seawall front slope makes only a minor contribution to the resulting variance. Unlike for the H&R model, the influence of the sea state does not show any obvious pattern with crest level or allowable discharge. However, as for the H&R model, the importance of the tide increases as the allowable discharge decreases, for each value of the seawall crest level; and for each allowable discharge, the tide's contribution decreases as the crest level increases.

- The above discussion of Figures 6.3 to 6.6 highlights how crucial it is to obtain reliable and sufficient data on wave, tide and e_B to enable the probability distributions of these variables to be determined accurately.

- <u>Figures 6.7 to 6.14</u> - The value of T_p at the design point for the H&R model does not follow any special trend with the crest level or the allowable discharge. T_m for Owen's model increases as the crest level and the allowable discharge increase. The values of H_S and Tide at the design point increase with increasing values of the crest level and of the allowable discharge for both overtopping models. Also for the two models, the value of e_B at the design point decreases as the crest level and the allowable discharge increase. The values of H_s and Tide at the design point are higher for the H&R model than for Owen's model despite the generally higher probabilities of failure associated with Owen's model (see Figure 6.2).

- From all figures, it can be seen that the choice of overtopping model is very important in the probability assessment of the safety of seawalls exposed to normal conditions. The two models lead to quite different results. Use of Owen's model in design would be more conservative than use of the H&R model. However, the conservative nature of Owen's model also implies that its use in design will result in more expensive structures than those designed using the H&R formulation.

Note that the results for an individual case study cannot be adopted in a general sense. Each situation has particular characteristics which make it unique.
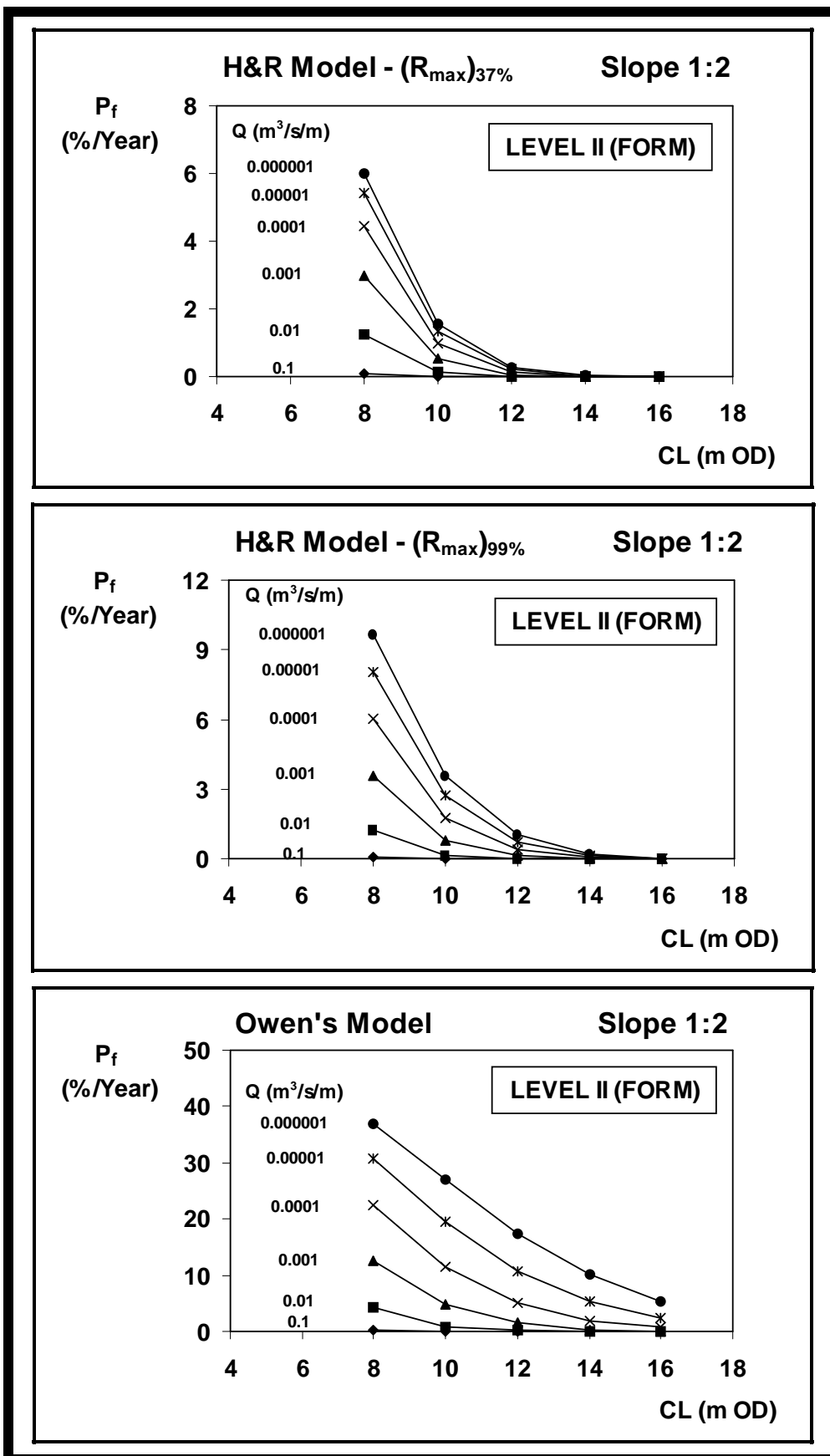
Figure 6.2: Probability of failure versus crest level for different values of the allowable discharge, for the H&R model and for Owen's model.
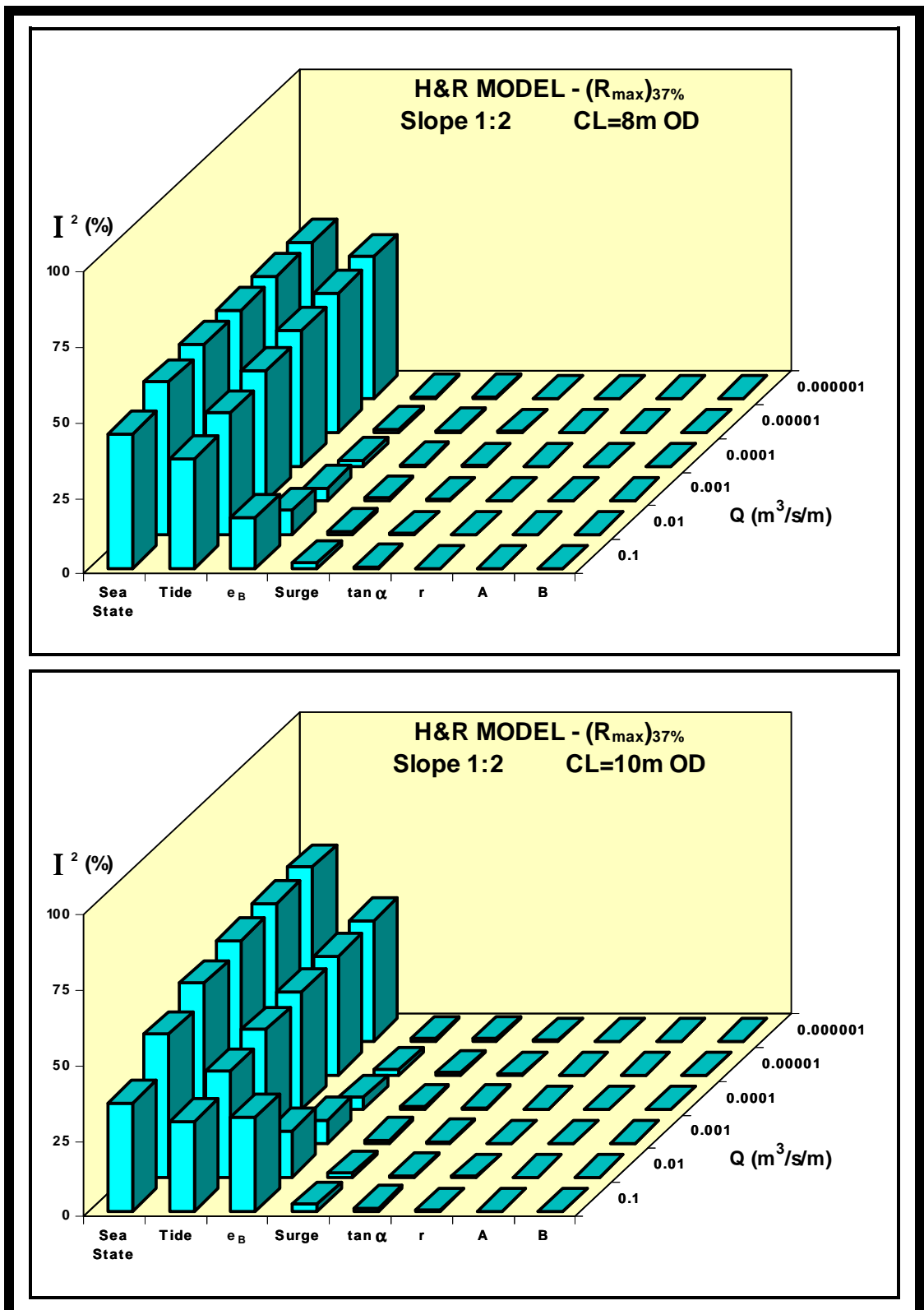
Figure 6.3: Sensitivity of the probability of failure to inaccuracies in the values of the H&R model basic variables at the design point as a function of the allowable discharge.
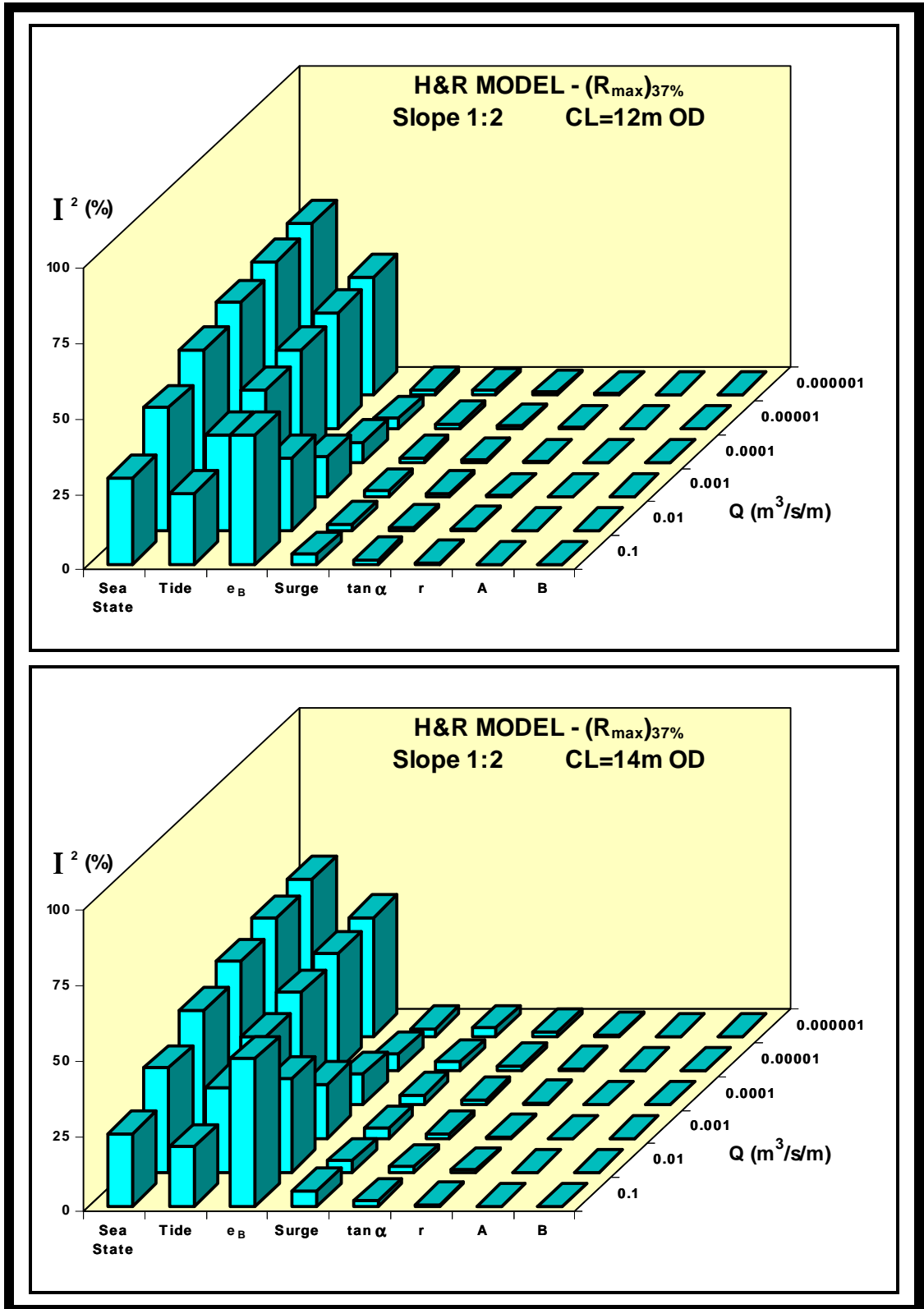
Figure 6.3: continued.
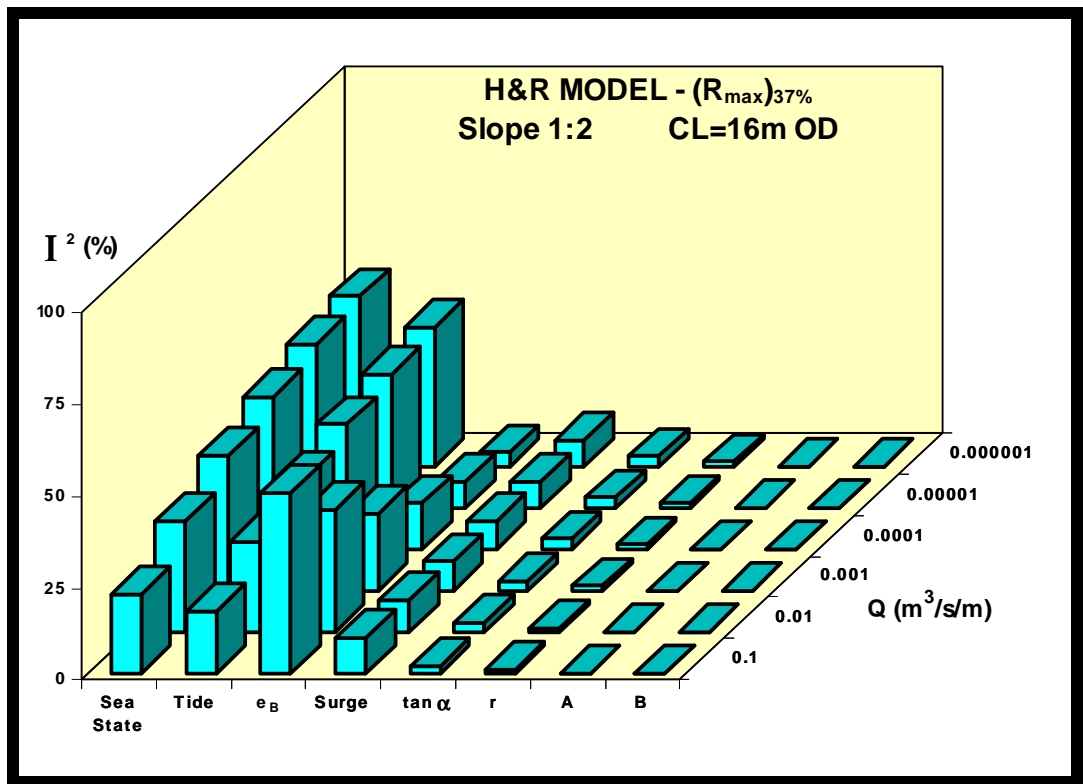
Figure 6.3: continued.
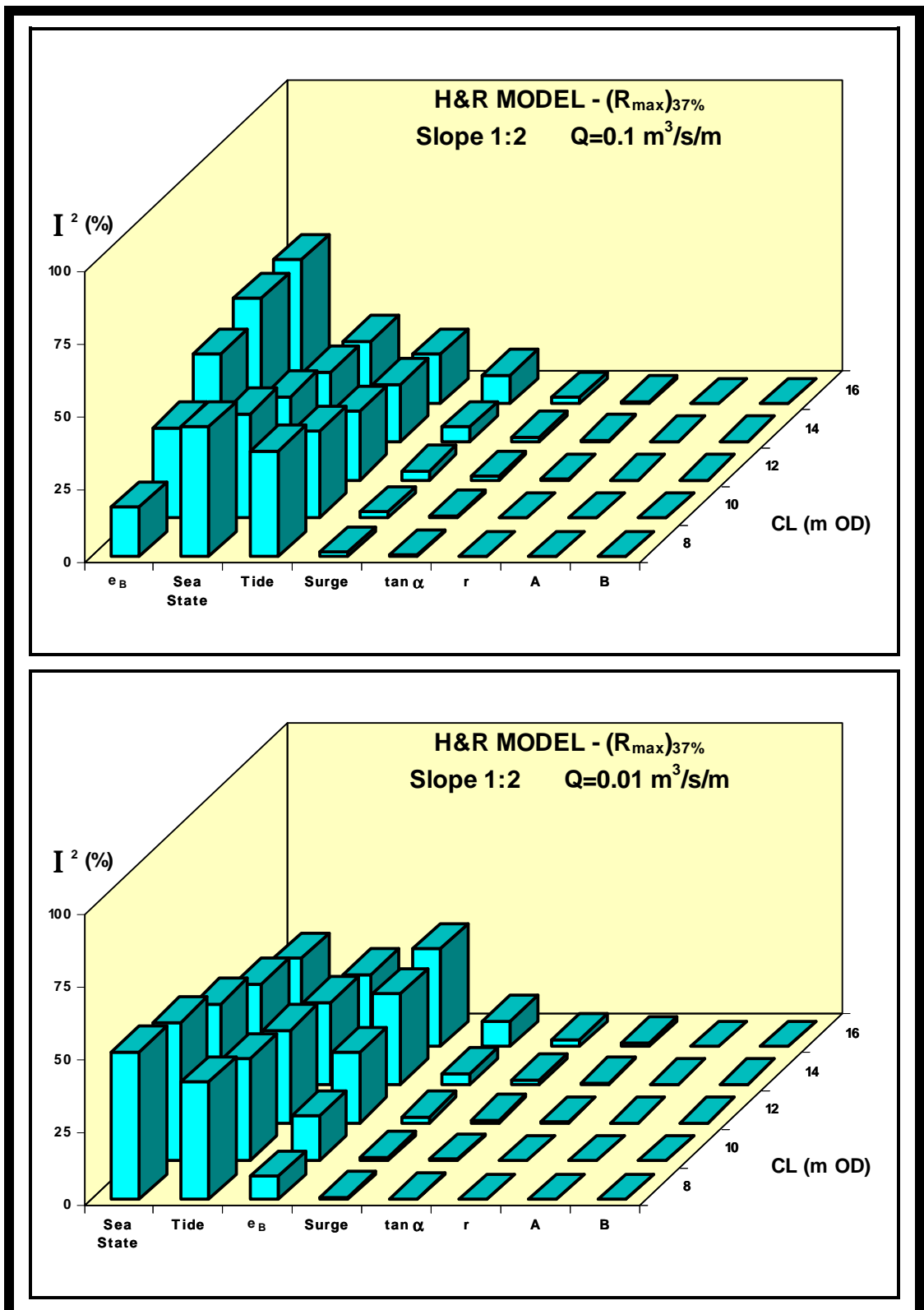
Figure 6.4: Sensitivity of the probability of failure to inaccuracies in the values of the H&R model basic variables at the design point as a function of the seawall crest level.
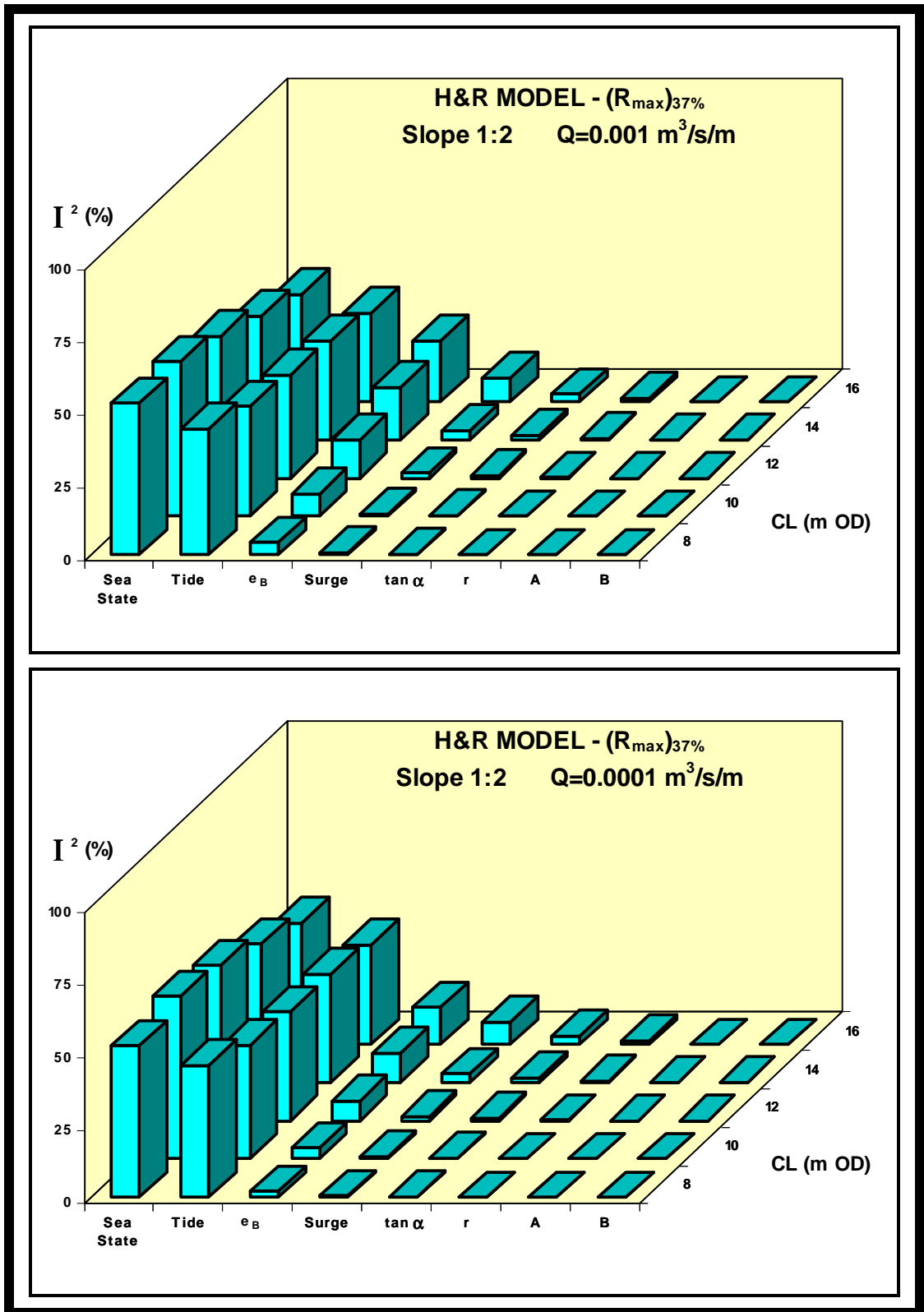
Figure 6.4: continued.
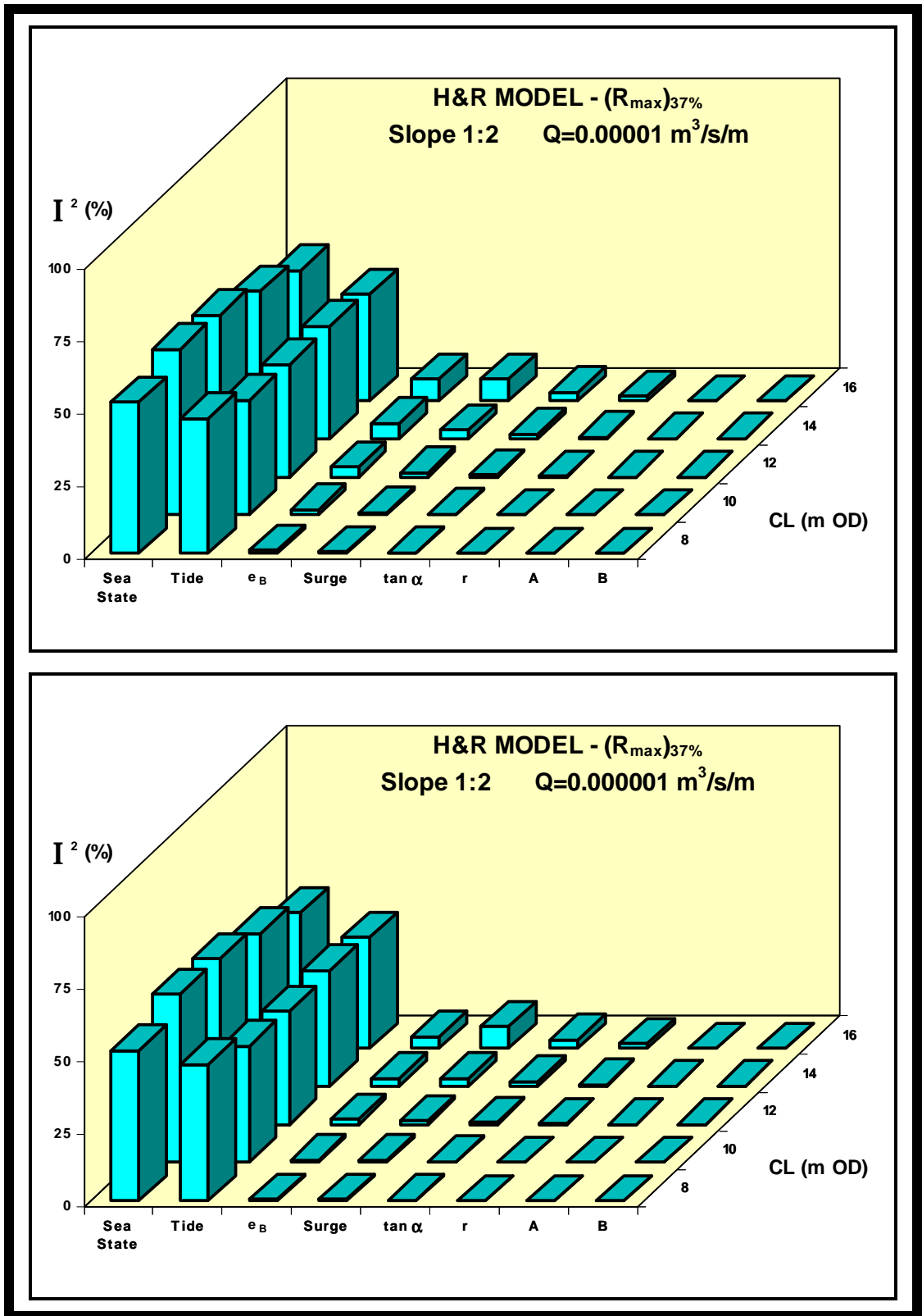
Figure 6.4: continued.
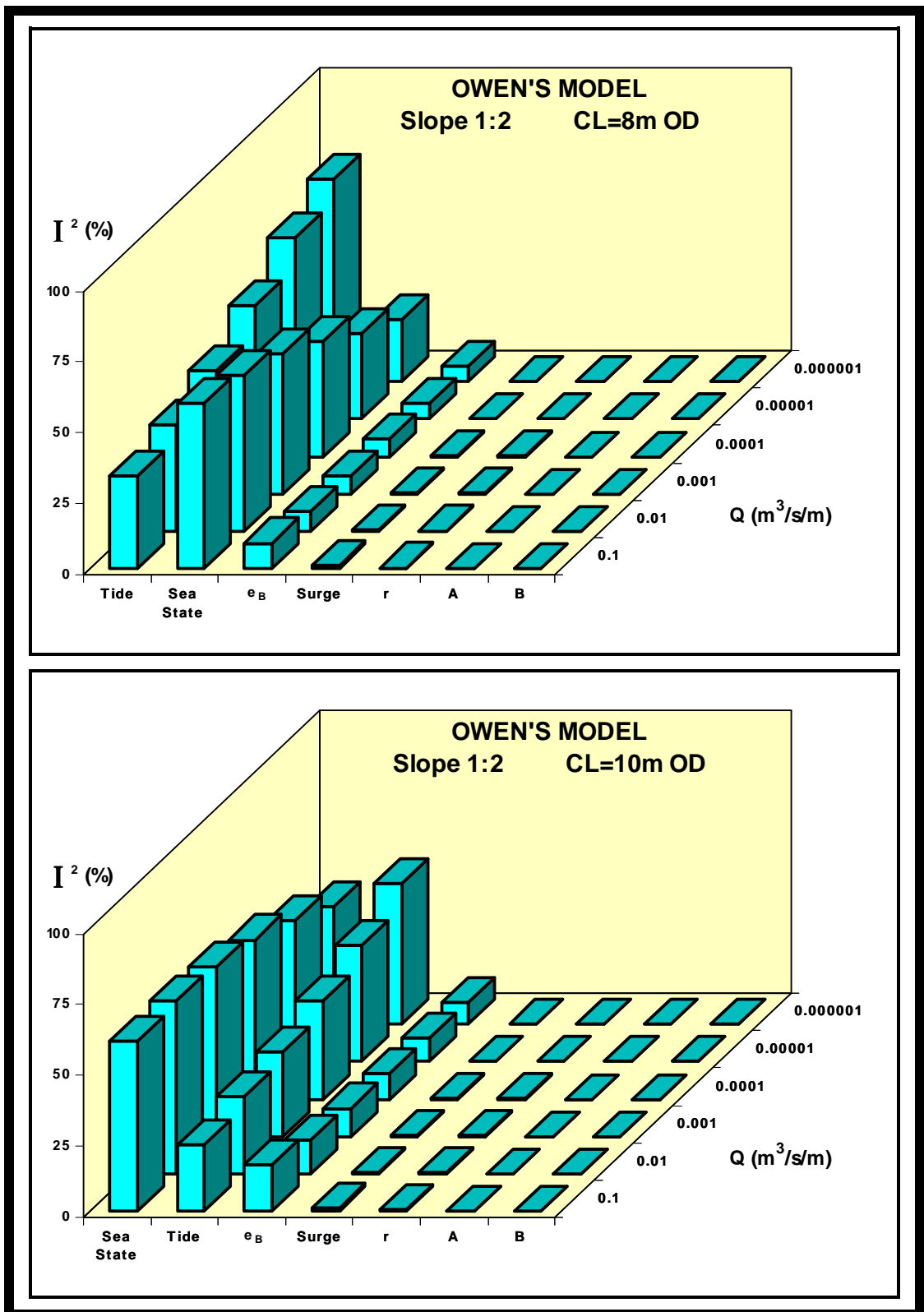
Figure 6.5: Sensitivity of the probability of failure to inaccuracies in the values of Owen's model basic variables at the design point as a function of the allowable discharge.
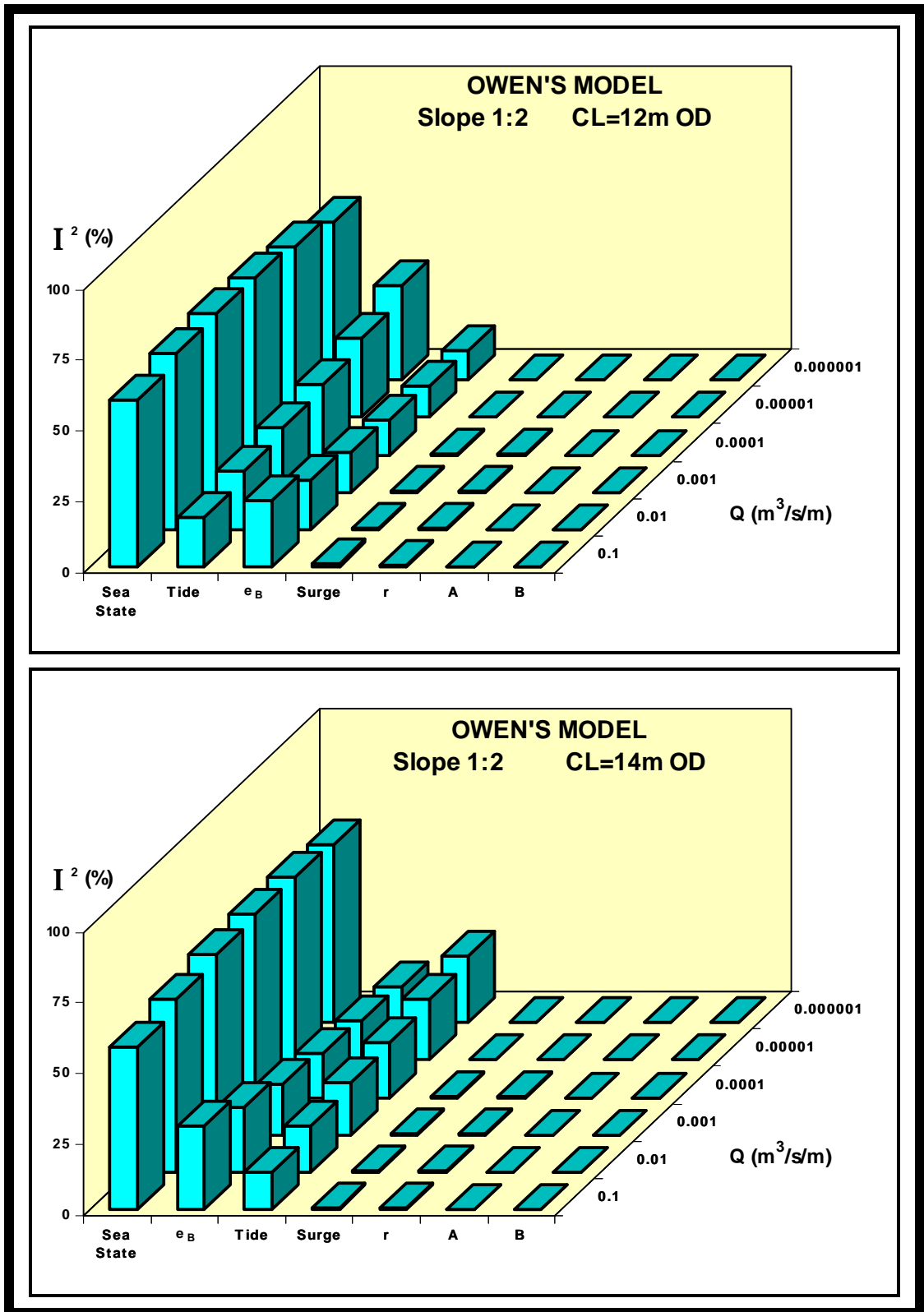
Figure 6.5: continued.
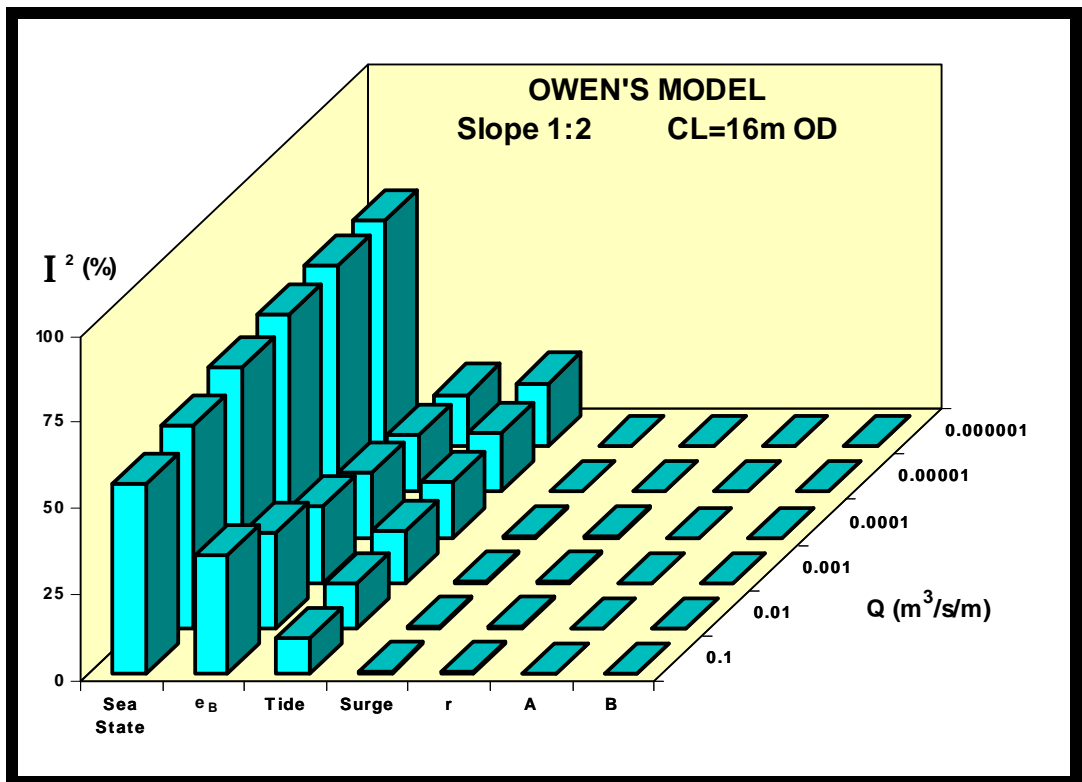
Figure 6.5: continued.
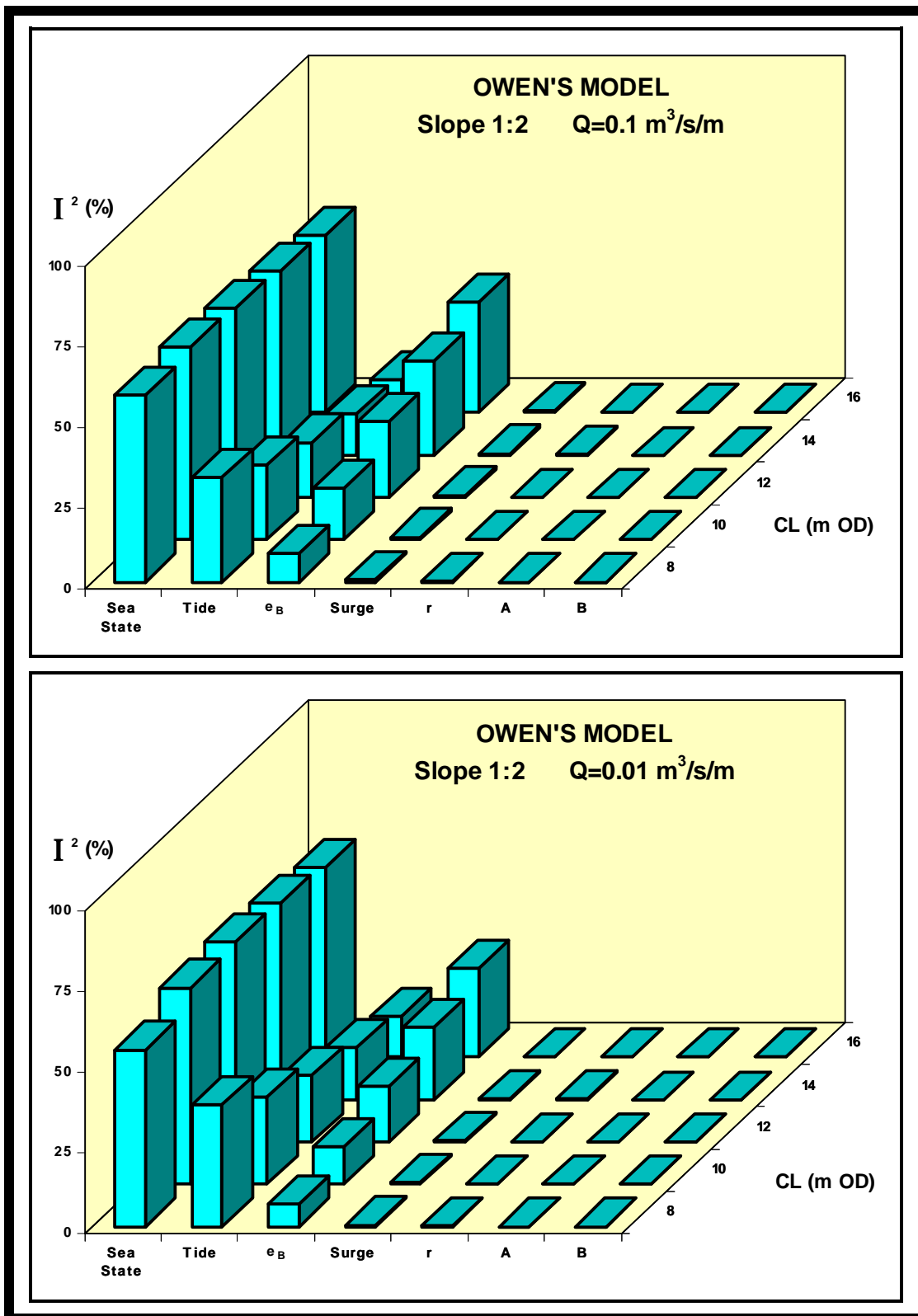
Figure 6.6: Sensitivity of the probability of failure to inaccuracies in the values of Owen's model basic variables at the design point as a function of the seawall crest level.
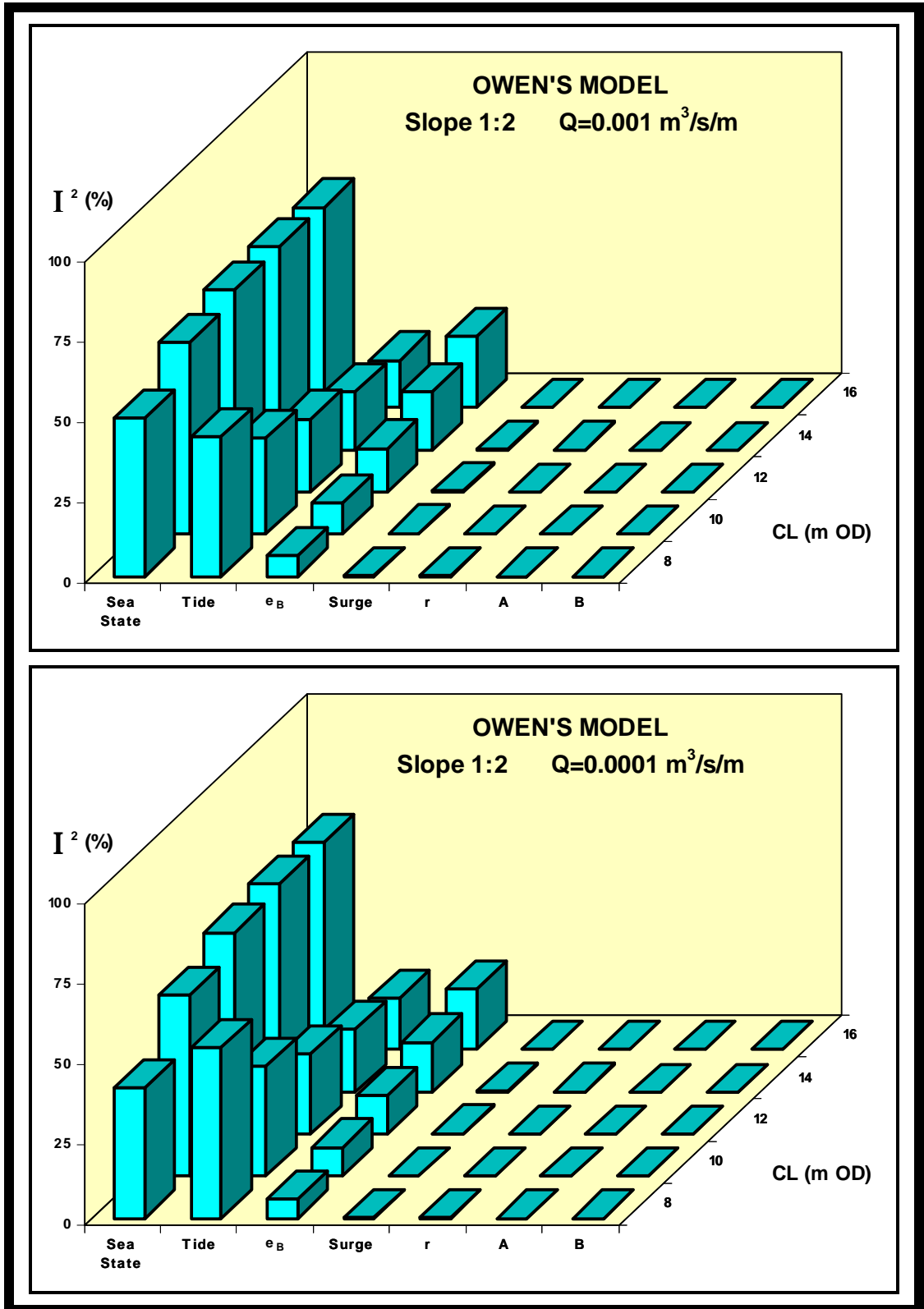
Figure 6.6: continued.
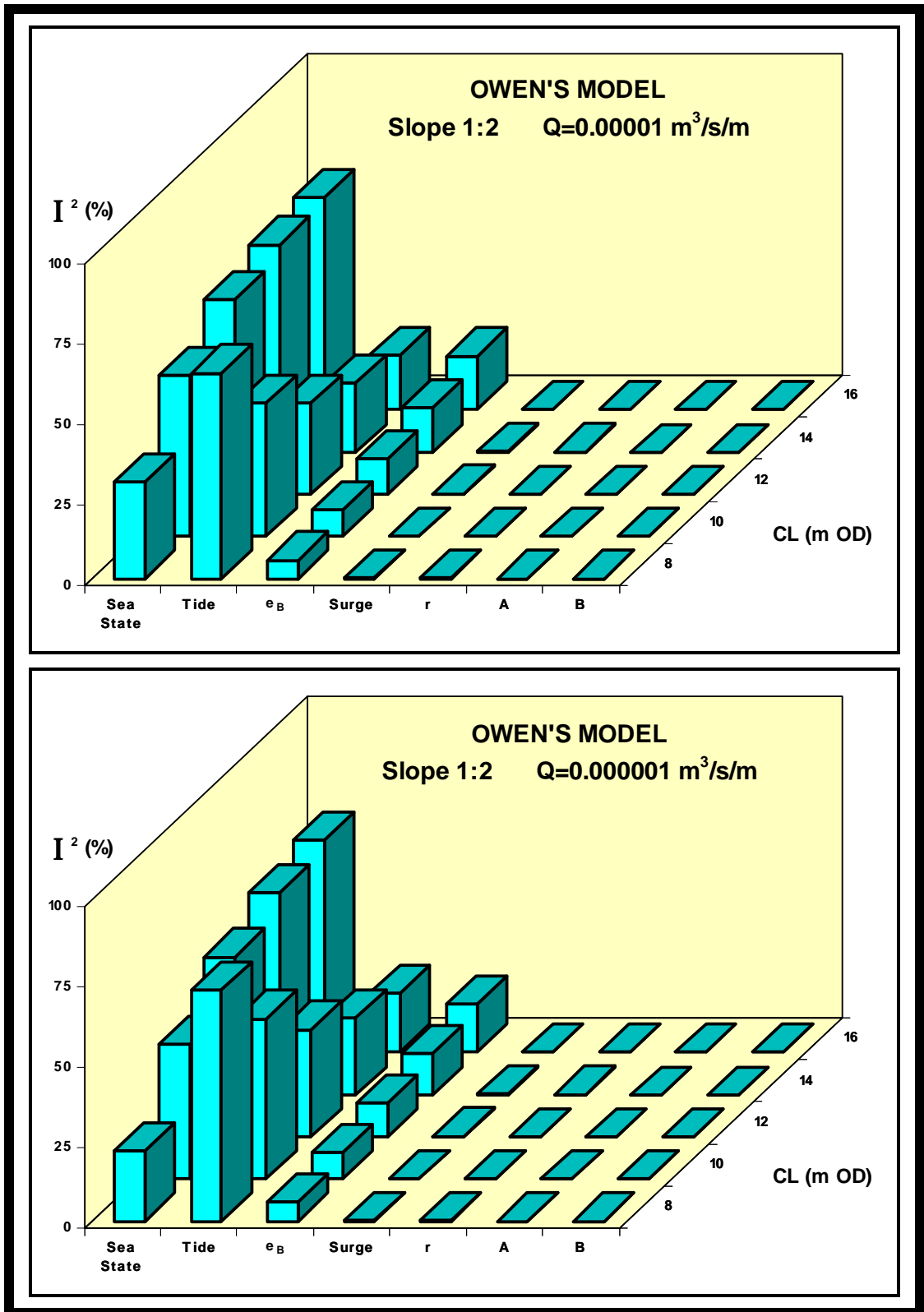
Figure 6.6: continued.
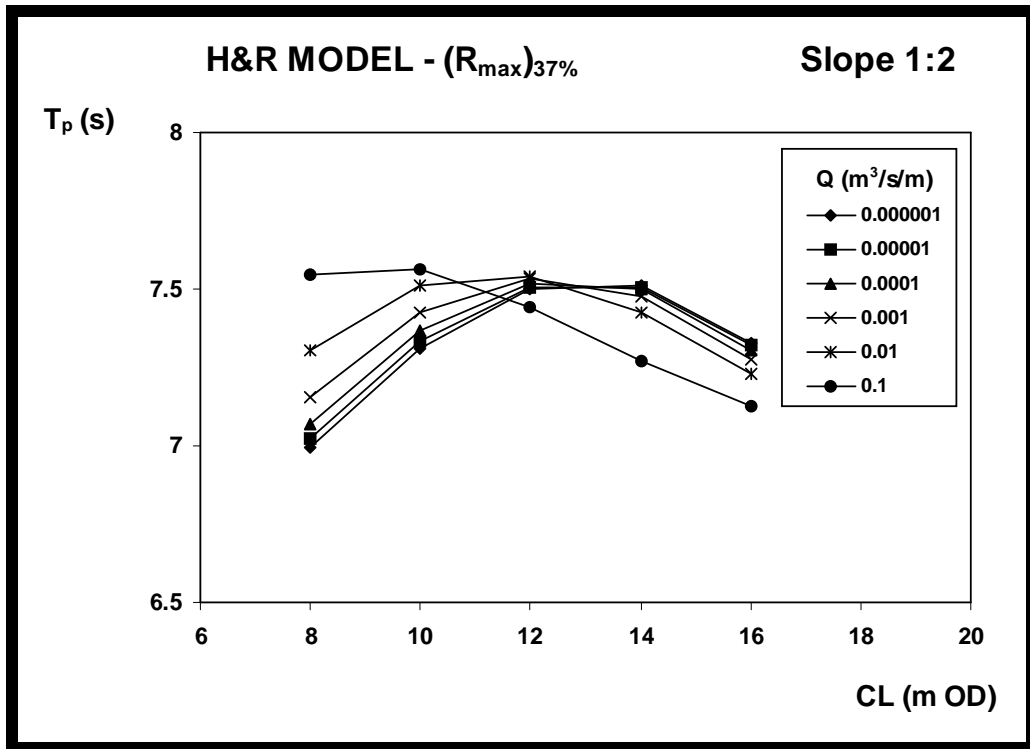
Figure 6.7: Value of T_p at the design point as a function of the seawall crest level and the allowable discharge, for the H&R model.
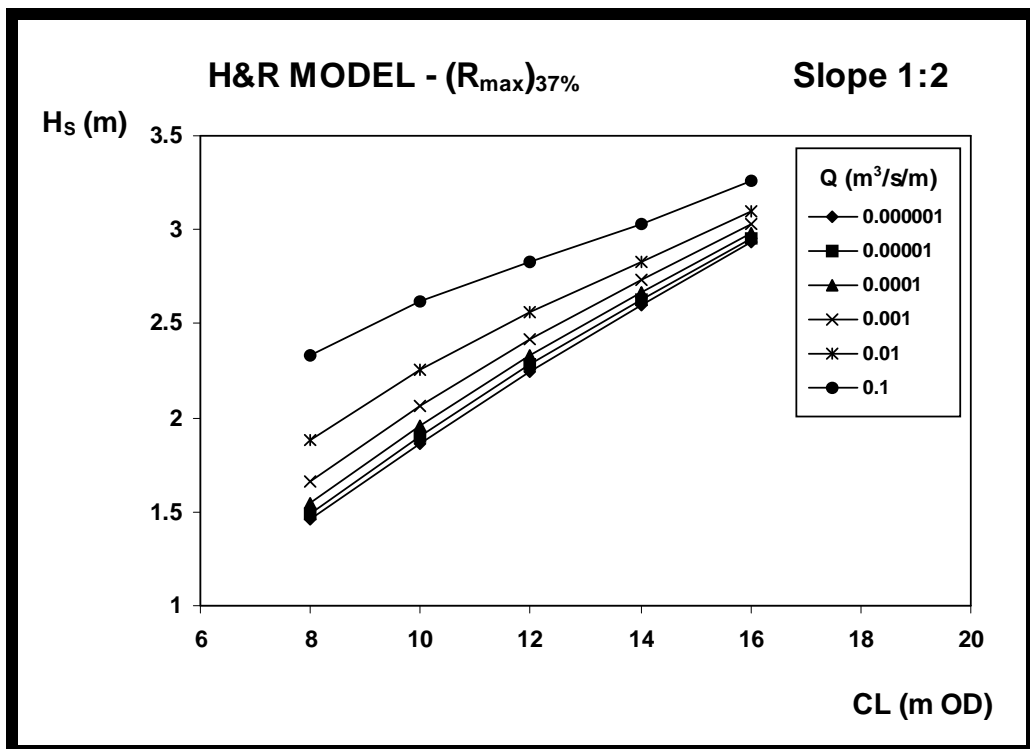
Figure 6.8: Value of H_S at the design point as a function of the seawall crest level and the allowable discharge, for the H&R model.
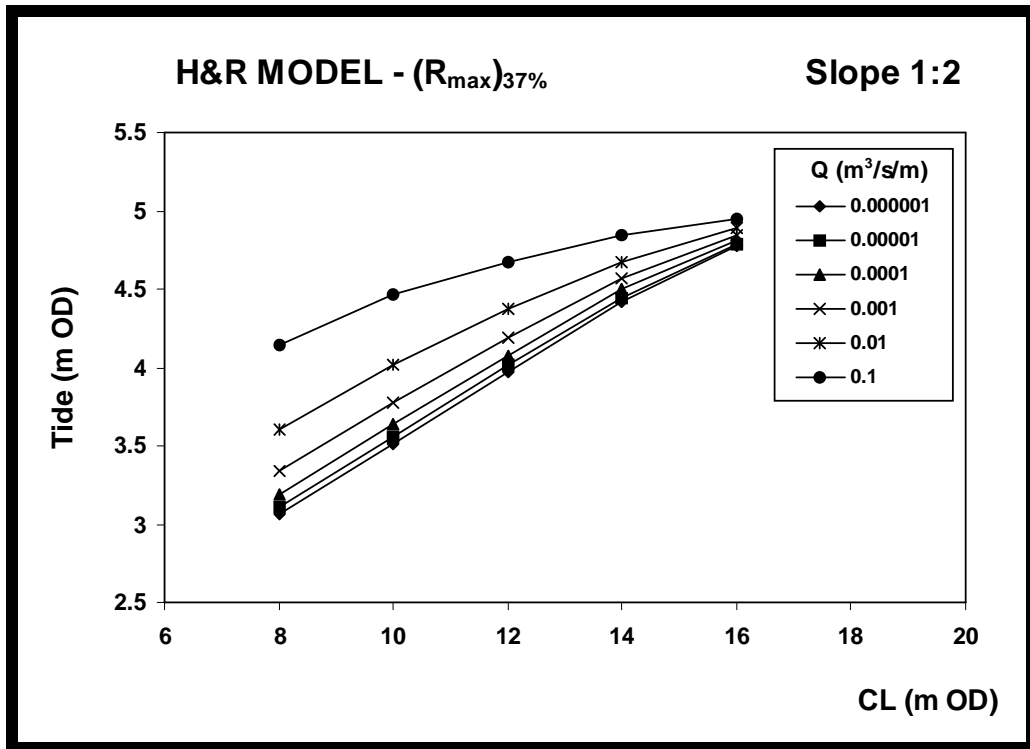
Figure 6.9: Value of Tide at the design point as a function of the seawall crest level and the allowable discharge, for the H&R model.
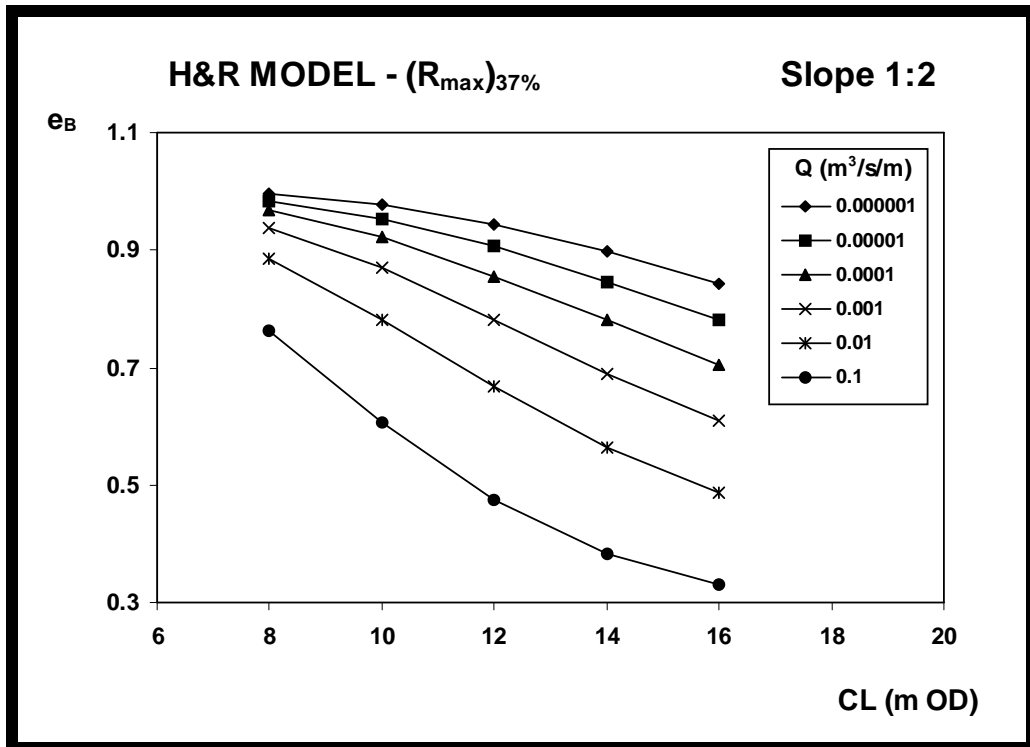
Figure 6.10: Value of e_B at the design point as a function of the seawall crest level and the allowable discharge, for the H&R model.
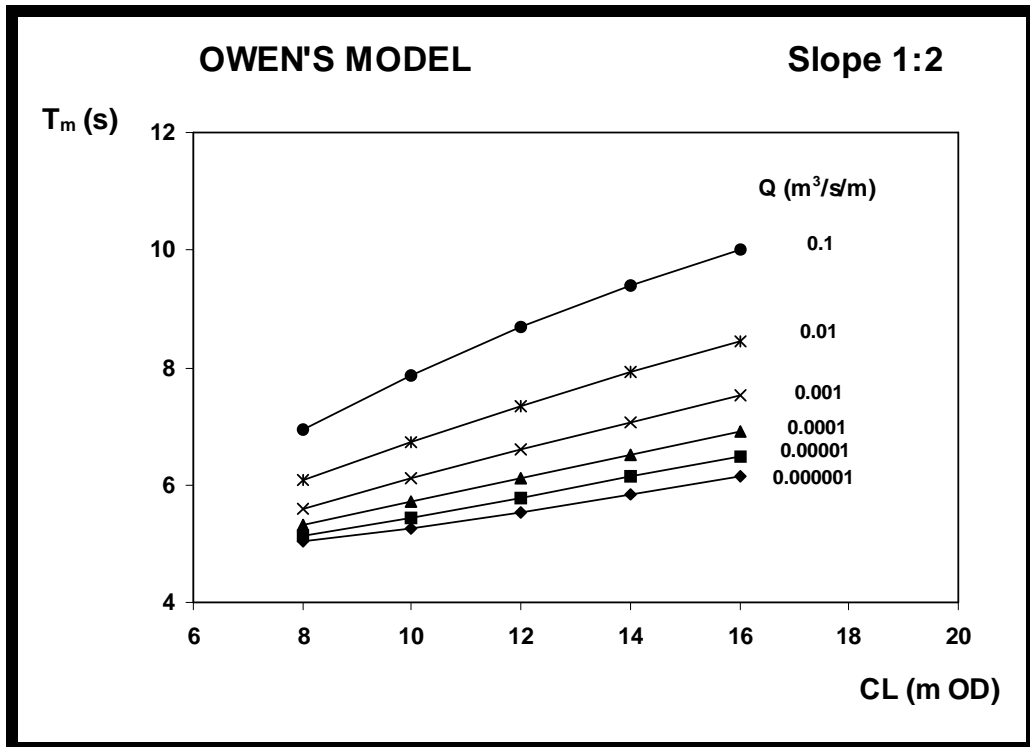
Figure 6.11: Value of T_m at the design point as a function of the seawall crest level and the allowable discharge, for Owen's model.
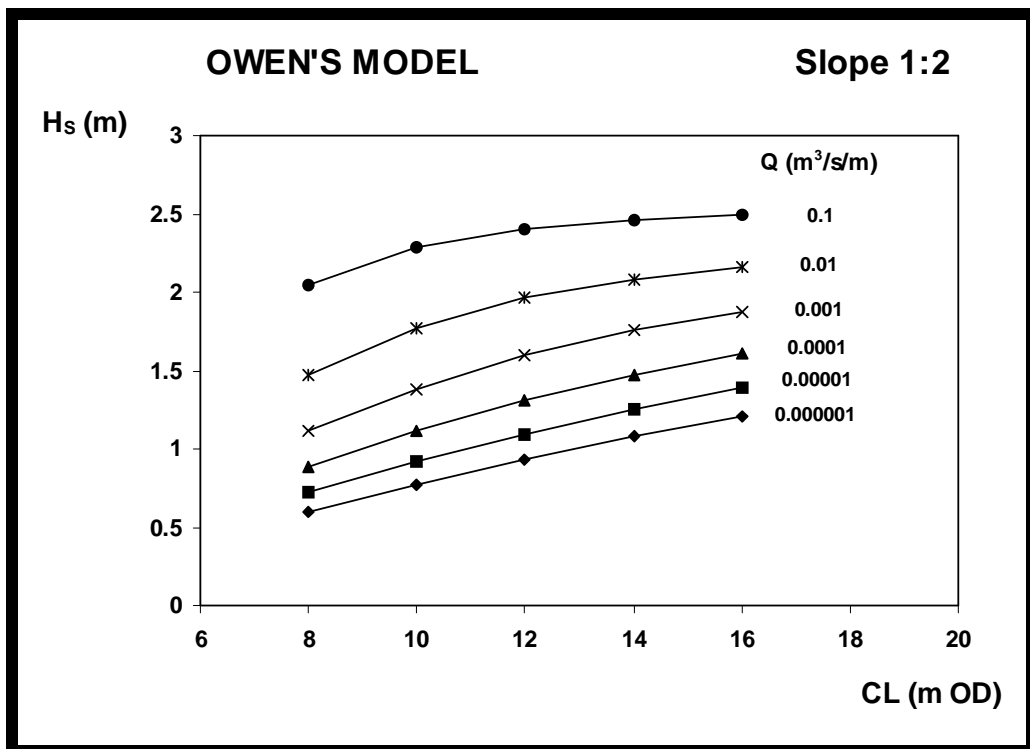
Figure 6.12: Value of H_S at the design point as a function of the seawall crest level and the allowable discharge, for Owen's model.
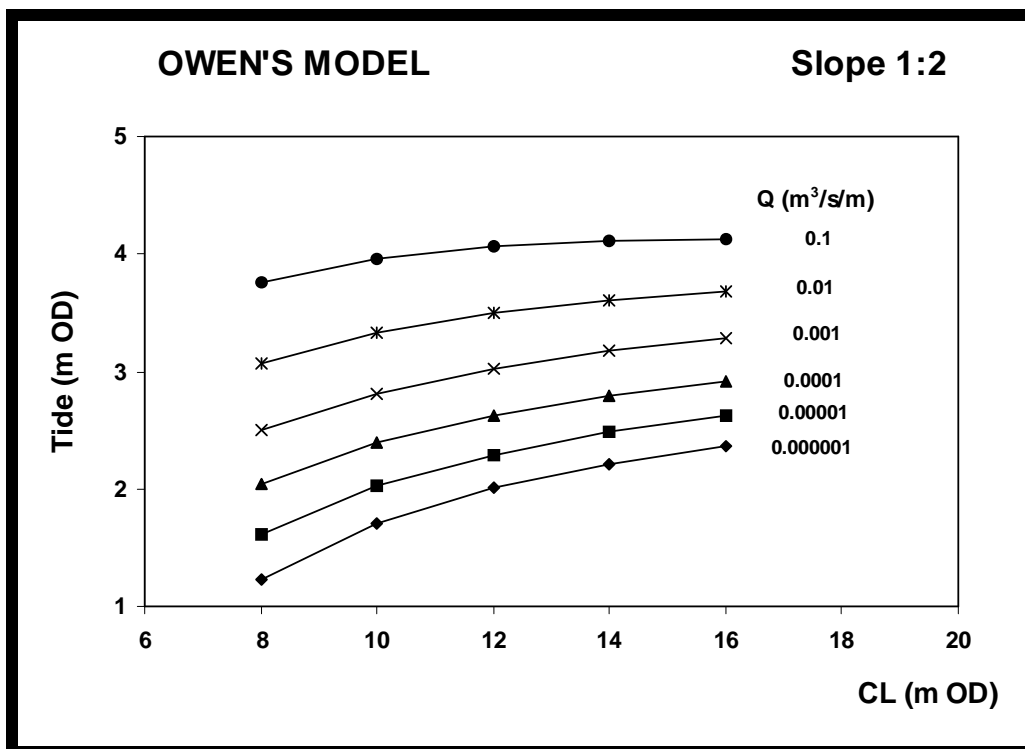
Figure 6.13: Value of Tide at the design point as a function of the seawall crest level and the allowable discharge, for Owen's model.
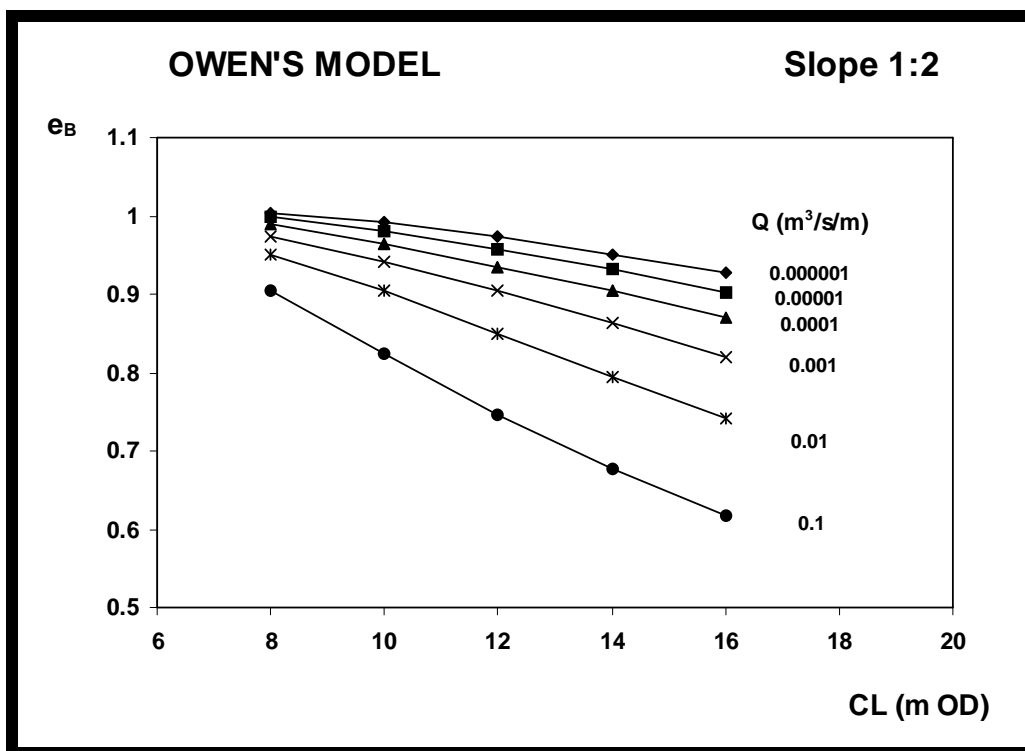
Figure 6.14: Value of e_B at the design point as a function of the seawall crest level and the allowable discharge, for Owen's model.

6.1.2.2 Extreme Conditions

The results produced by PARASODE for wave overtopping under extreme conditions are presented in tabular form in Appendix D2. These results could also have been presented graphically, as in Section 6.1.2.1. However, for simplicity, only the main conclusions are drawn here.

From the tables in Appendix D2, the following observations may be made:

- As anticipated, the probabilities of failure associated with extreme conditions are much higher than those for normal conditions. For the same values of the crest level and the allowable discharge, Owen's model still predicts higher probabilities of failure than the H&R model. For the H&R model the probabilities of failure are generally higher for $(R_{max})_{99\%}$ than for $(R_{max})_{37\%}$. Consequently, for the extreme conditions, a crest level greater than 14m would be required to satisfy structural safety according to the H&R model whilst Owen's model would demand a crest level greater than 16m. As for normal conditions, P_f (%/year) decreases as the crest level of the seawall increases. Likewise, for the same value of the crest level, P_f decreases as the allowable discharge increases.

- For the H&R model, the sensitivity parameters, α^2 (%), show again that the major influence on the probability of failure is generally provided by the uncertainty in the sea state (up to 99%). Unlike for normal conditions, the water level is much less important (up to 10%), whilst for the biggest allowable discharges, the model parameter, e_B, still occasionally plays a major role (up to 49%). The effect of the other variables is negligible. For each value of the seawall crest level, the importance of the sea state tends to increase as the allowable discharge decreases, while the effect of e_B decreases. For each allowable discharge, the behaviour of the sea state and e_B show no obvious relationship to the crest level.

- Once more, the sensitivity parameters for Owen's model show that the most important factor is the sea state (up to 95%). Model parameter e_B represents, in some instances, the most significant contribution (up to 58%); the effect of e_B is greatest for the smallest allowable discharges. The other variables make only minor contributions to the resulting variance. The influence of the sea state shows no obvious relationship to the crest level or allowable discharge.

- The behaviours at the design point of the values of the most important variables in relation to the crest level and allowable discharge are identical to the behaviours described for normal conditions.

- As for normal conditions, it can be seen that the choice of overtopping model is very important in the probability assessment of the safety of seawalls subjected to extreme conditions. The two models lead to quite different results. Once again, use of Owen's model in design would be more conservative than use of the H&R model.

6.1.3 Validating PARASODE Results Using @Risk

The accuracy of the Level II (FORM) reliability algorithms used in PARASODE has been evaluated by comparison with the results provided by the Level III method of Latin Hypercube Sampling (LHS) available in @Risk. This evaluation has been carried out for the Level II results of the H&R model, $(R_{max})_{37\%}$, and Owen's model, for normal conditions only. The results of the Level III method are used as a benchmark, since their only limitation is the computer time needed to perform a sufficiently large number of iterations (Jang et al, 1994).

In simulation, with more iterations, output distributions become increasingly stable as the statistics describing each input distribution change less with additional samples. It is important to run enough iterations so that the output statistics are reliable. However, there comes a point when the time spent on additional iterations is unnecessary because the output statistics are not significantly changed. The number of iterations required to generate stable output distributions varies depending on the model used in the simulation and the distribution functions in the model.

Like other research (Startzman & Wattenbarger, 1985; Super-Software, 1994), this work concentrates on the variability of the following statistics as measures of simulation convergence (Figure 6.15):

- mean;

- standard deviation;

- coefficient of variation.
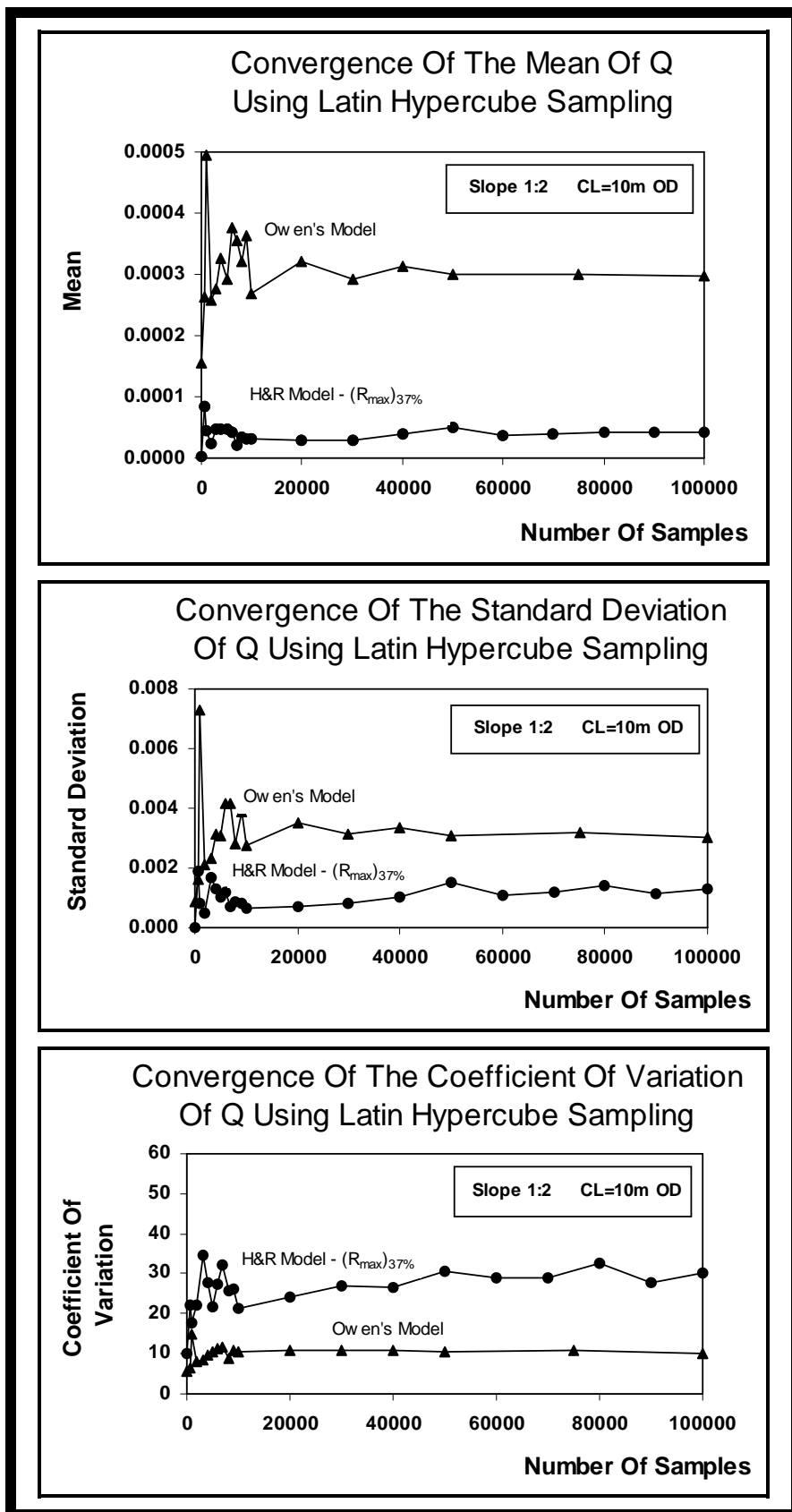
Figure 6.15: Example of the convergence of the mean, standard deviation and coefficient of variation of Q using Latin Hypercube Sampling for the H&R model and for Owen's model.

Other statistics, such as the skewness and the kurtosis, have also been analysed. The results are not shown here because they behaved in similar ways to the above statistics.

Figure 6.15 shows that estimates of the mean, standard deviation and coefficient of variation are sufficiently accurate after about 30000 samples for Owen's model, while the H&R model does not approach reasonably converged statistics until about 60000 samples are used. Simulating more samples would not introduce a noticeable improvement on the calculations, and would only require more time and computer memory. Several other LHS simulations were performed using random seeds and similar results were obtained. Therefore, it was decided to carry out the simulations for the H&R model and Owen's model using 60000 and 30000 samples, respectively. In practice, Owen's model required only about 35% of the computer time required by the H&R model, for the input conditions considered. However, even 30000 samples is a considerable number, especially when LHS has been used instead of the Traditional Sampling method.

Many other statistical properties may be determined by simulation including confidence limits. Often, the entire cumulative distribution of the result is required. However, there is no reason to believe that the convergence of the above statistics would not be accompanied by the corresponding convergence of other properties (Startzman & Wattenbarger, 1985; Law & Kelton, 1991).

The convergence of the probability of failure for different allowable discharges has also been analysed. Figure 6.16 suggests that 60000 samples for the H&R model and 30000 for Owen's model are more than sufficient.

The distributions of the basic variables obtained during the simulation were checked against the input distributions. Agreement was excellent, as expected with the large number of samples involved in each simulation for the H&R model and for Owen's model.

Figures 6.17 and 6.18 show the probability of failure obtained using the Level II and Level III methods plotted against the crest level of the seawall for particular values of the allowable discharge, both for the H&R model,

$(R_{max})_{37\%}$ and for Owen's model. The two methods give comparable results. However, as the crest level decreases, the FORM results diverge from the LHS results, generally overestimating the probability of failure. The differences in the probability of failure between FORM and LHS results suggest that the failure surfaces for both models are curved near the design point. In such cases, a Second Order Reliability Method (SORM) would be expected to account for the non-linearity of the failure surface and to remain accurate, giving results consistent with the LHS method, while still providing sensitivity factors and other details which do not depend on the magnitude of the probability of failure. Note that in the present work the FORM calculations took only between about 5% and 20% of the computer time required for the LHS computations.
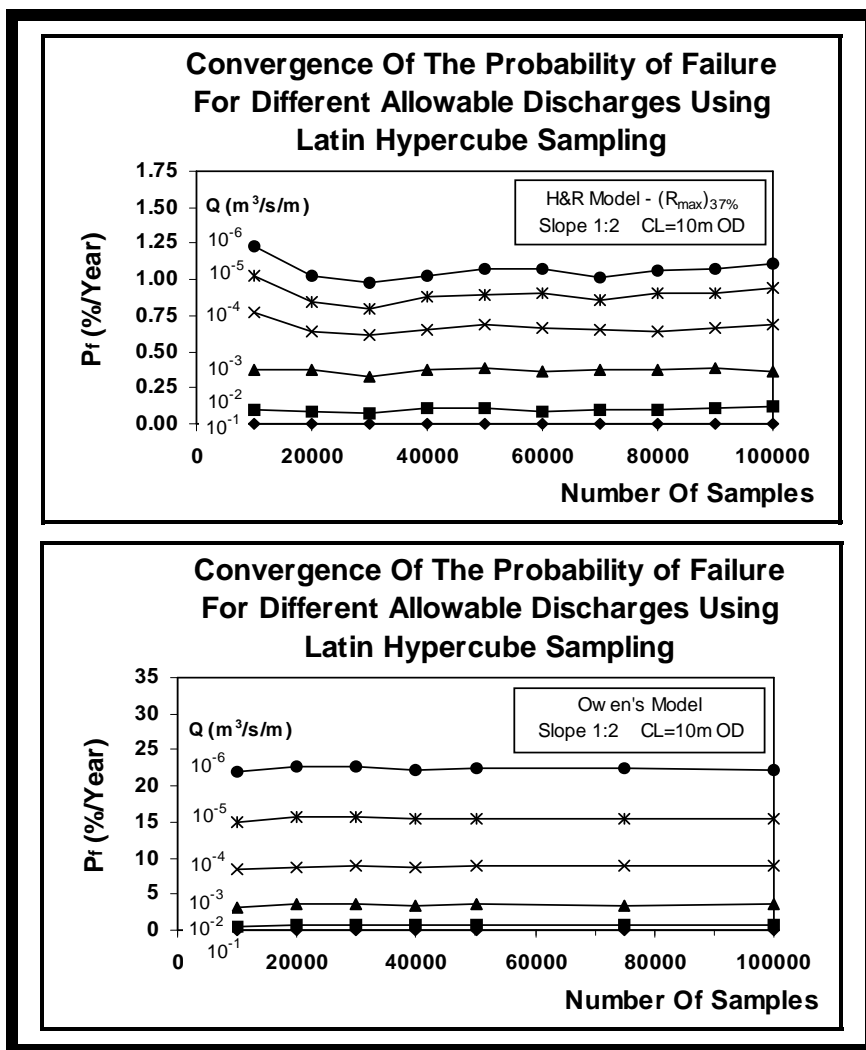
Figure 6.16: Example of the convergence of the probability of failure for different allowable discharges using Latin Hypercube Sampling for the H&R model and for Owen's model.
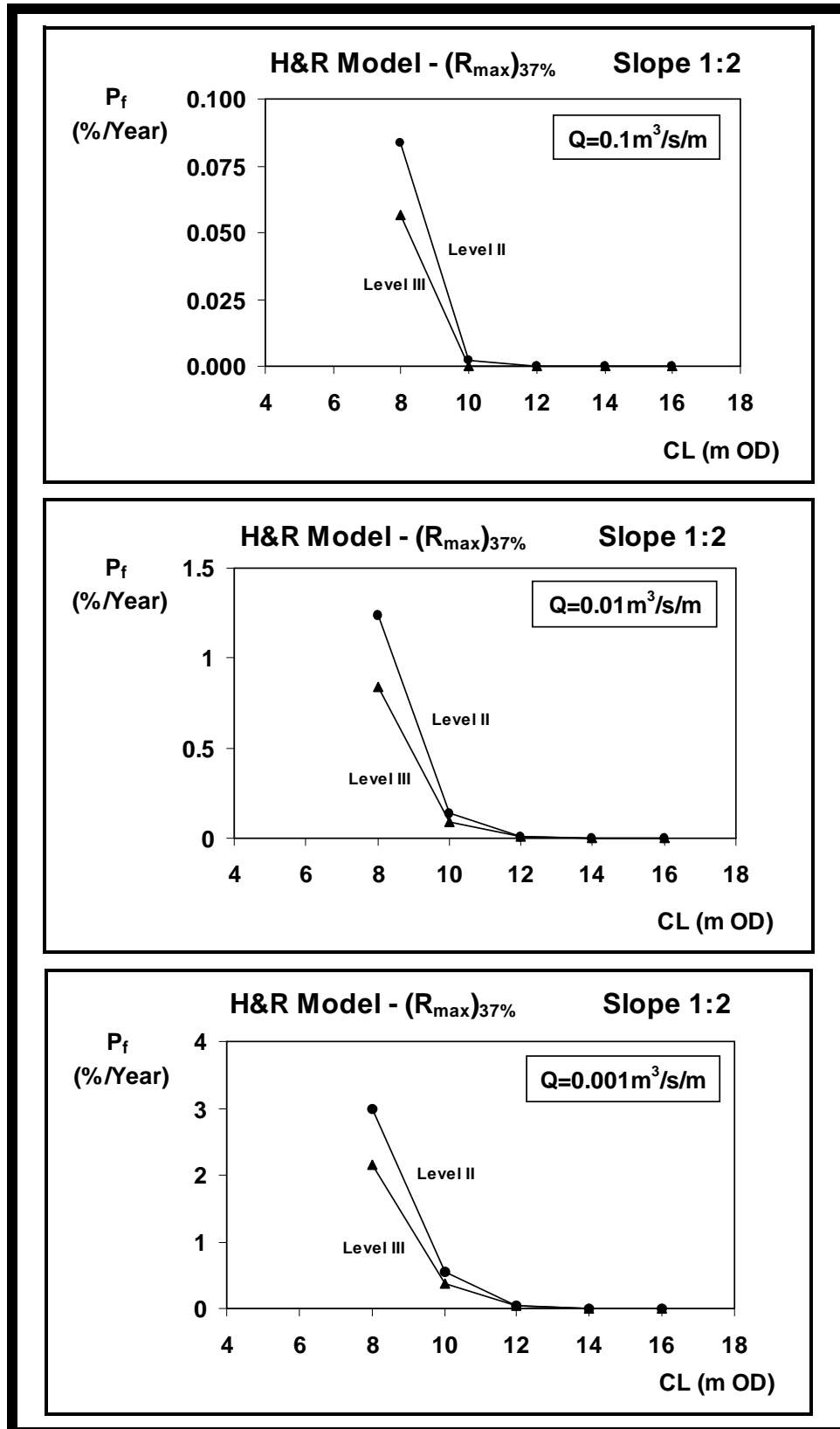
Figure 6.17: Comparison of the probability of failure obtained using Level II (FORM) and Level III (LHS) methods for particular values of the allowable discharge, for the H&R model, $(R_{max})_{37\%}$.

H&R Model - (R$_{max}$)$_{37\%}$ **Slope 1:2**

P_f (%/Year)

Q=0.0001m^3/s/m

Level II

Level III

CL (m OD)

H&R Model - (R$_{max}$)$_{37\%}$ **Slope 1:2**

P_f (%/Year)

Q=0.00001m^3/s/m

Level II

Level III

CL (m OD)

H&R Model - (R$_{max}$)$_{37\%}$ **Slope 1:2**

P_f (%/Year)

Q=0.000001m^3/s/m

Level II

Level III

CL (m OD)
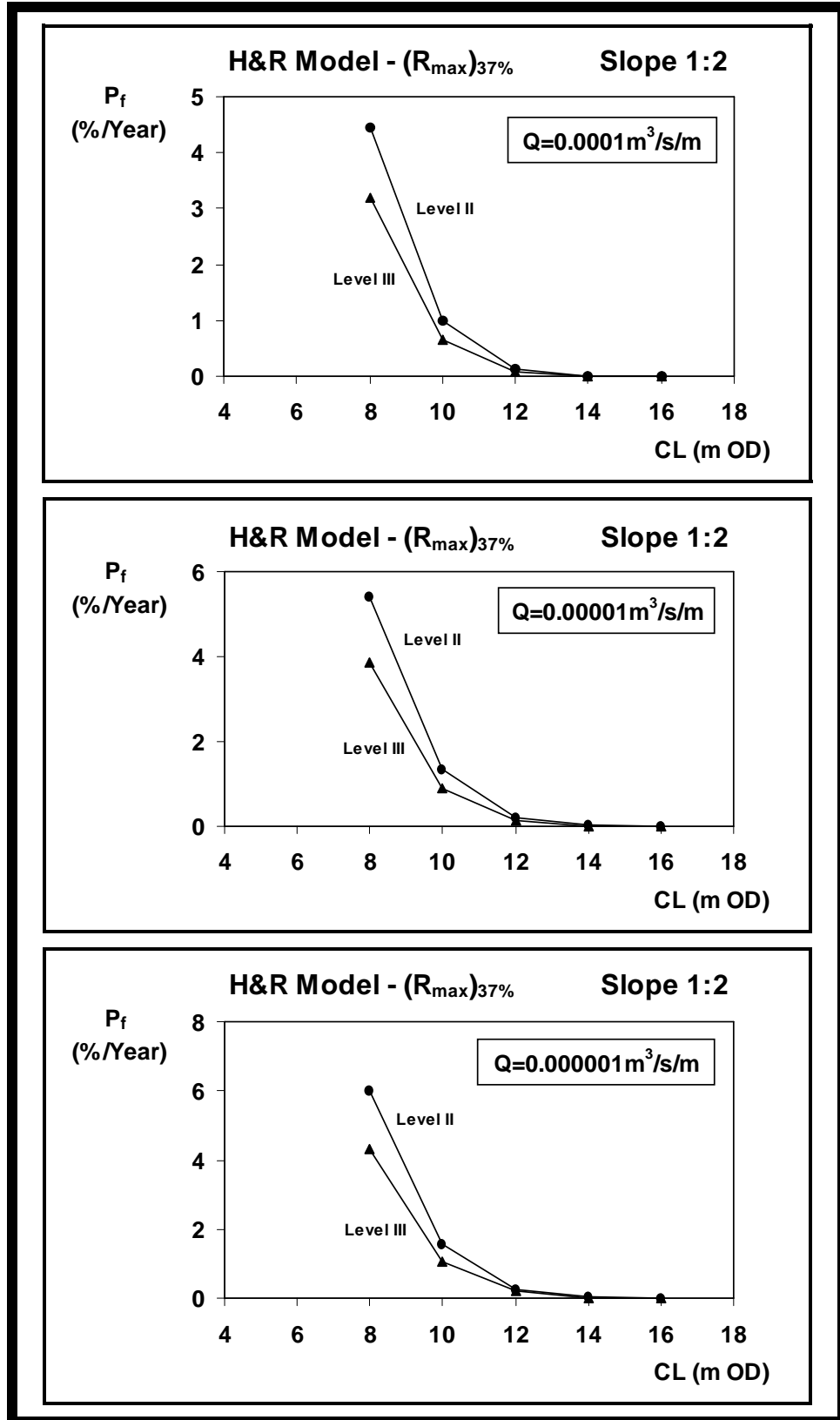
Figure 6.17: continued.
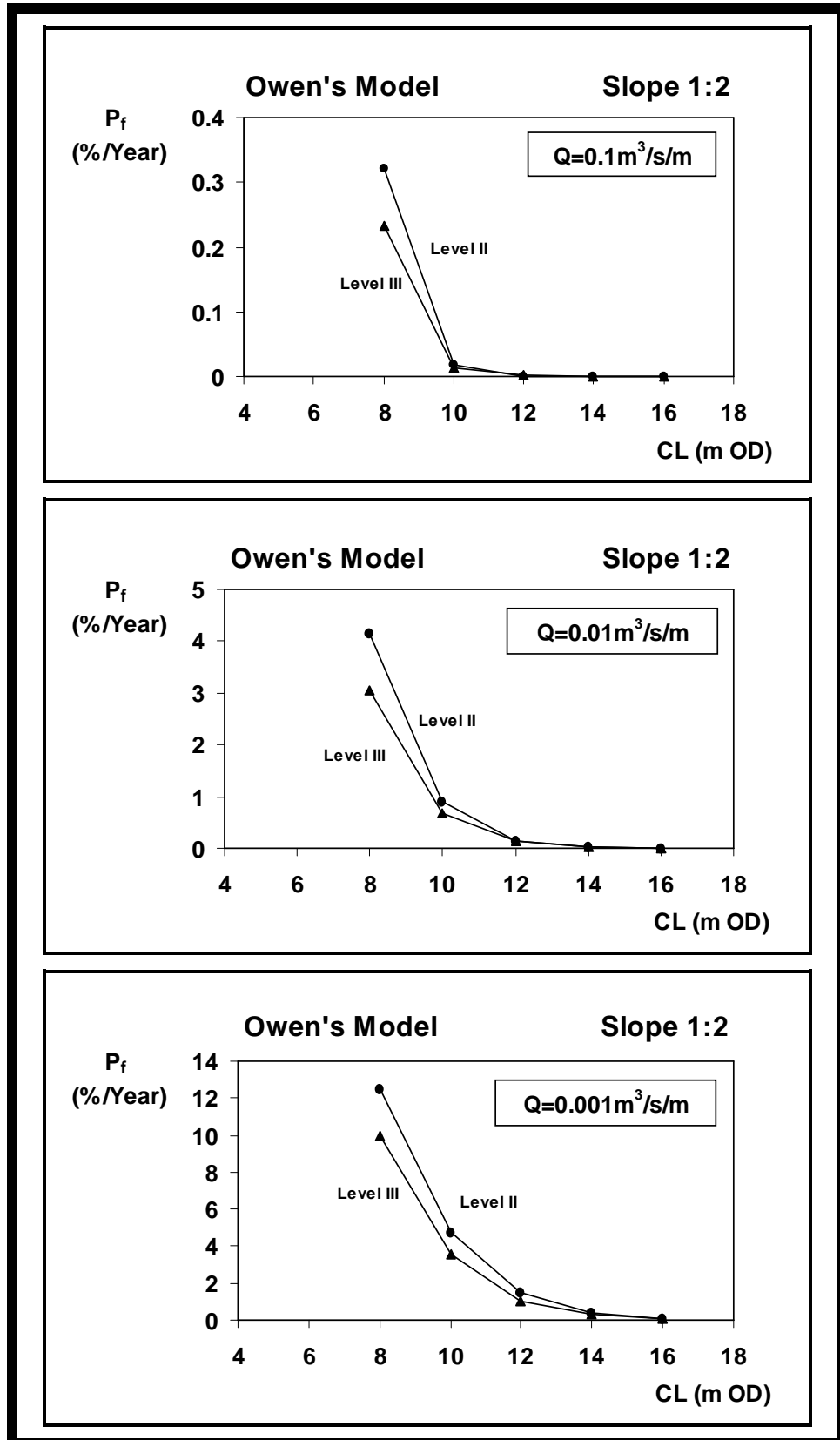
Figure 6.18: Comparison of the probability of failure obtained using Level II (FORM) and Level III (LHS) methods for particular values of the allowable discharge, for Owen's model.
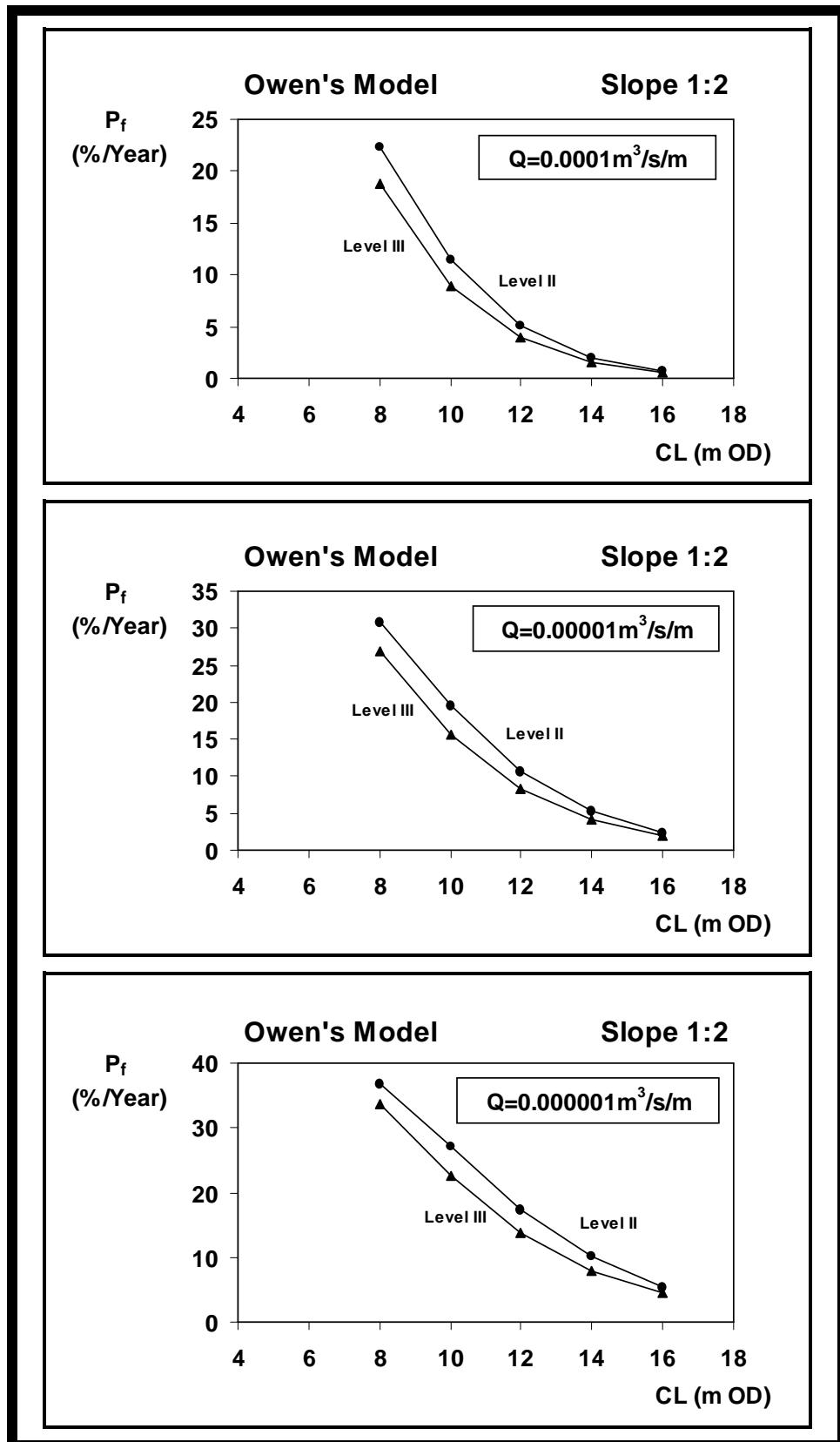
Figure 6.18: continued.

6.2   Examples - Dune Erosion

6.2.1   General Cases Of Dune Erosion

6.2.1.1 Input To PARASODE

Table 6.2 summarises the dune erosion examples studied using PARASODE. In each example, only one parameter (e.g. the initial profile or the allowable erosion distance) has been changed as shown in the table. The remaining input data is common to all examples. The input data files used in example 1 are provided in Appendix D3 as well as the corresponding output file *summary.dat*.

EXAMPLE	PROFILE	NOURISHMENT	ALLOWABLE EROSION DISTANCE, TR (m)	MODE
1	A1	No	90	1
2	B1	No	90	1
3	C1	No	90	1
4	D1	No	90	1
5	A1	Yes nourwidt=75m nourtlev=6m 1:mnour=1:1.5	90	1
6	A1	No	140	1
7	A1	No	130	1
8	A1	No	120	1
9	A1	No	110	1
10	A1	No	100	1
11	A1	No	80	1
12	A1	No	70	1
13	A1	No	60	1
14	A1	Yes nourtlev=6m 1:mnour=1:1.5	90	2 P_f(%)=0.001011

Table 6.2: Dune erosion examples (note that movements of sand have been allowed only seaward in these examples).

Profiles A1, B1, C1 and D1 (Figure 6.19) have been used to illustrate the main erosion situations which can be studied using PARASODE (see Section 5.2.5).
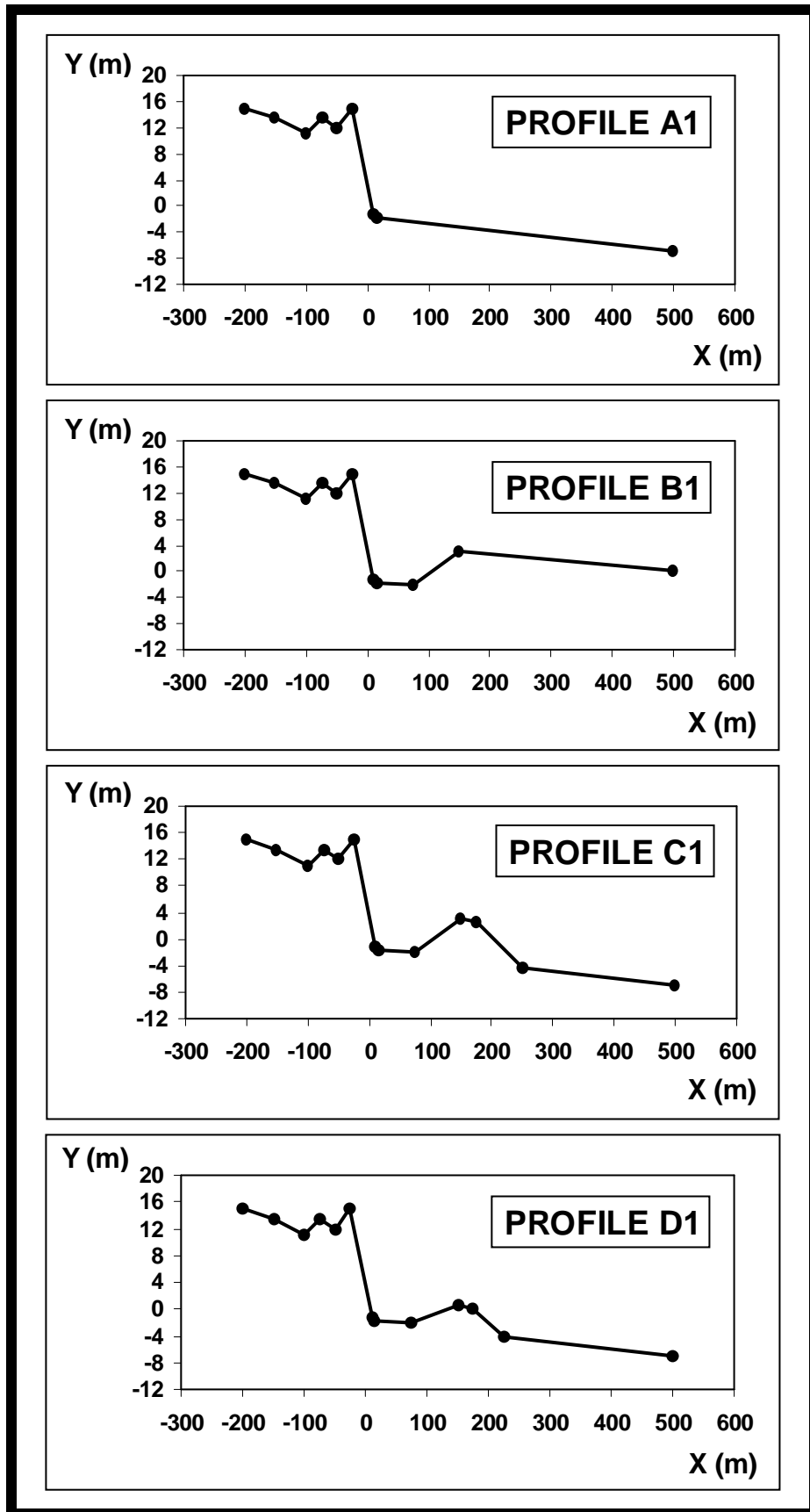
Figure 6.19: Initial profiles used in dune erosion examples 1 to 14.

The adopted characteristics of the random variables are shown in Table 6.3 and are typical of Dutch conditions (see Chapter 4 and Appendix D3).

Variable	Distribution	μ	σ	Lower Limit
H_S	Normal	0	0.6	----------
D_{50}	Normal	225E-6	225E-7	----------
DP	Normal	0	0.6	----------
SD	Normal	0	1	----------
GB	Normal	0.4	0.1	----------
Ac	Normal	0	1	----------
h	Weibull	2.52	0.33	2.19

Table 6.3: Characteristics of the random variables adopted for examples 1 to 14.

Correlation between h and H_S has been considered as follows (Van de Graaff, 1986):

$$\mu_{H_S|h} = 4.82 + 0.6h - 0.0063(7-h)^{3.13} \qquad 3 \leq h < 7$$
$$\mu_{H_S|h} = 4.82 + 0.6h \qquad\qquad\qquad\qquad h > 7 \qquad\qquad (6.1)$$

where $\mu_{H_S|h}$ is the mean value of H_S given a value for h. Hence, H_S has been modelled as the sum of $\mu_{H_S|h}$ and the variability of H_S about its mean value (see Section 4.5.2). This variability has been considered as Normal distributed with a mean of zero and a standard deviation of 0.6m.

Note that PARASODE also allows the user to consider correlation between h and H_S by providing a non-zero correlation coefficient between the two variables. Unlike DUNEPROB and DUNE, independence between h and H_S can be ensured by adopting a value of zero for the correlation coefficient.

6.2.1.2 Results And Discussion

The results from PARASODE for dune erosion are shown in tabular form in Appendix D4. This section provides a graphical illustration of some of the results.

Examples 1 to 4 show how different initial profiles can affect the final results (see Tables D4.1 to D4.4). From the sensitivity parameters, $\alpha^2(\%)$, the most important contribution to the resulting variance is given by the maximum water level during surge, h (Figure 6.20). Values for $\alpha^2(\%)$ of about 80% to 90% indicate that this variable is by far the most important one. The sediment size, D_{50}, and the accuracy of the computation, Ac, also make some significant contribution to the resulting variance. The contribution of the surge duration is less important and the significant wave height, H_S, the change in the initial profile, DP, and the gust bumps, GB, make only minor contributions. It might be expected that H_S would play a strong role in the effects of erosion. Note, however, that due to the relationship between h and H_S, the contribution of H_S shown in the tables represents only the effect of the variability in H_S about its expected value.
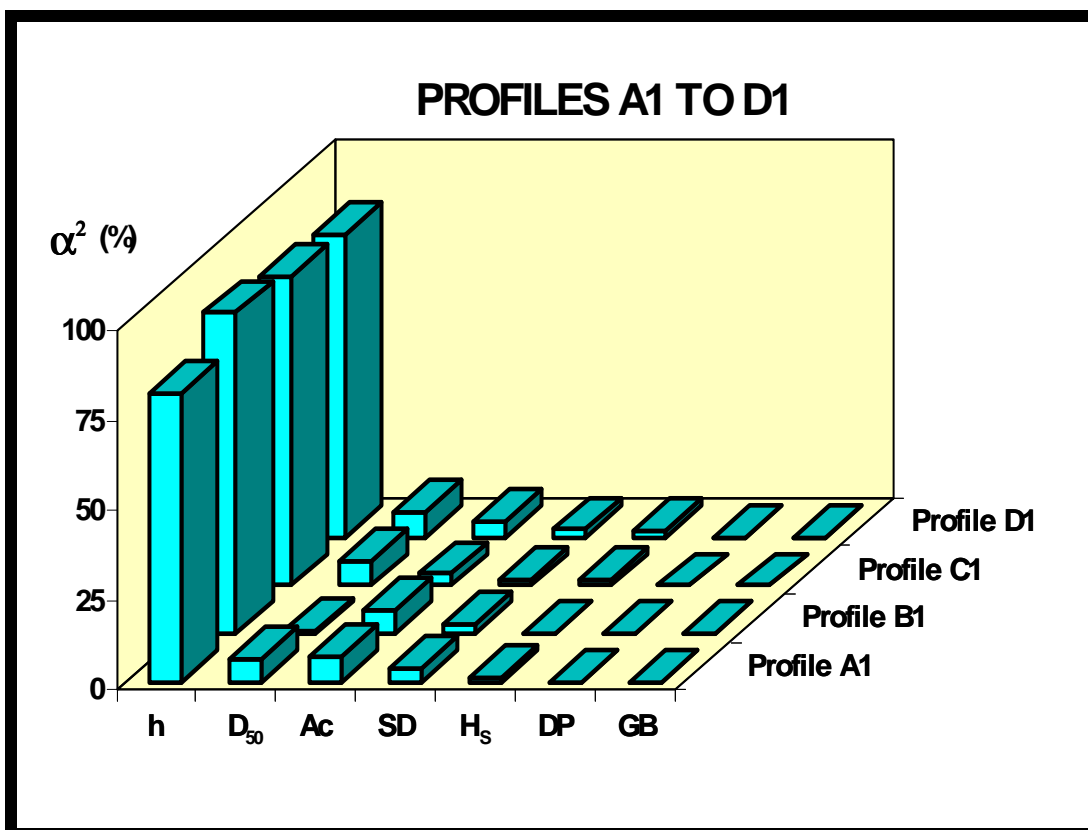
Figure 6.20: Sensitivity of the probability of failure to inaccuracies in the values of the erosion basic variables at the design point (examples 1 to 4).

Example 5 illustrates how nourishment can be used to decrease the dune's failure probability. The nourishment characteristics applied to profile A1 are: i) width at top level = 75m; ii) top level = 6m; and iii) gradient of the nourished face = 1:1.5 (Table 6.2). If nourishment is not provided (example 1), the probability of failure is approximately 0.06%/year (Table D4.1). This probability has been reduced to about 0.001%/year by nourishment. Example 14 represents the same conditions as example 5 but PARASODE has been run in mode 2; i.e. a target probability of failure of about 0.001%/year has been input and a corresponding nourishment width has been computed (74.87m). Examples 5 and 14 demonstrate the converse nature of modes 1 and 2 and show consistency between results obtained in running PARASODE in both modes (Tables D4.5 and D4.14).

Examples 1 and 6 to 13 illustrate how changes in the allowable erosion distance, TR, affect the results. Values of TR between 60m and 140m have been considered since they provide probabilities of failure between about P_f=2.6%/year and P_f=0.0006%/year, respectively. This range of probabilities covers all likely normal design cases, i.e. $10^{-2} \leq P_f / year \leq 10^{-5}$ (see Section 4.5.1). A plot has been produced of the probability of failure as a function of TR (Figure 6.21). For instance, the probability of failure per year associated with an allowable erosion distance of 100m is equal to about 0.02%/year. Hence, for Dutch conditions (see Chapter 4), if a 10^{-5}/year probability of exceedance is needed, TR=100m is unacceptable. However, if one considers a higher chance of failure, e.g. 10^{-3}/year, then TR=100m is acceptable. Note that a probability of failure associated with a value of TR represents a probability of exceedance of this TR value in one year.

Assuming statistical independence of each year, the probability of failure for a T_{ref} -year period can be obtained using (Van der Meer & Pilarczyk, 1987; Van der Meer, 1990; Van der Meer et al, 1994):

$$P(Z \leq 0; T_{ref} \text{ years}) = 1 - \left[1 - P(Z \leq 0; 1 \text{year}) \right]^{T_{ref}} \qquad (6.2)$$

Results derived from Figure 6.21 using eq. (6.2) are shown in Figure 6.22. Curves are drawn for three lifetimes: 20, 50 and 100 years. From this figure it follows, for example, that the allowable erosion distance TR=100m will be

exceeded with a probability $0.5 \leq P_f(\%) \leq 1$ during a lifetime of 20 years, whilst for lifetimes of 50 and 100 years, $1 \leq P_f(\%) \leq 5$.
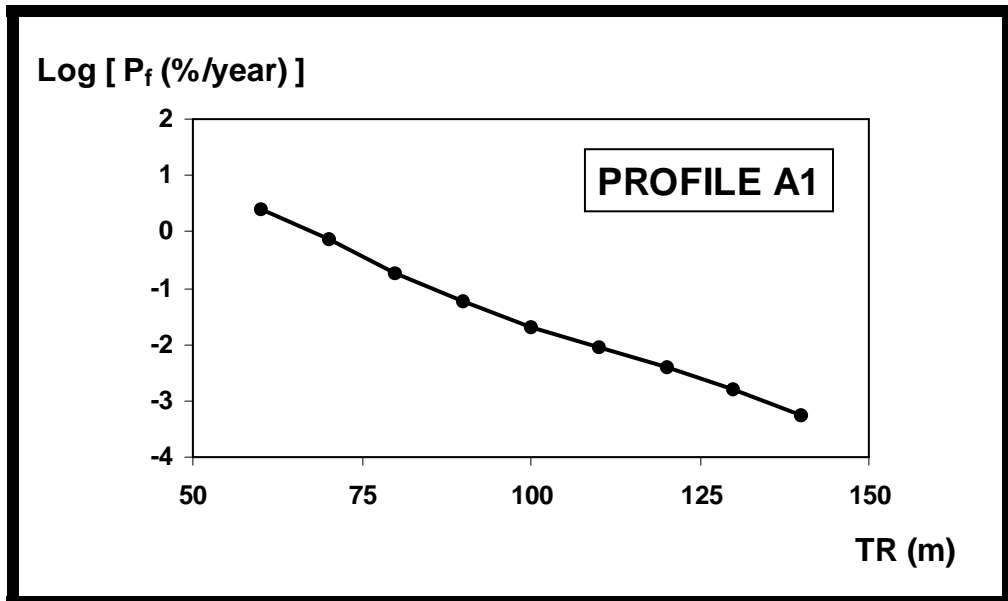
Figure 6.21: Probability of failure in one year of profile A1 as a function of allowable erosion distance (examples 1 and 6 to 13).
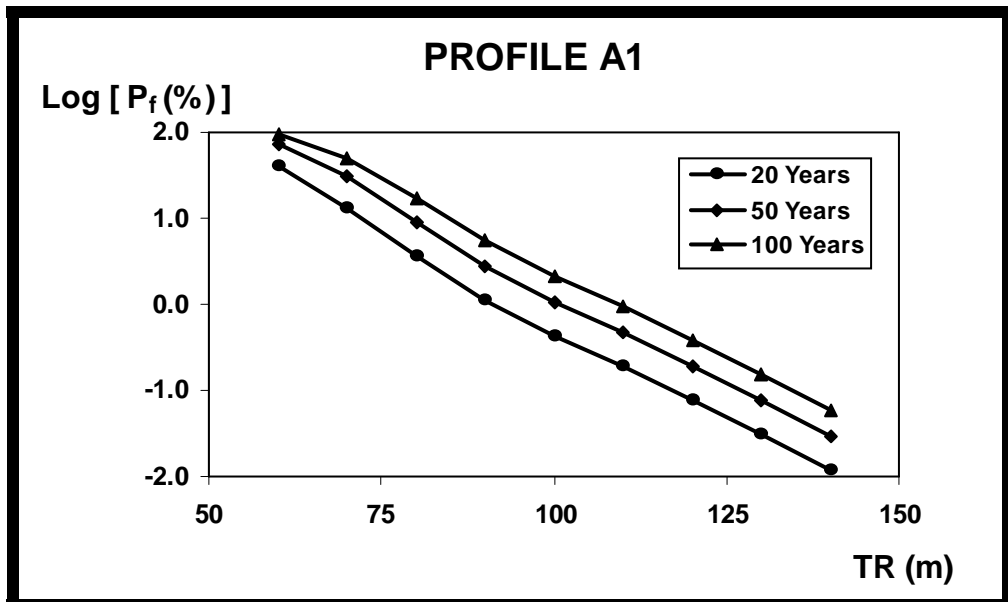
Figure 6.22: Lifetime probability of failure of profile A1 as a function of allowable erosion distance.

Plots like Figures 6.21 and 6.22 are valuable tools in preliminary design.

Tables D4.1 to D4.14 show that, for all examples, by far the most important contribution to the resulting variance is given by uncertainty in the maximum water level during surge followed by much smaller contributions from Ac, D_{50} and SD. The remaining variables make only minor contributions to the resulting variance. The dominant contribution of the surge is in accordance with Dutch studies (Van de Graaff, 1986; Van de Graaff, 1995). This fact is not surprising since the adopted characteristics of the random variables are typical of Dutch conditions.

Note that between 6 and 20 iterations have been required to run each of the above examples. The number does not depend on the probability of failure.

Finally, it is important to appreciate that the results of a numerical study are specific to the set of parameters used.

6.2.2 Particular Case Of Dune Erosion: The Sefton Coast, UK

6.2.2.1 Introduction

In The Netherlands, the narrow stretch of sandy beaches and dunes (in some places, the dunes are less than 200m wide) has to be maintained in order to protect people and property from damage. On the British Sefton coast, with a dune frontage up to 2km wide, the same problem does not arise. However, in view of increasing concern about the possibility that dune erosion may spread or accelerate in response to sea level rise or an increase in storminess associated with greenhouse warming, studies are in progress to achieve a better understanding of beach-dune interaction in this area (Pye, 1991). As mentioned by Pye & Neal (1994), erosion poses a significant management problem for the authorities responsible for the coast. Sefton Metropolitan Borough Council has statutory obligations to defend property from erosion and flooding, but it is also interested in preserving the natural character of the coast in order to maximise the recreational and nature conservation benefits. Large areas of the Sefton dune system lie within designated National and Local Nature Reserves, or are owned by the National Trust, while other areas are owned by private landowners and are used as caravan parks, golf courses, or for residential purposes.

It was of interest to evaluate the Dutch erosion prediction methods for use on this section of the British coast. The Dutch methods allow the retreat distance associated with a storm surge to be determined. Subsequently, a convenient position for erecting sand-trapping fences along the backshore to encourage foredune accretion (Thomas & Hall, 1992; Simm et al, 1996) may be fixed in such a way that the predicted erosion distance does not reach the fence line. Obviously, the methods' inherent limitations and assumptions must be borne in mind when applying the models. Furthermore, the statistical characteristics of the basic variables of the problem and the associated failure criteria used for Dutch conditions have to be analysed. They should not, in any circumstances, be adopted blindly for UK use.

6.2.2.2 General Description Of The Sefton Coastline

Figure 6.23 shows the Sefton coast, situated on the edge of Liverpool Bay between the Ribble and Mersey estuaries in the south-eastern corner of the Irish Sea.

The Irish Sea is almost completely enclosed, with two relatively narrow passages to oceanic waters: the North Channel between Scotland and Northern Ireland and St. George's Channel between Wales and Eire. There is a slow overall drift of water from south to north through the Irish Sea. As far as the coast is concerned, tidal currents, wave action and local sea-bed drifts are of much greater importance, together with fresh-water inputs from the main rivers. The Sefton coast is shielded from direct oceanic waves by the mainly enclosed nature of the Irish Sea and the additional blocking effects of the Isles of Anglesey and Man (see Figure 6.1). However, perhaps once or twice a year, during locally calm conditions, long-crested waves, or swells, are apparent and these have most likely been generated in the Atlantic Ocean.

Due to its natural state (unconsolidated sands, silts and perhaps a little clay with some outcrops of peat), the coast has very little strength. It has been moulded by the prevailing environment over many years to an overall form approaching equilibrium. This equilibrium is easily upset by very slight changes in the environment, leading on the one hand to accretion or equally on the other to erosion.
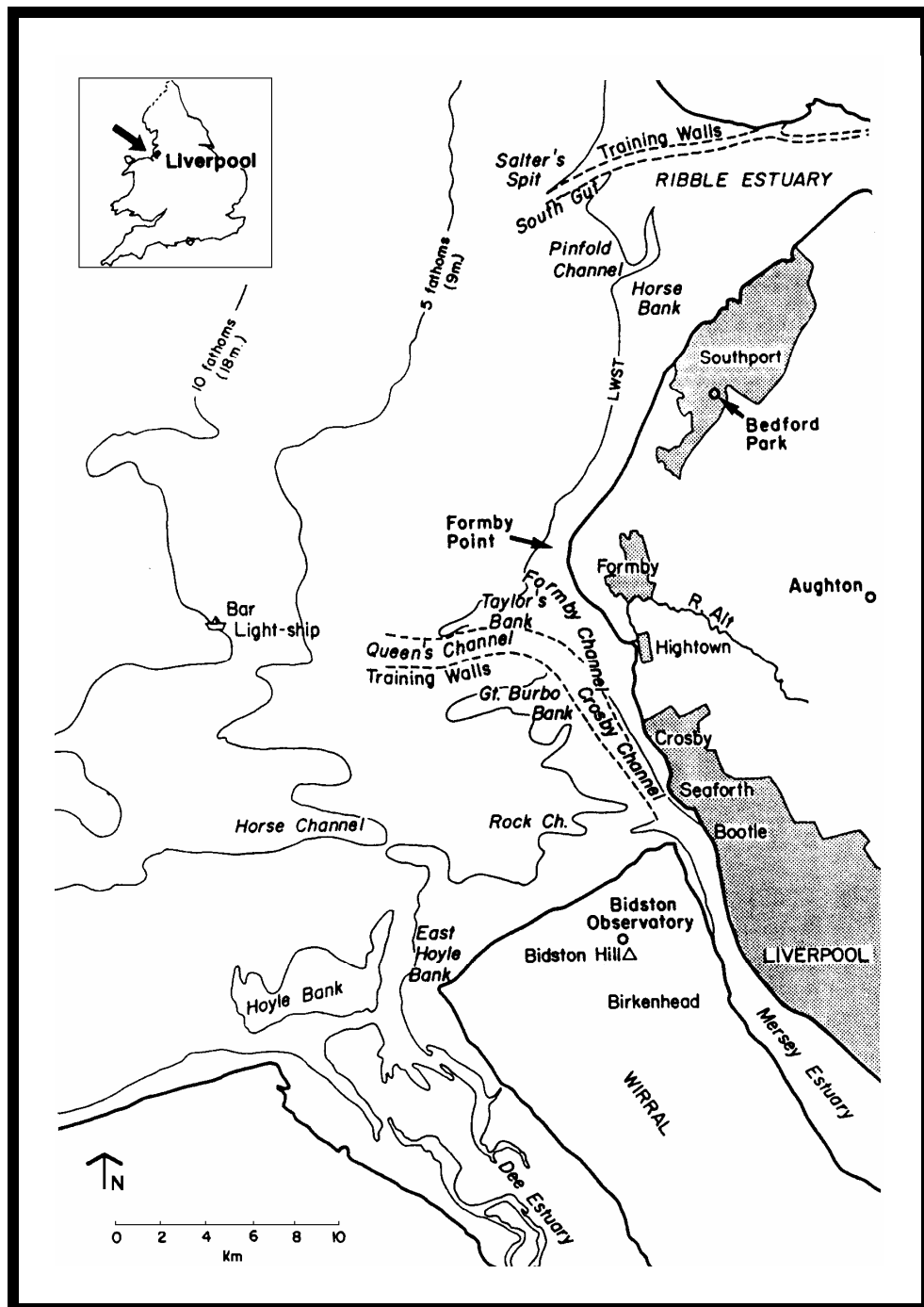
Figure 6.23: Location of the Sefton coast in north-west England (after Pye, 1991).

The area is notable for its high tidal range, in excess of 10m at maximum springs. It is occasionally subject to large meteorological surge contributions to high-water which when combined with strong wave activity cause severe erosion to the dune coast and some structural damage to coastal defence works. Very damaging storm tides occur on average every 5 or 6 years.

6.2.2.3 The Sefton Coast Dune System

The Sefton coastal dune complex is the largest in the British Isles and is of major significance in a European context (Doody, 1989; Atkinson & Houston, 1993; MAFF, 1993a; Pye & Neal, 1994). On the Sefton coast, dunes up to 2km wide are important not only for nature conservation and recreation, but also in terms of flood defence since they act as a natural barrier which prevents tidal inundation of a large area of West Lancashire and North Merseyside.

Following a period of rapid accretion in the second half of the last century, the dune frontage at Formby Point has been eroding since about 1906 at an average rate of up to 3m/year (Figure 6.24). The southern limit of erosion has remained roughly stable in the area of Lifeboat Road, but the northern erosion limit has gradually extended northwards. The rate of recovery of the frontal dunes following a storm has generally been insufficient to prevent a long-term net erosional trend between Lifeboat Road and Fisherman's Path. However, further north, between Ainsdale and Southport, and on the south side of Formby Point between Alexandra Road and the mouth of the River Alt, erosion during storms is generally less severe and the rate of dune recovery has been sufficiently rapid to maintain a net accretion throughout this century (Pye, 1991).

Foredune erosion at Formby Point was accelerated between 1900 and the mid 1970s by the abandonment of dune and foreshore management which had been extensively practised during the late 19th century, and by a significant increase in recreational pressure, sand mining and military activities (Pye & Neal, 1994). Since the establishment of a Coastal Management Scheme in 1977, damage to the dunes from these causes has been greatly reduced, but the dune protection and restoration works employed have had little effect on the problem of beach and foredune erosion by waves, especially during storm surges.

According to Pye & Neal (1994), possible factors contributing to the change from accretion to erosion around 1906 include:

- effects of dredging and training wall construction on sediment transport and wave regime;

- reduction or exhaustion of sand supply;

- effects of changes in bathymetry on wave climate;

- change in wind/wave climate;
- abandonment of beach and foredune management practices.
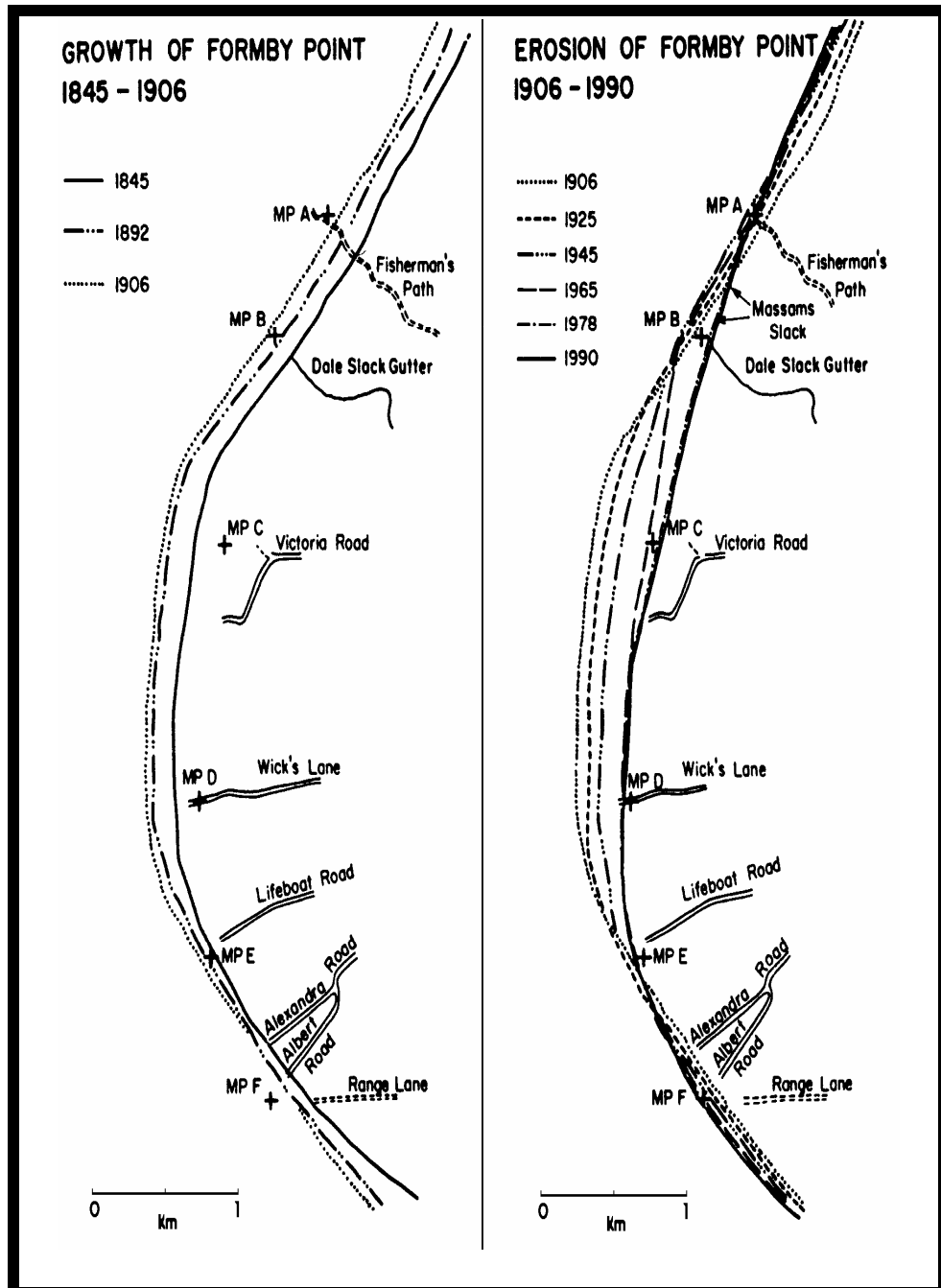
Figure 6.24: Plans showing the growth of Formby Point 1845-1906 and subsequent erosion in the period 1906-1990 (after Pye, 1991).

6.2.2.4 Storm Surge Of February 1990

In 1990, a major storm surge struck the Sefton coast. The storm surge was associated with the passage of a vigorous depression across northern Scotland and the North Sea during the period of 25-27 February. According to Pye (1991), on 26 February the mean hourly wind speed at Squiresgate Airport, Blackpool, increased from 33km/h to 73km/h, while the direction shifted from south-westerly to westerly. The strong onshore winds produced a surge which raised the height of predicted high water at 12.00hrs by approximately 1m along the coast between Morecombe Bay and North Wales. Strong westerly winds continued during the period 27-28 February, although a slight reduction in wind velocity and a reduction in wave height meant that the very severe conditions experienced on the morning of the 26 February were not repeated.

Structural damage was caused along the promenade at Southport and Crosby, and coastal defences were breached at Towyn in North Wales, causing flooding to several thousand homes. The storm surge and wave action also eroded large sections of the natural dune belt between Hightown and Southport (Pye, 1991). The greatest erosion was at Wick's Lane and Victoria Road (Table 6.4).

LOCATION	EROSION DISTANCE (m)
Albert Road	6.0
Lifeboat Road	8.3
Wick's Lane	11.1
Victoria Road	13.6
Fisherman's Path	7.5
Ainsdale-Southport	6.0

Table 6.4: Erosion between Hightown and Southport due to the 1990 storm surge (modified after Pye, 1991).

At Massam's Slack, where previous erosion had truncated the ends of dune ridges created artificially during the 1920s, waves overtopped the frontal ridge and flooded the slack behind.

6.2.2.5 Input To PARASODE

Due to the lack of data on the basic variables of dune erosion, PARASODE has been applied in a deterministic fashion to evaluate the erosion expected at Wick's Lane and Victoria Road due to the storm surge of February 1990. The data collected and used for calculations are presented in the following sections.

Selection Of Initial Profiles

Pye & Neal (1994) present a number of beach profiles, between Hightown and Southport, surveyed by the Sefton Borough Council Engineer and Surveyor's Department in August 1979. They also provide some information on dune heights. More detailed foreshore cross-sections for 1981 were obtained from the Metropolitan Borough of Sefton, Department of Technical Services. Unfortunately, no surveys were made shortly before or after the storm surge of 1990. The 1981 cross-sections are the latest available for the site and the ones used here for dune profile definition. Accretion and erosion records for the sand dune front at Formby Point are the only existing sources of information about the erosion caused by the storm surge (see Table 6.4). This information was also obtained from the Metropolitan Borough of Sefton.

The Metropolitan Borough of Sefton also provided maps from an aerial survey of July 1982, undertaken by Meridian Airmaps Limited. These maps provided information on the slopes and crest elevations of the foredunes. A considerable alongshore variation in the dune characteristics is evident in these maps. This variability caused difficulties in fixing a representative slope and crest elevation for the initial dune cross-sections.

A decision was finally made to study only the Wick's Lane and Victoria Road profiles. This selection was based, mainly, upon two factors:

- Vellinga's model, can be applied only to parts of the coast which are not strongly curved in plan (see Section 4.2.5).

- These two locations experienced the maximum erosion during the 1990 storm surge (see Table 6.4).

The two profiles are shown in Figure 6.25.
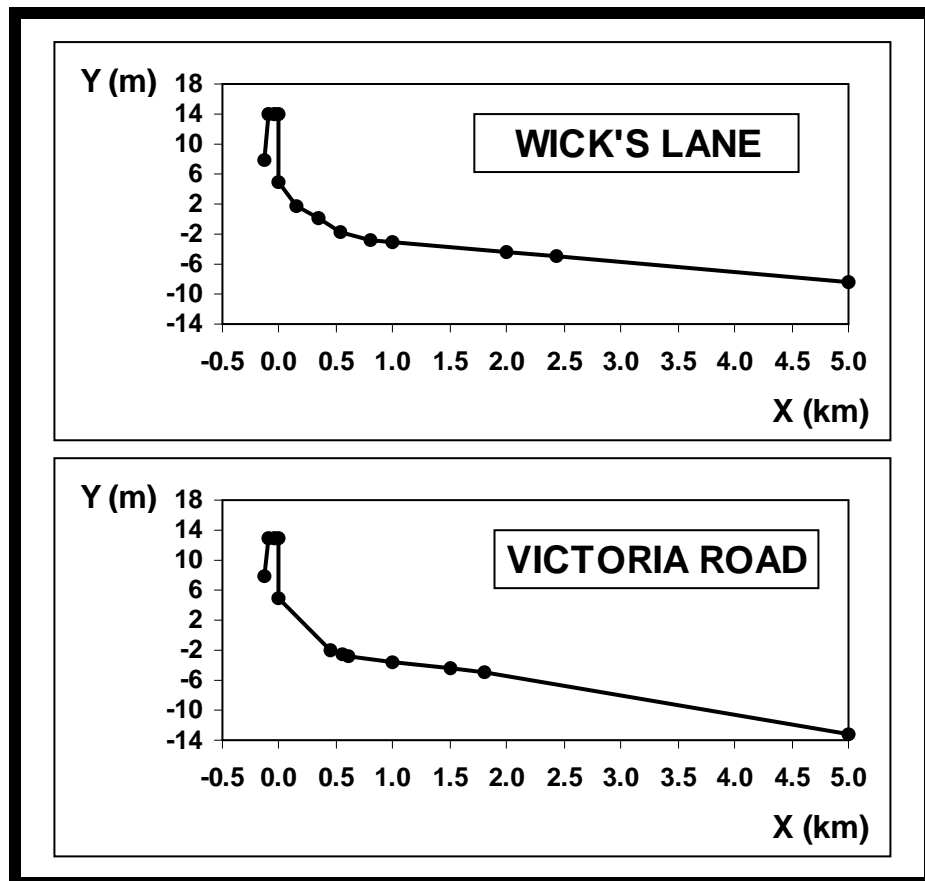
Figure 6.25: Initial profiles at Wick's Lane and Victoria Road.

Sand Grain Size

In the light of measurements reported by Pye (1991), the value adopted for the median sand diameter was $D_{50} = 215\,\mu m$.

As stated earlier, the amount of dune erosion depends on the particle diameter of the dune material via the fall velocity, w, which is calculated for a specific salt water temperature. In the case of the Sefton coast, the fall velocity for the entire period during which storm surges can be expected is calculated for a salt water temperature of 5°Celsi us using eq. (4.7).

Sea State

A significant wave height H_S=5.85m and an associated mean wave period, T_m=7.55s, were employed. These data were provided by Dr. Xiaoming Wu of

the Proudman Oceanographic Laboratory and are based on the computer model WAM. While these assumed wave conditions may not be exactly what occurred, they are considered a reasonable estimate.

Total Water Level

The maximum water level during surge is the sum of the tide level and surge. The mean of the values at Liverpool, 6.3m OD, and at Heysham and Southport, 6.4m OD (according to Pye, 1991) has been adopted: h=6.35m OD.

Interrelationship Between Sea State And Total Water Level

The occurrence of waves is likely to be at least partially correlated with the SWL, since both waves and storm surges are generated by meteorological conditions (Alcock, 1984), as shown in Figure 4.8. On the Sefton coast there is strong correlation between surges and waves. However, if the tidal range is very big, as it is on the Sefton coast (around 10m), the tide masks the correlation between waves and surge (Hawkes & Hague, 1994). Complete independence between sea states and total water levels is not expected, but the anticipated correlation is very weak. As part of Liverpool Bay (see Section 6.1), the Sefton coast falls between North Wales and Fleetwood and Cleveleys (see Figure 6.1). In the absence of data and following the comments in Section 6.1, it was assumed in this study that SWL and wave heights were independent. Note that PARASODE allows the user to consider correlation between h and H_S by providing a non-zero correlation coefficient between the two variables. Likewise, independence between h and H_S can be ensured by adopting a value of zero for the correlation coefficient.

6.2.2.6 Results And Discussion

According to Table 6.5, there is no difference in the computed erosion distance at Wick's Lane whether movements of sand are allowed only seaward or in both directions. In contrast, at Victoria Road the difference

amounts to about 22% of the value when movements are allowed seaward only.

LOCATION	EROSION DISTANCE FROM PARASODE (m)		MEASURED EROSION DISTANCE (m)
	Movements Of Sand Only Seaward	Movements Of Sand In Both Directions	
Wick's Lane	12.7	12.7	11.1
Victoria Road	6.8	5.3	13.6

Table 6.5: Computed and measured erosion distances at Wick's Lane and Victoria Road.

In the case of Wick's Lane, agreement between computed and measured erosion distances is very satisfactory. Agreement is much less satisfactory for Victoria Road. Two main reasons can be offered in order to explain the differences:

- The initial profiles used were unreliable because they had been taken nearly ten years before the storm surge of February 1990. Unfortunately, they were the latest profiles available.

- The measured erosion distance (the difference between the locations of the dune toe between measurements) did not relate solely to the surge event; the initial location of the dune toe had been measured eleven days before the surge and the location after the surge was taken three days later. Consequently, it would be expected that measurements would exceed computed values. This is the case for Victoria Road.

Given the potential errors, Vellinga's model can be regarded as having performed satisfactorily.

Finally, note that if the required data on the Sefton coast were available, then PARASODE could be run to produce probabilistic results for dune erosion due to the storm surge of February 1990. Unlike DUNE and DUNEPROB, PARASODE allows any degree of correlation between variables to be considered.

6.3   Summary

Chapter 6 illustrates the use of some features of PARASODE.

Firstly, the program has been used in a study of wave overtopping of a seawall to show the differences between the results of the H&R overtopping model and Owen's formulation. In this study, two main situations have been evaluated: i) the performance of the seawall for any possible value of total water level (tide plus surge); and ii) the ability of the structure to survive extreme total water levels. The first condition is relevant to design of retention and drainage systems, and for checking the safety of people and vehicles. The second provides peak values which are important for structural safety. Probabilities of failure per year versus the seawall crest level for different allowable discharges have been calculated. Sensitivity parameters have also been analysed, and the value of the design point has been examined as a function of the seawall crest level and the allowable discharge. Two main points are worth noting:

- The choice of overtopping model is very important in the probability assessment of the safety of seawalls. The two models lead to quite different results. Use of Owen's model in design would be more conservative than use of the H&R model. However, the conservative nature of Owen's model also implies that its use in design will result in more expensive structures than those designed using the H&R formulation.

- For both overtopping models, the sensitivity parameters show that the main influence on the variability of the probability of failure is generally provided by the uncertainty in the sea state.

Secondly, the accuracy of the Level II (FORM) reliability algorithms used in PARASODE has been evaluated by comparison with the results provided by the Level III method of Latin Hypercube Sampling (LHS) available in @Risk. This evaluation has been carried out for the Level II results of the H&R model, $(R_{max})_{37\%}$, and Owen's model, for normal design conditions only. The two methods give comparable results. However, as the seawall crest level decreases, the FORM results diverge from the LHS results, generally overestimating the probability of failure. The differences in the probabilities of failure suggest that the failure surfaces for both overtopping models are curved near the design point.

Thirdly, PARASODE has been used to study general examples of dune erosion during a storm surge. In these examples, movements of sand have been allowed only seaward. Four profiles have been considered for illustration of the main erosion situations which can be studied using PARASODE. Some examples illustrate how nourishment can be used to decrease the dune's failure probability caused by erosion due to a storm surge. The adopted characteristics of the random variables are typical of Dutch conditions. PARASODE has been run both in mode 1 (a nourishment width has been chosen and the corresponding probability of failure calculated) and in mode 2 (the computed probability of failure for mode 1 has been input and a corresponding nourishment width has been computed). These tests demonstrate the converse nature of modes 1 and 2 and show consistency between the results obtained.

Fourthly, an attempt has been made to apply PARASODE in a study of dune erosion on the Sefton coast, UK, due to the storm surge of February 1990. The study allowed the retreat distance associated with this storm surge to be estimated. However, due to lack of data, PARASODE could be applied only in a deterministic fashion to determine the erosion expected at Wick's Lane and Victoria Road. These two sections of the Sefton coast were selected because Vellinga's model can be applied only to parts of the coast which are not strongly curved in plan, and because these two locations experienced the maximum erosion during the 1990 storm surge. PARASODE results suggest that there is no difference in the computed erosion distance at Wick's Lane whether movements of sand are allowed only seaward or in both directions. In contrast, at Victoria Road the difference amounts to about 22% of the value when movements are allowed seaward only. In the case of Wick's Lane, agreement between computed and measured erosion distances is very satisfactory. Agreement is much less satisfactory for Victoria Road. Two main reasons can be offered in order to explain the differences: i) the initial profiles used were unreliable because they had been taken nearly ten years before the storm surge of February 1990; and ii) the measured erosion distance did not relate solely to the surge event. Given the potential for errors, Vellinga's model can be regarded as having performed satisfactorily.

7 CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER RESEARCH

The main objective of the present research was to assess the safety of coastal structures by means of probabilistic methods, with particular reference to wave overtopping of seawalls and to dune erosion. This chapter lists the principal conclusions of the research and, where appropriate, provides recommendations for further work.

Probabilistic methods

- Probabilistic methods provide a powerful framework for the design of coastal structures, accounting for the probability and consequences of failure as well as coping, to some degree, with variability and uncertainty. However, when assessing structural safety using probabilistic methods, it must be stressed that the process involves detailed knowledge about the individual structure. Therefore, confidence in the calculated value of the probability of failure must change with the amount and quality of the information used for its calculation. With these facts in mind, probabilistic methods may be seen simply as a design tool based on scientific methods which can facilitate good engineering decisions, but not a process which will necessarily provide a precise assessment of safety.

- Probabilistic approaches are increasingly being applied in engineering practice. This fact is apparent in civil engineering from the use of Level II calculations for determining the partial safety factors applied in standards and codes for the design of structures. Direct probabilistic approaches have increasingly become the rule in connection with the assessment of special structures (e.g. nuclear power stations and storage tanks for hazardous substances).

- In recent years, much has been learned by coastal engineers about probabilistic methods, but progress in formulating methods and gaining confidence in new design procedures is inevitably slow. At present, there is insufficient knowledge about coastal structures to enable a probability

analysis of failure mode systems to be carried out in full. However, instead of abandoning this "new" approach to design, efforts should be made to better identify the specific physical processes with which coastal engineers must deal, to better communicate their data requirements to researchers, to subsequently collect the required data sets, and to establish appropriate models for the complete implementation of the methods. Furthermore, it is important to incorporate as much experience as possible from failures.

- Diagrams like event trees, fault trees and cause-consequence charts have been presented for some coastal structures. However, such techniques have still almost always served essentially as schematic representations or research tools rather than as strict logical analyses of failure. Information on failures tends to concentrate on the consequences rather than on the causes of failure.

- Assessment of the safety of coastal structures depends fundamentally on assessment of individual failure modes. All single failure mode probabilistic methods have their advantages and disadvantages:

  - In the numerical integration method, the calculation of an N-fold integral may be extremely time-consuming and it usually requires a considerable computational effort, even with modern computer facilities. Traditional sampling is an acceptable alternative when dealing with simple failure functions and failure probabilities which are not very low. However, it suffers from the fact that if an "accurate" answer is desired for extreme conditions associated with relatively low probabilities of failure, many simulations are required. This is a drawback that recent methods, like Latin Hypercube sampling, may address to some extent by reducing the required number of simulations. In other cases, difficulties can be overcome by using Level II methods like FORM.

  - The main practical advantages of the FORM approach are that it is less time-consuming than Level III methods, the computational effort is independent of the probability level, it provides a rational basis for evaluating partial safety factors and it also provides an automatic procedure for determining the sensitivity of the computed failure probability to each of the basic design variables. This latter characteristic allows the designer to focus his attention on the parameters which are of greatest significance and shows where effort to reduce uncertainty should be concentrated. Due to their simplicity, these methods have become very popular, particularly in calibration work for codes of practice. However, these procedures also have their limitations. Amongst others, the main reason for

discrepancy between a Level II and a Level III method is that the failure function is usually non-linear. The stronger the non-linearity, the greater is the chance that the Level II results will differ considerably from the "exact" answer. However, the FORM results can be improved through a second or higher order approximation, but computational complications are increased considerably. It is more common to use the Level III methods, especially simulation, to validate the Level II results. Although a FORM method can provide an answer to a problem, it is never known how accurate the answer is unless a check is done using numerical integration or simulation techniques. Nevertheless, the FORM method is one of the most important tools in probabilistic design because one can rarely afford to make a million Level III calculations during preliminary design.

- Besides the calculations at Level III and Level II, there are those at Level I. Level I calculations are particularly suitable for everyday design (where a large body of previous experience of similar systems is available), although the determination of the partial coefficients must be based upon higher level results. Level I calculations are the basis of codes of practice.

- It is important to be aware of the characteristics of the various probabilistic methods, their applicability and their limitations, otherwise wrong conclusions can be drawn, incorrect decisions can be made and unsound action may be taken. If probabilistic methods are used with foresight and understanding, they are powerful and can provide reliable results. For example, comparison of design alternatives using these methods is a promising way in which to apply them.

Wave overtopping of seawalls

- Seawalls are expensive, and fixing a seawall freeboard at too large a value has both a financial penalty and is unnecessarily damaging to the natural environment owing to the increased impact of the structure on its surroundings. On the other hand, if the crest of a seawall is set too low, then there are problems with structural safety and potential social problems with flooding and with people's protection.

- Wave overtopping of seawalls has been the subject of many studies. Nevertheless, field measurements are scarce and numerical modelling of wave overtopping is not yet well developed. The calculation of overtopping discharge is based mainly on equations which have been

obtained from empirical fitting to hydraulic model test results. These equations have not been based upon any overtopping theory and no account has generally been taken of the physical boundary conditions.

- As part of this research, a new regression model (the H&R model) has been presented for describing wave overtopping data. Part of the motivation in deriving this new model was to improve the methods available to the designers of seawalls by developing a model closely related to the physics of wave overtopping. The main feature of the model is the fact that it satisfies the relevant physical boundary conditions, a feature which is especially important when the model is used near these boundaries.

- The H&R and Owen models have been used in a re-analysis of Owen's data for simple seawalls possessing uniform seaward slopes of 1:1, 1:2 and 1:4, subjected to random waves approaching normal to the slope. Both models represent part of the input to a FORTRAN Level II program, developed as part of this research, PARASODE. It is suggested that the regression coefficients contained within the models should be established using a robust regression technique such as the Least Absolute Deviations (LAD) method. The LAD regression coefficients are recommended for use both in the H&R and Owen models.

- For Owen's test results, the H&R model is little different from Owen's model in its ability to represent the data, except for small discharges for which the H&R model is better suited. An example of the application of the two models in predicting the freeboards necessary to limit overtopping to specified values shows that, for the small allowable discharges associated with normal design conditions, the H&R model predicts seawall crest elevations which may be several metres lower than values from Owen's model. Such differences may have very significant financial and environmental consequences and are worthy of further investigation.

- Whilst it is possible to use Owen's data to show the validity of the approach adopted in developing the new wave overtopping model, the data are far from ideal. Consequently, it is recommended that the present study on overtopping is extended:

- to encompass both the very small allowable discharges associated with normal design conditions and to permit proper evaluation of the empirical coefficients in the equations used to describe overtopping;

- to allow evaluation both of the probability distributions of the parameters involved in overtopping and of the horizontal distribution of the total overtopping volume;

- to collect data on the effects of wind on wave overtopping with the objective of permitting further development of overtopping equations.

- It is believed that the environmental, social and economic benefits likely to derive from implementing these recommendations would provide a very valuable return for the investment of the time, effort and costs involved, particularly for those countries with an exposed coast such as the United Kingdom, The Netherlands and Portugal. An important socio-economic justification for the work is the possibility of including information on the variability of overtopping volumes and their horizontal distribution not only in detailed design but also in political decision-making relating to urban planning issues such as the location of housing and basic infrastructure.

Dune erosion during a surge

- Dutch experience with regard to the probabilistic design of dunes has been examined. The computational model currently used throughout The Netherlands is based on Vellinga's equilibrium profile model. The more sophisticated time-dependent model developed by Steetzel is not yet used as the basis for probabilistic calculations.

- The Dutch programs are not directly applicable to conditions along coasts such as that in Sefton, UK, where there is a much weaker correlation than in The Netherlands between wave heights and water levels. Consequently, it was decided to introduce Vellinga's model and some features of the Dutch programs into PARASODE, and to carry out new probabilistic calculations.

PARASODE

- PARASODE concentrates on the failure modes of random wave overtopping of simple seawalls and dune erosion. The quantity of wave overtopping is calculated using both the H&R formula and Owen's formula. Dune erosion is calculated using Vellinga's model. However, much of the program is generic and can be adapted to other failure modes without undue difficulty. PARASODE has been applied to several different examples in order to illustrate the use of some of its features.

- The program has been used in a study of wave overtopping of a seawall to show the differences between the results of the H&R overtopping model and Owen's formulation. Two main points are worth noting:

  - The choice of overtopping model is very important in the probability assessment of the safety of seawalls. The two models lead to quite different results. Use of Owen's model in design would be more conservative than use of the H&R model. However, the conservative nature of Owen's model also implies that its use in design would result in more expensive structures than those designed using the H&R formulation.

  - For both overtopping models, the sensitivity parameters show that the main influence on the variability of the probability of failure is generally provided by the uncertainty in the sea state.

- The accuracy of the Level II (FORM) reliability algorithms used in PARASODE has been evaluated by comparison with the results provided by the Level III method of Latin Hypercube Sampling (LHS) available in the commercial software package @Risk. The two methods give comparable results. However, as the seawall crest level decreases, the FORM results diverge from the LHS results, generally overestimating the probability of failure. The differences in the probabilities of failure suggest that the failure surfaces for both overtopping models are curved near the design point.

- PARASODE has been used to study general examples of dune erosion during a storm surge. In these examples, movements of sand have been allowed only seaward. Four profiles have been considered for illustration of the main erosion situations which can be studied using PARASODE. Some examples illustrate how nourishment can be used to decrease the

dune's failure probability caused by erosion during a storm surge. The adopted characteristics of the random variables are typical of Dutch conditions. PARASODE has been run both in mode 1 (a nourishment width has been chosen and the corresponding probability of failure calculated) and in mode 2 (a probability of failure has been input and a corresponding nourishment width has been computed). These tests demonstrate the converse nature of modes 1 and 2 and show consistency between the results obtained.

- An attempt has been made to apply PARASODE in a study of dune erosion on the Sefton coast, UK, due to the storm surge of February 1990. The study allowed the retreat distance associated with this storm surge to be estimated. However, due to lack of data, PARASODE could be applied only in a deterministic fashion.

The above conclusions demonstrate that the principal objective of the research has been accomplished. Arrangements are already in hand to extend the work described here through international collaboration involving the University of Liverpool, the National Laboratory of Civil Engineering (LNEC), Portugal, and other partners.

REFERENCES

Abramowitz, M.; Stegun, I.A. (1964) (eds.). Handbook Of Mathematical Functions With Formulas, Graphs, And Mathematical Tables. National Bureau Of Standards, Applied Mathematics Series 55, New York.

Ahammed, M.; Melchers, R.E. (1993). Probabilistic assessment of pipeline service life based on pitting corrosion leaks. Proc. Conference On Probabilistic Risk And Hazard Assessment, September 22-23, 1993, Newcastle N.S.W., Australia (eds. Melchers, R.E.; Stewart, M.G.), Balkema, Rotterdam, The Netherlands, pp. 131-138.

Ahrens, J. (1977). Prediction Of Irregular Wave Overtopping. CETA 77-7, December 1977, U.S. Army Corps Of Engineers, Coastal Engineering Research Centre.

Ahrens, J.P. (1981). Irregular Wave Runup On Smooth Slopes. Coastal Engineering Technical Aid No. 81-17, US Army, Corps Of Engineers, Coastal Engineering Research Centre, Fort Belvoir.

Ahrens, J.P.; Bender, T. (1991). Evaluating the performance of seawalls. Proc. ICE Conference, Coastal Structures And Breakwaters, November 6-8, 1991, London, Thomas Telford Ltd, London, 1992, pp. 13-24.

Ahrens, J.P.; Heimbaugh, M.S. (1988). Seawall overtopping model. Proc. 21st ICCE, June 20-25, 1988, Malaga, Spain (ed. Edge, B.L.), ASCE, New York, 1989, pp. 795-806.

Alcock, G.A. (1984). Parameterizing Extreme Still Water Levels And Waves In Design Level Studies. Report No. 183, Institute Of Oceanographic Sciences (IOS), Wormley.

Alcock, G.A.; Carter, D.J.T. (1985). Extreme events in wave and water levels. Developments In Breakwaters, Proc. ICE Conference On Breakwaters '85, October 2-4, 1985, London, Thomas Telford Ltd, London, 1986, pp. 53-68.

Allen, D.E. (1981). Criteria for design safety factors and quality assurance expenditure. Proc. ICOSSAR '81, 3rd International Conference On Structural Safety And Reliability, June 23-25, 1981, Norway (eds. Moan, T.; Shinozuka, M.), Elsevier Scientific Publishing Company, Amsterdam, pp. 667-678.

Allsop, N.W.H. (1986). Sea Walls: A Literature Review. Report EX 1490, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Allsop, N.W.H. (1994). Breakwaters and coast defence: wave overtopping of seawalls, breakwaters and shoreline structures. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 355-357.

Allsop, N.W.H.; Franco, C. (1992). Performance of rubble mound breakwaters: singular points. MAST G6-S Coastal Structures, Final Overall Workshop, November 5-6, 1992, Lisbon, Portugal.

Allsop, N.W.H.; Meadowcroft, I.C. (1995). Probabilistic Assessment Of The Safety Of Breakwaters And Coastal Structures. Report SR 416 (Draft), Hydraulics Research Wallingford, Wallingford, UK.

Allsop, N.W.H.; Franco, L. & Hawkes, P.J. (1985a). Wave Run-Up On Steep Slopes - A Literature Review. Report SR1, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Allsop, N.W.H.; Hawkes, P.J.; Jackson, F.A. & Franco, L. (1985b). Wave Run-Up On Steep Slopes - Model Tests Under Random Waves. Report No. SR2, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Alonso, E.E. (1976). Risk analysis of slopes and its application to slopes in Canadian sensitive clays. Géotechnique, Vol. 26, pp. 453-472.

Aminti, P.; Franco, L. (1988). Wave overtopping on rubble mound breakwaters. Proc. 21st ICCE, June 20-25, 1988, Malaga, Spain (ed. Edge, B.L.), ASCE, New York, 1989, pp. 770-781.

Andrews, J.D.; Moss, T.R. (1993). Reliability And Risk Assessment. Longmans Scientific & Technical, Harlow.

Ang, A.H.-S.; Tang, W.H. (1975). Probability Concepts In Engineering Planning And Design - Vol. I: Basic Principles. John Wiley & Sons, Inc, New York.

Ang, A.H.-S.; Tang, W.H. (1984). Probability Concepts In Engineering Planning And Design - Vol. II: Decision, Risk And Reliability. John Wiley & Sons, Inc, New York.

Anscombe, F.J. (1960). Rejection of outliers. Technometrics, Vol. 2, pp. 123-147.

Anscombe, F.J. (1961). Examination of residuals. Proc. 4th Berkeley Symposium On Mathematical Statistics And Probability, 1, pp. 1-36.

Anscombe, F.J.; Tukey, J.W. (1963). The examination and analysis of residuals. Technometrics, Vol. 5, pp. 141-160.

Armstrong, R.D.; Beck, P.O. (1990). An algorithm to assist in the identification of multiple multivariate outliers when using a least absolute value criterion. Robust Regression: Analysis And Applications. Statistics - Textbooks And Monographs (eds. Lawrence, K.D.; Arthur, J.L.), Marcel Dekker, Inc, USA, pp. 89-103.

Ataa, M.A.S. (1994). Constrained And Unconstrained L1-Norm For Data Analysis And Some Comparisons With The Least Squares Approach. Ph.D. Thesis, University Of Liverpool, Department Of Statistics And Computational Mathematics, UK.

Atkinson, D.; Houston, J. (1993) (eds.). The Sand Dunes Of The Sefton Coast. Liverpool Museum, Liverpool.

Bai, Z.D.; Rao, C.R. & Yin, Y.Q. (1990). Least absolute deviations analysis. Sankhya, The Indian Journal Of Statistics, Vol. 52, pp. 166-177.

Barber, P.C. (1993). Economic appraisal of coastal engineering works. Coastal Estuarial And Harbour Engineer's Reference Book (eds. Abbott, M.B.; Price, W.A.), Chapman & Hall, London, pp. 707-714.

Bassett, G.; Koenker, R. (1978). Asymptotic theory of least absolute error regression. Journal Of The American Statistical Association, Vol. 73, pp. 618-622.

Battjes, J.A. (1974). Computation Of Set-Up, Longshore Currents, Run-Up And Overtopping Due To Wind-Generated Waves. Report 74-2, Communication On Hydraulics, Delft University Of Technology, Department Of Civil Engineering, The Netherlands.

Beaumont, G.P. (1986). Probability And Random Variables. Ellis Horwood Ltd, Chichester.

Bendat, J.S.; Piersol, A.G. (1971). Random Data: Analysis And Measurement Procedures. John Wiley & Sons, Inc, New York.

Berry, W.D. (1993). Understanding Regression Assumptions. Series: Quantitative Applications In The Social Sciences. Sage Publications, Inc, USA.

Besley, P.; Reeves, M.K. & Allsop, N.W.H. (1993). Single Layer Armour Research Club. Random Wave Model Tests: Overtopping And Reflection Performance. Report IT 384, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Birks, C.J. (1993). Flood defence in England and Wales: overview of flood problems. British-Hungarian Workshop On Flood Defence, September 6-10, 1993, Budapest (ed. Hegedus, M.), Water Resources Research Centre, pp. 57-78.

Bjerager, P. (1991). Methods for structural reliability computations. Reliability Problems: General Principles And Applications In Mechanics Of Solids And Structures, CISM Series No. 317 (eds. Casciati, F.; Roberts, J.B.), Springer-Verlag, New York, pp. 89-135.

Bjerager, P.; Skov, K. (1982). Reliability of series systems using the Rackwitz-Fiessler algorithm on the single mode safety margins. Reliability Theory Of Structural Engineering. Euromech 155, June 15-17, 1982, Mohr, G., Denmark, pp. 47-78.

Blackman, D.L. (1997). Private Communication.

Blockley, D.I. (1981). Reliability theory - incorporating gross errors. Proc. ICOSSAR '81, 3rd International Conference On Structural Safety And Reliability, June 23-25, 1981, Norway (eds. Moan, T.; Shinozuka, M.), Elsevier Scientific Publishing Company, Amsterdam, pp. 259-282.

Bradbury, A.P.; Allsop, N.W.H. (1988). Hydraulic effects of breakwater crown walls. Proc. ICE Conference, Breakwaters '88, May 4-6, 1988, London, Thomas Telford Ltd, London, 1988, pp. 385-396.

Bradbury, A.P.; Allsop, N.W.H. & Stephens, P.V. (1988). Hydraulic performance of breakwater crown walls. Report SR 146, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Breitung, K. (1983). Asymptotic approximations for multi-normal domain and surface integrals. 4th International Conference On Applications Of Statistics And Probability In Soil And Structural Engineering, Florence, Italy.

Breitung, K. (1984). Asymptotic approximations for multinormal integrals. Journal Of Engineering Mechanics, ASCE, Vol. 110, pp. 357-366.

Brennan, W.J.; Stickland, I.W. (1981). The probabilistic approach to the design of maritime structures. Transactions Of The Institution Of Engineers Of Ireland, Vol. 105, pp. 17-24.

Bruun, P. (1954). Coast Erosion And The Development Of Beach Profiles. Technical Memo No. 44, Beach Erosion Board, U.S. Army Corps Of Civil Engineers.

Bruun, P. (1962). Sea level rise as a cause of the shore erosion. Journal Of The Waterways & Harbors Division, ASCE, Vol. 88, pp. 117-130.

Bruun, P. (1985). Design And Construction Of Mounds For Breakwaters And Coastal Protection, Developments In Geotechnical Engineering, Vol. 37, Elsevier, Amsterdam, The Netherlands.

BSI (British Standards Institution) (1990). BSI Handbook 22 - Quality Assurance. BSI, London, UK.

BSI (British Standards Institution) (1991a). BS 4778: Part 3 - Availability, reliability and maintainability terms. Quality Vocabulary, BSI, London, UK.

BSI (British Standards Institution) (1991b). BS 6349: Part 7 - Guide to the design and construction of breakwaters. Code Of Practice For Maritime Structures, BSI, London, UK, pp. 1-89.

BSI (British Standards Institution) (1991c). BS 4778: Part 2 - Quality concepts and related definitions. Quality Vocabulary, BSI, London, UK.

Burcharth, H.F. (1983). The way ahead. Proc. ICE Conference, Breakwaters - Design And Construction, May 4-6, 1983, London, Thomas Telford Ltd, London, 1984, pp. 177-187.

Burcharth, H.F. (1985). On The Reliability Of Rubble Mound Breakwater Design Parameters. Hydraulics And Coastal Engineering Laboratory, University Of Aalborg, Department Of Civil Engineering, Denmark.

Burcharth, H.F. (1987). The Lessons From Recent Breakwater Failures. Developments In Breakwater Design. Hydraulics And Coastal Engineering Laboratory, University Of Aalborg, Department Of Civil Engineering, Denmark, 1988 (presented at the Technical Congress, World Federation Of Engineering Organisations, Vancouver).

Burcharth, H.F. (1990). Safety of marine structures: from failure experience to the development of design standards for rubble mound breakwaters. Hydraulics And Coastal Engineering Laboratory, University Of Aalborg, Department Of Civil Engineering, Denmark, 1990 (presented at the 1st International Conference On Ports And Marine Structures, Teheran, Iran).

Burcharth, H.F. (1991a). Introduction of partial coefficients in the design of rubble mound breakwaters. Proc. ICE Conference, Coastal Structures And Breakwaters, November 6-8, 1991, London, Thomas Telford Ltd, London, 1992, pp. 543-565.

Burcharth, H.F. (1991b). Recommendation Of A Partial Coefficient System For The Implementation Of Predetermined Safety In The Design Of Rubble Mound Breakwaters (Section 2.2). Report Of Subgroup F, PIANC, PTC II Working Group 12.

Burcharth, H.F. (1992). Reliability evaluation of a structure at sea. Proc. Short Course On Design And Reliability Of Coastal Structures (attached to the 23rd ICCE), Tecnoprint snc, Bologna, pp. 597-644.

Bury, K.V. (1975). Statistical Models In Applied Science. John Wiley & Sons, Inc, New York.

Cai, G.Q.; Elishakoff, I. (1994). Refined second-order reliability analysis. Structural Safety, Vol. 14, pp. 267-276.

Carvalho, M.M. (1982). On the specification of the wave input to use for the design of coastal structures. Jornadas Portuguesas De Engenharia De Estruturas, Laboratório Nacional de Engenharia Civil, Lisboa, pp. A.145-A.157.

Carvalho, M.M. (1992a). Dimensionamento Probabilístico De Quebra-Mares. Memória No. 786, Laboratório Nacional de Engenharia Civil, Lisboa.

Carvalho, M.M. (1992b). Statistical Treatment Of Extreme Sea Waves And Levels. Proc. International Conference On The Pearl River Estuary In The Surrounding Area Of Macau, October 19-23, 1992, Macau, LECM/PRWRC/LNEC, Macao, 1992, pp. 283-304.

Casciati, F.; Faravelli, L. (1985). Structural reliability and structural design optimization. 4th International Conference On Structural Safety And Reliability, Kobe, Japan, pp. 61-70.

Casciati, F.; Faravelli, L. (1991). Fragility Analysis Of Complex Structural Systems. Research Studies Press, Taunton.

Casciati, F.; Natale, L. (1992). Risk analysis for marine systems: an introduction. Proc. Short Course On Design And Reliability Of Coastal Structures (attached to the 23rd ICCE), Tecnoprint snc, Bologna, pp. 1-22.

Chang, C.-H.; Tung, Y.-K. & Yang, J.-C. (1994). Monte Carlo simulation for correlated variables with marginal distributions. Journal Of Hydraulic Engineering, ASCE, Vol. 120, pp. 313-331.

CIAD - Project Group (1985). Computer Aided Evaluation Of The Reliability Of A Breakwater Design. Association For Computer Applications In Applied Engineering (CIAD), Zoetermeer.

CIRIA (Construction Industry Research And Information Association) (1984). Rationalization Of Safety And Serviceability Factors In Structural Codes. CIRIA Report 63, London, UK.

CIRIA (Construction Industry Research And Information Association) (1986a). Maintenance Of Coastal Revetments. Technical Note 124, London, UK.

CIRIA (Construction Industry Research And Information Association) (1986b). Sea Walls: Survey Of Performance And Design Practice. Technical Note 125, London, UK.

CIRIA (Construction Industry Research And Information Association) (1996). Beach Recharge Material - Demand And Resources. CIRIA Report 154, London, UK.

CIRIA (Construction Industry Research And Information Association) / CUR (Centre For Civil Engineering Research And Codes) (1991). Manual On The Use Of Rock In Coastal And Shoreline Engineering. CIRIA Special Publication 83, London, UK; CUR Report 154, Gouda, The Netherlands.

Committee Of The Institution Of Structural Engineers (1955). Report On Structural Safety. The Structural Engineer, Vol. 33, pp. 141-149.

Cope, R.J.; Sawko, F. & Tickell, R.G. (1982). Computer Methods For Civil Engineers. McGraw-Hill Book Company Limited, London, UK.

Copeiro, E. (1978). Extremal prediction of significant wave height. Proc. 16th ICCE, August 27 to September 3, 1978, Hamburg, Germany, ASCE, New York, 1979, pp. 285-304.

Cullen, N. (1990). Risk Assessment Of Earth Dam Reservoirs. DoE 0002-SW/3, Final Report To The Department Of The Environment, UK.

Cumo, M.; Naviglio, A. (1987) (eds.). Safety Design Criteria For Industrial Plants. CRC Press, Boca Raton.

CUR (Centre For Civil Engineering Research Codes And Specifications); Rijkswaterstaat & Delft Hydraulics (1987). Manual On Artificial Beach Nourishment. Report 130, Gouda, The Netherlands.

CUR-TAW (Centre For Civil Engineering Research And Codes - Technical Advisory Committee On Water Defences) (1989). Guide To The Assessment Of The Safety Of Dunes As A Sea Defence. Report 140, Gouda, The Netherlands.

CUR-TAW (Centre For Civil Engineering Research And Codes - Technical Advisory Committee On Water Defences) (1990). Probabilistic Design Of Flood Defences. Report 141, Gouda, The Netherlands.

D'Agostino, R.B.; Stephens, M.A. (1986) (eds.). Goodness-Of-Fit Techniques. Statistics: Textbooks And Monographs. Marcel Dekker, Inc, New York.

D'Andrea, R.A.; Sangrey, D.A. (1982). Safety factors for probabilistic slope design. Journal Of The Geotechnical Engineering Division, ASCE, Vol. 108, pp. 1101-1118.

D'Angremond, K. (1992). Beach nourishment. Proc. Short Course On Design And Reliability Of Coastal Structures (attached to the 23rd ICCE), Tecnoprint snc, Bologna, pp. 415-432.

Daniel, C.; Wood, F.S. (1980) (eds.). Fitting Equations To Data: Computer Analysis Of Multifactor Data. John Wiley & Sons, New York.

Davis, G.B.; Hoffman, T.R. (1988). FORTRAN 77: A Structured, Disciplined Style. McGraw-Hill, New York.

Dean, R.G. (1976). Beach erosion: causes, processes, and remedial measures. CRC Critical Review In Environmental Control, Vol. 6, pp. 259-296.

Dean, R.G. (1977). Equilibrium Beach Profiles: US Atlantic And Gulf Coasts. Ocean Engineering Report No. 12, University Of Delaware, Department Of Civil Engineering, Newark.

Dean, R.G. (1982). Models For Beach Profile Response. Technical Report No. 30, University Of Delaware, Newark.

Dean, R.G. (1987). Coastal sediment processes: towards engineering solutions. Proc. Coastal Sediments '87, May 12-14, 1987, New Orleans, USA (ed. Kraus, N.C.), ASCE, New York, 1987, pp. 1-24.

Dean, R.G. (1991). Equilibrium beach profiles: characteristics and applications. Journal Of Coastal Research, Vol. 7, pp. 53-84.

De Gerloni, M.; Franco, L. & Passoni, G. (1991). The safety of breakwaters against wave overtopping. Proc. ICE Conference, Coastal Structures And Breakwaters, November 6-8, London, Thomas Telford Ltd, London, 1992, pp. 335-342.

De Haan, W. (1991). Deterministic computer-aided optimum design of rock rubble-mound breakwater cross-sections. Coastal Engineering, Vol. 15, pp. 3-19.

Delft Hydraulics (1991). DUROSTA: Time Dependent Cross-Shore Transport Model For Extreme Hydraulic Conditions. Manual Version 1.11, The Netherlands (in Dutch).

Delft Hydraulics (undated). PROBA2 - Probabilistic Analysis Of Reliability Or Failure Of A Structure System. Computer program leaflet, Delft Hydraulics, The Netherlands.

Delft Hydraulics Laboratory (1978). Dune Erosion During The Storm Surge On January 3rd 1976, Verification Of The Provisional Dune Erosion Prediction Model. Report R 587, Delft Hydraulics Laboratory, The Netherlands (in Dutch).

Delta Committee (1960). Final Report And Interim Recommendations. Report Deltacom. No. 1, The Hague, The Netherlands (in Dutch).

Der Kiureghian, A. (1980). Reliability analysis under stochastic loads. Journal Of The Structural Division, ASCE, Vol. 106, pp. 411-429.

Der Kiureghian, A. (1990). Bayesian analysis of model uncertainty in structural reliability. Proc. 3rd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '90, March 26-28, 1990, Berkeley, California, USA (eds. Der Kiureghian, A.; Thoft-Christensen, P.), Springer-Verlag, Berlin, 1991, pp. 211-221.

Der Kiureghian, A.; De Stefano, M. (1990). An Efficient Algorithm For Second-Order Reliability Analysis. Report No. UCB/SEMM-90/20, Department Of Civil Engineering, University Of California At Berkeley, California.

Der Kiureghian, A.; Liu, P.-L. (1986). Structural reliability under incomplete probability information. Journal Of Engineering Mechanics, ASCE, Vol. 112, pp. 85-104.

Der Kiureghian, A.; Lin, H.-Z. & Hwang, S.-J. (1987). Second-order reliability approximations. Journal Of Engineering Mechanics, ASCE, Vol. 113, pp. 1208-1225.

De Waal, J.P. (1993). Wave Overtopping Of Vertical Breakwaters: Analysis Of Tests Performed In July 1993. Proc. MCS Project 1st Workshop, October, 1993, Madrid.

De Waal, J.P.; Van der Meer, J.W. (1992). Wave runup and overtopping on coastal structures. Proc. 23rd ICCE, October 4-9, 1992, Venice, Italy (ed. Edge, B.L.), ASCE, New York, 1993, pp. 1758-1771.

De Waal, J.P.; Tonjes, P. & Van der Meer, J.W. (1996). The effect of wind on wave overtopping of vertical structures. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 154-155.

Dirkson, M. (1996). High water management of the rivers Rhine and Meuse: the Dutch policy on flood protection along the main rivers and on the use of the river bed. Proc. 31st MAFF Conference Of River And Coastal Engineers, July 3-5, 1996, Keele University, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 2.3.1-2.3.8.

Ditlevsen, O. (1976). A Collection Of Notes Concerning Structural Reliability. 2nd International Workshop Of Definition, Classification And Evolution Of Code Formats, January, 1976, Mexico City, Dialog 2-76, Danmarks Ingeniorakademi, Lyngby.

Ditlevsen, O. (1979a). Generalized second moment reliability index. Journal Of Structural Mechanics, Vol. 7, pp. 435-451.

Ditlevsen, O. (1979b). Narrow reliability bounds for structural systems. Journal Of Structural Mechanics, Vol. 7, pp. 453-472.

Ditlevsen, O.; Bjerager, P. (1986). Methods of structural systems reliability. Structural Safety, Vol. 3, pp. 195-229.

Ditlevsen, O.; Madsen, H.O. (1980). Discussion of 'Optimal Reliability Analysis by Fast Convolution' by N. C. Lind. Journal Of The Engineering Mechanics Division, ASCE, Vol. 106, pp. 579-583.

Ditlevsen, O.; Madsen, H.O. (1981). Probabilistic modeling of man-made load processes and their individual and combined effects. Proc. ICOSSAR '81, 3rd International Conference On Structural Safety And Reliability, June 23-25, 1981, Norway (eds. Moan, T.; Shinozuka, M.), Elsevier Scientific Publishing Company, Amsterdam, pp. 103-134.

Dodge, Y. (1987). An introduction to statistical data analysis L1-norm based. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 1-21.

Dong, P.; Riddell, K.J. (1996). Probabilistic risk assessment of beach erosion at Pevensey Bay, Southern England. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 694-695.

Doody, J.P. (1989). Conservation and development of the coastal dunes in Great Britain. Perspectives In Coastal Dune Management (eds. Meulen, F. van der; Jungerius, P.D. & Visser, J.H.), SPB Academic Publishing, The Hague, The Netherlands, pp. 53-67.

Doornkamp, J.C. (1990) (ed.). The Greenhouse Effect And Rising Sea Levels In The UK. M1 Press Limited, Nottingham, UK.

Douglass, S.L. (1986). Review And Comparison Of Methods For Estimating Irregular Wave Overtopping Rates. Technical Report CERC-86-12, U.S. Army Engineer Waterways Experiment Station, Vicksburg, Mississippi.

Dover, A.R.; Bea, R.G. (1979). Applications of reliability methods to the design of coastal structures. Proc. Coastal Structures '79: A Specialty Conference On The Design, Construction Maintenance And Performance Of Port And Coastal Structures, March 14-16, 1979, Alexandria, Virginia, ASCE, New York, 1979, pp. 747-763.

Draper, L.; Blakey, A. (1969). Waves At Mersey Bar Light Vessel. Internal Report No. A.37, National Institute Of Oceanography, Surrey, UK.

Draper, N.R.; Smith, H. (1981). Applied Regression Analysis. Wiley Series In Probability And Mathematical Statistics. John Wiley & Sons, Inc, New York.

Duggal, A.S.; Niedzwecki, J.M. (1994). Probabilistic collision model for a pair of flexible cylinders. Applied Ocean Research, Vol. 16, pp. 165-175.

Eberhardt, A. (1979). The game of engineering. International Water Power & Dam Construction, Vol. 31, pp. 31-34.

Edelman, T. (1968). Dune erosion during storm conditions. Proc. 11th ICCE, September, 1968, London (ed. Edge, B.L.), ASCE, New York, 1969, pp. 719-722.

Etter, D.M. (1992). FORTRAN77 With Numerical Methods For Engineers And Scientists. The Benjamin/Cummings Publishing Company, Inc, California.

Evans, M.; Hastings, N. & Peacock, B. (1993). Statistical Distributions. John Wiley & Sons, Inc, New York.

Everitt, B.S. (1995) (ed.). The Cambridge Dictionary Of Statistics In The Medical Sciences. Cambridge University Press, Cambridge, UK.

Everitt, B.S.; Dunn, G. (1991). Applied Multivariate Data Analysis. Arnold, E., London.

Ferry Borges, J.; Castanheta, M. (1971). Structural Safety. Course 101 (2nd edition), Laboratório Nacional De Engenharia Civil, Lisbon.

Ferry Borges, J.; Castanheta, M. (1974). Implementation Of Design Rules At Level II. Joint CEB - CECM - CIB - FIP - IABSE Committee On Structural Safety, Document V-2.2, Laboratório Nacional De Engenharia Civil, Lisbon.

Ferry Borges, J.; Castanheta, M. (1983). Structural Safety. Course 101 (3rd edition), Laboratório Nacional De Engenharia Civil, Lisbon.

Fiessler, B. (1979). Das Programmsystem FORM Zur Berechnung Der Versagenswahrscheinlichkeit Von Komponenten Von Tragsystemen. Berichte Zur Zuverlassigkeits Theorie Der Bauwerke, No. 43, Technical University, Munich.

Fiessler, B.; Neumann, H.-J. & Rackwitz, R. (1979). Quadratic limit states in structural reliability. Journal Of The Engineering Mechanics Division, ASCE, Vol. 105, pp. 661-676.

Fjeld, S. (1977). Reliability of offshore structures. Proc. 9th Offshore Technology Conference (OTC), May 2-5, 1977, Houston, Texas, OTC, Texas, 1977, pp. 459-471.

Fleming, C.A. (1990). Principles and effectiveness of groynes. Proc. Short Course On Coastal Protection (ed. Pilarczyk, K.W.), June 30 to July 1, 1990, Delft University Of Technology, The Netherlands, pp. 121-156.

Flint, A.R.; Baker, M.J. (1976). Risk analysis for offshore structures - the aims and methods. Proc. ICE Conference, Design And Construction Of Offshore Structures, October 27-28, 1976, London, ICE, London, 1977, pp. 87-96.

Flint, A.R.; Smith, B.W.; Baker, M.J. & Manners, W. (1981). The derivation of safety factors for design of highway bridges. Proc. Conference On The New Code For The Design Of Steel Bridges, Granada Publishing.

Franco, L.; De Gerloni, M. & Van der Meer, J.W. (1994). Wave overtopping at vertical and composite breakwaters. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 1030-1045.

Frangopol, D.M.; Corotis, R.B. & Rackwitz, R. (1996) (eds.). Proc. 7th International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '96, April 2-4, 1996, Boulder, Colorado, USA, Elsevier Science Ltd, Oxford, UK, 1997.

Freund, R.J.; Littell, R.C. (1991). SAS System For Regression. SAS Series In Statistical Applications, 2nd Edition, SAS Institute Inc, Cary, NC, USA.

Fukuda, N.; Uno, T. & Irie, I. (1974). Field observations of wave overtopping of wave absorbing revetment. Coastal Engineering In Japan, Vol. 17, pp. 117-128.

Gadd, P.E.; Potter, R.E.; Safaie, B. & Resio, D. (1984). Wave runup and overtopping: a review and recommendations. Proc. 16th Offshore Technology Conference (OTC), May 7-9, 1984, Houston, Texas, OTC, Texas, 1984, pp. 239-248.

Gentle, J.E. (1977). Least absolute values estimation: an introduction. Communications Statistics - Simulated Computations, Vol. B6, pp. 313-328.

Gentle, J.E.; Narula, S.C. & Sposito, V.A. (1987). Algorithms for unconstrained L1 linear regression. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 83-94.

Gilbert, R.; Antille, G. (1992). A robust look at the use of regression diagnostics. The Statistician, Vol. 41, pp. 41-53.

Goda, Y. (1971). Expected rate of irregular wave overtopping of seawalls. Coastal Engineering In Japan, Vol. 14, pp. 43-51.

Goda, Y. (1985). Random Seas And Design Of Maritime Structures. University Of Tokyo Press, Tokyo, Japan.

Goda, Y.; Kishiara, Y. & Kamiyama, Y. (1975). Laboratory investigation on the overtopping rates for seawalls by irregular waves. Journal Of Port & Harbour Research Institution, Vol. 14, pp. 3-44.

Godfrey, P. (1994). The control of risk. Risk, Management & Procurement In Construction - 7th Annual Conference, September 16, 1994, London, Centre Of Construction Law And Management / CIRIA, London, UK.

Grandhi, R.V.; Wang, L.P. (1996). Structural failure probability calculation using nonlinear approximations. Proc. 7th International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '96, April 2-4, 1996, Boulder, Colorado, USA, Elsevier Science Ltd, Oxford, UK, 1997, pp. 165-172.

Green, C.H.; Penning-Rowsell, E.C. (1989). Flooding and the quantification of 'intangibles'. Journal Of The Institute Of Water And The Environmental Management.

Groebner, D.F.; Shannon, P.W. (1993). Business Statistics: A Decision-Making Approach. MacMillan Publishing Company, New York.

Gunst, R.F.; Mason, R.L. (1980). Regression Analysis And Its Application: A Data-Oriented Approach. Marcel Dekker, Inc, New York.

Hadi, A.S. (1992). Identifying multiple outliers in multivariate data. Journal Of Royal Statistics Society, Vol. 54, pp. 761-771.

Hague, R.C. (1992). UK South Coast Shingle Study - Joint Probability Assessment. Report SR 315, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Hahn, G.F.; Shapiro, S.S. (1967). Statistical Models In Engineering. John Wiley & Sons, Inc, New York.

Halton, J.H. (1970). A retrospective and prospective survey of the Monte Carlo method. SIAM Review, Vol. 12, pp. 1-63.

Hammersley, J.M.; Handscomb, D.C. (1964). Monte Carlo Methods. Methuen & Co Ltd, London, UK.

Harlow, E.H. (1980). Large rubble mound breakwater failures. Journal Of The Waterway, Port, Coastal And Ocean Division, ASCE, Vol. 106, pp. 275-278.

Harlow, E.H. (1985). Breakwaters. Developments In Breakwaters, Proc. ICE Conference On Breakwaters '85, October 2-4, 1985, London, Thomas Telford Ltd, London, 1986, pp. 309-329.

Hasofer, A.M.; Lind,, NC (1974). Exact and invariant second-moment code format. Journal Of The Engineering Mechanics Division, ASCE, Vol. 100, pp. 111-121.

Haugen, E.B. (1968). Probabilistic Approaches To Design. John Wiley & Sons, Inc, New York.

Havno, K.; Lasarte, A. & Hvidberg-Knudsen, M. (1996). Flood mapping and risk assessment based on joint probability extreme value analyses. Proc. 31st MAFF Conference Of River And Coastal Engineers, July 3-5, 1996, Keele University, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 3.4.1-3.4.12.

Hawkes, P.; Hague, R. (1994). Validation Of Joint Probability Methods For Large Waves And High Water Levels. Report SR 347, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Hedges, T.S. (1977). Discussion of beach replenishment: sea defences and a review of the role of artificial beach replenishment. Proc. ICE, Part 1, Vol. 62, pp. 136-137.

Hedges, T.S. (1983). The core and underlayers of a rubble mound structure. Proc. ICE Conference, Breakwaters - Design And Construction, May 4-6, 1983, London, Thomas Telford Ltd, London, 1984, pp. 99-106.

Hedges, T.S. (1985). Workshop D, Geotechnics. Developments In Breakwaters, Proc. ICE Conference On Breakwaters '85, October 2-4, 1985, London, Thomas Telford Ltd, London, 1986, pp. 301-308.

Hedges, T.S. (1993). Global Warming: Some Implications For Beaches And Coastal Structures. Report No. CE/2/93, University Of Liverpool, Department Of Civil Engineering, (paper presented at the Irish Sea Forum, Seminar 4: Rising Sea Level And Coastal Defence).

Hedges, T.S. (1995). Private Communication.

Helton, J.C.; Breeding, R.J. (1993). Calculation of reactor accident safety goals. Reliability Engineering And System Safety, Vol. 39, pp. 129-158.

Herbert, D.M. (1993). Wave Overtopping Of Vertical Walls. Report SR 316, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Herbert, D.M.; Allsop, N.W.H. & Owen, M.W. (1994). Overtopping of sea walls under random waves. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 1130-1142.

Hogben, N. (1990). Long term wave statistics. The Sea (eds. Le Mehaute, B.; Hanes, D.M.), John Wiley & Sons, Inc, New York.

Hohenbichler, M.; Rackwitz, R. (1983). First-order concepts in system reliability. Structural Safety, Vol. 1, pp. 177-188.

Hohenbichler, M.; Rackwitz, R. (1990). Improvement of second-order reliability estimates by importance sampling. Journal Of Engineering Mechanics, ASCE, Vol. 116, pp. 1183-1197.

Hohenbichler, M.; Gollwitzer, S.; Kruse, W. & Rackwitz, R. (1987). New light on first- and second-order reliability methods. Structural Safety, Vol. 4, pp. 267-284.

Housley, J.G. (1996). Justification for beach nourishment. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 554-555.

HR Wallingford (1989). Wave Study For The Arun Coast. Report EX 1871, Hydraulics Research Wallingford Ltd, Wallingford, UK.

HR Wallingford (1994). Pensarn Coast Protection Scheme - Project Appraisal. Report EX 3022, Hydraulics Research Wallingford Ltd, Wallingford, UK.

HR Wallingford / Sir William Halcrow & Partners Ltd (1993). Risk Assessment For Sea And Tidal Defence Schemes: First Technical Progress Report. Report No. 459/2/Y, National Rivers Authority (NRA), UK.

HR Wallingford / Sir William Halcrow & Partners Ltd (1995). Risk Assessment For Sea And Tidal Defence Schemes - Discussion Document. Report No. 459/7/Y, National Rivers Authority (NRA), UK.

Hudson, R.Y. (1959). Laboratory investigation of rubble mound breakwaters. Journal Of The Waterways & Harbors Division, ASCE, Vol. 85, pp. 93-121.

Hughes, S.A.; Chiu, T.Y. (1981). Beach And Dune Erosion During Severe Storms. Report UFL/COEL-TR/043, University Of Florida, Coastal & Oceanographic Engineering Department, Gainsville, Florida.

Hulsbergen, C.H. (1974). Origin, effect and suppression of secondary waves. Proc. 14th ICCE, June 24-28, 1974, Copenhagen, Denmark, ASCE, New York, 1975, pp. 392-411.

Hutchinson, T.P. (1993). Essentials Of Statistical Methods, In 41 Pages. School Of Mathematics And Statistics, University Of Sydney, Rumsby Scientific Publishing, Sydney, Australia.

Hydraulics Research Station (1980). Design Of Seawalls Allowing For Wave Overtopping. Report EX 924, Hydraulics Research Station, Wallingford, UK.

IAHR (International Association Of Hydraulic Research) - Working Group On Wave Generation And Analysis (1989). List of sea-state parameters. Journal Of The Waterway, Port Coastal And Ocean Engineering, ASCE, Vol. 115, pp. 793-808.

Iman, R.L.; Conover, W.J. (1980). Risk Methodology For Geologic Disposal Of Radioactive Waste: A Distribution-Free Approach To Inducing Rank Correlation Among Input Variables For Simulation Studies. Technical Report SAND80-0157, Sandia Laboratories, Albuquerque, New Mexico.

Iman, R.L.; Conover, W.J. (1982). A distribution-free approach to inducing rank correlation among input variables. Communications Statistics - Simulated Computations, Vol. 11, pp. 311-344.

Iribarren, R. (1938). Una Formula Para El Calcula De Los Diques De Escollera. Revista De Obras Publicas, Madrid.

Isherwood, R.M. (1987). Technical Note: A revised parameterisation of the JONSWAP spectrum. Applied Ocean Research, Vol. 9, pp. 47-50.

Jang, Y.-S.; Sitar, N. & Der Kiureghian, A. (1994). Reliability analysis of contaminant transport in saturated porous media. Water Resources Research, Vol. 30, pp. 2435-2448.

Jensen, J.J.; Madsen, H.O. & Pedersen, P.T. (1990). The effect of a non-linear wave force model on the reliability of a jack-up platform. Proc. 3rd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '90, March 26-28, 1990, Berkeley, California, USA (eds. Der Kiureghian, A.; Thoft-Christensen, P.), Springer-Verlag, Berlin, 1991, pp. 185-195.

Jensen, O.J. (1983). Breakwater superstructures. Proc. Coastal Structures '83: A Specialty Conference On The Design, Construction Maintenance And Performance Of Coastal Structures (ed. Weggel, J.R.), March 9-11, 1983, Arlington, ASCE, New York, 1983, pp. 272-285.

Jensen, O.J. (1984). A Monograph On Rubble Mound Breakwaters. Danish Hydraulic Institute, Horsholm, Denmark, 1984.

Jensen, O.J.; Juhl, J. (1987). Wave overtopping on breakwaters and sea dikes. Proc. 2nd International Conference On Coastal And Port Engineering In Developing Countries (ed. Nanjing Hydraulic Research Institute), September 7-11, 1987, Beijing, China Ocean Press, Beijing, 1988, pp. 716-730.

Jensen, O.J.; Sorensen, T. (1979). Overspilling/overtopping of rubble mound breakwaters. Results of studies, useful in design procedures. Coastal Engineering, Vol. 3, pp. 51-65.

Joint Committee On Structural Safety (CEB/CECM/CIB/FIP/IABSE) (1978). Common Unified Rules For Different Types Of Construction And Material. Vol. I Of The International System Of Unified Standard Codes Of Practice For Structures, Bulletin D'Information No. 124/125, CEB, Paris.

Kikkawa, H.; Shi-igai, H. & Kono, T. (1968). Fundamental study of wave overtopping on levees. Coastal Engineering In Japan, Vol. 11, pp. 107-115.

Kobayashi, N.; Raichle, A.W. (1994). Irregular wave overtopping of revetments in surf zones. Journal Of The Waterway, Port Coastal And Ocean Engineering, ASCE, Vol. 120, pp. 56-73.

Kobayashi, N.; Wurjanto, A. (1989). Wave overtopping on coastal structures. Journal Of The Waterway, Port Coastal And Ocean Engineering, ASCE, Vol. 115, pp. 235-251.

Koenker, R. (1987). A comparison of asymptotic testing methods for L1-regression. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 287-295.

Koffman, E.B.; Friedman, F.L. (1987). Problem Solving And Structured Programming In FORTRAN 77. Addison-Wesley, Reading, Massachussets.

Kohsiek, L. (1984). Particle Sizes Along The Dutch Coast. Publication WWWKZ-84G.282, The Hague, The Netherlands (in Dutch).

Koyluoglu, H.U.; Nielsen, S.R.K. (1994). New approximations for SORM integrals. Structural Safety, Vol. 13, pp. 235-246.

Kreuzer, H. (1994). Uncertainty in dam safety evaluation: an economic aspect. International Journal In Hydropower & Dams, pp. 53-56.

Kriebel, D.L. (1990). Advances in numerical modelling of dune erosion. Proc. 22nd ICCE, July 2-6, 1990, Delft, The Netherlands (ed. Edge, B.L.), ASCE, New York, 1990, pp. 2304-2317.

Kriebel, D.L.; Dean, R.G. (1985). Numerical simulation of time-dependent beach and dune erosion. Coastal Engineering, Vol. 9, pp. 221-245.

Lafitte, R. (1993). Probabilistic risk analysis of large dams: its value and limits. International Water Power & Dam Construction, Vol. 45, pp. 13-16.

Lamberti, A. (1992). Example application of reliability assessment of coastal structures. Proc. Short Course On Design And Reliability Of Coastal Structures (attached to the 23rd ICCE), Tecnoprint snc, Bologna, pp. 669-698.

Larson, M. (1988). Quantification Of Beach Profile Change. Report No. 1008, Lund University, Institute Of Science And Technology, Sweden.

Larson, M.; Kraus, N.C. (1989). SBEACH, Numerical Model For Simulating Storm-Induced Beach Change, Empirical Foundation And Model Development. Technical Report CERC-89-9, US Army Engineer Waterways Experiment Station, Vicksburg, Mississippi.

Law, A.M.; Kelton, W.D. (1991). Simulation Modeling & Analysis. McGraw-Hill International Editions, New York.

Lee, H.-L.; Mays, L.W. (1983). Improved risk and reliability model for hydraulic structures. Water Resources Research, Vol. 19, pp. 1415-1422.

Lewis-Beck, M.S. (1980). Applied Regression - An Introduction. Series: Quantitative Applications In The Social Sciences. Sage Publications, Inc, USA.

Liu, P.-L.; Der Kiureghian, A. (1991). Optimization algorithms for structural reliability. Structural Safety, Vol. 9, pp. 161-177.

Liverpool / Thessaloniki Network (1996). European Coasts: An Introductory Survey. Prepared as a part of the ERASMUS Project Contract ICP 92-G-2013, Delft University Of Technology, Hydraulic Engineering Group, Department Of Civil Engineering, The Netherlands.

Lording, P.T.; Scott, J.R. (1971). Armor stability of overtopped breakwater. Journal Of The Waterways, Harbors & Coastal Engineering Division, ASCE, Vol. 97, pp. 341-354.

Lumbers, J.P.; Cook, S.C. (1993). Risk management in the water industry. The Institution Of Water And Environmental Management Yearbook (incorporating the National Water Industry Handbook), IWEM, pp. 39-47.

Madsen, H.O.; Tvedt, L. (1990). Methods for time-dependent reliability and sensitivity analysis. Journal Of Engineering Mechanics, ASCE, Vol. 116, pp. 2118-2135.

Madsen, H.O.; Krenk, S. & Lind, N.C. (1986). Methods Of Structural Safety. Prentice-Hall, Inc, Englewood Cliffs, New Jersey.

Madsen, K.; Nielsen, H.B. (1989). Finite Algorithms For Robust Linear Regression. Report No. 395, The Danish Center For Applied Mathematics, The Technical University Of Denmark, Denmark.

MAFF (Ministry Of Agriculture Fisheries And Food) (1993a). Coastal Defence And The Environment: A Guide To Good Practice. MAFF, London, UK.

MAFF (Ministry Of Agriculture Fisheries And Food) (1993b). Flood And Coastal Defence. Project Appraisal Guidance Notes. MAFF, London, UK.

MAFF (Ministry Of Agriculture Fisheries And Food) (1993c). Strategy For Flood And Coastal Defence In England And Wales. MAFF, London, UK.

Manners, W. (1990). Classification and analysis of uncertainty in structural systems. Proc. 3rd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '90, March 26-28, 1990, Berkeley, California, USA (eds. Der Kiureghian, A.; Thoft-Christensen, P.), Springer-Verlag, Berlin, 1991, pp. 251-260.

Massey, B.S. (1971). Units, Dimensional Analysis And Physical Similarity. Van Nostrand Reinhold Company, London, UK.

McKay, M.D.; Conover, W.J. & Beckman, R.J. (1979). A Comparison Of Three Methods For Selecting Values Of Input Variables In The Analysis Of Output From A Computer Code. Technometrics, Vol. 21, pp. 239-245.

McKean, J.W.; Schrader, R.M. (1987). Least Absolute Errors Analysis Of Variance. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 297-305.

Meadowcroft, I.C.; Von Lany, P.H.; Allsop, N.W.H. & Reeve, D.E. (1994). Risk assessment for coastal and tidal defence schemes. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 3154-3166.

Meadowcroft, I.C.; Reeve, D.E.; Allsop, N.W.H.; Diment, R.P. & Cross, J. (1995). Development of new risk assessment procedures for coastal structures. Proc. ICE Conference, Coastal Structures And Breakwaters, April 27-29, 1995, London, Thomas Telford Ltd, London, 1996.

Meadowcroft, I.C.; Allsop, N.W.H.; Reeve, D.E.; Deakin, R. & Cross, J. (1996). Risk assessment for coastal and tidal defences. Proc. 31st MAFF Conference Of River And Coastal Engineers, July 3-5, 1996, Keele University, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 3.3.1-3.3.12.

Melchers, R.E. (1987). Structural Reliability: Analysis And Prediction. Ellis Horwood Ltd, Chichester, West Sussex, England.

Melchers, R.E. (1993). On the treatment of uncertainty information in PRA. Proc. Conference On Probabilistic Risk And Hazard Assessment, September 22-23, 1993, Newcastle N.S.W., Australia (eds. Melchers, R.E.; Stewart, M.G.), Balkema, Rotterdam, The Netherlands, pp. 13-26.

Melchers, R.E.; Stewart, M.G. (1993). Proc. Conference On Probabilistic Risk And Hazard Assessment, September 22-23, 1993, Newcastle N.S.W., Australia, Balkema, Rotterdam, The Netherlands.

Ministerie Van Verkeer En Waterstaat (RIKZ/RIZA/DWW) (1994). Hydraulische Randvoorwaarden Voor Primaire Waterkeringen. Conceptrapport.

Mol, A.; Ligteringen, H. & Paape, A. (1983). Risk analysis in breakwater design. Proc. ICE Conference, Breakwaters - Design And Construction, May 4-6, 1983, London, Thomas Telford Ltd, London, 1984, pp. 133-138.

Mol, A.; Groeneveld, R.L. & Waanders, A.J. (1984). Safety and reliability of breakwaters. Proc. 19th ICCE, September 3-7, 1984, Houston, Texas (ed. Edge, B.L.), ASCE, New York, 1985, pp. 2451-2466.

Monso, J.L.; Vidaor, A.; Cadevall, C. & Garcia, C. (1996). Overtopping reduction in crownwall design. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 468-469.

Morgan, B.J.T. (1984). Elements Of Simulation. Chapman & Hall, London, UK.

Motyka, J.M.; Brampton, A.H. (1994). Effectiveness Of Beach Control Operations. R&D Note 283, National Rivers Authority, Bristol, UK.

Murtha, J.A. (1995). Decisions Involving Uncertainty: An @Risk Tutorial For The Petroleum Industry, Murtha, J.A., Houston, Texas.

Naess, A. (1987). Reliability estimates by quadratic approximation of the limit state surface. Proc. 1st International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '87, May 6-8, 1987, Aalborg, Denmark (ed. Thoft-Christensen, P.), Springer-Verlag, Berlin, 1988, pp. 287-295.

NAG (1993). The NAG FORTRAN Library Manual, Mark 16, Vol. 4, EO4 - Minimizing Or Maximizing A Function. The Numerical Algorithms Group Limited, Oxford, UK.

Narula, S.C.; Wellington, J.F. (1977). Multiple linear regression with minimum sum of absolute errors. Applied Statistics, Vol. 26, pp. 106-111.

Nguyen, V.U.; Chowdhury, R.N. (1985). Simulation for risk analysis with correlated variables. Géotechnique, Vol. 35, pp. 47-58.

Nielsen, S.R.K.; Burcharth, H.F. (1983). Stochastic Design Of Rubble Mound Breakwaters. Hydraulics And Coastal Engineering Laboratory, University Of Aalborg, Department Of Civil Engineering, Denmark.

NRA (National Rivers Authority) (1991). Sea Defence Survey. NRA, UK.

Ouellet, Y.O. (1974). On the need of wave data for the design of rubble mound breakwaters. Ocean Waves, pp. 500-522.

Owen, M.W. (1982a). The hydraulic design of seawall profiles. Proc. ICE Conference On Shoreline Protection, Thomas Telford Ltd, London, pp. 129-136.

Owen, M.W. (1982b). Overtopping of sea defences. International Conference On The Hydraulic Modelling Of Civil Engineering Structures, September 22-24, 1982, Coventry, England, BHRA Fluid Engineering, Bedford, England, 1982, pp. 469-480.

Owen, M.W.; Steele, A.A.J. (1991). Effectiveness Of Recurved Wave Return Walls. Report SR 261 (revised April 1993), Hydraulics Research Wallingford Ltd, Wallingford, UK.

Palisade Corporation (1994). Risk Analysis And Simulation Add-In For Microsoft Excel Or Lotus 1-2-3. Windows Version Release 3.1 User's Guide. Palisade Corporation, USA.

Palisade Corporation (1996). BestFit: Probability Distribution Fitting For Windows. User's Guide. Palisade Corporation, USA.

Paloheimo, E. (1975). A Generalisation Of The Weighted Fractiles Method. Memorandum presented to the Joint Committee On Structural Safety, Wiesbaden, Germany.

Parker, D.J. (1993). The economic assessment of urban flood defence projects. British-Hungarian Workshop On Flood Defence, September 6-10, 1993, Budapest (ed. Hegedus, M.), Water Resources Research Centre, pp. 349-378.

Parker, D.J.; Green, C.H. & Thompson, P.M. (1987). Urban Flood Protection Benefits: A Project Appraisal Guide (the "Red Manual"). Gower Technical Press, Aldershot, Hampshire.

Pate, M.E. (1981). Consistency of safety standards across the different economic sectors. Proc. ICOSSAR '81, 3rd International Conference On Structural Safety And Reliability, June 23-25, 1981, Norway (eds. Moan, T.; Shinozuka, M.), Elsevier Scientific Publishing Company, Amsterdam, pp. 505-514.

Paté-Cornell, M.E. (1984). Fault trees vs. event trees in reliability analysis. Risk Analysis, Vol. 4, pp. 177-186.

Pedersen, J.; Burcharth, H.F. (1992). Wave forces on crown walls. Proc. 23rd ICCE, October 4-9, 1992, Venice, Italy (ed. Edge, B.L.), ASCE, New York, 1993, pp. 1489-1502.

Penning-Rowsell, E.C.; Green, C.H.; Thompson, P.M.; Coker, A.M.; Tunstall, S.M.; Richards, C. & Parker, D.J. (1992). The Economics Of Coastal Management: A Manual Of Benefit Assessment Techniques (the "Yellow Manual"). Belhaven Press, London, UK.

PIANC (Permanent International Association Of Navigation Congresses) (1988). Construction Deviations And Reliability Of Construction. Report of Subgroup D (Final Draft), PIANC, PTC II Working Group 12.

PIANC (Permanent International Association Of Navigation Congresses) (1992). Analysis Of Rubble Mound Breakwaters. Supplement to Bulletin No. 78/79, PIANC, PTC II Working Group 12.

Pilarczyk, K.W. (1990). Design of sea walls and dikes - including overview of revetments. Proc. Short Course On Coastal Protection (ed. Pilarczyk, K.W.), June 30 to July 1, 1990, Delft University Of Technology, The Netherlands, pp. 197-288.

Pita, C. (1992). Maritime structures safety. Design and monitoring methods. Proc. International Conference On The Pearl River Estuary In The Surrounding Area Of Macao, October 19-23, 1992, Macao, LECM/PRWRC/LNEC, Macao, 1992, pp. 407-419.

Plate, E.J.; Duckstein, L. (1988). Reliability - based design concepts in hydraulic engineering. Water Resource Bulletin, Vol. 24, Part 2, pp. 235-245.

Portnoy, S. (1987). Using regression fractiles to identify outliers. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 345-356.

Potts, A.E. (1993). An overview of the applicability of PRA techniques to the design of offshore platforms and associated marine operations. Proc. Conference On Probabilistic Risk And Hazard Assessment, September 22-23, 1993, Newcastle N.S.W., Australia (eds. Melchers, R.E.; Stewart, M.G.), Balkema, Rotterdam, The Netherlands, pp. 27-42.

Press, W.H.; Teukolsky, S.A.; Vetterling, W.T. & Flannery, B.P. (1992). Numerical Recipes In FORTRAN: The Art Of Scientific Computing. Cambridge University Press, Cambridge, UK.

Pugh, D.T. (1987) (ed.). Tides, Surges And Mean Sea-Level: A Handbook For Engineers And Scientists. John Wiley & Sons Ltd, Chichester, UK.

Pye, K. (1991). Beach deflation and backshore dune formation following erosion under storm surge conditions: an example from Northwest England. Acta Mechanica, Vol. 2, pp. 171-181.

Pye, K.; Neal, A. (1994). Coastal dune erosion at Formby Point, North Merseyside, England: causes and mechanisms. International Journal Of Marine Geology, Geochemistry And Geophysics, Vol. 119, pp. 39-56.

Rackwitz (1976). Practical probabilistic approach to design. First Order Reliability Concepts For Design Codes. CEB Bulletin D'Information No. 112, Joint Committee On Structural Safety (CEB/CECM/CIB/FIP/IABSE).

Rackwitz, R.; Fiessler, B. (1978). Structural reliability under combined random load sequences. Computers & Structures, Vol. 9, pp. 489-494.

Roelvink, J.A.; Stive, M.J.F. (1989). Bar generating cross-shore flow mechanism on a beach. Journal Of Geophysical Research, Vol. 94, pp. 4785-4800.

Ronold, K.O. (1990). Long-term reliability of a jackup-platform foundation. Proc. 3rd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '90, March 26-28, 1990, Berkeley, California, USA (eds. Der Kiureghian, A.; Thoft-Christensen, P.), Springer-Verlag, Berlin, 1991, pp. 303-314.

Rosenblatt, M. (1952). Remarks on a multivariate transformation. The Annals Of Mathematical Statistics, Vol. 23, pp. 470-472.

Rousseeuw, P.J.; Leroy, A.M. (1987) (eds.). Robust Regression And Outlier Detection. Wiley Series In Probability And Mathematics Statistics. John Wiley & Sons, Inc, USA.

Royal Society (1992). Risk: Analysis, Perception And Management. Report Of A Royal Society Study Group. The Royal Society, London, UK.

Rubinstein, R.Y. (1981). Simulation And The Monte Carlo Method. John Wiley & Sons, Inc, New York.

Ruiz, L.F.V.; Quirós, F.B. (1994). Cost-benefit analysis of shore protection investments. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 3237-3250.

Ryu, C.R.; Kang, Y.K. & Kim, J.W. (1992). Optimal design of rubble mound structures under the irregular wave. Proc. 23rd ICCE, October 4-9, 1992, Venice, Italy (ed. Edge, B.L.), ASCE, New York, 1993, pp. 1503-1516.

Salih, B.A. (1989). Probabilistic Properties Of Wave Climates. Ph.D. Thesis, University Of Liverpool, Department Of Civil Engineering, UK.

Samuels, P.; Brampton, A.H. (1996). The effects of climate change on flood and coastal defence in the UK. Proc. 31st MAFF Conference Of River And Coastal Engineers, July 3-5, 1996, Keele University, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 2.2.1-2.2.10.

Sargent, F.E.; Birkemeier, W.A. (1985). Application Of The Dutch Method For Estimating Storm-Induced Dune Erosion. Instruction Report CERC-85-2, US Army Corps Of Engineer Waterways Experiment Station, Vicksburg, Mississippi.

SAS Institute Inc (1988). SAS/IML User's Guide. Release 6.03 Edition. SAS Institute Inc, Cary, NC, UK.

Sawaragi, T.; Deguchi, I. & San Park, I. (1988). Reduction of wave overtopping rate by the use of artificial reefs. Proc. 21st ICCE, June 20-25, 1988, Malaga, Spain (ed. Edge, B.L.), ASCE, New York, 1989, pp. 335-349.

Schittkowski, K. (1985). NLPQL: a FORTRAN subroutine solving constrained nonlinear programming problems. Algorithms And Software For Optimization, Annals Of Operations Research (ed. Monma, C.L.), Baltzer, Basel, 1985/1986, pp. 485-500.

Schueller, G.I.; Choi, H.S. (1977). Offshore platform risk based on reliability function model. Proc. 9th Offshore Technology Conference (OTC), May 2-5, 1977, Houston, Texas, OTC, Texas, 1977, pp. 473-482.

Schuttrumpf, H. (1997). Private Communication.

Shinozuka, M. (1983). Basic analysis of structural safety. Journal Of Structural Engineering, ASCE, Vol. 109, pp. 721-740.

Simm, J.D.; Brampton, A.H.; Beech, N.W. & Brooke, J.S. (1996). Beach Management Manual. CIRIA Report 153, London, UK.

Smith, G.M.; Seijffert, J.W.W. & Van der Meer, J.W. (1994). Erosion and overtopping of a grass dike: large scale model tests. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 2639-2652.

Smith, G.N. (1986). Probability And Statistics In Civil Engineering: An Introduction. Collins, London, UK.

Smith, O.P. (1987). Cost-effectiveness of breakwater cross sections. Journal Of The Waterway, Port Coastal And Ocean Engineering, ASCE, Vol. 113, pp. 447-460.

Smith, P.J.; Buckee, J.W. (1985). Calculating in-place and recoverable hydrocarbons: a comparison of alternative methods. Proc. SPE Hydrocarbon Economics And Evaluation Symposium, March 14-15, 1985, Dallas, Texas, Society Of Petroleum Engineers AIME, Texas, pp. 165-182.

Sorensen, R.M. (1991). Impact of expected climate change effects on coastal structure design. Proc. ICE Conference, Coastal Structures And Breakwaters, November 6-8, 1991, London, Thomas Telford Ltd, London, 1992, pp. 1-12.

Sorensen, T.; Jensen, O.J. (1985). Experience gained from breakwater failures. Developments In Breakwaters, Proc. ICE Conference On Breakwaters '85, October 2-4, 1985, London, Thomas Telford Ltd, London, 1986, pp. 103-117.

Sposito, V.A. (1990). Some Properties Of Lp-Estimators. Robust Regression: Analysis And Applications. Statistics - Textbooks And Monographs (eds. Lawrence, K.D.; Arthur, J.L.), Marcel Dekker, Inc, USA, pp. 23-58.

Sposito, V.A.; Kennedy, W.J. & Gentle, J.E. (1977). Lp norm fit of a straight line. Applied Statistics, Vol. 26, pp. 114-118.

SPSS (1993). SPSS For Windows: Base System User's Guide Release 6.0. Marija J. Norusis/SPSS Inc, USA.

Stangenhaus, G. (1987). Bootstrap and inference procedures for L1 regression. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 323-332.

Startzman, R.A.; Wattenbarger, R.A. (1985). An improved computation procedure for risk analysis problems with unusual probability functions. Proc. SPE Hydrocarbon Economics And Evaluation Symposium, March 14-15, 1985, Dallas, Texas, Society Of Petroleum Engineers AIME, Texas, pp. 123-132.

Steetzel, H.J. (1990). Cross-shore transport during storm surges. Proc. 22nd ICCE, July 2-6, 1990, Delft, The Netherlands (ed. Edge, B.L.), ASCE, New York, 1990, pp. 1922-1932.

Steetzel, H.J. (1993). Cross-Shore Transport During Storm Surges. Publication No. 476, Delft Hydraulics, The Netherlands (Ph.D. Thesis, Delft University Of Technology).

Stive, M.J.F.; Nicholls, R.J. & De Vriend, H.J. (1991). Sea-level rise and shore nourishment: a discussion. Coastal Engineering, Vol. 16, pp. 147-163.

Streeter, V.L.; Wylie, E.B. (1979). Fluid Mechanics. McGraw-Hill Kogakusha Ltd, Tokyo.

Super-Software (1994). User Manual Of SUPER-RISK 1.0. Super-Software, The Netherlands.

Swart, D.H. (1974). Offshore Sediment Transport And Equilibrium Beach Profiles. Ph.D. Thesis, Delft University Of Technology, The Netherlands.

TACPI (Technical Advisory Committee On Protection Against Inundation) (1974). Wave Run-Up And Overtopping. Government Publishing Office, The Hague, The Netherlands.

TAW (Technical Advisory Committee On Water Defences) (1972). Provisional Guideline For The Computation Of Dune Erosion During A Storm Surge. TAW, The Netherlands (in Dutch).

Technical University Of Braunschweig / SOGREAH Ingenierie / Centro De Estudios Y Experimentacion De Obras Publicas (1997). Performance Of Soft Beach Systems And Nourishment Measures For European Coasts (Contract No. MAS3-CT95-0004). Abstracts-In-Depth Of The First Annual Workshop.

Thoft-Christensen, P.; Baker, M.J. (1982). Structural Reliability Theory And Its Applications. Springer-Verlag, Berlin.

Thoft-Christensen, P.; Murotsu, Y. (1986). Application Of Structural Systems Reliability Theory. Springer-Verlag, Berlin.

Thomas, R.S. (1994). Breakwaters and coast defences: coastal defence structures. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 377-380.

Thomas, R.S.; Hall, B. (1992) (eds.). Seawall Design. CIRIA, Butterworth-Heinemann Ltd, London, UK.

Thompson, E.F.; Scheffner, N.W. (1996). Risk-based analysis of coastal projects. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 688-689.

Thornton, E.B.; Guza, R.T. (1983). Transformation of wave height distribution. Journal Of Geophysical Research, Vol. 88, pp. 5925-5938.

Townend, I.H. (1994a). Breakwaters and coast defences: risk assessment of coastal defences. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 381-384.

Townend, I.H. (1994b). Variation in design conditions in response to sea-level rise. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 205-213.

Townend, I.H.; Fleming, C.A. (1991). Beach nourishment and socio-economic aspects. Coastal Engineering, Vol. 16, pp. 115-127.

Turkstra, C.J. (1970). Theory Of Structural Design Decisions. Solid Mechanics Study 2. Solid Mechanics Division, University Of Waterloo, Ontario, Canada.

Turkstra, C.J.; Madsen, H.O. (1980). Load combinations in codified structural design. Journal Of The Structural Division, ASCE, Vol. 106, pp. 2527-2543.

Tvedt, L. (1983). Two Second-Order Approximations To The Failure Probability. Veritas Report RDIV/20-004-83, Det Norske Veritas, Oslo, Norway.

Tvedt, L. (1988). Second order reliability by an exact integral. Proc. 2nd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '88, September 26-28, 1988, London (ed. Thoft-Christensen, P.), Springer-Verlag, Berlin, 1989, pp. 377-384.

Tvedt, L. (1990). Distribution of quadratic forms in normal space - application to structural reliability. Journal Of Engineering Mechanics, ASCE, Vol. 116, pp. 1183-1197.

U.S. Army Corps Of Engineers - Coastal Engineering Research Center (1984). Shore Protection Manual (4th edition). U.S. Government Printing Office, Washington DC.

Van Aalst, W. (1983). Wave Height-Surge Level Relationships For The Dutch Coast. Note WWKZ-83G.218, Rijkswaterstaat, The Hague, The Netherlands (in Dutch).

Van de Graaff, J. (1977). Dune erosion during a storm surge. Coastal Engineering, Vol. 1, pp. 99-134.

Van de Graaff, J. (1983). Probabilistic design of dunes. Proc. Coastal Structures '83: A Specialty Conference On The Design, Construction Maintenance And Performance Of Coastal Structures (ed. Weggel, J.R.), March 9-11, 1983, Arlington, ASCE, New York, 1983, pp. 820-831.

Van de Graaff, J. (1984). Probabilistic Methods In Dune Design. Delft University Of Technology, The Netherlands (in Dutch).

Van de Graaff, J. (1986). Probabilistic design of dunes; an example from The Netherlands. Coastal Engineering, Vol. 9, pp. 479-500.

Van de Graaff, J. (1995). Private communication.

Van de Graaff, J.; Bijker, E.W. (1988). Seawalls and shoreline protection. Proc. 21st ICCE, June 20-25, 1988, Malaga, Spain (ed. Edge, B.L.), ASCE, New York, 1989, pp. 2090-2101.

Van de Graaff, J.; Koster, M.J. (1990). Dune and beach erosion and nourishment. Proc. Short Course On Coastal Protection (ed. Pilarczyk, K.W.), June 30 to July 1, 1990, Delft University Of Technology, The Netherlands, pp. 99-120.

Van de Graaff, J.; Niemeyer, H.D. & Van Overeem, J. (1991). Beach nourishment, philosophy and coastal protection policy. Coastal Engineering, Vol. 16, pp. 3-22.

Van der Meer, J.W. (1987). Risk Analysis In Breakwater Design. Report Of Subgroup C, PIANC, PTC II Working Group 12.

Van der Meer, J.W. (1990). Probabilistic design of breakwaters. Handbook Of Coastal And Ocean Engineering (ed. Herbich, J.B.), Gulf, Houston, 1990/1991, pp. 1051-1058.

Van der Meer, J.W. (1993). Conceptual Design Of Rubble Mound Breakwaters. Publication No. 483, Delft Hydraulics, The Netherlands.

Van der Meer, J.W. (1994). Breakwaters and coast defences: numerical modelling of wave-structure interaction. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 359-362.

Van der Meer, J.W.; Janssen, J.P.F.M. (1995). Wave run-up and wave overtopping at dikes. Wave Forces On Inclined And Vertical Wall Structures (eds. Kobayashi, N.; Demirbilek, Z.), ASCE, New York.

Van der Meer, J.W.; Pilarczyk, K.W. (1987). Stability Of Breakwater Armour Layers: Deterministic And Probabilistic Design. Publication No. 378, Delft Hydraulics Laboratory, The Netherlands.

Van der Meer, J.W.; D'Angremond, K. & Juhl, J. (1994). Probabilistic calculations of wave forces on vertical structures. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 1754-1767.

Van der Meulen, T.; Gourlay, M.R. (1968). Beach and dune erosion test. Proc. 11th ICCE, September, 1968, London (ed. Edge, B.L.), ASCE, New York, 1969, pp. 701-707.

Van Raalte, G.H.; Loman, G.J.A. (1993). Implementation of coastline recharge schemes. Proc. 28th MAFF Conference Of River And Coastal Engineers, July 5-7, 1993, Loughborough University Of Technology, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 3.3.1-3.3.20.

Vasco Costa, F. (1987). Reliability of partly damaged structures. Proc. 1st International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '87, May 6-8, 1987, Aalborg, Denmark (ed. Thoft-Christensen, P.), Springer-Verlag, Berlin, 1988, pp. 67-76.

Vasco Costa, F. (1990). Expectation ratio versus probability. Proc. 3rd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '90, March 26-28, 1990, Berkeley, California, USA (eds. Der Kiureghian, A.; Thoft-Christensen, P.), Springer-Verlag, Berlin, 1991, pp. 53-62.

Vellinga, P. (1982). Beach and dune erosion during storm surges. Coastal Engineering, Vol. 6, pp. 361-387.

Vellinga, P. (1983). Predictive computational model for beach and dune erosion during storm surges. Proc. Coastal Structures '83: A Specialty Conference On The Design, Construction Maintenance And Performance Of Coastal Structures (ed. Weggel, J.R.), March 9-11, 1983, Arlington, ASCE, New York, 1983, pp. 806-819.

Vellinga, P. (1986). Beach And Dune Erosion During Storm Surges. Publication No. 372, Delft Hydraulics, The Netherlands (Ph.D. Thesis, Delft University Of Technology).

Veneziano, D. (1976). Basic principles and methods of structural safety. First Order Reliability Concepts For Design Codes. CEB Bulletin D'Information No. 112, Joint Committee On Structural Safety (CEB/CECM/CIB/FIP/IABSE).

Vrijling, J.K. (1990). Probabilistic design of flood defences. Proc. Short Course On Coastal Protection (ed. Pilarczyk, K.W.), June 30 to July 1, 1990, Delft University Of Technology, The Netherlands, pp. 39-97.

Vrijling, J.K. (1993). Probabilistic methods in hydraulic engineering. Bulletin No. 80, PIANC (Permanent International Association For Navigation Congresses), pp. 56-61.

Vrijling, J.K.; Bruinsma, J. (1980). Hydraulic boundary conditions. Hydraulic Aspects Of Coastal Structures. Part 1 - Developments In Hydraulic Engineering, Delft University Press, The Netherlands, pp. 109-133.

Walpole, R.E.; Myers, R.H. (1993). Probability And Statistics For Engineers And Scientists. MacMillan Publishing Company, New York.

Wang, L.P.; Grandhi, R.V. (1995). Intervening variables and constraint approximations in safety index and failure probability calculations. Structural Optimization, Vol. 10, pp. 2-8.

Ward, D.L. (1992). Prediction Of Overtopping Rates For Irregular Waves On Riprap Revetments. Miscellaneous Paper CERC-92-4, US Army Engineer Waterways Experimental Station, Vicksburg, Mississippi.

Ward, D.L.; Wibner, C.G.; Zhang, J. & Edge, B. (1994). Wind effects on runup and overtopping. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 1687-1699.

Ward, D.L.; Zhang, J.; Wibner, C.G. & Cinotto, C.M. (1996). Wind effects on runup and overtopping of coastal structures. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 156-157.

Watanabe, A.; Shiba, K. & Isobe, M. (1994). A numerical model of beach change due to sheet-flow. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 2785-2798.

Wen, Y.K.; Chen, H.C. (1987). On fast integration for time variant structural reliability. Probabilistic Engineering Mechanics, Vol. 2, pp. 156-162.

Yen, B.C. (1989). Engineering approaches to risk and reliability analysis. Risk Analysis And Management Of Natural And Man Made Hazards, Proc. 3rd Conference Sponsored by the Engineering Foundation (eds. Haimes, Y.Y.; Stakhiv, E.Z.), ASCE, New York.

BIBLIOGRAPHY

Abdo, T.; Rackwitz, R. (1990). A new beta-point algorithm for large time-invariant and time variant reliability problems. Proc. 3rd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '90, March 26-28, 1990, Berkeley, California, USA (eds. Der Kiureghian, A.; Thoft-Christensen, P.), Springer-Verlag, Berlin, 1991, pp. 1-12.

Ahrens, J.P.; Heimbaugh, M.S. (1986). Irregular wave overtopping of seawalls. Proc. Conference Oceans '86, September, 1986, Institute Of Electrical And Electronic Engineers, IEEE, pp. 96-103.

Ang, A.H.-S.; Cornell, C.A. (1974). Reliability bases of structural safety and design. Journal Of The Structural Division, ASCE, Vol. 100, pp. 1755-1769.

Ashkar, F.; Rousselle, J. (1981). Design discharge as a random variable: a risk study. Water Resources Research, Vol. 17, pp. 577-591.

Atkinson, D.; Houston, J. (1991). The sand dunes of the Sefton Coast. Proc. Sefton Coast Research Seminar, May 31, 1991, Liverpool, National Museums & Galleries on Merseyside, 1991.

Bakker, W.T.; Vrijling, J.K. (1980). Probabilistic design of sea defences. Proc. 17th ICCE, March 23-28, 1980, Sydney, Australia, ASCE, New York, 1981, pp. 2040-2059.

Battjes, J.A. (1984). A review of methods to establish the wave climate for breakwater design. Coastal Engineering, Vol. 8, pp. 141-160.

Bjerager, P.; Krenk, S. (1987). Sensitivity measures in structural reliability analysis. Proc. 1st International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '87, May 6-8, 1987, Aalborg, Denmark (ed. Thoft-Christensen, P.), Springer-Verlag, Berlin, 1988, pp. 459-470.

Bjerager, P.; Krenk, S. (1989). Parametric sensitivity in first order reliability theory. Journal Of Engineering Mechanics, ASCE, Vol. 115, pp. 1577-1582.

Blackman, D.L. (1985). New estimates of annual sea level maxima in the Bristol channel. Estuarine, Coastal And Shelf Science, Vol. 20, pp. 229-232.

Burcharth, H.F. (1991). Design innovations, including recent research contributions. Proc. ICE Conference, Coastal Structures And Breakwaters, November 6-8, 1991, London, Thomas Telford Ltd, London, 1992, pp. 55-93.

Burgess, K.A.; Reeve, D.E. (1994). The development of a method for the assessment of sea defences and risk of flooding. Proc. 29th MAFF Conference Of River And Coastal Engineers - "A Strategic Approach", July 4-6, 1994, Loughborough University Of Technology, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 5.3.1-5.3.12.

Carter, D.J.T.; Challenor, P.G.; Ewing, J.A.; Pitt, E.G.; Srokosz, M.A. & Tucker, M.J. (1986). Estimating Wave Climate Parameters For Engineering Applications. Offshore Technology Report OTH 86 228, Institute Of Oceanographic Sciences, Surrey, UK.

Carvalho, M.M. (1990). Valores Extremos Da Agitação Maritima: Notas Sobre Periodos De Retorno. Relatório 126/90 - NPP, Laboratório Nacional De Engenharia Civil, Lisboa.

Carvalho, M.M. (1992). Probabilistic Methods For Structural Design. Programa COMETT, Laboratório Nacional De Engenharia Civil, Lisbon, pp. 1-30.

Chiu, T.Y. (1977). Beach and dune response to hurricane Eloise of September 1975. Proc. 5th Symposium Waterway, Port, Coastal And Ocean Division, ASCE - Coastal Sediments '77, 1977, Charleston, South Carolina, ASCE, New York, pp. 116-134.

Christiani, E.; Burcharth, H.F. & Sorensen, J.D. (1996). Reliability based optimal design of vertical breakwaters modelled as a series system of failure. Book Of Abstracts, 25th ICCE, September 2-6, 1996, Orlando, Florida, ASCE, New York, 1996, pp. 692-693.

Clemen, R.T. (1991). Making Hard Decisions: An Introduction To Decision Analysis. PWS-Kent, Boston.

Clifford, J.E. (1994). Breakwaters and coast defences: introduction to a suite of technical notes. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 353-354.

Coles, S.G.; Tawn, J.A. (1990). Statistics of coastal flood prevention. Philosophical Transactions Of The Royal Society Of London, Series A, Vol. 332, pp. 457-476.

Conte, S.D.; De Boor, C. (1980). Elementary Numerical Analysis: An Algorithmic Approach. Mathematics And Statistics Series. McGraw-Hill, Inc, Tokyo, Japan.

Cornell, C.A. (1969). Structural safety specifications based on second-moment reliability analysis. Symposium Of The International Association For Bridge And Structural Engineering (IABSE), London, 1969, IABSE, Zurich, 1969, pp. 235-245.

CUR - TAW (Centre For Civil Engineering Research And Codes - Technical Advisory Committee On Water Defences) (1984). Guide To The Assessment Of The Safety Of Dunes As A Sea Defence. Staatsuitgeverij, The Hague, The Netherlands (in Dutch).

CUR - TAW (Centre For Civil Engineering Research And Codes - Technical Advisory Committee On Water Defences) (1991). Guide For The Design Of River Dikes. Vol. I - Upper River Area. Report 142, Gouda, The Netherlands.

Dawdy, D.R.; Lettenmaier, D.P. (1987). Initiative for risk-based flood design. Journal Of Hydraulic Engineering, ASCE, Vol. 113, pp. 1041-1051.

Dean, R.G. (1991). Beach Profiles. Handbook Of Coastal And Ocean Engineering: Offshore Structures, Marine Foundations, Sediment Processes And Modelling (ed. Herbich, J.B.), Gulf Publishing Company, Houston, pp. 715-734.

Dean, R.G. (1992). Beach nourishment: design principles. Proc. Short Course On Design And Reliability Of Coastal Structures (attached to the 23rd ICCE), Tecnoprint snc, Bologna, pp. 301-349.

De Haan, T. (1994). Policy and economic aspects of dike-raising in The Netherlands. Proc. 29th MAFF Conference Of River And Coastal Engineers - "A Strategic Approach", July 4-6, 1994, Loughborough University Of Technology, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 8.4.1-8.4.13.

Delft Hydraulics Laboratory (1983). Settling Velocities Of Sand In Sea Water Of 5°C. Publication M1263IV-b, Delft Hydraulics Lab oratory, The Netherlands (in Dutch).

Dette, H.-H.; Uliczka, K. (1987). Prototype investigation on time-dependent dune recession and beach erosion. Proc. Coastal Sediments '87, May 12-14 1987, New Orleans, USA (ed. Kraus, N.C.), ASCE, New York, 1987, pp. 1430-1444.

Douglass, S.L. (1984). Irregular wave overtopping rates. Proc. 19th ICCE, September 3-7, 1984, Houston, Texas (ed. Edge, B.L.), ASCE, New York, 1985.

Enevoldsen, I.; Sorensen, J.D. (1994). Reliability-based optimization in structural engineering. Structural Safety, Vol. 15, pp. 169-196.

Ferry Borges, J.; Castanheta, M. (1973a). Checking Safety At Level II. Joint CEB - CECM - CIB - FIP - IABSE Committee On Structural Safety, Document IV-2.4, Laboratório Nacional De Engenharia Civil, Lisbon.

Ferry Borges, J.; Castanheta, M. (1973b). Generalized Theory Of Structural Safety. CEB International Course On Structural Concrete, Specialized Course - C1 On Structural Safety, Laboratório Nacional De Engenharia Civil, Lisbon.

Fleming, C.A.; Reeve, D.E. & Osment, J. (1993). Coastal modelling in design and shoreline management. Proc. 28th MAFF Conference Of River And Coastal Engineers, July 5-7, 1993, Loughborough University Of Technology, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 1.2.1-1.2.10.

Frangopol, D.M.; Corotis, R.B. (1996). Reliability-based structural system optimization: state-of-the-art versus state-of-the-practice. Analysis And Computation (ed. Cheng, F.Y.), ASCE, New York, pp. 67-78.

Freudenthal, A.M. (1968-1970). Probabilistic evaluation of design criteria for maritime structures. Bulletin No. 2, PIANC (Permanent International Association Of Navigation Congresses), pp. 17-31.

Gill, P.E.; Murray, W. & Wright, M.H. (1981). Practical Optimization. Academic Press, London, UK.

Gittler, W.E.; Krumrine, P.H. (1985). A novel approach for risk assessment in chemical EOR projects. Proc. SPE Hydrocarbon Economics And Evaluation Symposium, March 14-15, 1985, Dallas, Texas, Society Of Petroleum Engineers AIME, Texas, pp. 67-79.

Graff, J. (1981). An investigation of the frequency distributions of annual sea level maxima at ports around Great Britain. Estuarine, Coastal And Shelf Science, Vol. 12, pp. 389-449.

Hedges, T.S.; Reis, M.T. (1996). A New Regression Model For Wave Overtopping Data. Proc. 1st International Conference On Environmental Problems In Coastal Regions, August, 1996, Rio de Janeiro, Brazil, Computational Mechanics Publications, Southampton, UK, 1996.

Hedges, T.S.; Reis, M.T. (1998). Random Wave Overtopping Of Simple Seawalls: a New Regression Model. Proc. ICE Water, Maritime & Energy, Vol. 130, pp. 1-10.

Hertz, D.B.; Thomas, H. (1984). Practical Risk Analysis: An Approach Through Case Histories. John Wiley & Sons Ltd, New York.

Hohenbichler, M.; Rackwitz, R. (1981). Non-normal dependent vectors in structural safety. Journal Of The Engineering Mechanics Division, ASCE, Vol. 107, pp. 1227-1238.

Iman, R.L.; Davenport, J.M. & Zeigler, D.K. (1980). Latin Hypercube Sampling (A Program Users Guide). Technical Report SAND79-1473, Sandia Laboratories, Albuquerque, New Mexico.

Juhl, J.; Sloth, P. (1994). Wave overtopping of breakwaters under oblique waves. Proc. 24th ICCE, October 23-28, 1994, Kobe, Japan (ed. Edge, B.L.), ASCE, New York, 1995, pp. 1182-1196.

Kobayashi, N.; Reece, A.M. (1983). Irregular wave overtopping on gravel islands. Journal Of The Waterway, Port Coastal And Ocean Engineering, ASCE, Vol. 109, pp. 429-444.

Kooman, D. (1980). Risk analysis and design criteria in breakwater design. Proc. International Seminar On Criteria For Design And Construction Of Breakwaters And Coastal Structures, 1980, pp. 288-300.

Liu, P.-L.; Der Kiureghian, A. (1988). Optimization algorithms for structural reliability. Computational Probabilistic Methods. AMD-Vol. 93 (eds. Liu, B. et al.), ASME, New York, pp. 185-196.

Longuet-Higgins, M.S. (1952). On the statistical distribution of the heights of the sea. Journal Of Marine Research, Vol. XI, pp. 245-266.

Losada, M.A.; Curto, L.A.G. (1982). An approximation to the failure probability of maritime structures under a sea state. Coastal Engineering, Vol. 5, pp. 147-157.

MAFF (Ministry Of Agriculture Fisheries And Food) (1993). Coastal Defence And The Environment: A Strategic Guide For Managers And Decision Makers In The National Rivers Authority, Local Authorities And Other Bodies With Coastal Responsibilities. MAFF, London, UK.

Marsland, A. (1966). The design and construction of earthen floodbanks. River Engineering And Water Conservation Works (ed. Thorn, R.B.), The University Press, Aberdeen, UK, pp. 361-392.

Mettam, J.D. (1994). Breakwaters and coast defences: probabilistic design of rubble mound breakwaters - the way ahead. Proc. ICE Water, Maritime & Energy, Vol. 106, pp. 367-370.

Mettam, J.D.; Berry, J.G. (1982). Factors of safety for the design of breakwaters. Proc. 18th ICCE, November 14-19, 1982, Cape Town, South Africa (ed. Edge, B.L.), ASCE, New York, 1983, pp. 2097-2106.

Mettam, J.D.; Berry, J.G. (1983). Factors of safety for the design of breakwaters. The Dock & Harbour Authority, Vol. 63, pp. 319-322.

Mizumura, k; Yamamoto, M.; Endo, T. & Shiraishi, N. (1988). Reliability analysis of rubble mound breakwaters. Proc. 21st ICCE, June 20-25, 1988, Malaga, Spain (ed. Edge, B.L.), ASCE, New York, 1989, pp. 2053-2062.

Mulder, T.; Vrijling, J.K. (1980). Probabilistic load determination. Hydraulic Aspects Of Coastal Structures. Part 1 - Developments In Hydraulic Engineering, Delft University Press, The Netherlands, pp. 91-133.

Narule, S.C.; Wellington, J.F. (1990). On the robustness of the simple linear minimum sum of absolute errors regression. Robust Regression: Analysis And Applications. Statistics - Textbooks And Monographs (eds. Lawrence, K.D.; Arthur, J.L.), Marcel Dekker, USA, pp. 129-141.

NERC (Natural Environment Research Council) (1975). Flood Studies Report. Vol. 1 - Hydrological Studies. London, UK.

Niedoroda, A.W.; Swift, D.J.P. (1991). Shoreface processes. Handbook Of Coastal And Ocean Engineering: Offshore Structures, Marine Foundations, Sediment Processes And Modelling (ed. Herbich, J.B.), Gulf Publishing Company, Houston, pp. 735-770.

Nolte, K.G. (1974). Statistical methods for determining extreme sea states. 2nd International Conference On Port And Ocean Engineering, Under Arctic Conditions, August 27-30, 1973, Reykjavik, Iceland, University Of Iceland, Department Of Engineering And Science, pp. 705-742.

O'Connor, P.D.T. (1992). Practical Reliability Engineering. John Wiley & Sons, Chichester.

Paloheimo, E.; Hannus, M. (1974). Structural design based on weighted fractiles. Journal Of The Structural Division, ASCE, Vol. 100, pp. 1367-1378.

Penning-Rowsell, E.C.; Chatterton, J.B. (1977). The Benefits Of Flood Alleviation: A Manual Of Assessment Techniques (the "Blue Manual"). Gower Technical Press, Aldershot, Hampshire.

Pilarczyk, K.W. (1990). Data collection and prediction methods - an overview. Proc. Short Course On Coastal Protection (ed. Pilarczyk, K.W.), June 30 to July 1, 1990, Delft University Of Technology, The Netherlands, pp. 405-422.

Pouliquen, L.Y. (1970). Risk Analysis In Project Appraisal. World Bank Occasional Papers Number Eleven. The Johns Hopkins University Press, Baltimore.

Pugh, D.T.; Vassie, J.M. (1978). Extreme sea levels from tide and surge probability. Proc. 16th ICCE, August 27 to September 3, 1978, Hamburg, Germany, ASCE, New York, 1979, pp. 911-930.

Pye, K. (1990). Physical and human influences on coastal dune development between the Ribble and Mersey estuaries, northwest England. Coastal Dunes: Form And Processes (eds. Nordstrom, K.F.; Psuty, N.P. & Carter, R.W.G.), John Wiley & Sons Ltd, UK, pp. 339-359.

Raiffa, H. (1968) (ed.). Decision Analysis: Introductory Lectures On Choices Under Uncertainty. Addison-Wesley, Reading, Massachusetts.

Reeve, C.; Bettess, R. & Walden, Y. (1993). Investigation of flood embankments. British-Hungarian Workshop On Flood Defence, September 6-10, 1993, Budapest (ed. Hegedus, M.), Water Resources Research Centre, pp. 225-244.

Ress, S. (1993). Flood defence economics. British-Hungarian Workshop On Flood Defence, September 6-10, 1993, Budapest (ed. Hegedus, M.), Water Resources Research Centre, pp. 339-346.

Riddell, K. (1993). Probabilistic methods in scheme design. Proc. 28th MAFF Conference Of River And Coastal Engineers, July 5-7, 1993, Loughborough University Of Technology, Ministry Of Agriculture Fisheries And Food (MAFF) - Flood And Coastal Defence Division, London, pp. 1.3.1-1.3.10.

Ronold, K.O. (1990). Reliability analysis of a coastal dike. Coastal Engineering, Vol. 14, pp. 43-56.

Rousseeuw, P.J. (1987). An application of L1 to astronomy. Statistical Data Analysis Based On The L1-Norm And Related Methods (ed. Dodge, Y.), Elsevier Science Publishers B.V., Amsterdam, pp. 437-445.

Sears, G.F.; Phillips, N.V. (1985). Fractured reservoir evaluation using Monte Carlo techniques. Proc. SPE Hydrocarbon Economics And Evaluation Symposium, March 14-15, 1985, Dallas, Texas, Society Of Petroleum Engineers AIME, Texas, pp. 183-188.

Selivanov, A.O. (1994). Risk assessment analysis for possible modifications of the Russian coasts. Proc. LITTORAL 94, September 26-29, 1994, Lisbon, Portugal (eds. Carvalho, S.; Gomes, V.), Association EUROCOAST, Lisbon, Portugal, pp. 291-300.

Sorensen, J.D. (1986). Reliability-based optimization of structural elements. Structural Reliability Theory, Paper No. 18, University Of Aalborg, Denmark.

Sorensen, J.D.; Burcharth, H.F. & Christiani, E. (1994). Reliability analysis and optimal design at monolithic vertical wall breakwaters. Proc. 6th International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '94, September, 1994, Assisi, Italy (eds. Rackwitz, R.G.; Augusti, E.B.), Chapman & Hall, New York, 1995, pp. 257-264.

Stans, J.C. (1979). Model investigations and probabilistic design. Proc. Coastal Structures '79: A Specialty Conference On The Design, Construction Maintenance And Performance Of Port And Coastal Structures, March 14-16, 1979, Alexandria, Virginia, ASCE, New York, 1979, pp. 764-777.

Steele, A.A.J.; Owen, M.W. (1985). Prestatyn Coast Defence: Prediction Of Discharge Over A Proposed Seawall. Report EX 1335, Hydraulics Research Wallingford Ltd, Wallingford, UK.

Steetzel, H.J. (1987). A model for beach and dune profile changes near dune revetments. Proc. Coastal Sediments '87, May 12-14 1987, New Orleans, USA (ed. Kraus, N.C.), ASCE, New York, 1987, pp. 87-97.

Sundararajan, C.R. (1991). Guide To Reliability Engineering: Data Analysis, Applications, Implementation, And Management. Van Nostrand Reinhold, New York.

Turner, R.C.; Baker, M.J. (1988). A probabilistic wave model for reliability analysis. Proc. 2nd International Federation For Information Processing (IFIP) WG7.5 Conference On Reliability And Optimization Of Structural Systems '88, September 26-28, 1988, London (ed. Thoft-Christensen, P.), Springer-Verlag, 1989, pp. 357-375.

Van der Meer, J.W. (1990). Static and dynamic stability of loose materials. Proc. Short Course On Coastal Protection (ed. Pilarczyk, K.W.), June 30 to July 1, 1990, Delft University Of Technology, The Netherlands, pp. 157-195.

Van der Meer, J.W.; Pilarczyk, K.W. (1990). Stability of low-crested and reef breakwaters. Proc. 22nd ICCE, July 2-6, 1990, Delft, The Netherlands (ed. Edge, B.L.), ASCE, New York, 1990, pp. 1375-1388.

Van der Meer, J.W.; Stam, C.-J.M. (1992). Wave runup on smooth and rock slopes of coastal structures. Journal Of The Waterway, Port Coastal And Ocean Engineering, ASCE, Vol. 118, pp. 534-550.

Van der Meer, J.W.; Juhl, J. & Van Driel, G. (1992a). Probabilistic calculations of wave forces on vertical structures. MAST G6-S Coastal Structures, Final Overall Workshop, November 5-6, 1992, Lisbon, Portugal.

Van der Meer, J.W.; Petit, H.A.H.; Van den Bosh, P.; Klopman, G. & Broekens, R.D. (1992b). Numerical simulation of wave motion on and in coastal structures. Proc. 23rd ICCE, October 4-9, 1992, Venice, Italy (ed. Edge, B.L.), ASCE, New York, 1993, pp. 1772-1784.

Van Oorschot, J.H.; Van Raalte, G.H. (1991). Beach nourishment; execution methods and developments in technology. Coastal Engineering, Vol. 16, pp. 23-42.

Vasco Costa, F. (1982). Breakwaters safety as affected by the dispersion of armour units. Jornadas Portuguesas De Engenharia De Estruturas, Laboratório Nacional De Engenharia Civil, Lisboa, pp. A.99-A.111.

Vrijling, J.K.; De Swart, P.F. (1991). Berm breakwater design - the longshore transport case: a probabilistic approach. Proc. ICE Conference, Coastal Structures And Breakwaters, November 6-8, 1991, London, Thomas Telford Ltd, London, 1992, pp. 513-525.

Wang, H.; Peene, S.J. (1990). A probabilistic model of rubble mound armor stability. Coastal Engineering, Vol. 14, pp. 307-331.

Ward, D.L.; Ahrens, J.P. (1992). Overtopping Rates For Seawalls. Miscellaneous Paper CERC-92-4, US Army Engineer Waterways Experimental Station, Vicksburg, Mississippi.

Watt, W.E.; Wilson, K.C. (1978). Allowance for risk and uncertainty in the optimal design of hydraulic structures. International Symposium On Risk And Reliability In Water Resources (ed. McBeantal, E.A.), pp. 575-591.

Weggel, J.R. (1976). Wave overtopping equation. Proc. 15th ICCE, July 11-17, 1976, Honolulu, Hawaii, ASCE, New York, 1977, pp. 2737-2755.

Wen, Y.K. (1987). Approximate methods for nonlinear time-variant reliability analysis. Journal Of Engineering Mechanics, ASCE, Vol. 113, pp. 1826-1839.

Yamamoto, M.; Mizura, K.; Endo, T. & Shiraishi, N. (1988). Reliability based design of rubble mound breakwaters. Proc. 21st ICCE, June 20-25, 1988, Malaga, Spain (ed. Edge, B.L.), ASCE, New York, 1989, pp. 2062-2074.

Yamamoto, M.; Mizumura, K.; Endo, T. & Shiraishi, N. (1990). Reliability analysis of composite breakwaters protected with armor blocks. Proc. 22nd ICCE, July 2-6, 1990, Delft, The Netherlands (ed. Edge, B.L.), ASCE, New York, 1990, pp. 1402-1416.

Yamamoto, Y.; Horikawa, K. (1992). New methods to evaluate wave runup height and wave overtopping rate. Proc. 23rd ICCE, October 4-9, 1992, Venice, Italy (ed. Edge, B.L.), ASCE, New York, 1993, pp. 1734-1747.

APPENDIX A

Analysis Of Overtopping Data

APPENDIX A1 - Regression Theory

Those concepts of regression analysis applied in this study to the re-analysis of Owen's data are presented here. The main references are Lewis-Beck (1980), Gunst & Mason (1980), Draper & Smith (1981), Rousseeuw & Leroy (1987), McKean & Schrader (1987), Freund & Littell (1991), Law & Kelton (1991), SPSS (1993), Groebner & Shannon (1993), Ataa (1994) and Everitt (1995).

The linear model with one independent variable, X, can be written as follows (Gunst & Mason, 1980):

$$Y = \beta_0 + \beta_1 X + \varepsilon \qquad (A1.1)$$

where Y is the dependent variable, and β_0 and β_1 are unknown parameters of the model. Sometimes X is called the predictor or regressor variable, Y the predicted or response variable, and β_0 and β_1 the intercept and the slope, respectively. $\beta_0 + \beta_1 X$ represents the systematic component of the variability of Y, and ε is a random error term which takes into account the fact that the model does not exactly describe the behaviour of the response. The linear model can be expressed as follows:

$$Y = b_0 + b_1 X + e \qquad (A1.2)$$

where, based on a sample of N observed (X,Y) values, b_0 and b_1 are estimates of parameters β_0 and β_1. These estimates are used to construct the fitted model (also called the predictive or prediction equation, or the linear regression line for Y on X) as follows:

$$\hat{Y} = b_0 + b_1 X \qquad (A1.3)$$

where $\hat{Y}$ is the predictive value of Y. From eqs. (A1.2) and (A1.3), the residual from the regression is defined by $e = Y - \hat{Y}$ (see Figure A1.1).
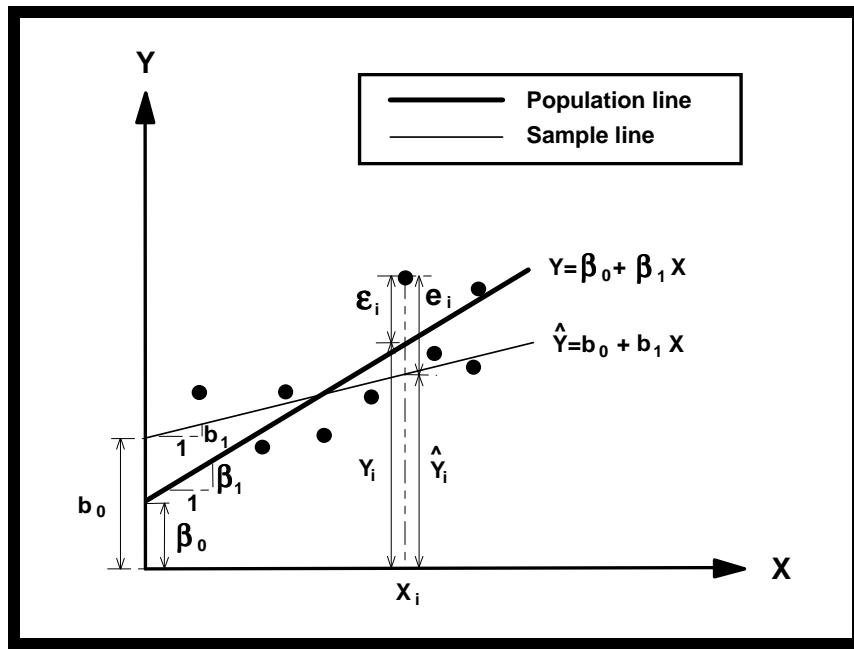
Figure A1.1: Regression residuals and error term for a regression model with a single independent variable (modified after Berry, 1993).

A1.1 Least-Squares (LS) Method

A1.1.1 Basic LS Principle

There are many criteria for determining the regression line which best fits the observed data, i.e. for determining estimates b_0 and b_1 (Draper & Smith, 1981; Dodge, 1987). The most common criterion, the basis of the least-squares method, minimises the sum of the squares of the residuals. b_0 and b_1 are then calculated as follows (Gunst & Mason, 1980):

$$b_1 = \frac{\sum_{i=1}^{N}(X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{N}(X_i - \overline{X})^2} \qquad b_0 = \overline{Y} - b_1\overline{X} \qquad (A1.4)$$

where

$$\overline{X} = \frac{1}{N}\sum_{i=1}^{N}X_i \quad \text{and} \quad \overline{Y} = \frac{1}{N}\sum_{i=1}^{N}Y_i \qquad (A1.5)$$

are the mean values of X and Y, respectively. b_0 and b_1 are called the LS estimates. The slope estimate, b_1, represents the estimated change in Y

associated with a unit change in X. The intercept, b_0, estimates the value of Y when X equals zero. Substituting b_0 into eq. (A1.3):

$$\hat{Y} = \overline{Y} + b_1(X - \overline{X}) \qquad\qquad (A1.6)$$

Note that when $X = \overline{X}$, $\hat{Y} = \overline{Y}$ so that $(\overline{X}, \overline{Y})$ lies on the fitted line.

In determining the LS estimates, b_0 and b_1, the following assumptions have to be made (Lewis-Beck, 1980):

1) No specification error:
   i. the relationship between X and Y is linear
   ii. no relevant independent variables have been excluded
   iii. no irrelevant independent variables have been included

2) No measurement error, i.e. X and Y are accurately measured.

3) For each observation, the expected value of the error term is zero ($E[\varepsilon_i] = 0$).

4) The variance of the error term is constant for all values of X ($E[\varepsilon_i^2] = \sigma^2$, known as homoscedasticity).

5) The error terms for different observations are uncorrelated ($E[\varepsilon_i \varepsilon_j] = 0$ for $i \neq j$).

6) The independent variable is uncorrelated with the error term ($E[\varepsilon_i X_i] = 0$).

Assumption 1 asserts that the theoretical model embodied in the equation is correct, or almost so, over the range of observed values; that is, the functional form of the relationship is actually a straight line and no variables have been improperly included or excluded. The need for assumption 2 is self-evident: if the measurements are inaccurate, then the estimates could be inaccurate. If only the dependent variable is measured with error, then the LS estimates may remain unbiased[1], provided the error is random (Lewis-Beck, 1980). If the independent variable is measured with any error, then the LS estimates will be biased. However, if the variance of the error is small compared to the variance of the true X values, this error is usually ignored (Draper & Smith, 1981) and this policy is adopted in this study. Assumption 3 is not restrictive for regression models containing an intercept,

---

[1] An unbiased estimator correctly estimates the population parameter on average, i.e. $E[b] = \beta$ (Beaumont, 1986).

because if the intercept term is included in the equation, the mean of the residuals is always zero so that the mean of the residuals provides no information about the mean of the errors (Gunst & Mason, 1980; SPSS, 1993). If this assumption is not met and the model includes the intercept, the intercept is biased but the slope estimate is unchanged. If assumption 4 is violated, the LS estimates remain unbiased, but they do not have minimum variance and the significance tests and confidence intervals, usually given in statistical packages, are invalid (Lewis-Beck, 1980). Assumption 5 requires that there is no correlation between errors. If correlation is present, the LS estimates are still unbiased but the significance tests and confidence intervals are invalid (Lewis-Beck, 1980). However, if the number of observations is large when compared to the number of independent variables, the dependency can be ignored for practical purposes (SPSS, 1993) and this is the case for Owen's data set. Note that since the sum of the residuals is constrained to be zero, the residuals cannot be strictly independent. If assumption 6 does not hold, the LS estimates are biased.

In general terms, when assumptions 1 to 6 are met, desirable estimates of the population parameters β_0 and β_1 are obtained, i.e. they are unbiased and, of all the estimates that are unbiased, they have minimum variance (Lewis-Beck, 1980; Everitt & Dunn, 1991). Furthermore, if the error term, ε_i, is normally distributed, or almost so, the estimates are maximum likelihood estimates. In this case, significance tests carried out in order to determine how likely it is that the population parameter values differ from zero, are reliable. Confidence intervals can also be relied upon. If the error term is not normally distributed, significance tests and confidence intervals should not be interpreted in the usual fashion (Lewis-Beck, 1980; SPSS, 1993). Rarely are assumptions not violated in one way or another in regression analyses (SPSS, 1993). However, ignoring the assumptions can lead to results which are improperly used and interpreted. One should not rush into using statistics lightly without proper understanding and without verifying the underlying assumptions (Carvalho, 1982).

Knowing the estimates of the parameters, Y can be predicted for a given X value. It is important to know how good is a prediction of Y provided by the prediction equation, i.e. how well does the equation account for variations in the dependent variable. The total observed variability in the dependent variable from its mean value, $\overline{Y}$ (Figure A1.2), can be divided conveniently

into two components (Lewis-Beck, 1980; Freund & Littell, 1991; SPSS, 1993):

i) $\hat{Y}_i - \overline{Y}$, the variability accounted for by the regression line;

ii) $Y_i - \hat{Y}_i$, the variability unexplained by regression.
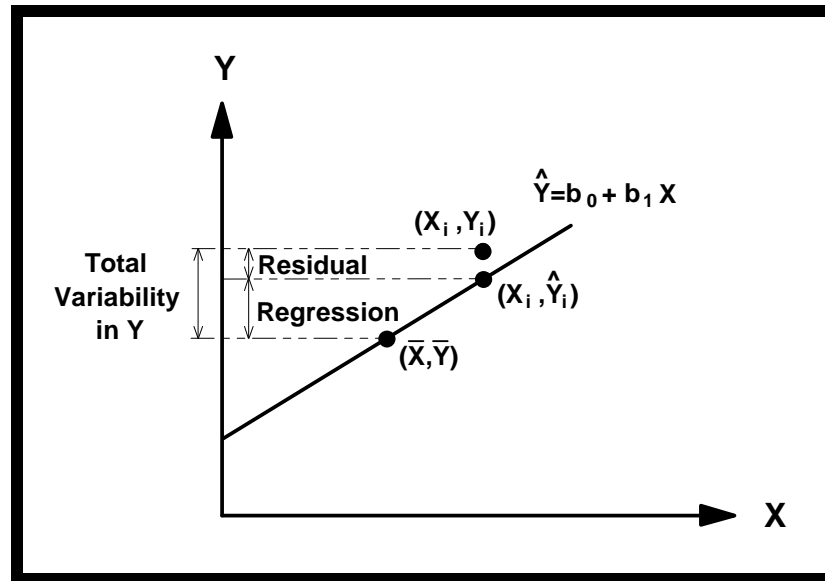
Figure A1.2: Components of the variability in the dependent variable, Y , for the LS method (modified after SPSS, 1993).

It can be shown (Gunst & Mason, 1980; Draper & Smith, 1981) that:

$$\sum_{i=1}^{N}(Y_i - \overline{Y})^2 = \sum_{i=1}^{N}(\hat{Y}_i - \overline{Y})^2 + \sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 \tag{A1.7}$$

The quantity before the equals sign is the total sum of squares, TSS, and it is invariant for a given set of data (X_i, Y_i). The first quantity following the equals sign is the regression sum of squares, SS_{Reg}, and the second quantity is the residual sum of squares, SS_{Res}. It is recalled here that the LS method guarantees that SS_{Res} is at its minimum. Clearly, SS_{Reg} should be large relative to SS_{Res}, since then the regression line is explaining the majority of the variability in Y about its mean. The sum of squares has associated with it the degrees of freedom, DF. DF indicates how many independent pieces of information involving the N independent numbers, $Y_1,...,Y_N$, are needed to

compile the sum of squares (Draper & Smith, 1981). These definitions form the basis of some of the tests used to establish the goodness of fit of the regression line.

A1.1.2 Goodness Of Fit

Summary Statistics For The Regression Line

A commonly used measure of the goodness of fit of the regression line (Gunst & Mason, 1980; Draper & Smith, 1981; Groebner & Shannon, 1993) is the coefficient of determination, R^2:

$$R^2 = \frac{SS_{Reg}}{TSS} = \frac{\sum_{i=1}^{N}(\hat{Y}_i - \overline{Y})^2}{\sum_{i=1}^{N}(Y_i - \overline{Y})^2} \qquad (A1.8)$$

R^2 measures the proportion of total variation about the mean of Y explained by the regression line. The possible values of R^2 range from 0 to 1. If $R^2=1$, all observations fall on the regression line, so that Y can be predicted from X without error. If $R^2=0$, there is no linear relationship between X and Y. The closer R^2 is to unity, the better the fit of the regression line to the data points.

R^2 tends to be an optimistic estimate of how well the model fits the population (SPSS, 1993). The model usually does not fit the population as well as it fits the sample from which it is derived. The adjusted statistic R_a^2 attempts to correct R^2 to more closely reflect the goodness of fit of the model in the population (Freund & Littell, 1991; SPSS, 1993):

$$R_a^2 = R^2 - \frac{p(1-R^2)}{N-p-1} \qquad (A1.9)$$

where p is the number of independent variables in the model.

The square root of R^2, R, equals the correlation coefficient between X and Y, ρ_{XY}, and that between Y and $\hat{Y}$, $\rho_{Y\hat{Y}}$. The sign of the square root is the sign of b_1.

Another commonly used measure of the goodness of fit of the regression line is the sample standard error of the estimate, S:

$$S = \sqrt{\frac{SS_{Res}}{N-p-1}} = \sqrt{\frac{\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2}{N-p-1}} \qquad (A1.10)$$

The population variance of the residuals, σ^2, is generally not known and S^2 is its usual estimate (SPSS, 1993). S is the measure of deviation of the observed Y values around the regression line and so the smaller S is, the more reliable are the predictions (Gunst & Mason, 1980). S^2 is also known as the mean square residual, MS_{Res} (see Table A1.1).

Analysis Of Variance - ANOVA Table

One method of testing if the regression line is statistically significant (i.e. that predicting Y based on X and using the LS regression line is preferable to just using the overall mean of Y) is to use analysis of variance of Y from its mean value, as shown in Table A1.1.

Source of Variation	Degrees of Freedom, DF	Sum of Squares	Mean Square	F
Regression	p	$SS_{Reg} = \sum_{i=1}^{N}\left(\hat{Y}_i - \overline{Y}\right)^2$	$MS_{Reg} = \dfrac{SS_{Reg}}{p}$	$\dfrac{MS_{Reg}}{S^2}$
Residual	N-p-1	$SS_{Res} = \sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2$	$MS_{Res} = S^2 = \dfrac{SS_{Res}}{N-p-1}$	_____

Table A1.1: ANOVA table (modified after Draper & Smith, 1981).

If the regression assumptions are met, the ratio of the mean square regression to the mean square residual is distributed as a one tailed F statistic with p and N-p-1 degrees of freedom (SPSS, 1993). Generally, F is a statistic used to test the null hypothesis (Figure A1.3) Ho: *the regression model does not explain any of the total variation in the dependent variable* (Groebner & Shannon, 1993). In the case of a single independent variable, it tests the null hypothesis Ho: $\beta_1 = 0$ (SPSS, 1993). If $F = MS_{Reg} / S^2 > F_{Crit} = F(p, N-p-1; 1-\alpha)$, the null hypothesis can be rejected at

the α level of significance on the basis of the data used and it can be inferred that the LS equation appears to be a suitable predictor (see Figure A1.4). Very often α is chosen as 0.05 (Hutchinson, 1993), although there is no logical reason for using this convention (see Appendix A5 for more details on significance level, α).

HYPOTHESIS TESTING

Basic idea

- State a particular hypothesis (known as the null hypothesis and denoted Ho).

- Assume that the hypothesis is true and determine the probability of getting a result at least as far from the hypothesised value as is the observed value.

- If this probability is very low, say less than α, then reject the hypothesis (α is known as the significance level).

Type I and Type II errors

- In reality, the null hypothesis is either true or false.

- The conclusion at the end of a hypothesis test is to reject or not reject Ho.

- Type I error: Ho is rejected when it is true (the probability of doing this, given that Ho is correct, is the significance level, α).

- Type II error: Ho is not rejected when it is false.

- Choosing a very low value of α gives a very low probability of a Type I error, but gives a high probability of a Type II error.

Figure A1.3: Hypothesis testing.
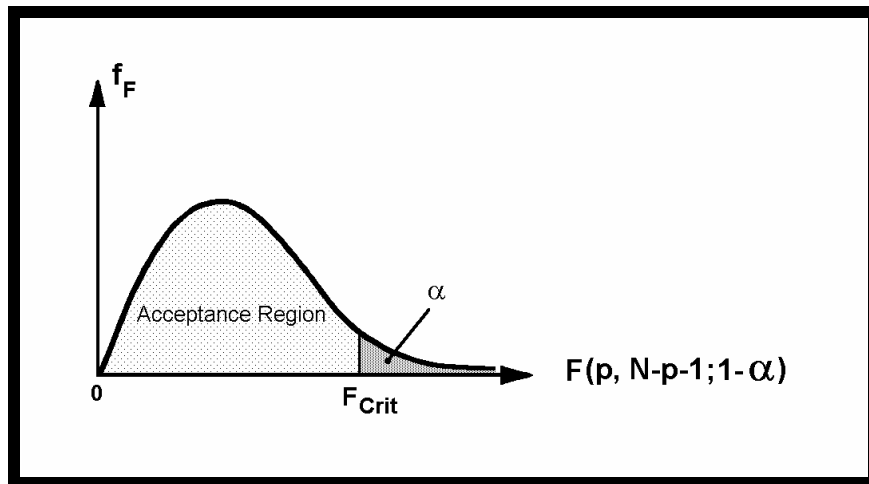
Figure A1.4: The F distribution used as a significance test for LS regression (modified after Groebner & Shannon, 1993).

Statistics Of The Parameters In The Regression Line

Since b_0 and b_1 are estimates of the population's parameters, they typically differ from the population values and vary from sample to sample (SPSS, 1993). When the assumptions of linear regression are met, the distributions of b_0 and b_1 are Normal, with means β_0 and β_1. The standard deviations of the parameters are, respectively:

$$SEb_0 = \sigma \sqrt{\frac{1}{N} + \frac{\overline{X}^2}{(N-1)S_X^2}} \qquad SEb_1 = \frac{\sigma}{\sqrt{(N-1)S_X^2}} \qquad (A1.11)$$

where S_X is the standard deviation of the X values. SEb_0 and SEb_1 are commonly referred to as the standard errors of b_0 and b_1, respectively. They are useful measures of the dispersion of the parameters' estimates (Groebner & Shannon, 1993). Since σ is usually not known, S is used instead and SEb_0 and SEb_1 are then called estimates of the standard errors of b_0 and b_1 (Figure A1.5). As S_X^2 is the sample variance,

$$S_X^2 = \sum_{i=1}^{N} \frac{(X_i - \overline{X})^2}{N-1} \qquad (A1.12)$$
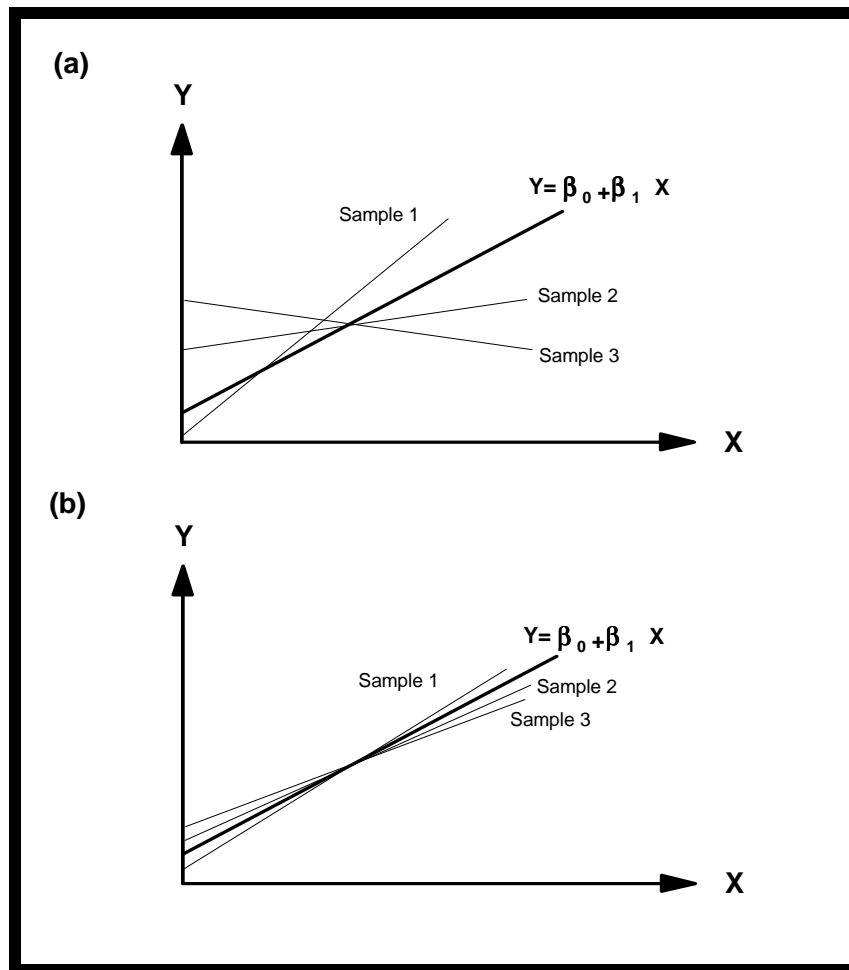
Figure A1.5: Large (a) and small (b) standard error of the slope of the LS regression line (modified after Groebner & Shannon, 1993).

Another way of testing if the intercept and/or the slope of the regression line are zero, i.e. Ho: $\beta_0 = 0$ and/or Ho: $\beta_1 = 0$, is to use the following statistic (SPSS, 1993):

$$t = \frac{b_0}{SEb_0} \qquad \text{or} \qquad t = \frac{b_1}{SEb_1} \qquad\qquad \text{(A1.13)}$$

The distribution of the statistic, when the assumptions are met, is Student's t distribution with N-p-1 degrees of freedom, and is used in a two-tailed test (Lewis-Beck, 1980). So if $|t| > t_{Crit} = t(N-p-1; 1-0.5\alpha)$, then the parameter estimate is significant and Ho can be rejected at the α level of significance (Figure A1.6).
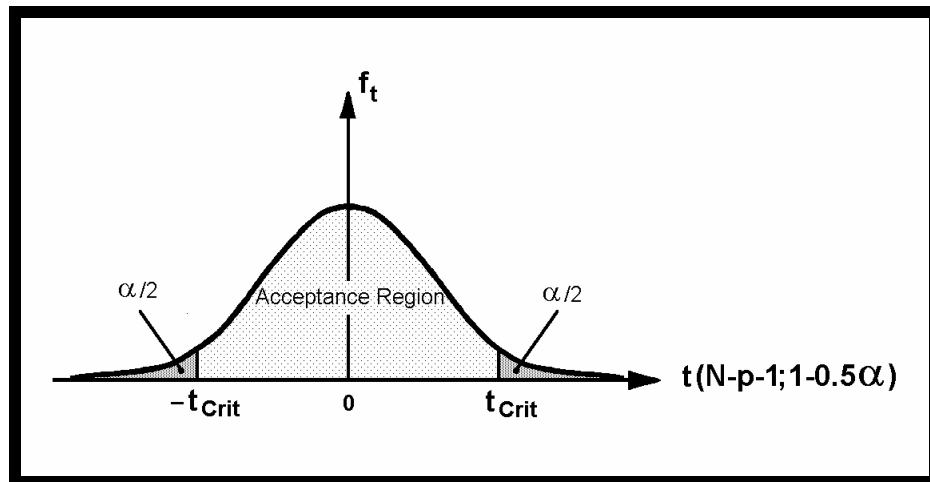
Figure A1.6: Student's t distribution used as a significance test for LS regression (modified after Groebner & Shannon, 1993).

The values of b_0 and b_1 are point estimates of the unknown parameters. However, if the assumptions of regression are not violated, a range of values can be calculated for each parameter which, within a designated likelihood, α, includes the population value. This range is called the confidence interval (Groebner & Shannon, 1993). A $(1-\alpha)\%$ confidence interval for the parameters is defined as follows:

$$b \pm t(N-p-1; 1-0.5\alpha)\,SEb \qquad (A1.14)$$

where b is the parameter considered (b_0 or b_1) and SEb is its standard error. The most widely used interval is a two-tailed 95% confidence interval, i.e. $\alpha = 0.05$: if repeated samples are drawn from a population under the same conditions and if 95% confidence intervals are calculated from each sample, then 95% of the intervals will contain the unknown parameter, β (Law & Kelton, 1991). As there is no logical reason for using specifically a 95% confidence interval, the 90, 95 and 99% confidence intervals are given here for completeness.

The confidence interval not only allows an interval estimate of the parameters (as opposed to a point estimate) but also provides a test of the null hypothesis Ho: $\beta = 0$, where β can be either β_0 or β_1. If the value of zero does not fall within the interval, the null hypothesis can be rejected on the basis of the data used, and $\beta \neq 0$ is accepted (Draper & Smith, 1981).

Predicted Values And Their Standard Errors

A major use of regression is prediction (Gunst & Mason, 1980). However, the fact that a regression model explains a significant proportion of variation in the dependent variable, does not mean that it is satisfactory for prediction (Groebner & Shannon, 1993). It is important to know how useful the independent variable is for predicting Y. Furthermore, it must be stressed that inferences made from a regression line only apply over the range of the data contained in the sample.

One may wish to predict the mean of Y for all cases with a given value of X, Xo, or to predict the value of Y for a single case. In both instances, the predicted value is the same (Draper & Smith, 1981); what differs is the standard error (Groebner & Shannon, 1993). The standard error of the predicted mean value of Y at a specific value Xo of X is:

$$SE\hat{Y} = \sigma \sqrt{\frac{1}{N} + \frac{(Xo - \overline{X})^2}{(N-1)S_X^2}} \qquad (A1.15)$$

Replacement of σ by S provides the corresponding estimated standard error. It can be seen that the smallest value occurs when $Xo = \overline{X}$. The larger the distance from the mean, the greater the standard error. Thus, the mean of Y for a given value of X is better estimated for central values of the observed X's than for extreme values (Draper & Smith, 1981). Prediction intervals for the mean predicted Y at Xo are calculated in the standard way: $\hat{Y} \pm t(N-p-1; 1-0.5\alpha) SE\hat{Y}$.

The variance of the individual prediction is the variance of the mean prediction plus the variance of Y, for a given X (Groebner & Shannon, 1993). Hence, the standard error of the individual prediction for a given value of X, Xo, is:

$$SEInd\hat{Y} = \sigma \sqrt{1 + \frac{1}{N} + \frac{(Xo - \overline{X})^2}{(N-1)S_X^2}} \qquad (A1.16)$$

with the corresponding estimated value obtained by inserting S for σ. Prediction intervals for the value of Y at Xo are calculated in the standard way: $\hat{Y} \pm t(N-p-1; 1-0.5\alpha) SEInd\hat{Y}$.

Finally, note that a regression equation is determined from data which "cover" certain areas of the X-space. Suppose the point Xo lies outside the region covered by the original data. While a predicted value at the point Xo can be obtained mathematically, reliance on such a prediction is dangerous. It becomes increasingly so the further Xo lies from the original region, unless there is additional knowledge indicating that the regression equation is valid over a wider region of the X-space.

Searching For Violation Of Assumptions

Testing the goodness of fit of the regression model includes detecting possible violations of the assumptions relating to the data being analysed (SPSS, 1993). The most common way to look for evidence that the necessary assumptions are violated is a search focused on residuals (Everitt & Dunn, 1991). As noted earlier, a residual is the difference between an observed value and the value predicted by the regression equation. It represents the amount which the regression equation has not been able to explain. In performing the regression analysis, certain assumptions about the errors have been made. Thus if the fitted model is correct, the residuals should exhibit tendencies confirming the assumptions made for the errors. At the very least, they should not negate these assumptions.

There are several different types of residuals (Gunst & Mason, 1980), although each is a function of the difference between the observed and predicted values. For example, the standardised residuals, Z_{Resid}, are the values of the residuals divided by the sample standard deviation of the residuals. They have a mean of zero and a standard deviation of 1. These residuals are used in this study for plotting purposes.

The behaviour of the residuals can be analysed numerically and/or graphically. Certain statistics provide a numerical check on some of the assumptions (see, for example, Anscombe, 1961, Anscombe & Tukey, 1963, Gunst & Mason, 1980, and Draper & Smith, 1981). However, this study generally adopts the graphical approach as it is good at revealing violations of the assumptions. The lack of emphasis on numerical tests is deliberate, since a detailed examination of the plots of the residuals is usually more informative. Plots reveal violations of the assumptions serious enough to require corrective action (Draper & Smith, 1981).

A plot of the residuals against the predicted values and against the values of the independent variable can be used to check the assumptions of homoscedasticity and the linear relationship between X and Y (SPSS, 1993). The reason the residuals are plotted against the predicted values, $\hat{Y}_i$, and not the measured values, Y_i, is because the residuals and the measured values are usually correlated. The residuals and the predicted values are not correlated, for prediction equations with intercepts (Gunst & Mason, 1980; Draper & Smith, 1981). A satisfactory residuals plot should give the overall impression of a horizontal "band" equally spread about the horizontal line through the zero value of the residuals. Systematic patterns in these plots indicate a changing variance or inadequate model.

In detecting homoscedasticity, it is also helpful to plot squared residuals versus $\hat{Y}_i$ and X_i. The advantage of plotting the squared residuals instead of the residuals is that such a plot can accentuate some types of trends existing between the residuals and $\hat{Y}_i$ or X_i (Gunst & Mason, 1980).

The normality of the residuals can be checked by superimposing the probability density function (PDF) of a Normal distribution on the histogram of the residuals. A major problem of using PDF (or cumulative distribution function, CDF) plots to judge the correctness of a specific distribution is the curvature of these functions. It is easier to assess departures from a straight line. In probability plots of the data, the points lie on a straight line if the hypothesised distribution is the actual underlying distribution (D'Agostino & Stephens, 1986). There are several possible probability plots. In this study, the P-P (probability-probability) and the Q-Q (quantile-quantile) plots are used to compare the observed distribution of residuals with the expected distribution under the assumption of normality.

A P-P plot is a graph of the fitted cumulative distribution function, F_X^F, versus the input data cumulative distribution function, F_X. The basis of this type of plot is illustrated in Figure A1.7 and examples are shown in Appendices A2 and A5. If the two cumulative distribution functions are close together, then the P-P plot will be approximately linear with an intercept of 0 and a slope of 1. The P-P plot amplifies differences between the two distributions in the middle values (Law & Kelton, 1991).
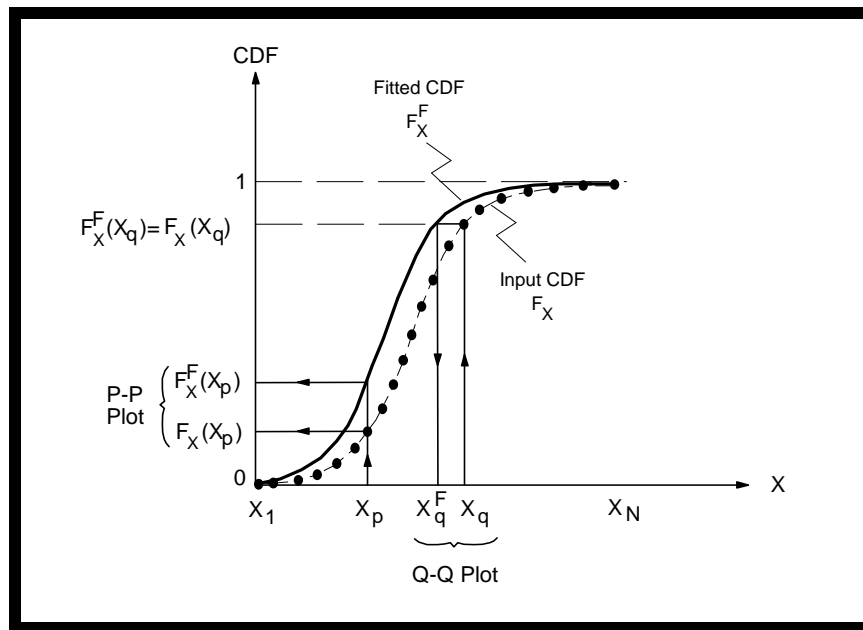
Figure A1.7: Definition of P-P and Q-Q plots (modified after Law & Kelton, 1991).

A Q-Q plot is a graph of the X_q^F quantile of a fitted cumulative distribution function versus the X_q quantile of the input data cumulative distribution function. The basis of this type of graph is also illustrated in Figure A1.7 and examples are shown in Appendices A2 and A5. If the hypothesised distribution is the actual distribution of the input data and if the sample size is large, then the Q-Q plot will also be approximately linear. The Q-Q plot amplifies differences in the tails between the two distributions (Law & Kelton, 1991).

The linearity, or lack of linearity, in a P-P or Q-Q graph, can be used as a basis for determining whether the input data could reasonably have been drawn from the hypothesised distribution. The variance of the points in the tails of the P-P or Q-Q plots is generally larger than that of the points at the centre of the distribution. Thus the relative linearity of the plots near the tails of the distributions will often seem poorer than at the centre, even if the "correct" model has been chosen (Hahn & Shapiro, 1967; Gunst & Mason, 1980). As the plotted points are ordered, they are not independent, and so they are not randomly scattered about the straight line. Therefore, even if the chosen model is appropriate, the plots may consist of a series of successive points above and below the straight line (Hahn & Shapiro, 1967; D'Agostino & Stephens, 1986). Finally, it is unreasonable to expect the observed

residuals to be exactly Normal; some deviation is expected because of sampling variation (SPSS, 1993). Even if the errors are normally distributed in the population, sample residuals are only approximately Normal.

Residuals are also used to detect outliers in the Y direction. An outlier in the Y direction among residuals is one that is far greater than the rest in absolute value and lies some standard deviations from the mean of the residuals (Draper & Smith, 1981). In this study, data with absolute standardised residual values greater than 2.5 are considered outliers (Rousseeuw & Leroy, 1987; Hadi, 1992; Gilbert & Antille, 1992) although other limits, such as 3, have also been suggested (SPSS, 1993). Outliers can be spotted readily on residual plots: they are not typical of the rest of the data. The presence of these outliers can bias the estimates of the regression parameters and make the resulting analysis less useful (Daniel & Wood, 1980; D'Agostino & Stephens, 1986; Freund & Littel, 1991).

Residuals can also be used to detect points that have unusual values for the independent variable and which can have a substantial impact on the results of the analysis (SPSS, 1993). These points are known as leverage points, outliers in the X direction, or influential points.

Basically, there are two ways of dealing with outliers (Rousseeuw & Leroy, 1987). The first is to use regression diagnostics in which certain statistics are computed from the data to detect deviating points, after which the outliers are deleted or corrected. An LS analysis is carried out on the remaining data. Rules exist for rejecting outliers (D'Agostino & Stephens, 1986). However, automatic rejection is not always wise, because the outliers might provide information which other data points do not. They might arise from an unusual combination of circumstances which are of vital interest and require further investigation rather than rejection. As a general rule, outliers should be rejected only if they can be traced to causes such as recording errors or mistakes in setting up the apparatus (Anscombe, 1960). When there is only a single outlier, some of these methods work quite well by looking at the effect of deleting one point at a time. Unfortunately, it is much more difficult to diagnose outliers when there are several of them (Portnoy, 1987).

Using LS residuals is not a good way of identifying real outliers (Rousseeuw & Leroy, 1987): the LS fit is pulled considerably in the direction of the deviating points and so the real outliers may possess quite small LS residuals. The alternative approach is to use robust regression, i.e. data-fitting procedures which are less sensitive than LS to typical departures from the idealised assumptions for which the estimators are optimised, and where possible outliers among the data points are identified and given less weight (Madsen & Nielsen, 1989). Hence, when LS assumptions are not met and/or when outliers are present, a more robust technique should be applied to confirm or refute LS results. There are several robust regression techniques (Rousseeuw & Leroy, 1987).

A1.2   Least Absolute Deviations Method (LAD)

A1.2.1 Basic LAD Principle

To define more robust regression alternatives, statisticians have exploited the fact that the median is a better measure of central tendency than the mean if the variable can take very large or very small values. Extreme values can greatly affect the mean even if they are very unlikely to occur; this is not the case with the median (Law & Kelton, 1991).

Suppose that errors $\varepsilon_i, i = 1,...,N$ are independent and follow the Double Exponential (or Laplace) distribution:

$$f_{\varepsilon_i} = \frac{1}{2\sigma} \exp\left[-\frac{|\varepsilon_i|}{\sigma}\right] \quad \text{for} \quad -\infty \leq \varepsilon_i \leq +\infty \qquad (A1.17)$$

rather than the Normal distribution:

$$f_{\varepsilon_i} = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{\varepsilon_i^2}{2\sigma^2}\right] \quad \text{for} \quad -\infty \leq \varepsilon_i \leq +\infty \qquad (A1.18)$$

as is assumed for the LS method. The Double Exponential probability density function has a pointed peak of height $1/2\sigma$ at $\varepsilon_i = 0$ and tails off to zero as ε_i goes to both plus and minus infinity. Then, application of the maximum likelihood principle for estimating the parameters β_0 and β_1, assuming σ fixed, involves minimisation of the sum of absolute errors, $\sum |\varepsilon_i|$, i=1,...,N,

and not the minimisation of the sum of squares of errors, $\sum \varepsilon_i^2$, i=1,...,N (Draper & Smith, 1981). Since the errors are not squared, the influence of the outliers in the estimators is less.

The method of robust regression which minimises the sum of the absolute deviations is known as the least absolute deviations method or the ℓ_1-norm. The LAD method does not rely upon the Normal error assumption: it assumes a Double Exponential distribution of the errors which has thicker tails than the Normal distribution. Hence, the LAD method provides maximum likelihood estimates when the errors follow a Double Exponential distribution with zero mean and with mean absolute deviation σ. This method can be seen as a generalisation of the concept of the median, because minimisation of the sum of the absolute errors defines the median of N observations of Y (Rousseeuw & Leroy, 1987).

Figure A1.8 illustrates the different results obtained from using the LS and LAD methods. Figure A1.8(a) is a plot of five points which lie almost on a straight line: the LS fit and the LAD fit are essentially the same. Figure A1.8(b) displays a situation where, for some reason, point 4 has been wrongly moved from its original position (indicated by the dashed circle). This point is an outlier in the Y direction and it has a rather strong influence on the LS line, which is quite different from the line in Figure A1.8(a). Figure A1.8(c) shows the robustness of the LAD fit with respect to such an outlier: the line remains approximately where it was when observation 4 was correct.

The LAD method is recommended when outliers exist in the Y direction but it does not accommodate outliers in the X direction (Rousseeuw & Leroy, 1987). Since, in Owen's data set, outliers in the Y direction are the most important, the LAD method is used here. There is no need to use a robust technique which is computationally more costly (Rousseeuw & Leroy, 1987).

LAD is probably the oldest of all robust methods for estimating regression coefficients[1] (McKean & Schrader, 1987). However, two major difficulties have impeded its general adoption (Dodge, 1987). These are computational difficulties due, firstly, to the lack of closed form formulae for estimating the

---

[1]   A brief review and comprehensive bibliography of LAD estimation until 1977 is given in Gentle (1977).

parameters and, secondly, to the lack of accompanying statistical inference procedures (e.g. for testing the general linear hypothesis, for obtaining confidence intervals and for constructing analysis of variance tables). Fortunately, interest has increased recently in developing statistical methods based on LAD rather than on LS (McKean & Schrader, 1987; Gentle et al, 1987; Sposito, 1990; Armstrong & Beck, 1990; Bai et al, 1990; Ataa, 1994). The motivation for this interest has arisen largely from the recognition that many real data sets do not conform to the Normal error assumption, but also from the conceptual simplicity of the LAD estimates and their increasingly competitive computational cost (Ataa, 1994).
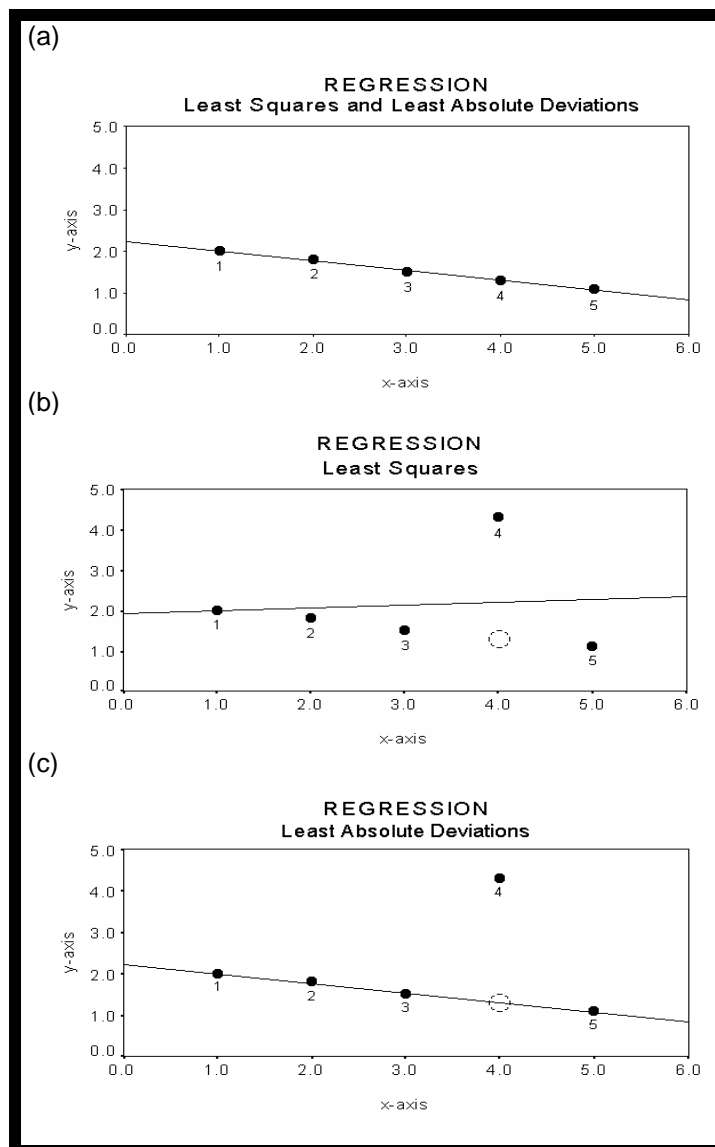
Figure A1.8: LS and LAD regression lines (modified after Rousseeuw & Leroy, 1987).

To infer accurately the values of β_0 and β_1 from the LAD values of b_0 and b_1, respectively, the six assumptions mentioned in Section A1.1.1 have to be valid. For further details, see McKean & Schrader (1987) and Koenker (1987).

A1.2.2 Goodness Of Fit

Summary Statistics For The Regression Line

The measure of the goodness of fit of the LAD regression line is the sample standard error of the estimate, S, defined as (McKean & Schrader, 1987):

$$S = \frac{2\,SAD_{Res}}{N-p-1} = \frac{2\sum_{i=1}^{N}\left|Y_i - \hat{Y}_i\right|}{N-p-1} \tag{A1.19}$$

where SAD_{Res} is the sum of the absolute deviations of the residuals (see Table A1.2). Since the population variance of the residuals, σ^2, is generally not known, S^2 is its usual estimate.

Analysis Of Variance - ANOLAD Table

The statistical methods of hypothesis testing associated with the LAD method have been discussed recently by a number of authors (McKean & Shrader, 1987; Bai et al, 1990) and the suggested tests are similar to those for the LS method. For testing a general linear hypothesis, the LS minimisation in sums of squared residuals is replaced by minimisation in sums of absolute deviations. By using such a direct analogy, an analysis of variance table based on the LAD method (an ANOLAD table) can be constructed (Table A1.2). It summarises the tests of the hypothesis in a form similar to an ANOVA table. The ANOLAD table shows whether predicting Y based on X using the LAD regression line is preferable to just using the overall median of Y, Y_{med}. Note that in the LAD method, the median is used as the centre of location instead of the mean used in the LS method.

Source of Variation	Degrees of Freedom, DF	Sum of Absolute Deviations	Mean Absolute Deviations	χ^2
Regression	p	$SAD_{Reg} = \sum_{i=1}^{N} \left| \hat{Y}_i - Y_{med} \right|$	$MAD_{Reg} = \dfrac{SAD_{Reg}}{p}$	$\dfrac{2\,MAD_{Reg}}{S}$
Residual	N-p-1	$SAD_{Res} = \sum_{i=1}^{N} \left| Y_i - \hat{Y}_i \right|$	$MAD_{Res} = \dfrac{S}{2} = \dfrac{SAD_{Res}}{N-p-1}$	————

Table A1.2: ANOLAD table (modified after McKean & Schrader, 1987).

If the assumptions are met, the ratio of the mean absolute regression to the mean absolute residual is distributed as a χ^2 statistic with N-p-1 degrees of freedom (McKean & Schrader, 1987; Koenker, 1987; Ataa, 1994). χ^2 is a statistic used to test the null hypothesis Ho: *the regression model does not explain any of the total variation in the dependent variable*. For a single independent variable, it tests the null hypothesis Ho: $\beta_1 = 0$. If $\chi^2 = 2MAD_{Reg} / S > \chi^2_{Crit} = \chi^2 (N\text{-}p\text{-}1; \alpha)$, the null hypothesis can be rejected at the α level of significance on the basis of the data used and it can be inferred that the LAD equation appears to be a good predictor, running a risk of less than α of being wrong (Figure A1.9).
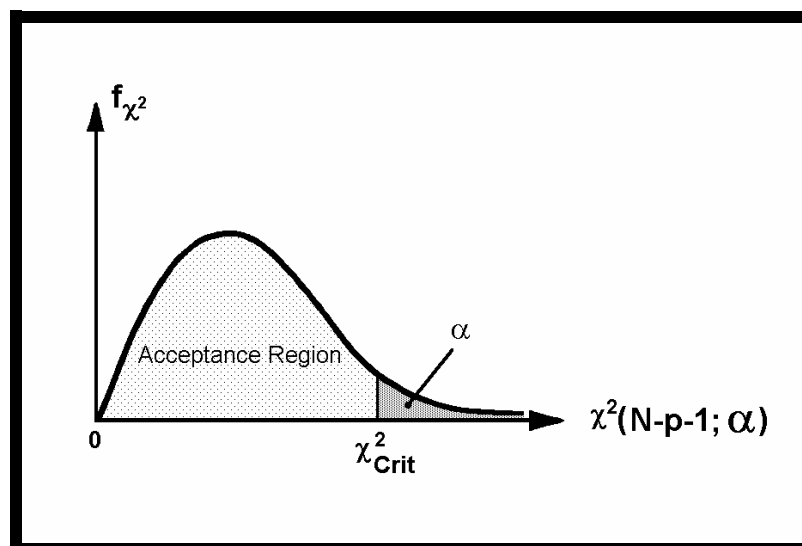
Figure A1.9: The χ^2 distribution used as a significance test for LAD regression (modified after Bendat & Piersol, 1971).

Statistics Of The Parameters In The Regression Line

Under the above assumptions, Bassett & Koenker (1978) showed that the distributions of b_0 and b_1 are approximately Normal, with means of β_0 and β_1 and with standard deviations (or standard errors of the estimates) as defined in eq. (A1.11), where σ is estimated by S calculated from eq. (A1.19).

Another way of testing if the intercept and/or the slope of the regression line are zero, i.e. Ho: $\beta_0 = 0$ and/or Ho: $\beta_1 = 0$, is to use the following statistic:

$$z = \frac{b_0}{SEb_0} \quad \text{or} \quad z = \frac{b_1}{SEb_1} \tag{A1.20}$$

The distribution of the statistic, when the assumptions are met, is standard Normal. So if $\left| z \right| > z_{Crit} = z(\alpha / 2)$, then the parameter estimate is significantly non-zero at the α level (Figure A1.10): that is, Ho can be rejected at the α level of significance.
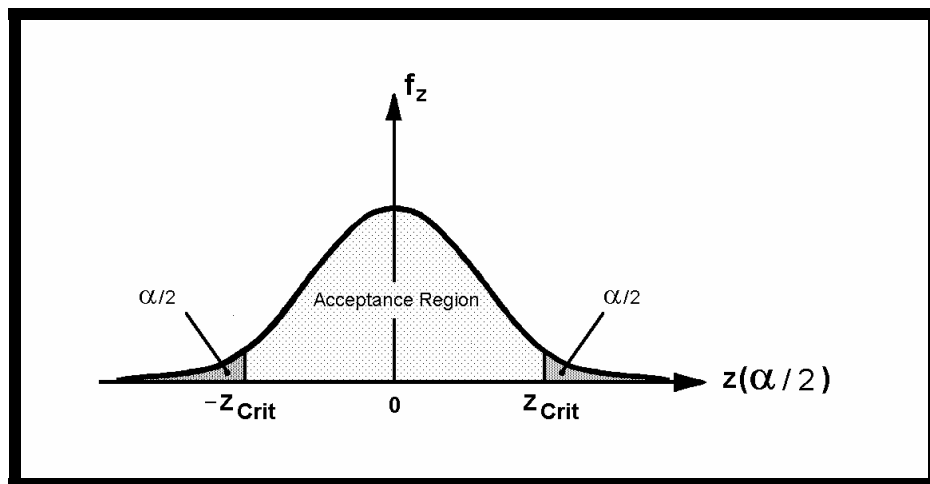
Figure A1.10: The standard Normal distribution used as a significance test for LAD regression (modified after Bendat & Piersol, 1971).

Based on the LAD estimates of b_0 and b_1, and on the underlying assumptions, a $(1-\alpha)\%$ confidence interval similar to the LS counterparts can be calculated for each parameter. Here, SEb is calculated as for eq. (A1.20) and the t-critical values in eq. (A1.14) are replaced by z-critical values (Stangenhaus, 1987):

$$b \pm z(\alpha / 2) \, SEb \qquad\qquad (A1.21)$$

where $z(\alpha / 2)$ is the upper tailed standard Normal critical value (Figure A1.10). As for the LS method, the confidence intervals can be used to test the null hypothesis Ho: $\beta = 0$, where β can be either β_0 or β_1. If zero does not fall within the interval, the null hypothesis can be rejected.

Searching For Violation Of Assumptions

Analogous to plots of the LS residuals, the LAD residuals may also be plotted. If ε follows a Double Exponential distribution with zero mean and mean absolute deviation σ, then the probability density function of $|\varepsilon|$ follows an Exponential distribution with a mean and standard deviation equal to σ (Ataa, 1994):

$$f_{|\varepsilon|} = \frac{1}{\sigma} \exp\left[-\frac{|\varepsilon|}{\sigma}\right] \qquad\qquad (A1.22)$$

This fact can be exploited by plotting the absolute values of the residuals against a theoretical Exponential fit, allowing the suitability of the Double Exponential distribution to be assessed visually.

APPENDIX A2 - Examples Of Regression Analysis

This appendix describes how regression equations have been fitted to Owen's data. The H&R model (employing both $(R_{max})_{37\%}$ and $(R_{max})_{99\%}$ in defining the value of C) and Owen's model have been considered. Similar inferences were made in all cases. Here, an illustrative example is given for each model using data for the embankment slope of 1:2.

The H&R and Owen models are represented by eqs. (3.16) and (3.6), respectively (Chapter 3). These equations contain the parameters A and B to be estimated by regression. They can be expressed, by suitable transformation, in the form of the standard linear regression model (see eq. (A1.3) in Appendix A1):

$$H\&R \text{ Model:} \quad Ln(Q_*) = Ln\,A + B\,Ln(1 - R_*)$$

$$\text{Owen's Model:} \quad Ln(Q_*) = Ln\,A - BR_* \tag{A2.1}$$

For both models, $Ln(Q_*)$ is the dependent variable, $Ln\,A$ is the intercept and B the slope; for the H&R model, $Ln(1 - R_*)$ is the independent variable and for Owen's model, R_* is the independent variable. Note that R_* and Q_* are defined differently in the two models (see Chapter 3).

In this study, regression analysis has started by applying the least-squares (LS) method (see Appendix A1). The following results have been obtained using the statistical software package SPSS for Windows (SPSS, 1993).

A2.1   LS Regression

A2.1.1 H&R Model

From the scatterplot in Figure A2.1, the relationship between $Ln(Q_*)$ and $Ln(1 - R_*)$ appears to be approximately linear and the values of R^2 and R_a^2 in Table A2.1 suggest that no relevant independent variable has been excluded: these values reveal that $Ln(1 - R_*)$ accounts for about 81% of the variation in $Ln(Q_*)$, which can be considered satisfactory. This is reflected in the ANOVA results (Table A2.1) where the mean square regression is much bigger than the mean square residual.
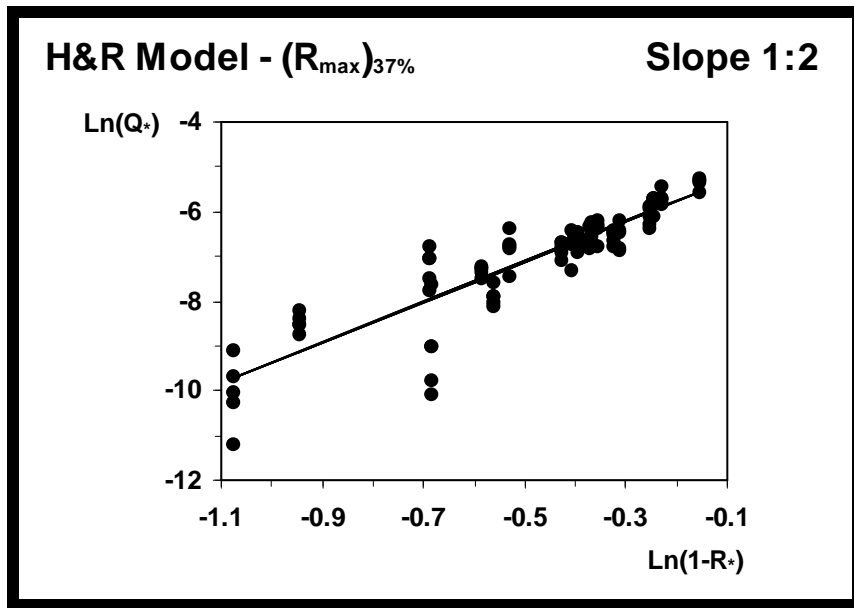
Figure A2.1: Scatterplot and LS regression line for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

SUMMARY STATISTICS		
Coefficient of Determination, R^2	**Adjusted Coefficient of Determination, R_a^2**	**Standard Error of the Estimate, S**
0.811	0.809	0.523

ANOVA				
Source of Variation	**Degrees of Freedom, DF**	**Sum of Squares**	**Mean Square**	**F**
Regression	1	109.239	109.239	399.588
Residual	93	25.424	0.273	----------

STATISTICS OF LS PARAMETERS						
	Parameter, b	**Estimated Standard Error of b, SEb**	**Confidence Interval of b**			**t**
			90%	**95%**	**99%**	
b_0	-4.841	0.120	-5.041 -4.641	-5.079 -4.603	-5.157 -4.525	-40.466
b_1	4.548	0.228	4.168 4.928	4.095 5.000	3.947 5.149	19.990

Table A2.1: Summary of LS statistics for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

Since F>F_{Crit} for α equal to 0.01, 0.05 and 0.1 (as tabulated in statistical books) and $|t| > t_{Crit}$ for b_1 for any value of α between 0.001 and 0.9 (as tabulated in statistical books), the slope estimate b_1=4.548 differs significantly from zero. Equivalently, the 90, 95 and 99% confidence intervals in Table A2.1 do not include zero. The same conclusions apply to the parameter b_0.

The above results, although pleasing, should not be accepted too readily, because the applicability of the LS method has not yet been assessed. This analysis should be an indispensable part of regression, even when R^2 (or R_a^2) is large and the statistical tests are significant (Rousseeuw & Leroy, 1987).

Figure A2.2 appears to show a tendency for the variation (or spread) in the standardised residuals to decrease as the predicted values of $Ln(Q_*)$ and $Ln(1-R_*)$ increase. The assumption of homoscedasticity is unlikely to be valid. Although unbiased, the estimated parameters would not have minimum variance and the confidence intervals and significance tests obtained from regression are not reliable (see Appendix A1).

Figure A.2.3, a plot of the squared standardised residuals versus the predicted values of the dependent variable and versus the independent variable, accentuates the fact that there does not appear to be a constancy of the error variances. Yet what is most evident from this figure is that three of the standardised residuals look suspiciously large in magnitude. These points are clearly outliers (see Table 2.2).
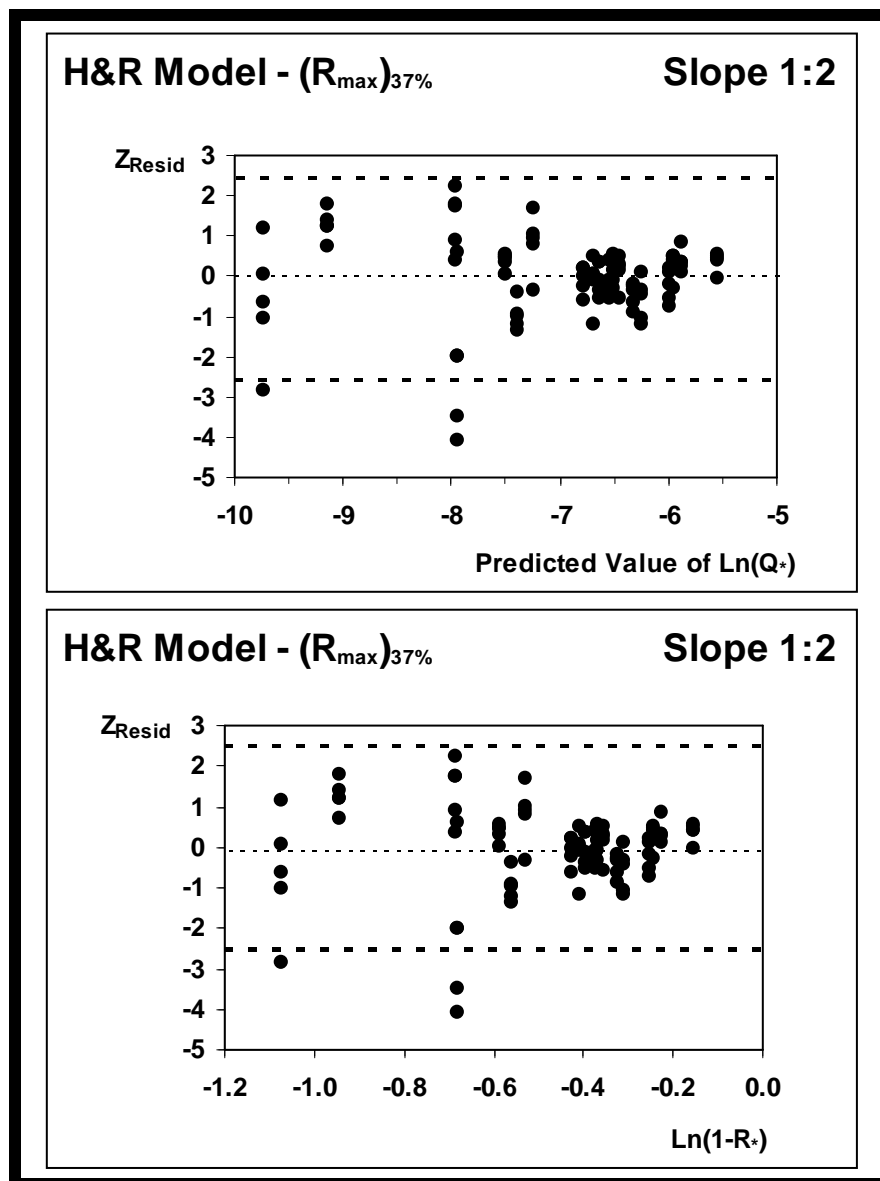
Figure A2.2: Scatterplots of the standardised residuals against the predicted values of $Ln(Q_*)$ and against $Ln(1-R_*)$: LS regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
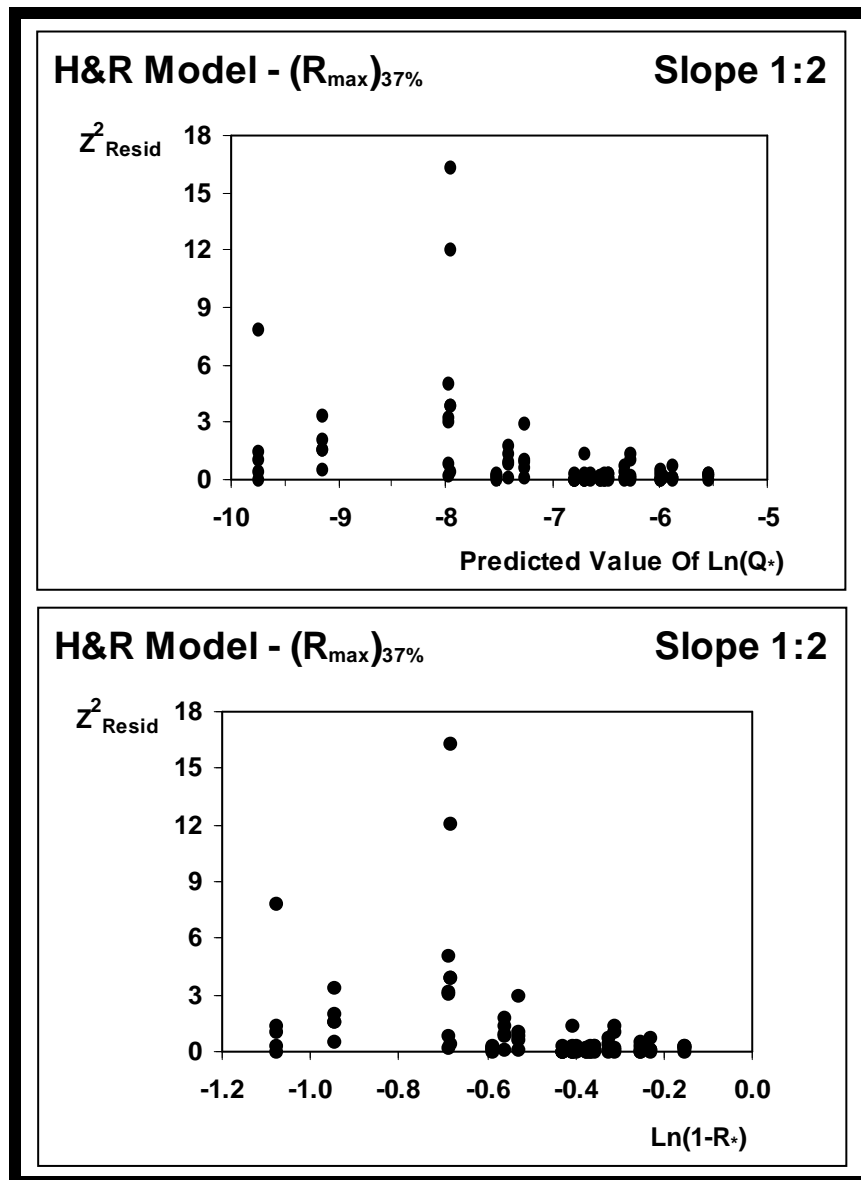
Figure A2.3: Scatterplots of the squared standardised residuals against the predicted values of $\text{Ln}(Q_*)$ and against $\text{Ln}(1-R_*)$: LS regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

Independent Variable	Dependent Variable	Predicted Dependent Variable	Residuals	Standardised Residuals
-0.683	-10.061	-7.949	-2.113	-4.041
-0.683	-9.765	-7.949	-1.816	-3.473
-1.075	-11.200	-9.734	-1.466	-2.804
-0.687	-6.790	-7.966	1.176	2.249

Table A2.2: Details of the four highest residuals from LS regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

From Figures A2.4 to A2.6, it seems that the Normal distribution fits the residuals poorly; the residuals appear to follow a distribution with thicker tails than the Normal distribution. The non-normality has been confirmed by the results of three goodness-of-fit tests available in the software package BestFit (see Appendices A4 and A5 for more details of BestFit and goodness-of-fit tests, respectively) which all rejected the hypothesis of the residuals being Normal distributed at any α level of significance considered (these results are not presented here, but similar analyses are shown in Appendix A5). Since the assumption of normality seems to be violated, the confidence intervals and results of significance tests are unreliable. Even if the parameters are still unbiased, the violation of assumptions can be important (Draper & Smith, 1981).
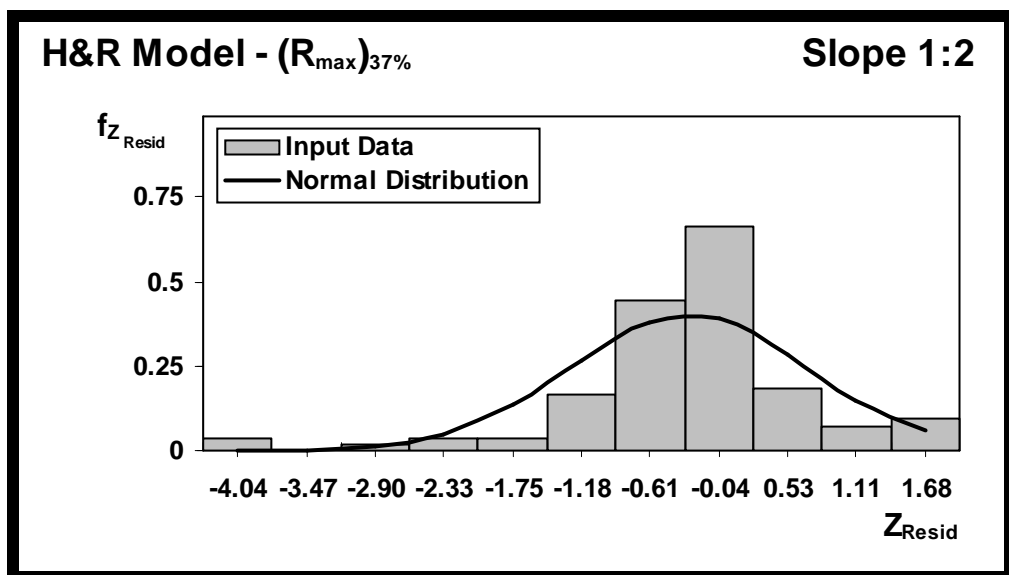
Figure A2.4: Comparison of the Normal probability density function for standardised residuals with data from LS regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
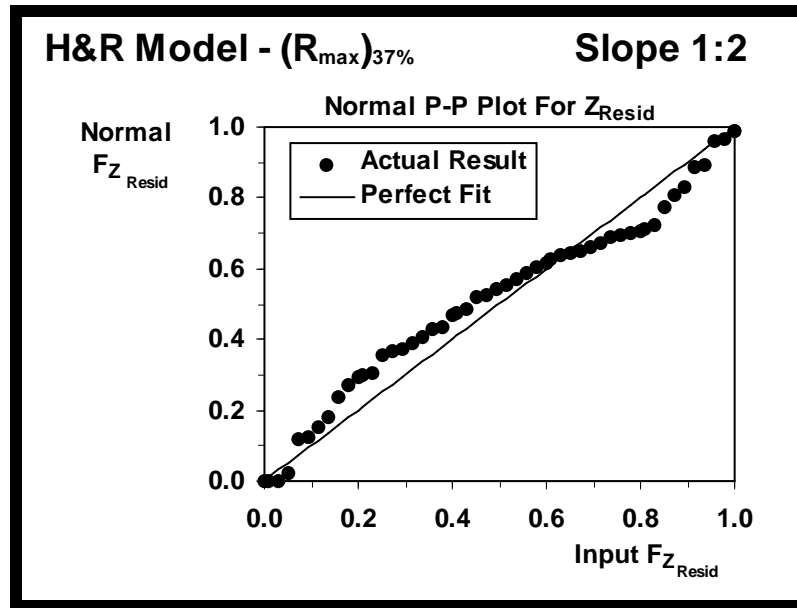
Figure A2.5: Comparison of the Normal cumulative distribution function for standardised residuals with data from LS regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
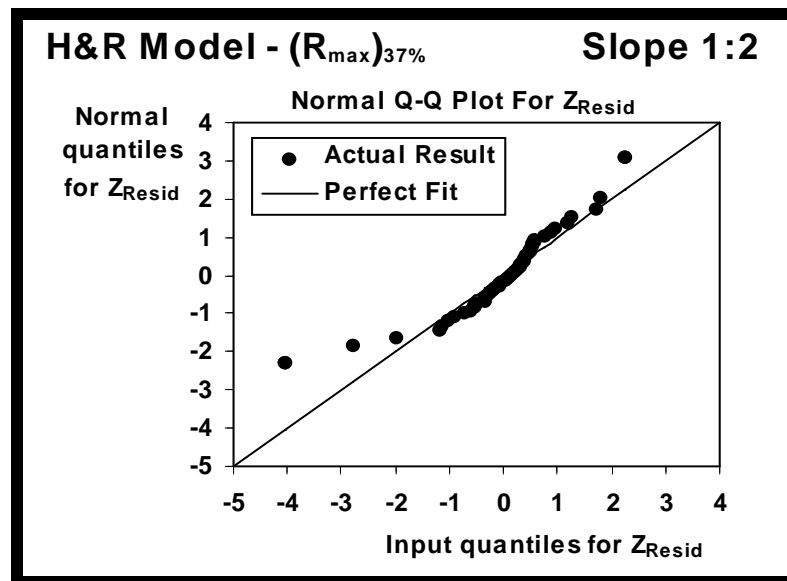
Figure A2.6: Comparison of the Normal quantiles for standardised residuals with data from LS regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

A2.1.2 Owen's Model

Since the results obtained for the Owen model are very similar to those for the H&R model, they are not discussed here. Nevertheless, the corresponding figures are presented for completeness.
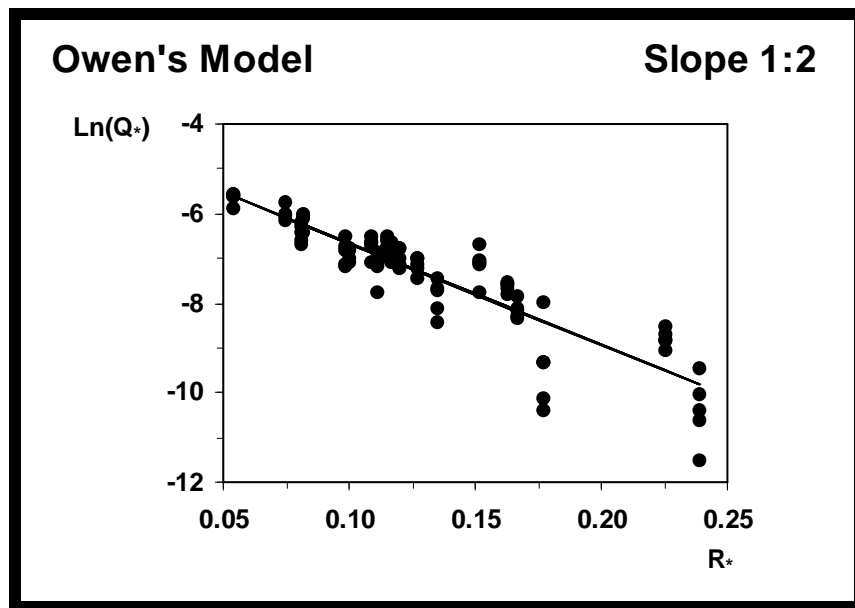
Figure A2.7: Scatterplot and LS regression line for Owen's model, slope 1:2.

SUMMARY STATISTICS		
Coefficient of Determination, R^2	**Adjusted Coefficient of Determination, R_a^2**	**Standard Error of the Estimate, S**
0.814	0.812	0.522

ANOVA				
Source of Variation	**Degrees of Freedom, DF**	**Sum of Squares**	**Mean Square**	**F**
Regression	1	111.198	111.198	407.613
Residual	93	25.371	0.273	----------

STATISTICS OF LS PARAMETERS						
	Parameter, b	**Estimated Standard Error of b, SEb**	**Confidence Interval of b**			**t**
			90%	**95%**	**99%**	
b_0	-4.385	0.155	-4.643 -4.126	-4.693 -4.077	-4.793 -3.976	-28.325
b_1	-22.798	1.129	-24.685 -20.912	-25.046 -20.551	-25.780 -19.817	-20.189

Table A2.3: Summary of LS statistics for Owen's model, slope 1:2.
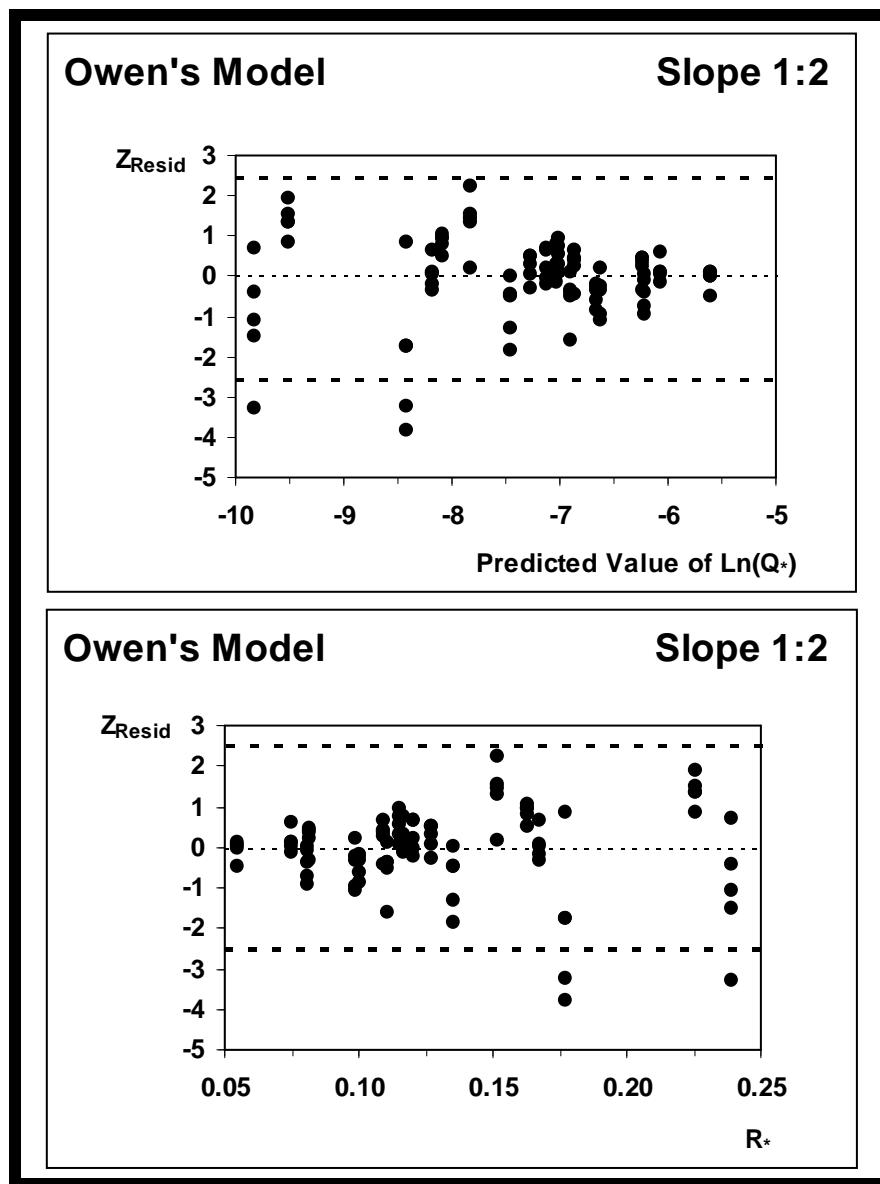
Figure A2.8: Scatterplots of the standardised residuals against the predicted values of $Ln(Q_*)$ and against R_* : LS regression for Owen's model, slope 1:2.
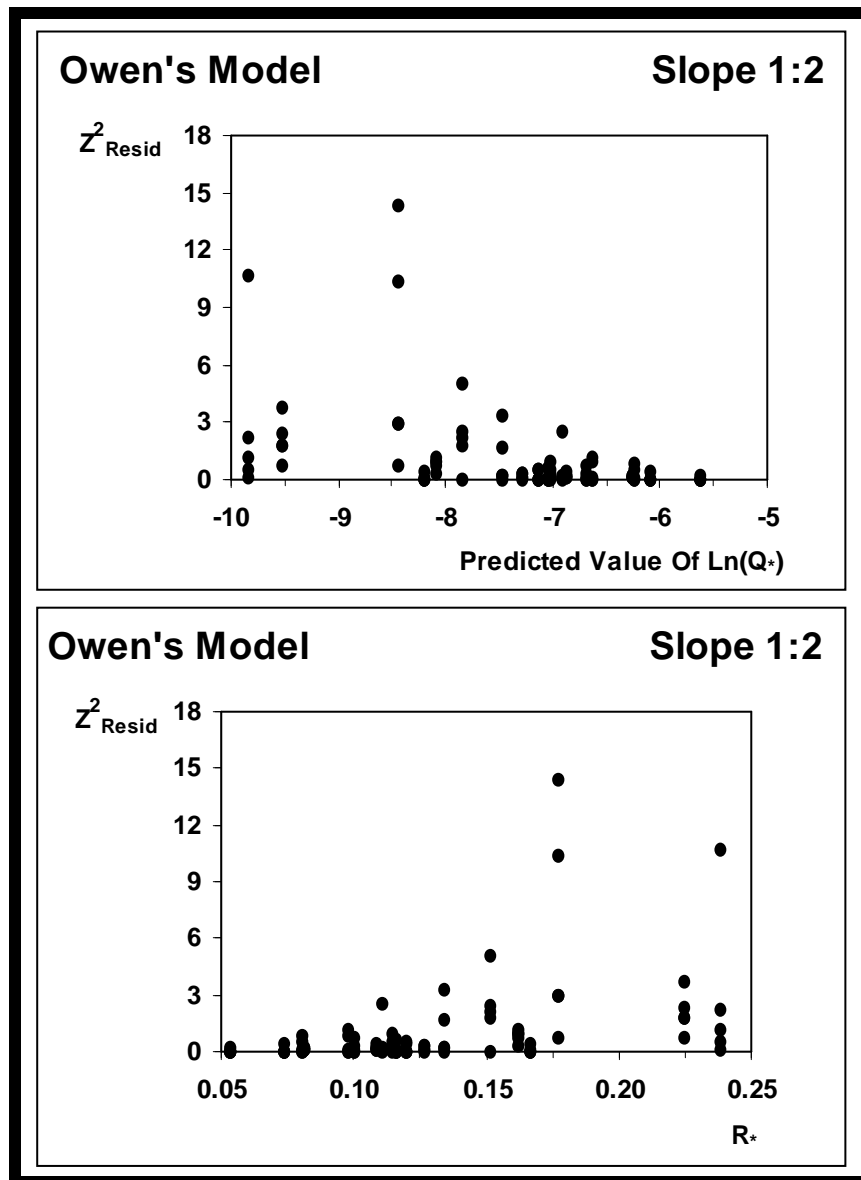
Figure A2.9: Scatterplots of the squared standardised residuals against the predicted values of Ln(Q*) and against R* : LS regression for Owen's model, slope 1:2.

Independent Variable	Dependent Variable	Predicted Dependent Variable	Residuals	Standardised Residuals
0.177	-10.404	-8.425	-1.979	-3.790
0.239	-11.530	-9.824	-1.706	-3.267
0.177	-10.107	-8.425	-1.682	-3.221
0.152	-6.666	-7.839	1.173	2.247

Table A2.4: Details of the four highest residuals from LS regression for Owen's model, slope 1:2.
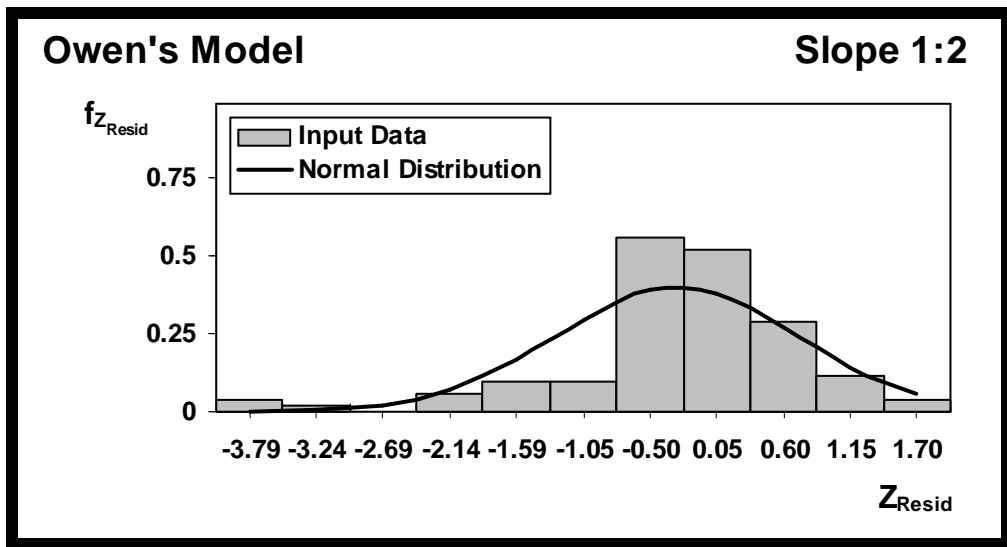
Figure A2.10: Comparison of the Normal probability density function for standardised residuals with data from LS regression for Owen's model, slope 1:2.
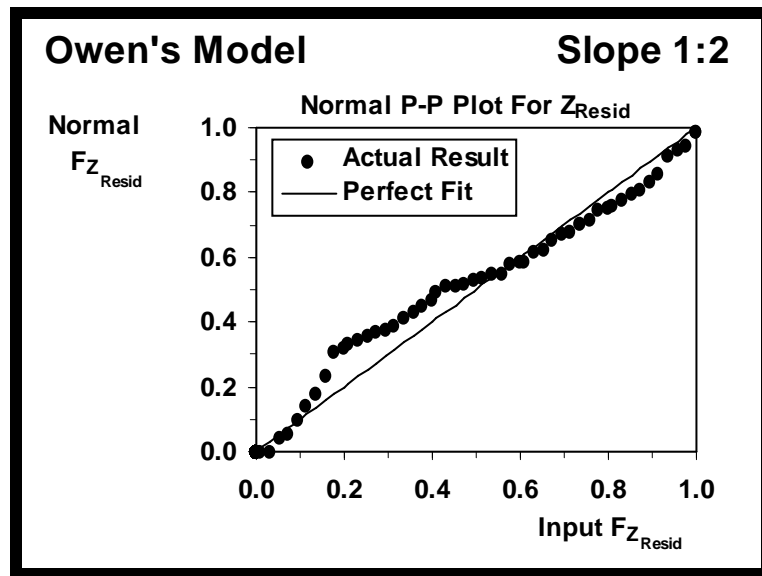
Figure A2.11: Comparison of the Normal cumulative distribution function for standardised residuals with data from LS regression for Owen's model, slope 1:2.
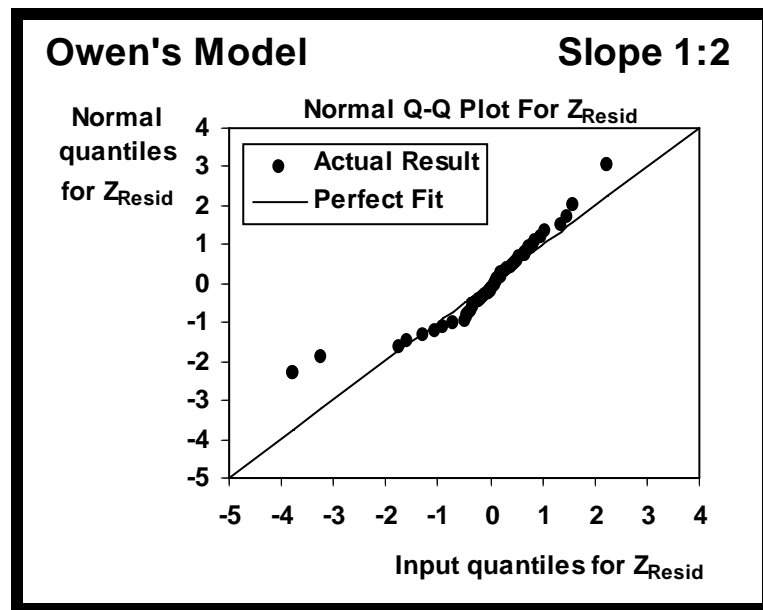
Figure A2.12: Comparison of the Normal quantiles for standardised residuals with data from LS regression for Owen's model, slope 1:2.

A2.2   LAD Regression

From the above LS regression results, the H&R and Owen models appear to perform equally well. This is not surprising. The main differences between the models are expected to exist for very small discharges (see Chapter 3) which are not part of Owen's measured conditions.

The above departures from the idealised LS assumptions and the occurrence of several outliers suggest that it might be appropriate to apply robust regression. In this study, the least absolute deviations (LAD) method has been adopted (see Appendix A1). It is more likely than the LS method to accommodate outlying data points and to provide reliable estimates of the parameters. The results for the illustrative example given here have been obtained using the statistical software package SAS for Windows (Freund & Littell, 1991) and using the program presented in Appendix A3 written in SAS/IML language (SAS Institute Inc, 1988).

A2.2.1 H&R Model
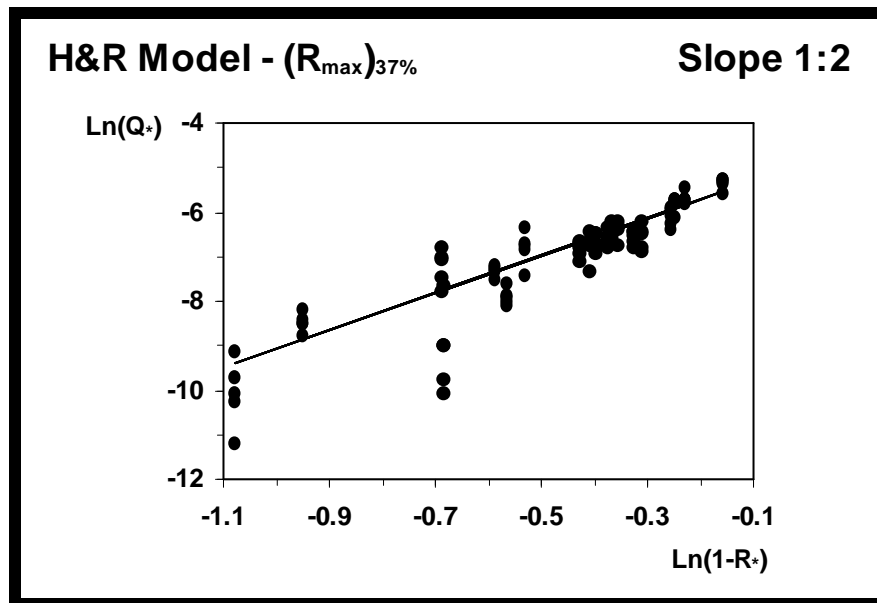
Figure A2.13: Scatterplot and LAD regression line for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

Standard Error of the Estimate, S
0.726

ANOLAD				
Source of Variation	**Degrees of Freedom, DF**	**Sum of Absolute Deviations**	**Mean Absolute Deviations**	χ^2
Regression	1	68.623	68.623	189.109
Residual	93	33.747	0.363	----------

STATISTICS OF LAD PARAMETERS						
	Parameter, b	**Estimated Standard Error of b, SEb**	**Confidence Interval of b**			**z**
			90%	**95%**	**99%**	
b_0	-4.889	0.166	-5.162 -4.616	-5.214 -4.563	-5.316 -4.461	-29.440
b_1	4.174	0.316	3.654 4.693	3.555 4.793	3.361 4.987	13.217

Table A2.5: Summary of LAD statistics for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
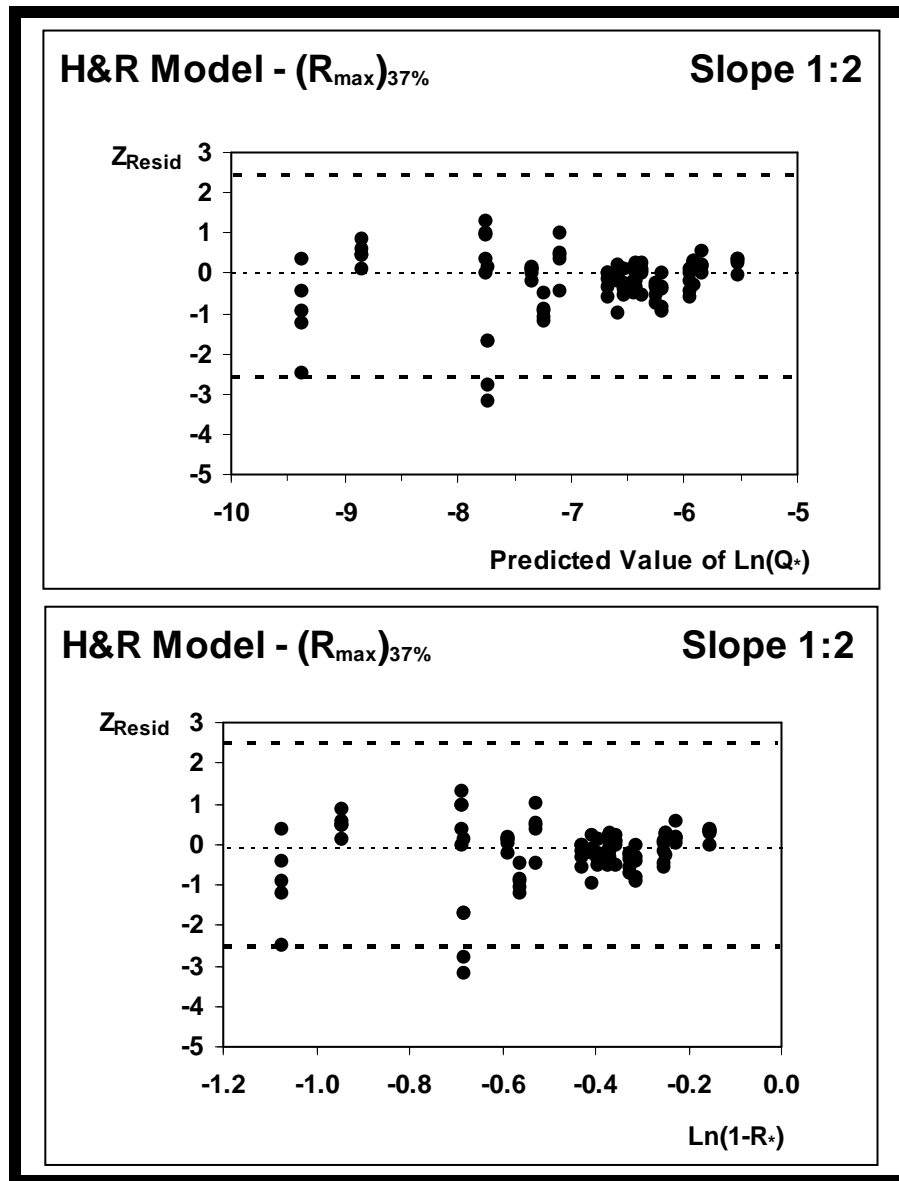
Figure A2.14: Scatterplots of the standardised residuals against the predicted values of $Ln(Q_*)$ and against $Ln(1-R_*)$: LAD regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
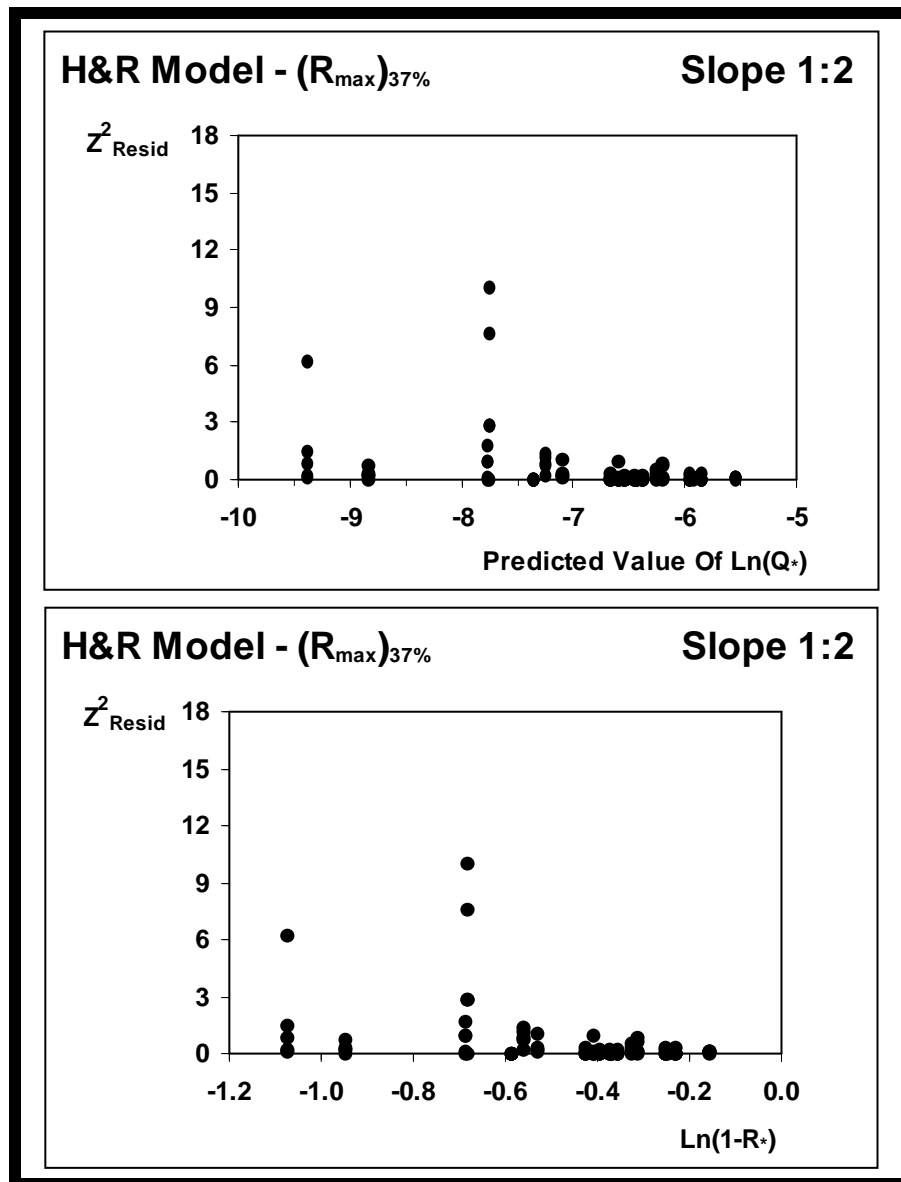
Figure A2.15: Scatterplots of the squared standardised residuals against the predicted values of $Ln(Q_*)$ and against $Ln(1-R_*)$: LAD regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

Independent Variable	Dependent Variable	Predicted Dependent Variable	Residuals	Standardised Residuals
-0.683	-10.061	-7.741	-2.321	-3.163
-0.683	-9.765	-7.741	-2.024	-2.758
-1.076	-11.200	-9.379	-1.821	-2.482
-0.683	-8.984	-7.741	-1.243	-1.694

Table A2.6: Details of the four highest residuals from LAD regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
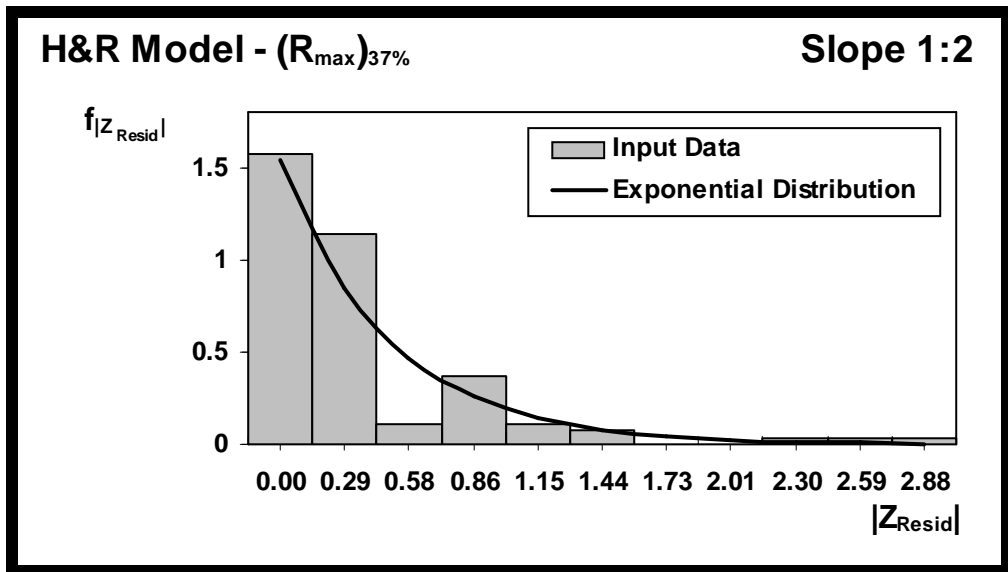
Figure A2.16: Comparison of the Exponential probability density function for absolute values of the standardised residuals with data from LAD regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
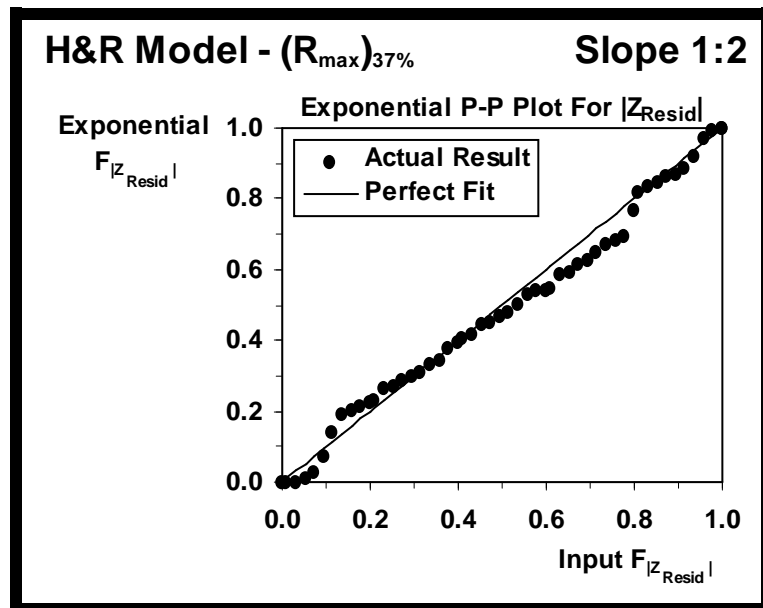
Figure A2.17: Comparison of the Exponential cumulative distribution function for absolute values of the standardised residuals with data from LAD regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.
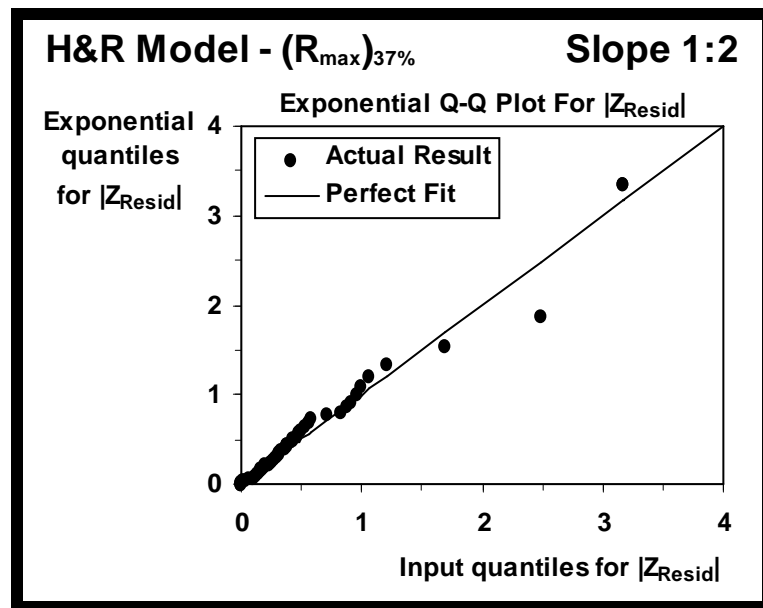
Figure A2.18: Comparison of the Exponential quantiles for absolute values of the standardised residuals with data from LAD regression for the H&R model, $(R_{max})_{37\%}$, slope 1:2.

A number of points are worth noting in relation to the H&R model:

- The LAD results support a linear relationship between $Ln(Q_*)$ and $Ln(1-R_*)$ (see Figure A2.13; $\chi^2 > \chi^2_{Crit}$ for any value of α and $SAD_{Reg} >> SAD_{Res}$ in Table A2.5).

- The facts that $\chi^2 > \chi^2_{Crit}$ for any value of α, $|z| > z_{Crit}$ for b_1 for any value of α and the 90, 95 and 99% confidence intervals for b_1 do not contain the value zero (Table A2.5), confirm that the slope estimate $b_1=4.174$ is significantly different from zero. For the same reasons, the results in Table A2.5 also confirm that $b_0 \neq 0$.

- The LAD estimates b_0 and b_1 (Table A2.5) differ from the corresponding LS estimates (Table A2.1), showing the influence of the outlying data points in the LS results.

- Comparison of Figure A2.2 with Figure A2.14 and of Figure A2.3 with Figure A2.15 shows that the LAD method accommodated extreme points much better than the LS method. Figure A2.14 also shows points being better balanced above and below the zero line. This is reflected in the smaller values of the standardised residuals shown in Table A2.6 compared with the corresponding values in Table A2.2.

- • Figures A2.16 to A2.18 indicate that the absolute errors may follow an Exponential distribution (i.e. the errors may follow a Double Exponential distribution, as hypothesised). In view of the approximately linear behaviour shown in the P-P and the Q-Q graphs, the LAD method seems suitable for fitting the H&R model to this data set. When similar graphs have been constructed as a diagnostic tool for the LS method, some systematic non-linear behaviour has clearly been identified (see Figures A2.5 and A2.6). This improvement has been confirmed by running BestFit to check the goodness of fit of the Exponential distribution.

A2.2.2 Owen's Model

The remarks made with regard to the H&R model also seem to apply to Owen's model. Indeed, the improvement obtained with the LAD method for the latter is even more evident (see Figures A2.22 to A2.24) than for the former.

As in the case of the LS analysis, the difference in performance between the H&R model and the Owen model does not seem important for the range of discharges tested, as expected.
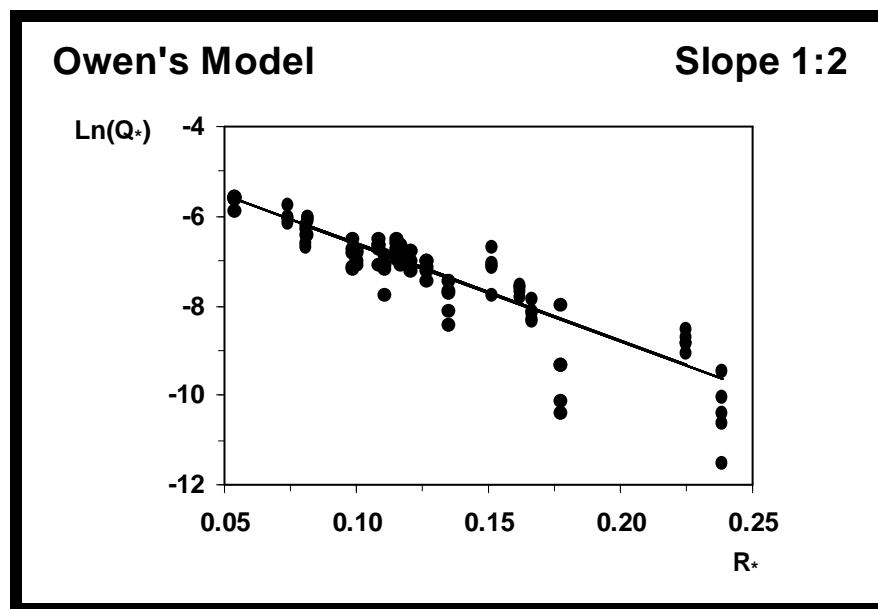
Figure A2.19: Scatterplot and LAD regression line for Owen's model, slope 1:2.

Standard Error of the Estimate, S				
0.718				
ANOLAD				
Source of Variation	**Degrees of Freedom, DF**	**Sum of Absolute Deviations**	**Mean Absolute Deviations**	χ^2
Regression	1	73.742	73.742	205.350
Residual	93	33.397	0.359	----------

STATISTICS OF LAD PARAMETERS						
	Parameter, b	**Estimated Standard Error of b, SEb**	**Confidence Interval of b**			**z**
			90%	**95%**	**99%**	
b_0	-4.445	0.213	-4.795 -4.095	-4.862 -4.028	-4.993 -3.897	-20.883
b_1	-21.706	1.553	-24.260 -19.152	-24.749 -18.663	-25.703 -17.709	-13.979

Table A2.7: Summary of LAD statistics for Owen's model, slope 1:2.
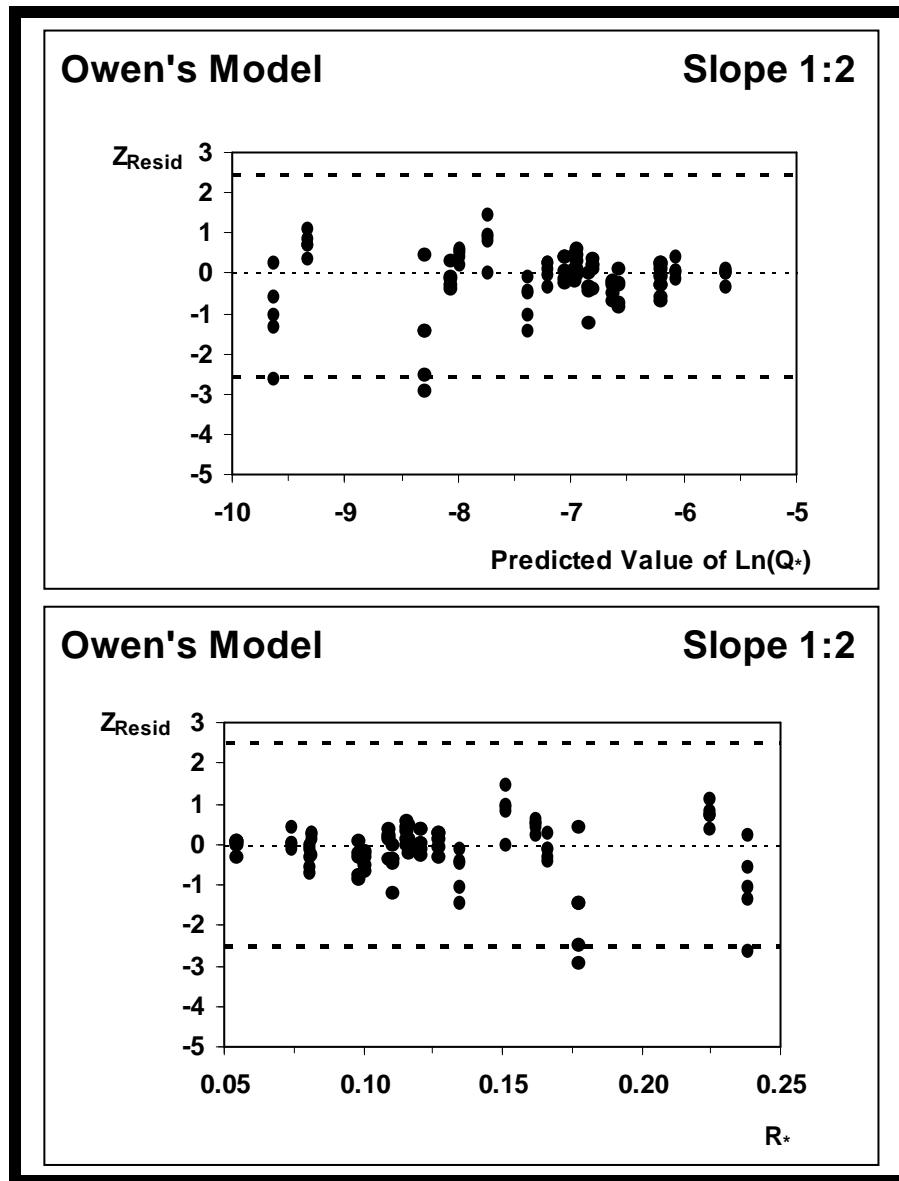
Figure A2.20: Scatterplots of the standardised residuals against the predicted values of $Ln(Q_*)$ and against R_* : LAD regression for Owen's model, slope 1:2.
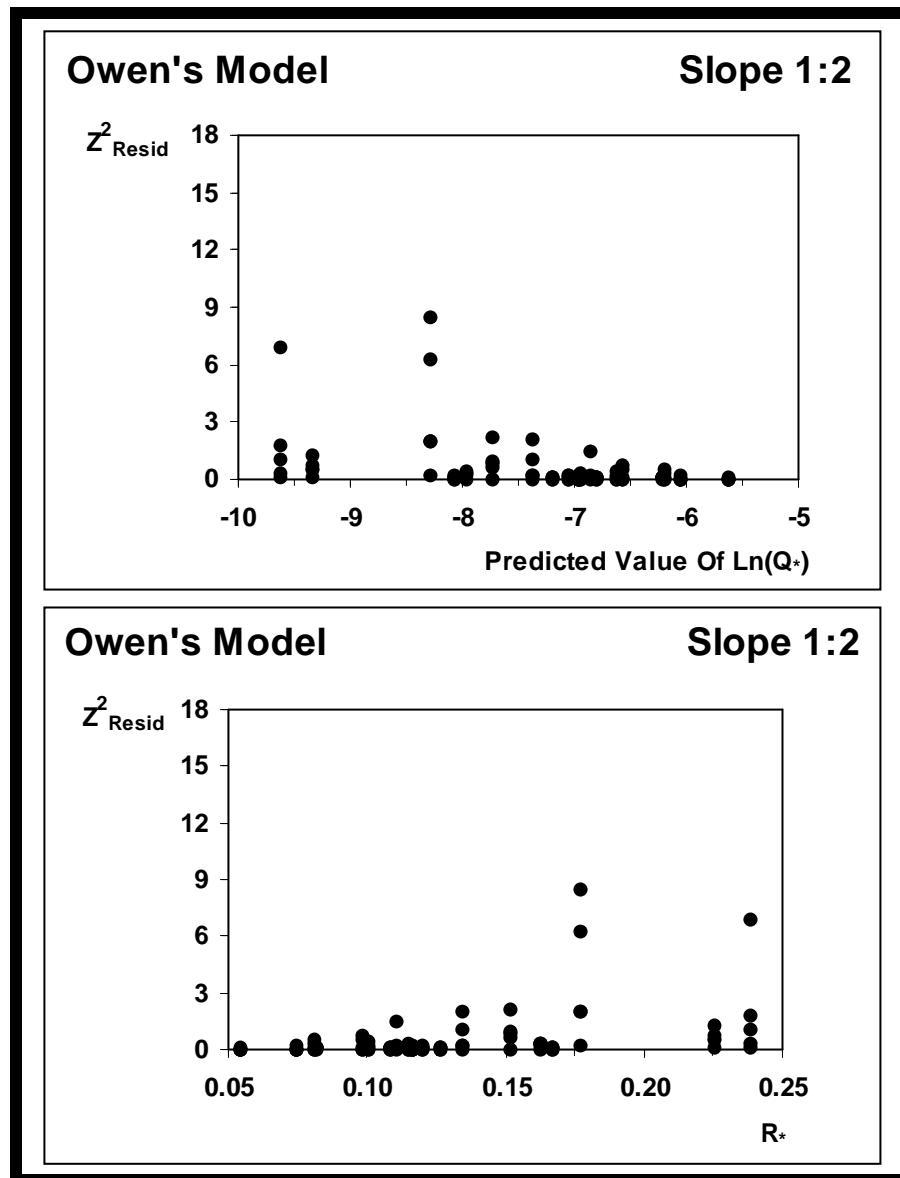
Figure A2.21: Scatterplots of the squared standardised residuals against the predicted values of Ln(Q$_*$) and against R$_*$: LAD regression for Owen's model, slope 1:2.

Independent Variable	Dependent Variable	Predicted Dependent Variable	Residuals	Standardised Residuals
0.177	-10.404	-8.291	-2.112	-2.910
0.239	-11.530	-9.624	-1.906	-2.626
0.177	-10.107	-8.291	-1.816	-2.501
0.152	-6.666	-7.734	1.068	1.472

Table A2.8: Details of the four highest residuals from LAD regression for Owen's model, slope 1:2.
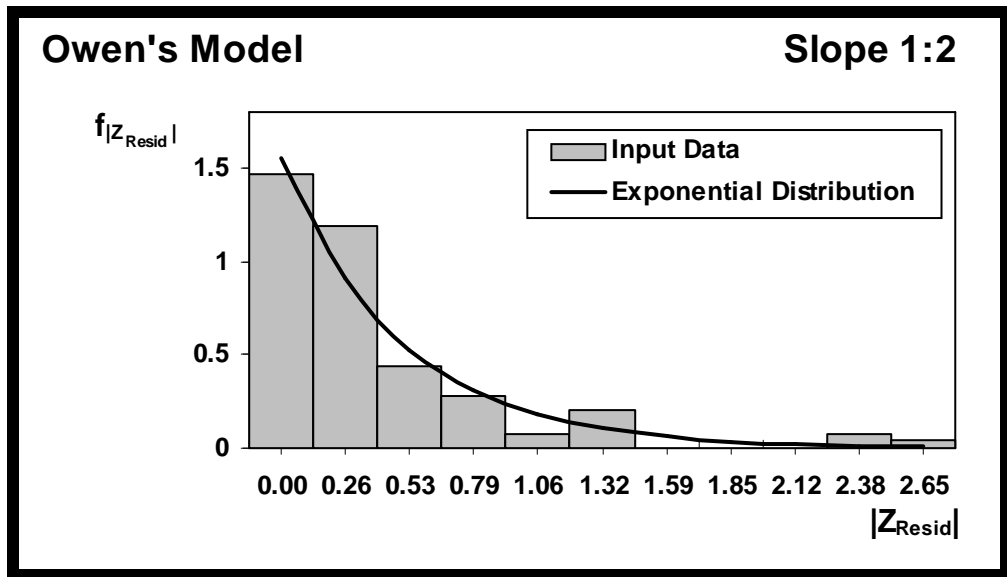
Figure A2.22: Comparison of the Exponential probability density function for absolute values of the standardised residuals with data from LAD regression for Owen's model, slope 1:2.
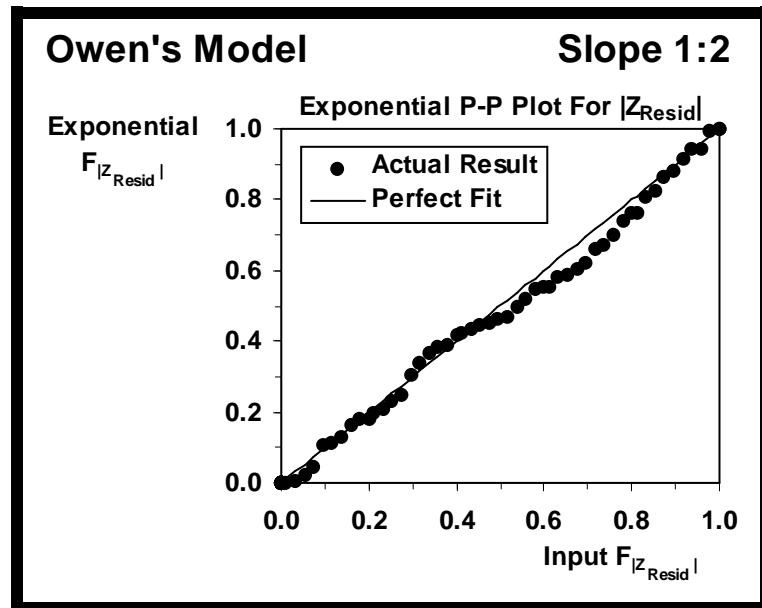
Figure A2.23: Comparison of the Exponential cumulative distribution function for absolute values of the standardised residuals with data from LAD regression for Owen's model, slope 1:2.
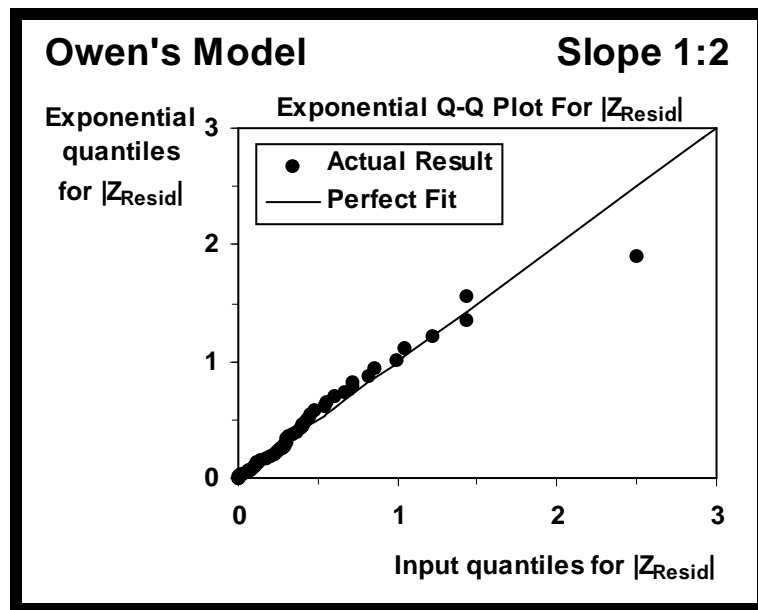
Figure A2.24: Comparison of the Exponential quantiles for absolute values of the standardised residuals with data from LAD regression for Owen's model, slope 1:2.

A2.3   Main Conclusions Of The Analysis

Regression equations were fitted to Owen's data using the LS method. The presence of outliers and violation of the Normal error LS assumption lead to the subsequent use of the more robust technique of LAD. The LAD assumption that errors have a Double Exponential distribution appears much more plausible than the Normal error assumption for Owen's data set. The results obtained from the LAD regression seem to be more sensible than those of the LS method. As expected, the parameters are less affected by the outliers identified in the diagnostic plots. Thus, the LAD estimates of the coefficients are recommended both for the H&R and Owen models.

The LAD regression results suggest that the H&R model and the Owen model perform almost equally well for the range of conditions tested. This is not surprising since the main differences between the two models are expected to exist for very small discharges in the ranges of practical interest. These conditions are not covered by Owen's data set.

LS parameter values can be calculated explicitly from the data but the LAD parameters cannot. However, efficient algorithms have now eliminated this computational difficulty. Statistical inference procedures for LAD have recently been developed but not completely implemented in the usual statistical packages. LAD forms an attractive robust alternative to LS for data sets with outlying observations which violate the LS assumption of normality. The author hopes that this study will encourage engineers to use robust techniques in general; in particular, the LAD method is an attractive alternative to the classical LS technique.

APPENDIX A3 - SAS/IML Program

```
################################################################################

SAS/IML PROGRAM FOR REGRESSION

################################################################################
```

This program uses the SAS/IML language (SAS Institute Inc, 1988) to perform
regression analysis on Owen's data, using the least-squares (LS) method and the
least-absolute-deviations (LAD) method. Both the H&R model and Owen's model are used
in the analyses. For the program to work, two files are needed on the A drive:

    i)   file SLOPE12.POR which is the data file that has to contain the X and Y values
        for each regression line. Here, for the H&R model, X is called "lnkrstar" and
        Y is called "lnkqstar"; for the Owen model, X is called "rstarow" and Y is
        called "lnqstow". The file must contain the values of these four variables.
        The extension *.POR allows this file to be used as a data file in SAS and
        SPSS.

    ii) file CODE.SAS which contains this program.

Also needed on the A drive are two directories called A:\SASFILES and A:\SASDESC.

When this program has been run in SAS, two files called A:\REG_RES.DBF and
A:\SASDESC\EXCHANGE.SA2 have been created. The former consists of variables saved
during the regression process and which are then used for plotting purposes (plots
can be done, for example, in Excel). The latter is of no importance for the user.
These two files cannot be overwritten, so after each run of the program, the user has
to rename the file A:\REG_RES.DBF and has to delete the file A:\SASDESC\EXCHANGE.SA2.

Note that the C drive can be used instead of the A drive if the code in this program
is changed accordingly.

For more details on this program, the user is referred to SAS Institute Inc (1988)
and Freund & Littel (1991).

```
################################################################################


libname slope12 spss 'a:slope12.por';
libname tete1 'a:\sasfiles';
libname tete2 'a:\sasfiles';
libname tete3 'a:\sasfiles';
libname tete5 'a:\sasfiles';
libname tete6 'a:\sasfiles';
libname tete4 'a:\sasdesc';
data sastemp;
   set slope12._f_;
run;


====================================================================================
rq performs the LAD calculations.
====================================================================================
proc iml;
        start rq(yname,Y,xname,X,b,predict,error,zres,q);
        bound=1.0e10;
        coef=X`;
        m=nrow(coef);
        n=ncol(coef);
        r=repeat(0,m+2,1);
        L=repeat(q-1,1,n) || repeat(0,1,m) || -bound || -bound ;
        u=repeat(q,1,n) || repeat(.,1,m) || {..} ;
        a=(y` || repeat(0,1,m) || {-1 0}) //
          (repeat(0,1,n) || repeat(-1,1,m) || {0 -1}) //
          (coef || I(m) || repeat(0,m,2));
        basis=n+m+2-(0:n+m+1);
        call lp(rc,p,d,a,r,,u,L,basis);
        L=repeat(q-1,1,n) || repeat(0,1,m) || -bound || {0};
        u=repeat(q,1,n) || repeat(0,1,m) || {. 0};
        call lp(rc,p,d,a,r,n+m+1,u,L,basis);
        variable=xname`;b=d[3:m+2];
        predict=X*b;
        error=y-predict;
        wsum=sum(choose(error<0,(q-1)*error,q*error));
        SA=2*wsum;
        MSA=SA/(n-m);
        s1=2*MSA;
        zres=error/s1;
        OBS=X[,2]||Y||predict||error||zres;
        print ,,,,'LEAST ABSOLUTE REGRESSION',
```

```
                      'Dependent Variable: ' yname,
                      'Regression Quantile: ' q,
                      'Number of Observations: ' n,
                      'Sum of Absolute Errors: ' wsum,
                      'Mean Absolute Deviations: ' MSA,
                      'Estimate of s: ' s1,
                       variable b,,,
                      'PREDICTED VALUES AND RESIDUALS',
                       OBS (|colname={X Y YPred Resid ZResid} format=9.5|);
        finish;


==========================================================================================
Calls rq to perform the LAD calculations for the H&R model.
==========================================================================================
        use sastemp;
        read all var {'lnkrstar'} into X;
        read all var {'lnkqstar'} into Y;
        n=nrow(X);
        m=ncol(X);
        X=repeat(1,n,1) || X;
        run rq('lnkqstar',Y,{'lnA' 'ln(1-R*)'},X,b,pre,res,zres,.5);


==========================================================================================
LS calculations for the H&R model.
==========================================================================================
        c=inv(X`*X);
        b=c*X`*Y;
        dfe=n-m-1;
        SSE=Y`*Y-b`*X`*Y;
        MSSE=SSE/dfe;
        s2=sqrt(MSSE);
        L={0 1};
        dfr=m;
        SSR=(L*b)`*inv(L*c*L`)*(L*b);
        MSSR=SSR/dfr;
        F=MSSR/MSSE;
        ProbF=1-PROBF(F#F,dfr,dfe);
        SOURCE=(m||SSR||MSSR||F||ProbF)//(dfe||SSE||MSSE||{.}||{.});
        seb=sqrt(vecdiag(c)#MSSE);
        T=b/seb;
        ProbT=1-PROBF(T#T,1,dfe);
        VARIABLE=b||seb||T||ProbT;
        pre2=X*b;
        stpre2=pre2;
        create tete1.new var{pre2};
        append;
        summary var {pre2} stat {mean std} opt {save};
        zpre2=(stpre2-pre2[,1])/pre2[,2];
        close tete1.new sastemp;
        res2=Y-stpre2;
        zres2=res2/s2;
        OBS=X[,2]||Y||stpre2||zpre2||res2||zres2;
        print ,,,'LEAST SQUARES REGRESSION - HR Model',
                 SOURCE (|colname={DF SS MSS F ProbF} rowname={Reg Resid} format=9.5|),,
              'PARAMETER ESTIMATES',
                 VARIABLE (|colname={B SEB T ProbT} rowname={Int Slope} format=10.6|),,,
              'PREDICTED VALUES AND RESIDUALS',
                 OBS (|colname={X Y YPred ZYPred Resid ZResid} format=9.5|);
        create tete5.new var{stpre2 zpre2 res2 zres2};
        append;
        sstpre2=stpre2;
        szpre2=zpre2;
        sres2=res2;
        szres2=zres2;
        summary var {stpre2 zpre2 res2 zres2} opt {save};
        close tete5.new sastemp;


==========================================================================================
Calls rq to perform the LAD calculations for the Owen model.
==========================================================================================
        use sastemp;
        read all var {'rstarow'} into XOW;
        read all var {'lnqstow'} into YOW;
        n=nrow(XOW);
        m=ncol(XOW);
        XOW=repeat(1,n,1) || XOW;
        run rq('lnqstow',YOW,{'lnA' 'R*'},XOW,b,preow,resow,zresow,.5);


==========================================================================================
LS calculations for the Owen model.
==========================================================================================
        cow=inv(XOW`*XOW);
        bow=cow*XOW`*YOW;
```

```
        dfeow=n-m-1;
        SSEow=YOW`*YOW-bow`*XOW`*YOW;
        MSSEow=SSEow/dfeow;
        s2ow=sqrt(MSSEow);
        L={0 1};
        dfrow=m;
        SSRow=(L*bow)`*inv(L*cow*L`)*(L*bow);
        MSSRow=SSRow/dfrow;
        Fow=MSSRow/MSSEow;
        ProbFow=1-PROBF(Fow#Fow,dfrow,dfeow);
        SOURCE=(m||SSRow||MSSRow||Fow||ProbFow)//(dfeow||SSEow||MSSEow||{.}||{.});
        sebow=sqrt(vecdiag(cow)#MSSEow);
        Tow=bow/sebow;
        ProbTow=1-PROBF(Tow#Tow,1,dfeow);
        VARIABLE=bow||sebow||Tow||ProbTow;
        pre2ow=XOW*bow;
        stpre2ow=pre2ow;
        create tete3.new var{pre2ow};
        append;
        summary var {pre2ow} stat {mean std} opt {save};
        zpre2ow=(stpre2ow-pre2ow[,1])/pre2ow[,2];
        close tete3.new sastemp;
        res2ow=YOW-stpre2ow;
        zres2ow=res2ow/s2ow;
        OBS=XOW[,2]||YOW||stpre2ow||zpre2ow||res2ow||zres2ow;
        print ,,,'LEAST SQUARES REGRESSION - Owen Model',
                SOURCE (|colname={DF SS MSS F ProbF} rowname={Reg Resid} format=9.5|),,
                'PARAMETER ESTIMATES',
                VARIABLE (|colname={B SEB T ProbT} rowname={Int Slope} format=10.6|),,,
                'PREDICTED VALUES AND RESIDUALS',
                OBS (|colname={X Y YPred ZYPred Resid ZResid} format=9.5|);
        create tete6.new var{stpre2ow zpre2ow res2ow zres2ow};
        append;
        sspre2ow=stpre2ow;
        szpre2ow=zpre2ow;
        sres2ow=res2ow;
        szres2ow=zres2ow;
        summary var {stpre2ow zpre2ow res2ow zres2ow} opt {save};
        close tete6.new sastemp;


========================================================================================
Creates the results file A:\REG_RES.DBF.
========================================================================================
        create tete2.new var{Y pre res zres sstpre2 szpre2 sres2 szres2 YOW preow
                        resow zresow sspre2ow szpre2ow sres2ow szres2ow};
        append;
        close tete2.new sastemp;
proc dbload dbms=dbf data=tete2.new;
        path='a:\reg_res.dbf';
        accdesc=tete4.exchange;
        load;
run;
```

APPENDIX A4 - BestFit

BestFit is a software package for fitting probability distributions to data (Palisade Corporation, 1996). Selected statistical distributions can be fitted to a maximum of 30 000 data points. Results may be displayed numerically and graphically. The package allows the user to transfer all results, including graphs, statistics and distribution functions, to other programs (e.g. Excel, @Risk) for further analysis and presentation. Examples of output which can be generated are given in Appendix A5.

BestFit goes through the following steps when attempting to find the "best fit" to the input data:

- For each distribution type selected, a first guess for its parameters is made using Maximum Likelihood Estimators - MLEs (Bury, 1975; Law & Kelton, 1991; Walpole & Myers, 1993; Palisade Corporation, 1996).

- The fit is optimised (if the user wishes) using the Levenberg-Marquardt method (Press et al, 1992; Palisade Corporation, 1996). Otherwise, the maximum likelihood estimates for the parameters are adopted. The Levenberg-Marquardt method takes an iterative approach to minimise a goodness-of-fit statistic. The goodness-of-fit test used by BestFit for optimisation is the Chi-Square test. Note that the Levenberg-Marquardt method does not necessarily find the absolute minimum for the Chi-Square value; rather, it may find a local minimum.

- The goodness of fit is measured for the distribution functions. For continuous distributions, BestFit can use the Kolmogorov-Smirnov (K-S) statistic and the Anderson-Darling (A-D) statistic, as well as the Chi-Square statistic (Bendat & Piersol, 1971; D'Agostino & Stephens, 1986; Law & Kelton, 1991; Press et al, 1992; Groebner & Shannon, 1993; Walpole & Myers, 1993; Palisade Corporation, 1996). These tests provide an idea of how well a certain distribution fits the input data.

- For continuous distributions, twenty distribution functions are compared and the one with the lowest goodness-of-fit value can be considered the "best fit" to the input data. Including continuous and discrete distributions, there are 26 statistical distributions available in BestFit (Evans et al, 1993; Palisade Corporation, 1996).

Despite its name, BestFit does not provide an absolute answer to the question of which distribution best fits the data. It identifies the likelihood that the input data is taken from a selected distribution. Before employing any fitted distribution obtained from BestFit, the user should assess the results quantitatively and qualitatively by examining the summary statistics and the results of the goodness-of-fit tests which BestFit produces, and by considering the graphs.

APPENDIX A5 - Fitting A Distribution To Data

The choice of probability distribution to describe the randomness of a variable can have a large impact on the accuracy of the results of probabilistic calculations and on the quality of the decisions made using these results (Burcharth, 1992). Consequently, the task of selecting a distribution deserves special consideration, particularly if a sensitivity analysis of the problem confirms a significant contribution of the distribution to the final results (see Chapters 2 and 6).

If it is possible to collect data on the input random variable of interest, these data can be used in one of the following two ways to specify a distribution (Law & Kelton, 1991):

- The data values themselves are used to define an empirical distribution function.

- Standard techniques of statistical inference are used to "fit" a theoretical distribution (e.g. Normal, Log-Normal) to the data, to calculate its parameters and to determine its goodness of fit.

If a theoretical distribution can be found that fits the observed data reasonably well, then this approach is preferable to using an empirical distribution (Law & Kelton, 1991): i) a theoretical distribution "smooths out" the data and may provide information on the overall underlying distribution; ii) with a fitted theoretical distribution, values outside the range of the observed data can be generated, though care should be taken when considering extrapolated values (Copeiro, 1978; Carvalho, 1992b).

This appendix investigates which distributions (if any) available in PARASODE and BestFit (Normal, Log-Normal, Gumbel, Rectangular, Gamma, Beta, Exponential, Rayleigh and Weibull) provide a reasonable model for the e_B variable (Chapter 3). It discusses the three basic steps in specifying theoretical distributions on the basis of field and/or laboratory data (Law & Kelton, 1991):

Step I   To hypothesise the form of the distributions.
Step II   To estimate the parameters of the distributions.
Step III To test the appropriateness of the fitted distributions.

The theoretical distribution of e_B has been separately analysed for the three slopes of 1:1, 1:2 and 1:4, both for the H&R model and for Owen's model. The H&R model employed both $(R_{max})_{37\%}$ and $(R_{max})_{99\%}$ in defining the value of C. Similar conclusions were drawn in all cases. In this appendix, an illustrative example is provided for the H&R model, $(R_{max})_{37\%}$, for slope 1:1.

A5.1   Hypothesise The Form Of The Distributions

In modelling the physical world, the form of the probability distribution of a random variable may be deduced theoretically on the basis of assumptions about the reality. However, the process is often difficult. Firstly, the probability model needed to describe a physical phenomenon may not be formulated readily (Ang & Tang, 1975). Secondly, the functional form of the probability distribution may not be easily derived, although under certain circumstances, the properties of the physical process may suggest the form of the distribution. Therefore, on many occasions, the required probability distribution has to be inferred empirically, that is, based entirely on available observational data. This is the case in this study in hypothesising the form of the distribution of e_B.

In practice, the choice of the probability distribution may also be dictated by mathematical tractability and convenience. For example, because of the mathematical simplifications possible with the Normal distribution, and the wide availability of information associated with this distribution, it is frequently used to model non-deterministic problems, even when there is no clear basis for such a choice (Ang & Tang, 1975).

The first stage in hypothesising a particular form of distribution from observed data is to decide what general characteristics the distribution is expected to have. In the second column of Table A5.1, summary statistics are presented for the input data corresponding to the H&R model, $(R_{max})_{37\%}$, slope 1:1. In Figure A5.1, a corresponding histogram based on 11 classes of width 0.23 is shown. Other graphs could have been used for general guidance (see, for example, Law & Kelton, 1991, for further details).

	INPUT DISTRIBUTION	LOG NORMAL	MAXIMA TYPE I (GUMBEL)	GAMMA
Parameter 1		-0.02065	0.888504	10.08094
Parameter 2		0.318767	0.269917	0.102194
Minimum	0.358773			
Maximum	2.879325			
Mean	1.030215	1.030615	1.044302	1.030215
Mode	0.931626	0.884916	0.888504	0.928021
Median	1	0.979562	0.987432	0.996358
Standard Deviation	0.345835	0.337051	0.346182	0.324472
Variance	0.119602	0.113603	0.119842	0.105282
Skewness	2.560238	1.016094	1.139547	0.629911
Kurtosis	10.41889	4.890328	5.4	3.595182
Minimum	0.358773	0.358773	0.358773	0.358773
Maximum	2.879325	2.879325	2.879325	2.879325
P1	5	4.263141	3.569606	5.289339
P2	20	22.51086	21.91579	20.26023
P3	31	28.88299	29.31184	27.94029
P4	26	20.36826	20.42403	21.86319
P5	6	10.73151	10.76643	11.92440
P6	4	4.857546	5.036838	5.063927
P7	1	2.028647	2.239037	1.789538
P8	1	0.813057	0.973808	0.549417
P9	0	0.319959	0.419589	0.150933
P10	0	0.125347	0.180068	0.037892
P11	1	0.049301	0.077145	8.83E-02
#Classes	11			
Interval Width	0.229141			
Results				
Chi-Square Test Value		23.69993	17.00955	116.4673
Confidence		Rejected	>0.07	Rejected
Rank		2	1	3
Kolmogorov-Smirnov Test Value		0.077729	0.086047	0.086107
Confidence		>0.15 *	>0.05	>0.15 *
Rank		1	2	3
Anderson-Darling Test Value		0.845229	0.811245	0.782497
Confidence		>0.15 *	>0.025	>0.15 *
Rank		3	2	1
Confidence				
Chi-Square Adjusted Value		23.69993	17.00955	116.4673
Critical Value @ .75		6.737201	6.737201	6.737201
Critical Value @ .5		9.341818	9.341818	9.341818
Critical Value @ .25		12.54886	12.54886	12.54886
Critical Value @ .1		15.98717	15.98717	15.98717
Critical Value @ .05		18.30703	18.30703	18.30703
Critical Value @ .025		20.48317	20.48317	20.48317
Critical Value @ .01		23.20925	23.20925	23.20925
Kolmogorov-Smirnov Adjusted Value		0.767816	0.838685	0.850573
Critical Value @ .15		1.138	0	1.138
Critical Value @ .1		1.224	0.803	1.224
Critical Value @ .05		1.358	0.874	1.358
Critical Value @ .025		1.48	0.939	1.48
Critical Value @ .01		1.628	1.007	1.628
Anderson-Darling Adjusted Value		0.845229	0.81291	0.782497
Critical Value @ .15		1.61	0.474	1.61
Critical Value @ .1		1.933	0.637	1.933
Critical Value @ .05		2.492	0.757	2.492
Critical Value @ .025		3.07	0.877	3.07
Critical Value @ .01		3.857	1.038	3.857
Targets				
#1 Value	0.683968	0.651058	0.663384	0.642357
#1 Percentile%	10%	10%	10%	10%
#2 Value	0.852544	0.828771	0.8384	0.838712
#2 Percentile%	30%	30%	30%	30%
#3 Value	1	0.979562	0.987432	0.996358
#3 Percentile%	50%	50%	50%	50%
#4 Value	1.133459	1.157788	1.166769	1.172682
#4 Percentile%	70%	70%	70%	70%
#5 Value	1.382523	1.473818	1.495916	1.461741
#5 Percentile%	90%	90%	90%	90%
#6 Value	1.571693	1.654788	1.69021	1.61543
#6 Percentile%	95%	95%	95%	95%

Table A5.1: Statistical report obtained from BestFit for the H&R model, $(R_{max})_{37\%}$, slope 1:1.
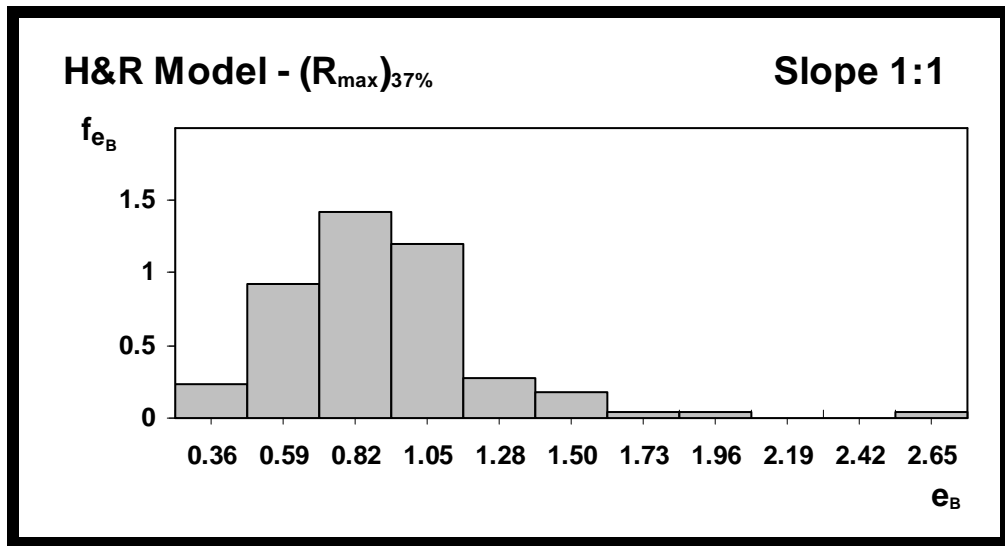
Figure A5.1: Histogram of the input data for the H&R model, $(R_{max})_{37\%}$, slope 1:1.

The shape of the histogram strongly suggests that the actual underlying distribution is skewed to the right (i.e. its right tail is longer than its left tail). This fact is supported by the values in Table A5.1 which show that the mean (1.0302) is 10.6% bigger than the mode (0.9316), and the skewness is 2.56 (the skewness is zero for a symmetric distribution). These characteristics of the input data appear to rule out symmetric distributions (in this case, the Normal and the Rectangular distributions). Furthermore, the kurtosis of the input distribution is 10.42, bigger than 3, which indicates distributions with "thicker" tails than for a Normal distribution. The coefficient of variation, $\sigma/\mu = 0.336$, makes it fairly unlikely that the actual distribution could be Exponential (Law & Kelton, 1991) which has a coefficient of variation of 1. On the other hand, the Log-Normal, Gumbel, Gamma, Beta, Rayleigh and Weibull distributions can all take on shapes similar to that of the histogram shown in Figure A5.1. So, it seems that, potentially, e_B may be described by any of these distributions and they are proposed here as candidate distributions.

A5.2   Estimate The Parameters Of The Distributions

After one or more candidate distributions have been hypothesised, the values of the parameters of these distributions are estimated so that the

distributions may be used in the FORM analysis and in simulation. BestFit uses both the Maximum Likelihood Estimators (MLEs) and the Levenberg-Marquardt method to estimate the parameters of the distributions from the input data (see Appendix A4 for further details).

A5.3   Test The Appropriateness Of The Fitted Distributions

After identifying the candidate distributions and estimating their parameters, they must be examined to assess how well they represent the data. If an objective assessment is desired, statistical goodness-of-fit tests should be used in conjunction with probability plots (Ang & Tang, 1975). This approach has been used here. The Log-Normal, the Gumbel and the Gamma distributions performed better than the others in describing e_B. So, for convenience, the results shown below are limited to these three distributions. The fact that the analysis did not suggest one "best" distribution is not surprising: there are insufficient data to identify unambiguously one distribution as the most appropriate (Melchers, 1987).

A5.3.1 Probability Plots

First, graphs comparing the histogram of the data with the probability density functions of the fitted distributions have been plotted (Figure A5.2). In general, the agreement between the theoretical distributions and the input data seems good.

Figures A5.3 and A5.4 show the P-P and the Q-Q graphs plotted for the three distributions (see Appendix A1 for explanations of P-P and Q-Q plots). Clearly, the three hypothesised distributions appear to fit the input data equally well both in the body and at the tails. As expected (Appendix A1), the data points are not randomly scattered about the straight lines.
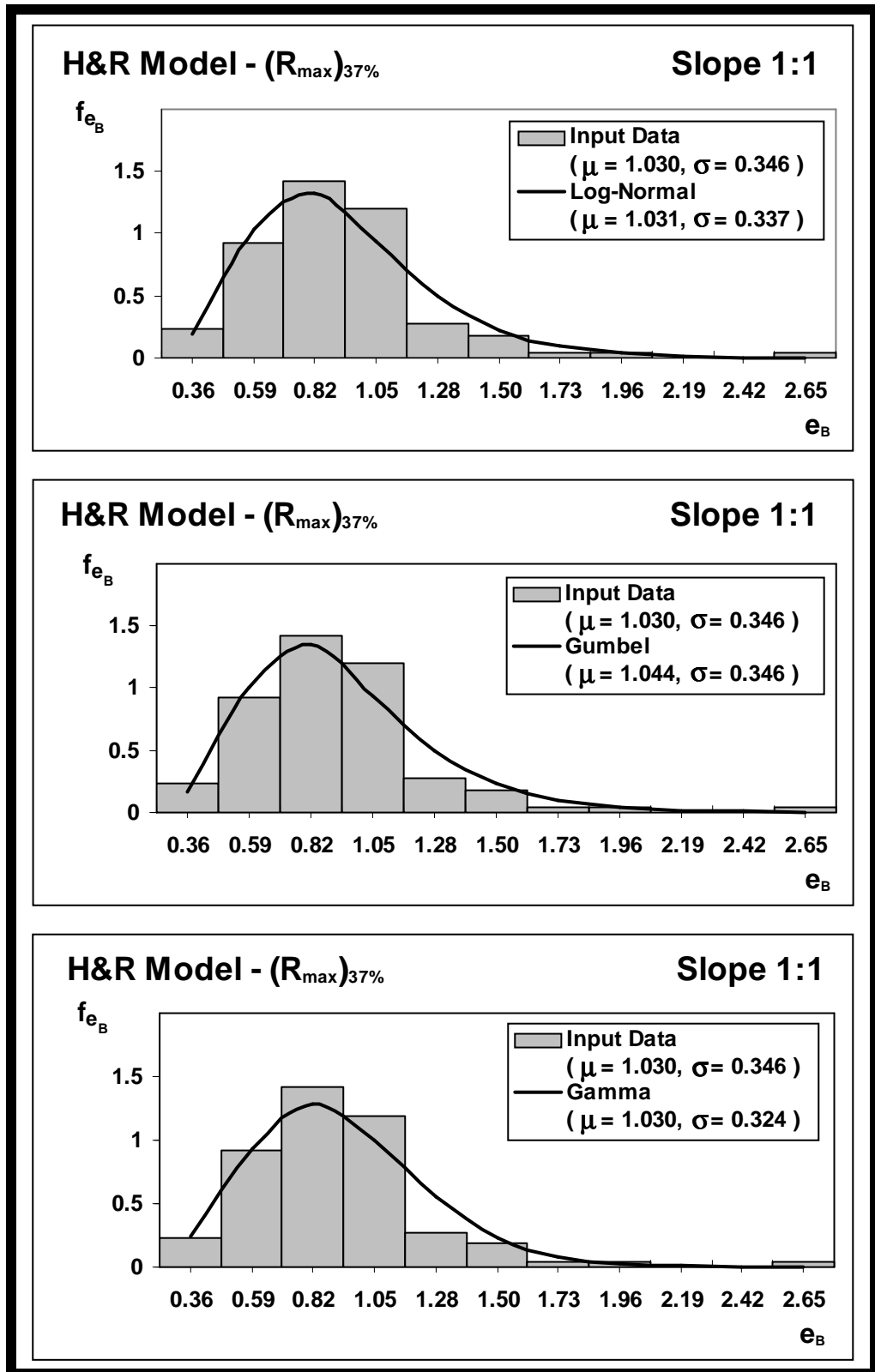
Figure A5.2: Comparison of theoretical probability density functions for e_B with input data.
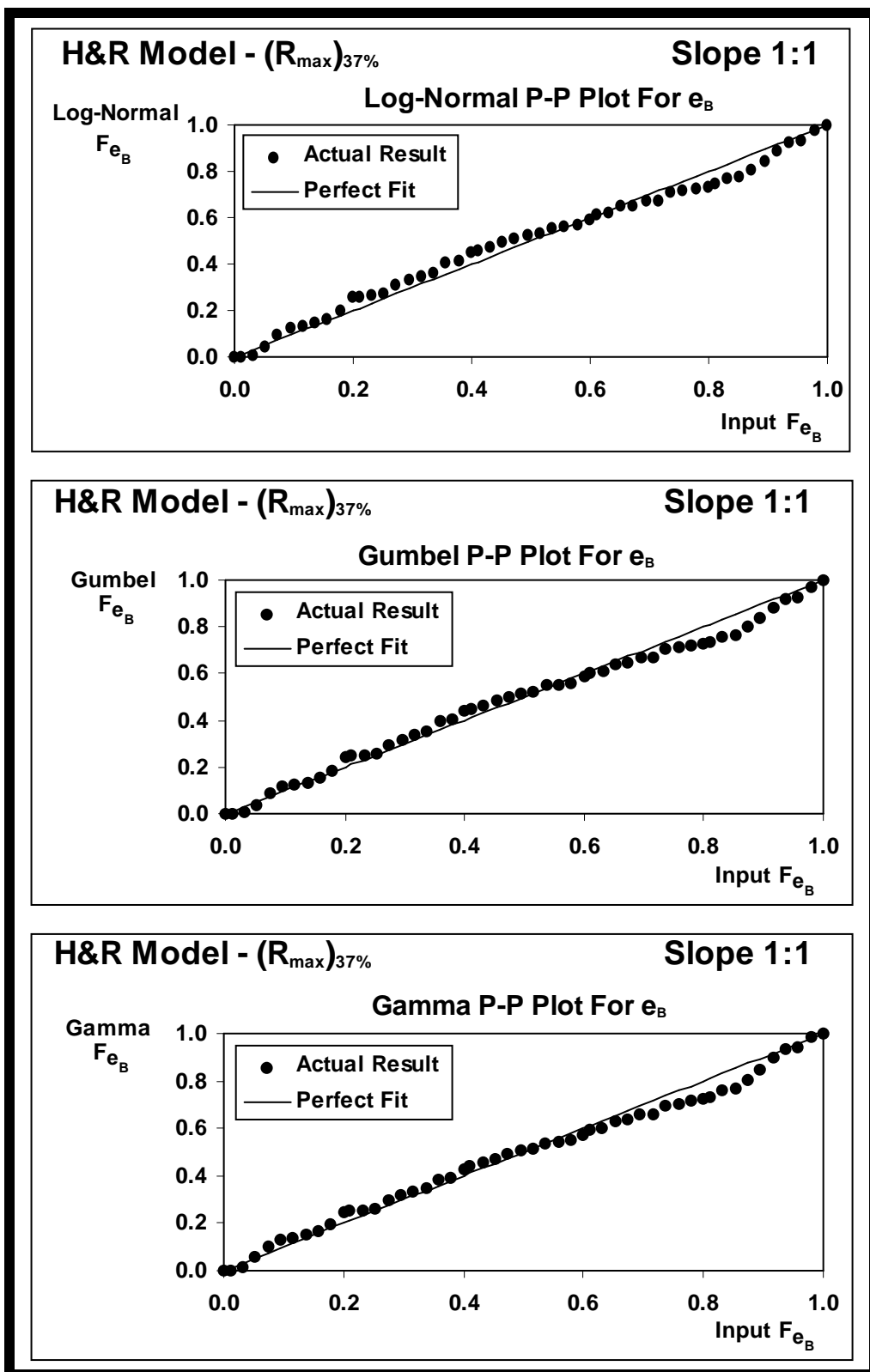
Figure A5.3: Comparison of theoretical cumulative distribution functions for e_B with input data.
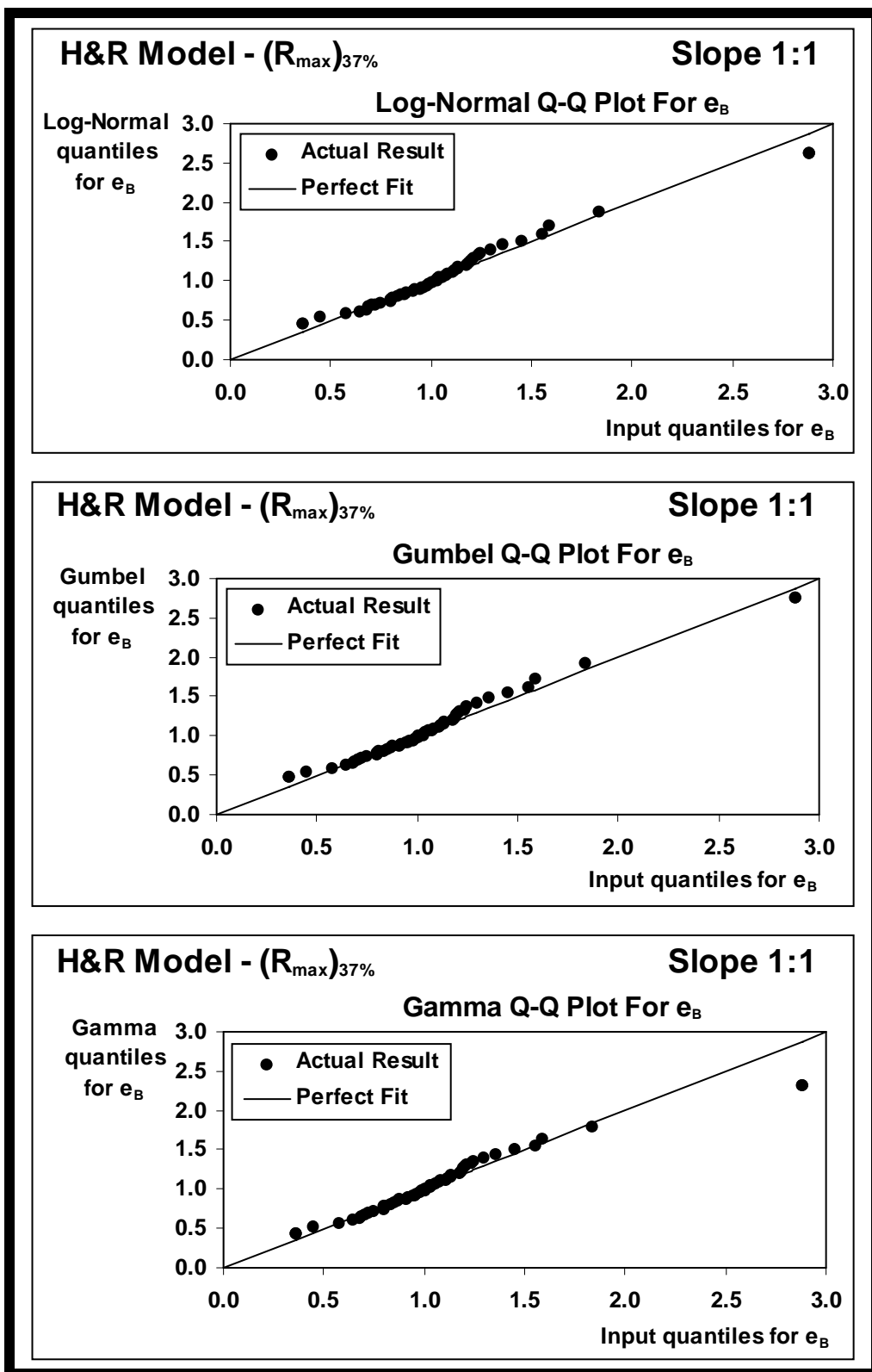
Figure A5.4: Comparison of theoretical quantiles for e_B with input data quantiles.

A5.3.2 Goodness-Of-Fit Tests

A graphical approach for testing hypothesised distributions should always be complemented by statistical goodness-of-fit tests. A goodness-of-fit test assesses formally whether the observations $X_1,...,X_N$ can be considered to be a sample from a particular distribution with cumulative distribution function F_X^F. It can be used to test the null hypothesis Ho: *The X_i's are independent random variables with cumulative distribution function F_X^F* (Law & Kelton, 1991). The acceptance or rejection of Ho depends on the significance level, α. Ho is accepted at the α level of significance if the test value is less than or equal to the critical value associated with α. If the test value is greater than the critical value, then the assumed distribution is rejected at the α level of significance. A distribution that is acceptable at one significance level may be unacceptable at another significance level.

The choice of α is largely a subjective matter (Ang & Tang, 1975). If the chosen α is very small (say, 0.001), the null hypothesis might not be rejected even if appreciable differences exist between the two distributions (type II error). If the chosen α is large (say, 0.20), rejection of the null hypothesis could well occur even when the two distributions are almost identical (type I error). So, there is the need to compromise between type I and type II errors when choosing a significance level (Groebner & Shannon, 1993). α is commonly set at 0.05 (Hutchinson, 1993), although there is no logical reason to select this particular value. Preferably, goodness-of-fit tests should be performed for several values of α, allowing the user to determine, approximately, the highest level of significance at which the hypothesised distribution should not be rejected. In BestFit the significance level, α, is indicated as "confidence" and the critical values for each test are indicated as "Critical Value @ α" (see Table A5.1).

Importantly, goodness-of-fit tests may reject a model as inadequate, but they can never prove that a model is correct.

Sample size is crucial in influencing decisions based on statistical tests. Such tests are often not very powerful for small to moderate sample sizes: they might not be sensitive to subtle disagreements between the data and the fitted distribution. They should be regarded as a systematic approach for detecting fairly gross differences. On the other hand, if the sample is very large, these tests will almost always reject Ho (Law & Kelton, 1991). Since

Ho is virtually never exactly true, even a minor departure from the fitted distribution will be detected for large samples.

The tests mentioned here are those used by BestFit: the Chi-Square test, the Kolmogorov-Smirnov (K-S) test and the Anderson-Darling (A-D) test. D'Agostino & Stephens (1986) describe other goodness-of-fit tests and stress that no one test is optimal for all possible deviations from the hypothesised distribution. In fact, tests might not provide a unique answer. The ranking of the distributions for one test is unlikely to be identical with that from another test due to their different characteristics:

- The Chi-Square test is the most common goodness-of-fit test due to its flexibility and ease of use; but it requires the input data to be grouped into classes and there are no firm guide-lines for selecting the number and size of these classes. Unfortunately, results depend, to some degree, on the number and size of the classes adopted. BestFit uses, as the default, Scott's Normal approximation (Law & Kelton, 1991) to calculate the number of classes, k, from the number of data points, N: $k=(4N)^{2/5}$. But other formulations can be applied (see, for example, Hahn & Shapiro, 1967; Law & Kelton, 1991). For the Chi-Square test, the distribution which has the smallest value of the test statistic is ranked in first place. The Chi-Square statistic is defined as (Law & Kelton, 1991):

$$\chi^2 = \sum_{j=1}^{k} \frac{(N_j - Np_j)^2}{Np_j} \qquad (A5.1)$$

 where N_j is the number of X_i's in the jth class (note that $\sum_{j=1}^{k} N_j = N$); p_j is the expected proportion of the X_i's that would fall in the jth class if sampling was done from the fitted distribution.

- The K-S test does not require the data to be grouped into classes but is poor of detecting tail discrepancies. The K-S statistic is (Law & Kelton, 1991):

$$D_N = \sup\left[\left| F_X(x) - F_X^F(x) \right| \right] \qquad (A5.2)$$

 where $F_X^F(x)$ is the hypothesised cumulative distribution function evaluated at X=x; $F_X(x)=N_X/N$ is the empirical cumulative distribution; N_X is the number of X_i's less than or equal to x, i=1,...,N; "sup" means the smallest upper bound on members of the set [...]. "sup" is used here instead of the more familiar "max" since,

> in some cases, the maximum may not be well defined (Law & Kelton, 1991).

- • The A-D test does not require the data to be grouped into classes and places more emphasis on tail values than the K-S test. The form of the A-D statistic used for computations is (Law & Kelton, 1991):

$$A_N^2 = \frac{-\left\{\sum_{i=1}^{N}(2i-1)\left[\ell n Z_i + \ell n(1-Z_{N+1-i})\right]\right\}}{N} - N \qquad (A5.3)$$

> where $Z_i = F_X^F(x_i)$ and the other variables are defined as previously.

For the K-S and the A-D tests, the distribution which has the smallest value of the adjusted test statistic is ranked in first place (see Table A5.1). The adjustment performed to the actual test statistic depends on the distribution (for more details, see Law & Kelton, 1991; Palisade Corporation, 1996).

In BestFit, no account is taken of the fact that the parameters of the distributions are estimated from the input data: for the Chi-Square test, the critical value used for comparison with the actual value of the test is the upper α critical point for a Chi-Square distribution with k-1 degrees of freedom, $\chi^2(k-1;\alpha)$, and not k-m-1 degrees of freedom (Palisade Corporation, 1996), where k is the number of classes and m is the number of parameters of the distribution estimated from the data. Ho is rejected only if $\chi^2 > \chi^2(k-1;\alpha)$. This is often recommended (Law & Kelton, 1991) since it is conservative. Thus, the actual probability of rejecting Ho when it is true (type I error) is at least as small as the stated probability, α. On the other hand, the probability of a type II error (the probability of not rejecting a false Ho) is increased.

Note that, unlike the Chi-Square test, where the critical values are the same for all distributions, the K-S and the A-D tests include special cases depending on the distribution (for more details, see Palisade Corporation, 1996).

In Table A5.1, the Chi-Square test rejects the Log-Normal and the Gamma distributions at any of the usual α levels. On the other hand, the values of $\chi^2(10;\alpha)$ for $\alpha < 0.07$ for the Gumbel distribution, are not exceeded by the

test value $\chi^2 = 17$; so, Ho for the Gumbel distribution should not be rejected at the $\alpha < 0.07$ level. Thus, this test gives no reason to conclude that the input data is poorly fitted by the Gumbel(0.889, 0.270) distribution but it suggests that the probability is very low that the input data came from the Log-Normal(-0.021, 0.319) or the Gamma(10.08, 0.102) distributions.

According to the K-S test, the Log-Normal and the Gamma distributions should not be rejected for any of the levels of significance listed ($\alpha < 0.15$) and the Gumbel distribution should not be rejected for $\alpha < 0.05$. Thus this test gives no reason to conclude that the input data is poorly fitted by any of the three distributions.

The A-D test also suggests that the Log-Normal and the Gamma distributions should not be rejected at any of the levels of significance listed ($\alpha < 0.15$). The Gumbel distribution should not be rejected at the 2.5% level (since the adjusted value 0.813 is less than the critical A-D test value for $\alpha < 0.025$) but it should be rejected for larger values of α .

A5.4   Main Conclusions Of The Analysis

This appendix provides some guidance on how a theoretical distribution should be fitted to data. An illustrative example is given for the H&R model, $(R_{max})_{37\%}$, slope 1:1.

From the statistics presented in Table A5.1 and from Figures A5.1 to A5.4, it can be inferred that the Log-Normal, Gumbel and Gamma distributions provide reasonable representations of the input data. It can also be inferred that there is little to choose between the three distributions since they perform almost equally well.

In summary, there is no reason to believe, based on the figures shown above and on the statistical tests performed, that any of the three distributions should be rejected as adequate models for describing the randomness of e_B. Similar conclusions may be drawn by looking at all other cases.

APPENDIX B

Example Of Input And Output For DUNEPROB And DUNE

The computer programs DUNEPROB and DUNE introduced in Chapter 4 have been run for the same data set representing conditions near Hook of Holland (Figure B.1).
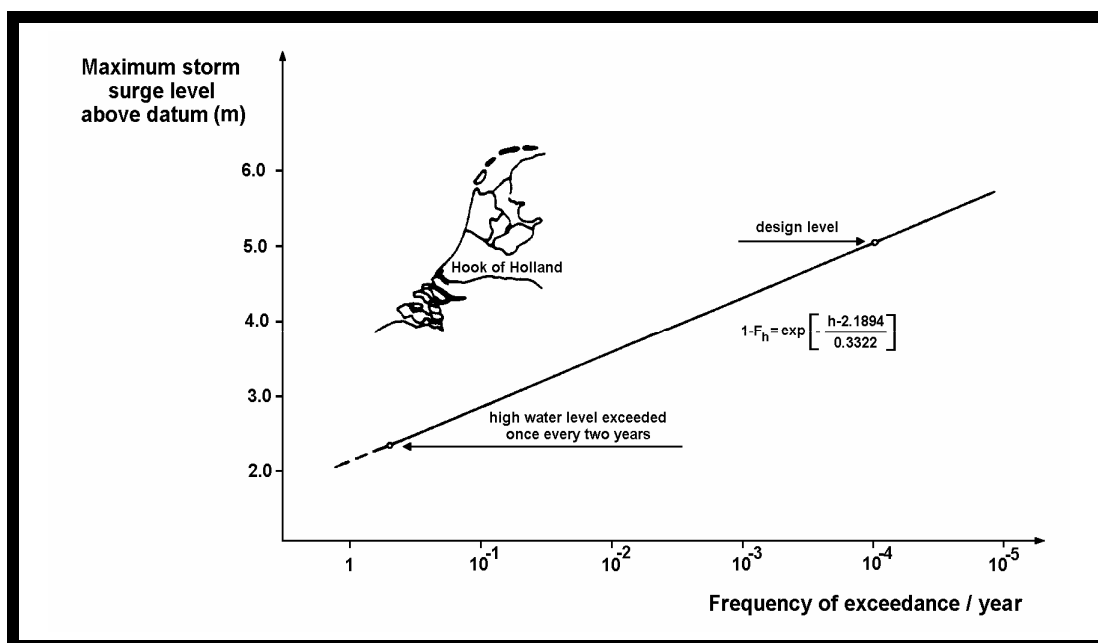
Figure B.1: Exceedance curve for the maximum storm surge level, Hook of Holland (modified after Van de Graaff, 1983).

The main purpose of this appendix is to show, with an example, that the input data related to the seven main erosion parameters is strongly orientated to conditions in The Netherlands. For this reason, only the data related to these seven parameters is shown in detail. Any additional input is mentioned only briefly. Results for the current example which are relevant to the discussion are also presented and compared.

DUNEPROB

The program starts by asking the user for a filename (e.g. *Hook*). It uses this information to name the four required input files:

Hook.Pro - contains the pairs of coordinates (X,Y) of the initial profile.

Hook.For - contains the information required for a FORM calculation (e.g. maximum number of iterations, number of FORM calculations).

> *Hook.Ran* - contains the random variables and their statistical properties.
>
> *Hook.AB* - contains the parameters required to define the distribution of the significant wave height as a function of the surge level during storm conditions.

Table B.1 shows the data provided in this example for the input file *Hook.Ran*.

Variable	Distribution	μ	σ
h (m)	Exponential	2.1894	0.3322
D_{50} (m)	Deterministic	225E-6	0
H_S_Inacc (m)	Normal	0	0.6
Surcharge (-)	Normal	0	1

Table B.1: Input file *Hook.Ran* for the current example.

The following data have been provided for the input file *Hook.AB* (see Figure B.1 and eqs. (4.5) and (4.6) in Section 4.5.2.2 of the main text):

- a=3.795
- b=0.444
- $H_{wavemax}$=10m

Table B.2 shows the main results obtained for a retreat distance with a probability of exceedance of approximately 10^{-5}/year. The retreat distance is 118.5m. From the sensitivity parameters, α^2, the most important contribution to the resulting variance is given by the surge parameter, a figure of 95%, indicating that this parameter is by far the most important one. The minor contribution of H_S to the resulting variance seems strange. Note, however, that due to the relationship between the maximum surge level and the significant wave height, $\alpha^2_{H_s\_Inacc} = 0.01$ represents only the effect of the variability in H_S about its expected value. In this case, $\sigma_{H_S} = 0.60\,m$.

Variable	Design Point	$\partial Z / \partial X$	α^2
h (m)	5.483E+00	-3.484E+01	0.95
$H_S\_$Inacc (m)	1.652E-01	-5.592E+00	0.01
Surcharge (-)	8.083E-01	-9.861E+00	0.04

Target Coordinate = -1.185E+02
P(Z $\leq$ 0) = 3.367E-05

Table B.2: Relevant DUNEPROB results for the current example.

DUNEPROB can also output results in the form of figures. For example, the probability of failure can be plotted as a function of the retreat distance.

DUNE

To run the program DUNE for the present example, the following two input data files are required:

Hook.Pro **-** contains the pairs of coordinates (X,Y) of the initial profile.

Hook.In **-** contains the values adopted for the maximum water level during surge, significant wave height, peak wave period and sand diameter (see Section 4.5.3); it also contains the parameter G_O which allows longshore transport to be considered (see Table 4.4); some settings for the dune erosion calculation are also part of this file.

For the present example (Ministerie van Verkeer en Waterstaat, 1994):

- Maximum water level during surge   5.7m
- Significant wave height   8.6m
- Peak wave period   12s
- Sand diameter   225E-06m
- G_O 0m³/m

The main results are presented in Table B.3.

Erosion Quantities (m³/m)

Erosion above surge level	= -219.52
Surcharge above surge level	= -74.88

Distances (m)

Erosion distance	= 88.34
Shift distance for the surcharge	= 30.53
Total retreat distance	= 118.87

Table B.3: Relevant DUNE results for the current example.

DUNE gives a retreat distance of 118.87m associated with a probability of failure of 10^{-5}/year. As expected, this value is almost the same as the one obtained with DUNEPROB (118.5m).

Note that DUNE produces more numerical output than is illustrated here. It also provides graphical output showing the initial dune profile, Vellinga's post-storm profile and the limiting profile.

APPENDIX C

PARASODE
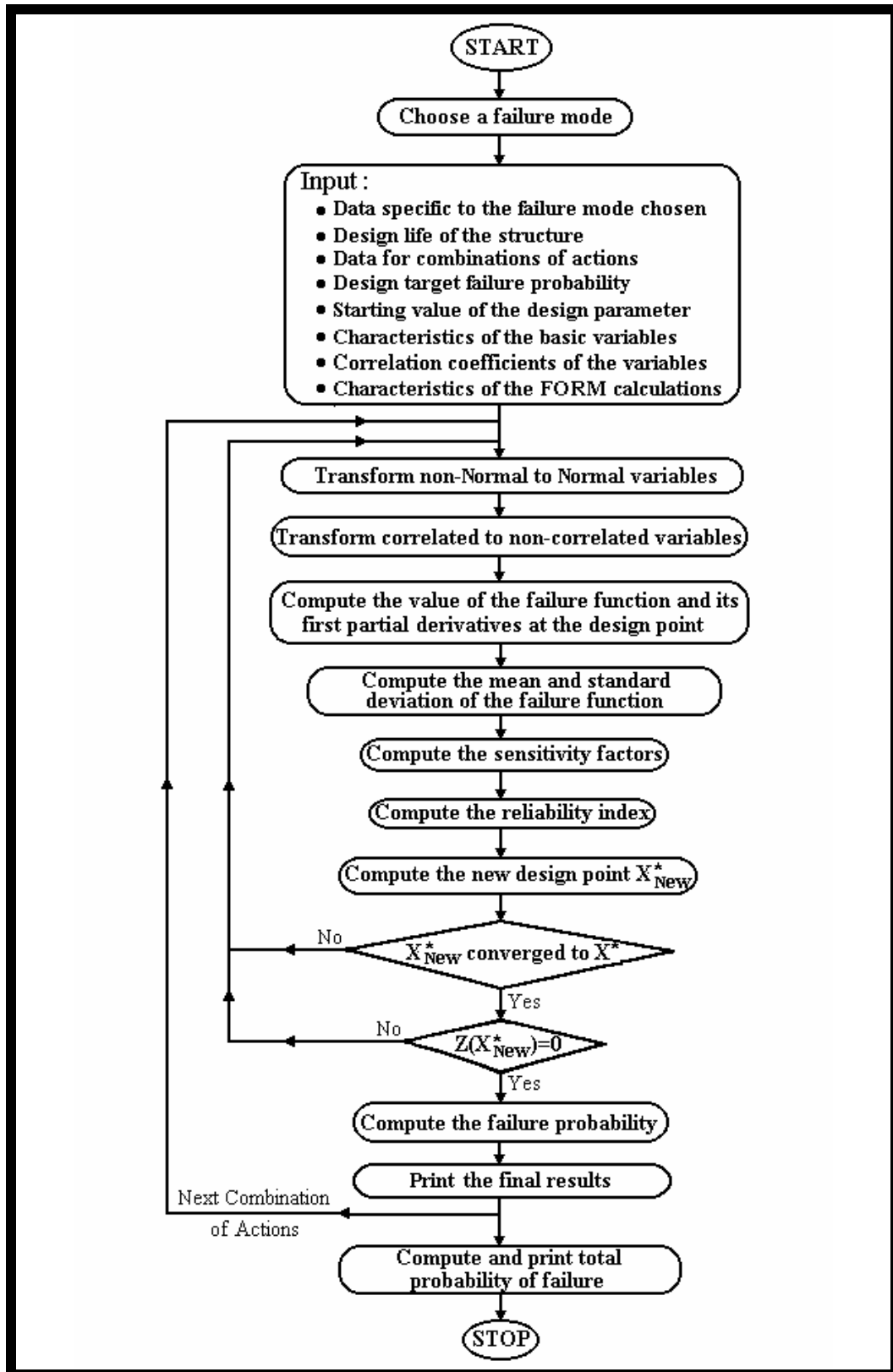
Figure C1.1: Simplified flowchart of PARASODE for Mode 1.

START

Choose a failure mode

Input :
- Data specific to the failure mode chosen
- Design life of the structure
- Design target failure probability
- Starting value of the design parameter
- Characteristics of the basic variables
- Correlation coefficients of the variables
- Characteristics of the FORM calculations

Transform non-Normal to Normal variables

Transform correlated to non-correlated variables

Compute the value of the failure function and its first partial derivatives at the design point

Compute the mean and standard deviation of the failure function

Compute the sensitivity factors

Compute the new design point X^*_{New}

No — X^*_{New} converged to X^*

Yes

No — $Z(X^*_{New})=0$

Calculation of a better estimate of the design parameter

Yes

Print the final results

STOP

Figure C1.2: Simplified flowchart of PARASODE for Mode 2.
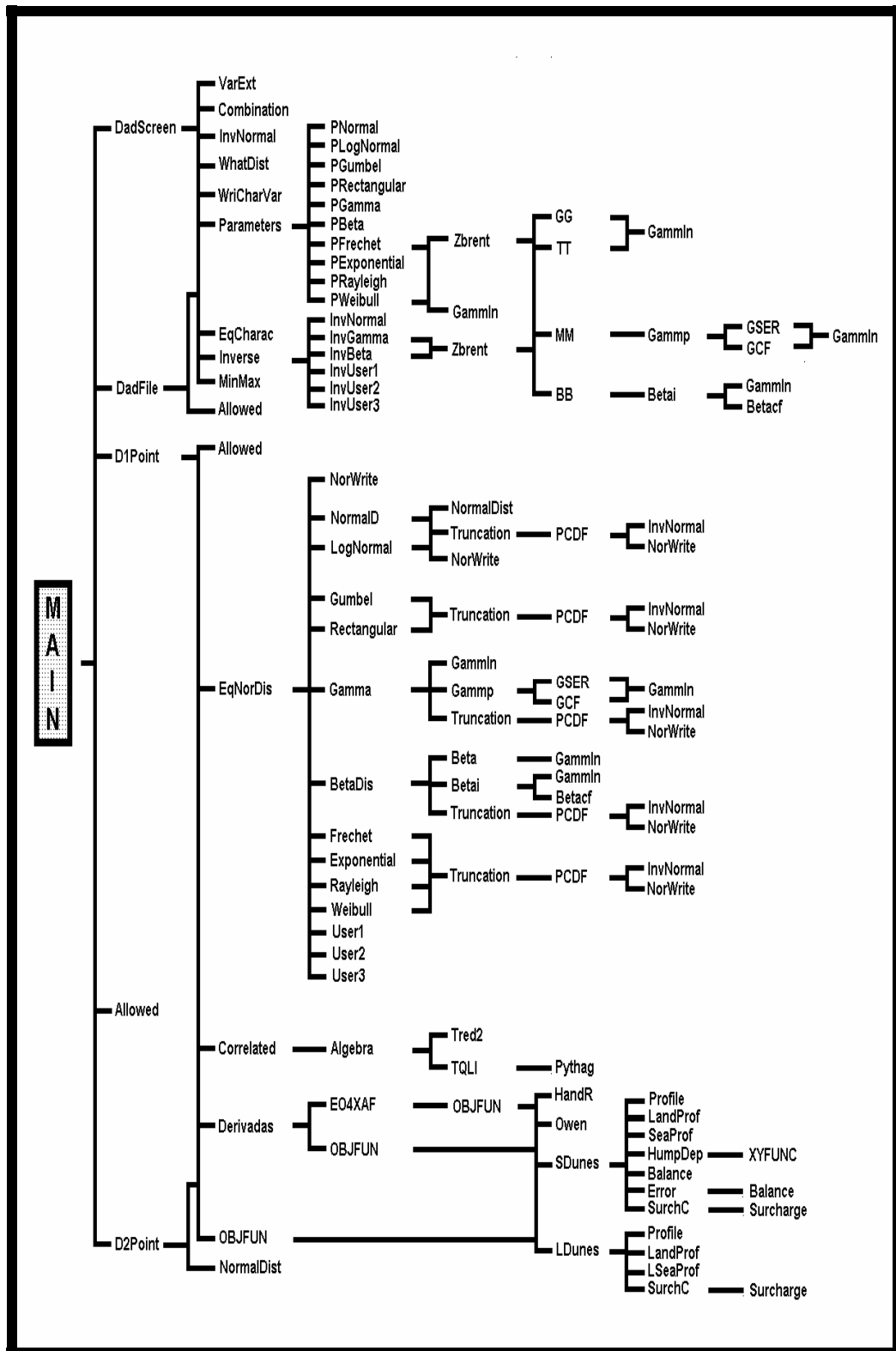
Figure C2.1: Flowchart showing the subroutines used in PARASODE.

APPENDIX C3 - Tables Of Pre-Defined Distributions

DISTRIBUTION	CDF	PDF	INVERSE OF CDF	RESTRICTIONS	SUBROUTINES
NORMAL	$\Theta\left(\dfrac{x-\zeta}{\lambda}\right)$	$\dfrac{1}{\sqrt{2\pi}\lambda}\exp\left[-\dfrac{1}{2}\left(\dfrac{x-\zeta}{\lambda}\right)^2\right]$	$\zeta + \lambda\Theta^{-1}[F_x]$	$\lambda > 0$	NormalD; NormalDist Inverse; InvNormal
LOG-NORMAL	$\Theta\left(\dfrac{\ln(x)-\zeta}{\lambda}\right)$	$\dfrac{1}{\sqrt{2\pi}\lambda x}\exp\left[-\dfrac{1}{2}\left(\dfrac{\ln(x)-\zeta}{\lambda}\right)^2\right]$	$\exp\left[\zeta + \lambda\Theta^{-1}[F_x]\right]$	$x > 0$ $\lambda > 0$	LogNormal; NormalDist Inverse; InvNormal
MAXIMA TYPE I (GUMBEL)	$\exp[-\exp[-\zeta(x-\lambda)]]$	$\zeta\exp[-\zeta(x-\lambda)]F_x$	$\lambda - \dfrac{\ln[-\ln[F_x]]}{\zeta}$	$\zeta > 0$	Gumbel Inverse
RECTANGULAR (UNIFORM)	$\dfrac{x-\zeta}{\lambda-\zeta}$	$\dfrac{1}{\lambda-\zeta}$	$\zeta + (\lambda-\zeta)F_x$	$\zeta \le x \le \lambda$ $\zeta < \lambda$	Rectangular Inverse
GAMMA	$\dfrac{\Gamma(\zeta,\lambda x)}{\Gamma(\zeta)}$	$\dfrac{\lambda^\zeta x^{\zeta-1}\exp(-\lambda x)}{\Gamma(\zeta)}$	No explicit form	$x \ge 0$ $\zeta > 0; \lambda > 0$	Gamma; Gammp; GSER GCF; Gammln; Inverse InvGamma; Zbrent MM
BETA	$\dfrac{\beta_u(\zeta,\lambda)}{\beta(\zeta,\lambda)}$	$\dfrac{(x-x_1)^{\zeta-1}(x_2-x)^{\lambda-1}}{\beta(\zeta,\lambda)(x_2-x_1)^{\zeta+\lambda-1}}$	No explicit form	$x_1 \le x \le x_2$ $x_1 < x_2$ $\zeta > 0; \lambda > 0$	BetaDis; Beta; Betai Gammln; Betacf Inverse; InvBeta Zbrent BB
MAXIMA TYPE II (FRECHET)	$\exp\left[-\left(\dfrac{\lambda}{x}\right)^\zeta\right]$	$\dfrac{\zeta}{\lambda}\left(\dfrac{\lambda}{x}\right)^{\zeta+1}\exp\left[-\left(\dfrac{\lambda}{x}\right)^\zeta\right]$	$\lambda\left[-\ln[F_x]\right]^{-1/\zeta}$	$x > 0$ $\zeta > 2; \lambda > 0$	Frechet Inverse
EXPONENTIAL	$1-\exp\left[-\left(\dfrac{x-\zeta}{\lambda}\right)\right]$	$\dfrac{1}{\lambda}\exp\left[-\left(\dfrac{x-\zeta}{\lambda}\right)\right]$	$\zeta - \lambda\ln[1-F_x]$	$x \ge \zeta$ $\lambda > 0$	Exponential Inverse
RAYLEIGH	$1-\exp\left[-\dfrac{x^2}{2\zeta^2}\right]$	$\dfrac{x}{\zeta^2}\exp\left[-\dfrac{x^2}{2\zeta^2}\right]$	$\zeta\sqrt{-2\ln[1-F_x]}$	$x \ge 0$ $\zeta > 0$	Rayleigh Inverse
MINIMA TYPE III (WEIBULL)	$1-\exp\left[-\left(\dfrac{x-\zeta}{\lambda}\right)^\eta\right]$	$\dfrac{\eta}{\lambda}\left(\dfrac{x-\zeta}{\lambda}\right)^{\eta-1}\exp\left[-\left(\dfrac{x-\zeta}{\lambda}\right)^\eta\right]$	$\zeta + \lambda\left[-\ln[1-F_x]\right]^{1/\eta}$	$x \ge \zeta$ $\lambda > 0; \eta > 0$	Weibull Inverse

Table C3.1: Cumulative distribution function, probability density function, inverse function and restrictions for the distributions available in PARASODE (the subroutines used for their calculation are also tabulated).

DISTRIBUTION	μ	σ	RESTRICTIONS
NORMAL	ζ	λ	$\lambda > 0$
LOG-NORMAL	$\exp\left(\zeta + \dfrac{\lambda^2}{2}\right)$	$\exp\left(\zeta + \dfrac{\lambda^2}{2}\right)\sqrt{\exp(\lambda^2) - 1}$	$X > 0$ $\lambda > 0$
MAXIMA TYPE I (GUMBEL)	$\lambda + \dfrac{0.57722}{\zeta}$	$\dfrac{\pi}{\sqrt{6}\,\zeta}$	$\zeta > 0$
RECTANGULAR (UNIFORM)	$\dfrac{\zeta + \lambda}{2}$	$\dfrac{\lambda - \zeta}{\sqrt{12}}$	$\zeta \leq X \leq \lambda$ $\zeta < \lambda$
GAMMA	$\dfrac{\zeta}{\lambda}$	$\dfrac{\sqrt{\zeta}}{\lambda}$	$X \geq 0$ $\zeta > 0; \lambda > 0$
BETA	$x_1 + \dfrac{\zeta}{\zeta + \lambda}(x_2 - x_1)$	$\dfrac{(x_2 - x_1)}{(\zeta + \lambda)}\sqrt{\dfrac{\zeta\lambda}{(\zeta + \lambda + 1)}}$	$x_1 \leq X \leq x_2$ $x_1 < x_2$ $\zeta > 0; \lambda > 0$
MAXIMA TYPE II (FRECHET)	$\lambda\Gamma\left(1 - \dfrac{1}{\zeta}\right)$	$\sqrt{\lambda^2\Gamma\left(1 - \dfrac{2}{\zeta}\right) - \left(\lambda\Gamma\left(1 - \dfrac{1}{\zeta}\right)\right)^2}$	$X > 0$ $\zeta > 2; \lambda > 0$
EXPONENTIAL	$\zeta + \lambda$	λ	$X \geq \zeta$ $\lambda > 0$
RAYLEIGH	$\zeta\sqrt{\dfrac{\pi}{2}}$	$\zeta\sqrt{2 - \dfrac{\pi}{2}}$	$X \geq 0$ $\zeta > 0$
MINIMA TYPE III (WEIBULL)	$\lambda\Gamma\left(1 + \dfrac{1}{\eta}\right) + \zeta$	$\sqrt{\lambda^2\Gamma\left(1 + \dfrac{2}{\eta}\right) - \lambda^2\Gamma^2\left(1 + \dfrac{1}{\eta}\right)}$	$X \geq \zeta$ $\lambda > 0; \eta > 0$

Table C3.2: Mean, standard deviation and restrictions for the distributions available in PARASODE.

DISTRIBUTION	ζ	λ	η	RESTRICTIONS	SUBROUTINES
NORMAL	μ	σ	--------	$\lambda > 0$	PNormal
LOG-NORMAL	$\ell n(\mu) - \dfrac{\lambda^2}{2}$	$\sqrt{\ell n\left(1 + \dfrac{\sigma^2}{\mu^2}\right)}$	--------	$x > 0$ $\lambda > 0$	PLogNormal
MAXIMA TYPE I (GUMBEL)	$\dfrac{\pi}{\sqrt{6}\,\sigma}$	$\mu - \dfrac{0.57722}{\zeta}$	--------	$\zeta > 0$	PGumbel
RECTANGULAR (UNIFORM)	$\mu - \sqrt{3}\sigma$	$\mu + \sqrt{3}\sigma$	--------	$\zeta \leq x \leq \lambda$ $\zeta < \lambda$	PRectangular
GAMMA	$\dfrac{\mu^2}{\sigma^2}$	$\dfrac{\mu}{\sigma^2}$	--------	$x \geq 0$ $\zeta > 0; \lambda > 0$	PGamma
BETA	$\dfrac{(x_1 - \mu)}{(x_2 - x_1)} + \dfrac{(x_2 - \mu)(\mu - x_1)^2}{\sigma^2(x_2 - x_1)}$	$\dfrac{(x_2 - x_1)\zeta}{(\mu - x_1)} - \zeta$	--------	$x_1 \leq x \leq x_2$ $x_1 < x_2$ $\zeta > 0; \lambda > 0$	PBeta
MAXIMA TYPE II (FRECHET)	No explicit form	$\dfrac{\mu}{\Gamma\left(1 - \dfrac{1}{\zeta}\right)}$	--------	$x > 0$ $\zeta > 2; \lambda > 0$	PFrechet Gamma Zbrent GG
EXPONENTIAL	$\mu - \sigma$	σ	--------	$x \geq \zeta$ $\lambda > 0$	PExponential
RAYLEIGH	$\mu\sqrt{\dfrac{2}{\pi}}$	--------	--------	$x \geq 0$ $\zeta > 0$	PRayleigh
MINIMA TYPE III (WEIBULL)	Given	$\left[\dfrac{\mu - \zeta}{\Gamma\left(1 + \dfrac{1}{\eta}\right)}\right]$	No explicit form	$x \geq \zeta$ $\lambda > 0; \eta > 0$	PWeibull Gamma Zbrent TT

Table C3.3: Parameters and restrictions for the distributions available in PARASODE (the subroutines used for their calculation are also tabulated)

APPENDIX C4 - User-Defined Distributions

C4.1 Observed Water Levels At Liverpool
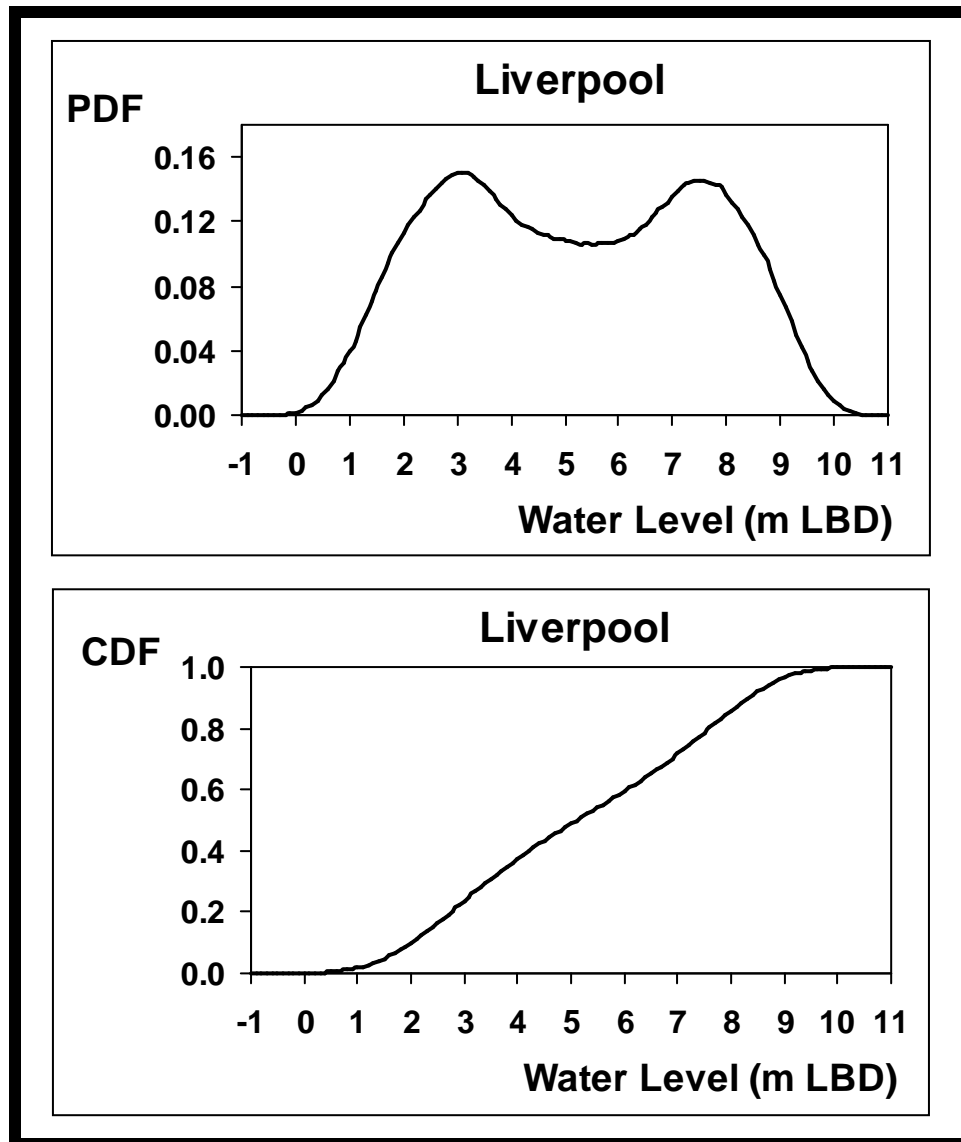
Figure C4.1: Probability density function and cumulative distribution function of observed water levels at Liverpool (LBD=Liverpool Bay Datum; OD=Ordnance Datum; OD=LBD-4.93m).

The PDF and CDF are based upon 45992 values collected into classes of 0.1m interval by Proudman Oceanographic Laboratory (POL), UK. Results have been smoothed using a nine-point moving average.
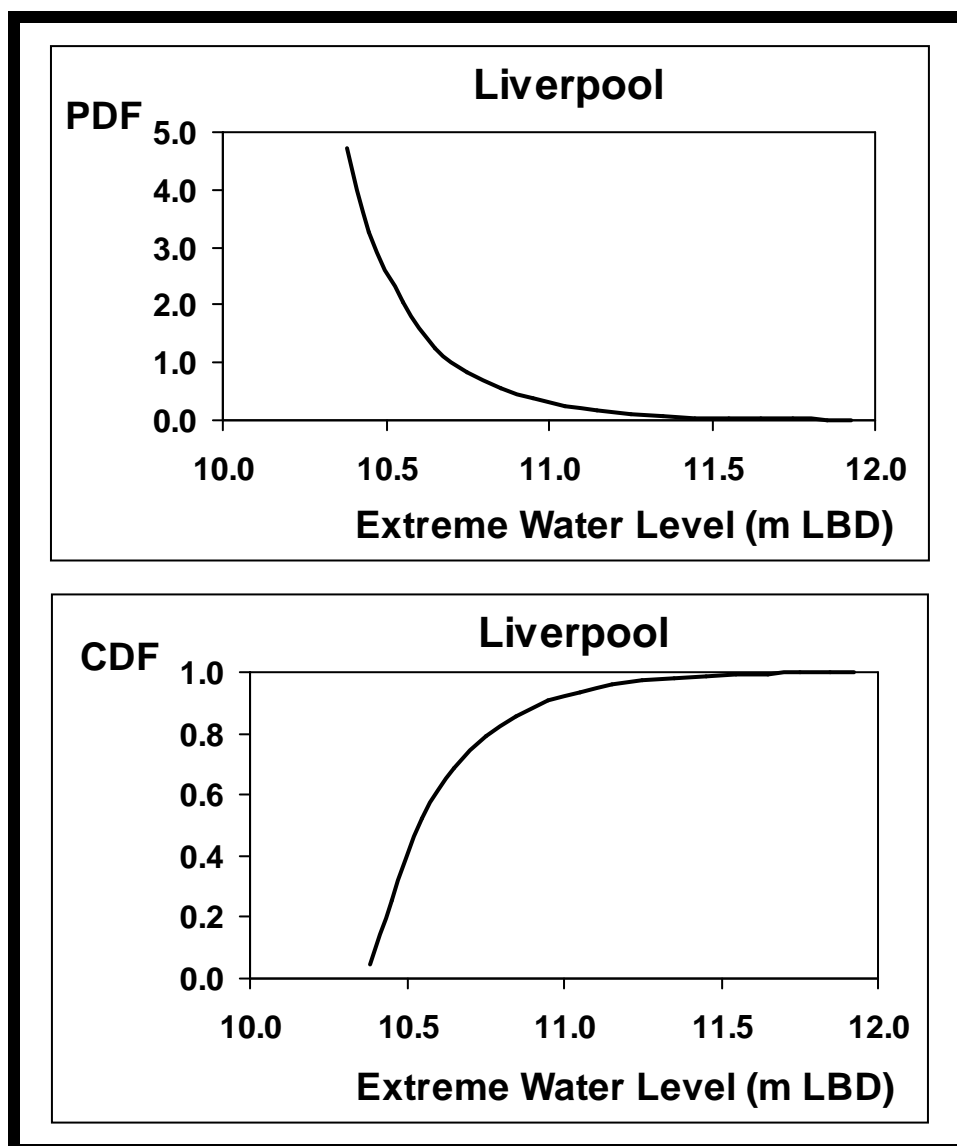
Figure C4.2: Probability density function and cumulative distribution function of observed extreme water levels at Liverpool (LBD=Liverpool Bay Datum; OD=Ordnance Datum; OD=LBD-4.93m).

The distributions of extreme water levels are based on information provided by Blackman (1997), POL.

C4.2   Predicted Tide Levels At Liverpool
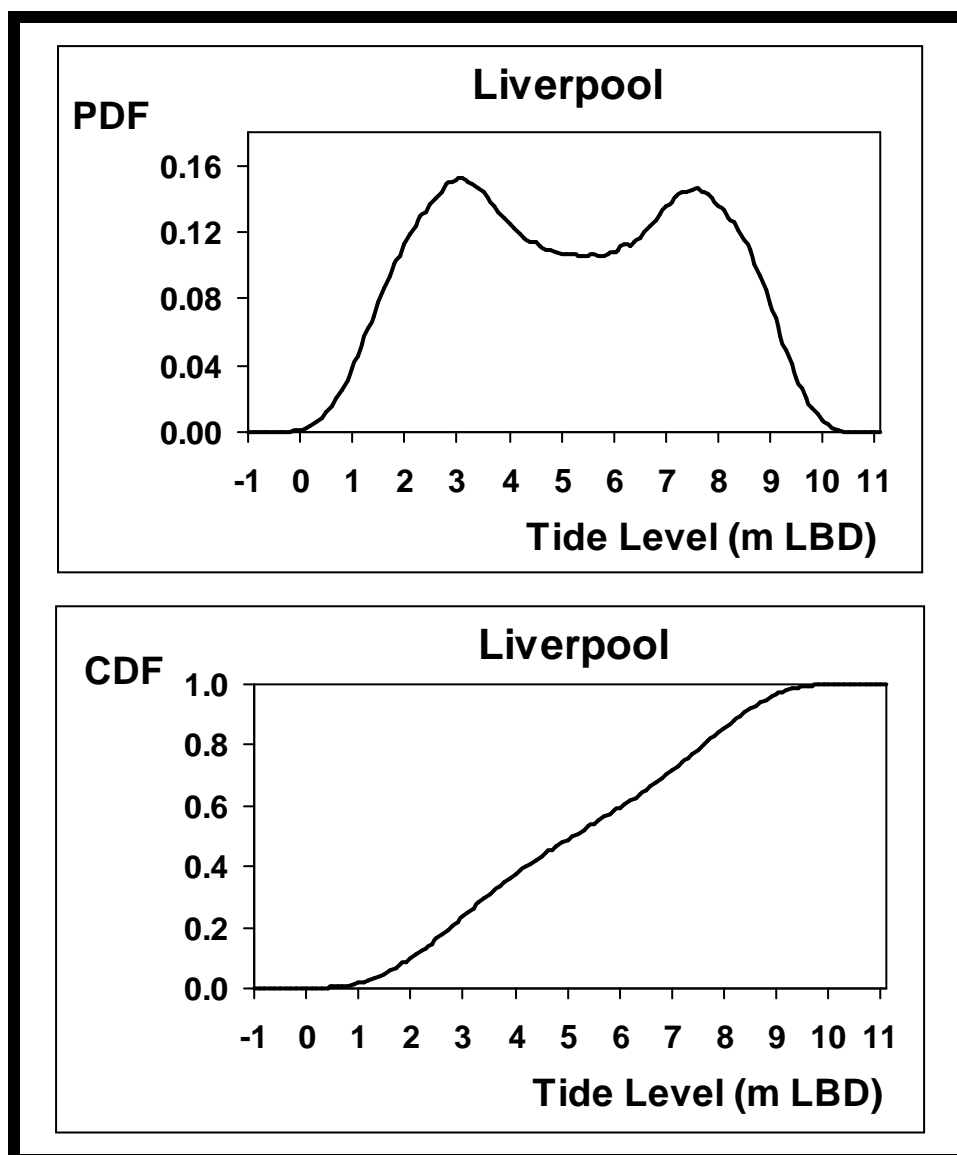
Figure C4.3: Probability density function and cumulative distribution function of predicted tide levels at Liverpool (LBD=Liverpool Bay Datum; OD=Ordnance Datum; OD=LBD-4.93m).

The PDF and CDF are based upon 47076 values collected into classes of 0.1m interval by POL. Results have been smoothed using a nine-point moving average.

APPENDIX C5 - Details Of Dune Erosion Calculations

In PARASODE, the methods used to calculate the areas of erosion and accretion and to test the required balance between these areas depend on the direction of the sand movements:

- If movements are allowed only seaward (see Figures C5.1 and C5.2), PARASODE:

    1) Calculates area C which lies between the surge level, the nourished profile above surge level and the gradient 1:md. Note that if (S1,T1)=(S3,T3) then area C is zero (see Figures C5.1(a) and C5.2(a)).

    2) Calculates area B which lies between the surge level and the gradient 1:mt. Note that if (S9,T9) is below the nourished profile, area B is zero (see Figure C5.2).

    3) Calculates area E which lies between points (S9,T9), (S2,T2), the surge level and the nourished profile. Note that if (S9,T9) is below the nourished profile, area E is zero (see Figure C5.2).

    4) Calculates the Y-coordinates, YPV, of Vellinga's points, (XPV,YPV), corresponding to the X-values of the points in the nourished profile; the total number of points in Vellinga's profile is NPV.

    5) Determines the intersection points between the two profiles. Where the nourished profile is above Vellinga's profile, it is considered a hump. A hump starts at (XHStart,YHStart) and finishes landward at (XHEnd,YHEnd). If the nourished profile is below Vellinga's profile, it is considered a depression. A depression starts at (XDStart,YDStart) and finishes landward at (XDEnd,YDEnd). The total number of humps is NumHump and the total number of depressions is NumDep.

    6) Calculates the areas of the humps, BH, and of the depressions, BD. The corresponding cumulative areas, SHump and SDep, from the seaward to the landward limits of the profiles, are also calculated. These areas include area C and area E-B. Note that if (S9,T9) is below the nourished profile (see Figure C5.2), since movements of sand are allowed only seaward, the first hump and its area have to be neglected, and the number of humps, together with areas BH and SHump have to be adjusted.

    7) Calculates the error, Err, in the balance between erosion and accretion, i.e. determines the difference in the areas of the humps and the areas of the depressions. The idea is that sand in a hump is used to fill the depressions situated seaward of the

       hump until it is depleted. A value for the absolute error of less than 1m³/m is required. If Err>1m³/m, then accretion exceeds erosion and Vellinga's profile has to be moved landward. If Err<-1m³/m, then erosion exceeds accretion and Vellinga's profile has to be moved seaward if $S8 \neq S1$; if S8=S1, no erosion of the nourished profile is expected (C=0).

8) If a balance has not yet been achieved, the new X-coordinate of the starting point of the parabolic part of the profile, S8, is calculated, along with the new points (S9,T9), (S2,T2), and (S3,T3). Calculations 1) to 7) are then repeated until a balance is achieved.

- If movements are allowed in both directions (see Figures C5.3 and C5.4), PARASODE:

  1) Calculates area A which lies between the surge level and the parabolic part of Vellinga's profile.

  2) Calculates area C which lies between the surge level, the nourished profile above surge level and the gradient 1:md. Note that if (S1,T1)=(S3,T3) then area C is zero (see Figures C5.3(a) and C5.4(a)).

  3) Calculates area B which lies between the surge level and the gradient 1:mt. Note that if (S9,T9) is below the nourished profile, area B is zero (see Figure C5.4).

  4) Calculates area Q which lies between the surge level and the nourished profile below surge level.

  5) Calculates the error, Err, in the balance between erosion and accretion (Err=Q-B-A-C). A value for the absolute error of less than 1m³/m is required. If Err>1m³/m, then accretion exceeds erosion and Vellinga's profile has to be moved landward. If Err<-1m³/m, then erosion exceeds accretion and Vellinga's profile has to be moved seaward if $S8 \neq S1$; if S8=S1, no erosion of the nourished profile is expected (C=0).

  6) If a balance has not yet been achieved, the new X-coordinate of the starting point of the parabolic part of the profile, S8, is calculated, along with the new points (S9,T9), (S2,T2), and (S3,T3). Calculations 1) to 5) are then repeated until a balance is achieved.

Once the final position of Vellinga's profile and erosion area C are known, a surcharge on C, TSurch, is required (see Figure C5.5) as follows:

$$TSurch = SurchEros + SurchLongT$$

(C5.1)

SurchEros is a surcharge on erosion area C which takes account of the effects of the storm surge duration, of the gust bumps, and of the accuracy of the computation (see Section 4.5.2). SurchLongT is a surcharge on erosion area C which accounts for the effect of a gradient in the longshore transport rate (according to eq.(4.9)). These surcharges are expressed as an additional movement, SurD, of the dune face. Note that TSurch can be positive (see Figure C5.5(a)) or negative (see Figure C5.5(b)) since the effects of the surge duration and of the accuracy of the computation can take positive or negative values.

The surcharge distance, SurD, is determined such that the area D between the two 1:md gradients, the surge level and the nourished profile, is equal to TSurch. An absolute error, Err1, of $1m^3/m$ in the difference between D and the absolute value of TSurch is allowed. If Err1>$1m^3/m$, then the surcharge distance is too big, and a new calculation of D is required for a smaller value of SurD. If Err1<-$1m^3/m$, then the surcharge distance is too small, and a new calculation of D is required for a bigger value of SurD.

The X-coordinate of the point of intersection between the surge level and the gradient 1:md of the surcharge is S10. The intersection point between the nourished profile and the surcharge gradient 1:md has coordinates (S4,T4).
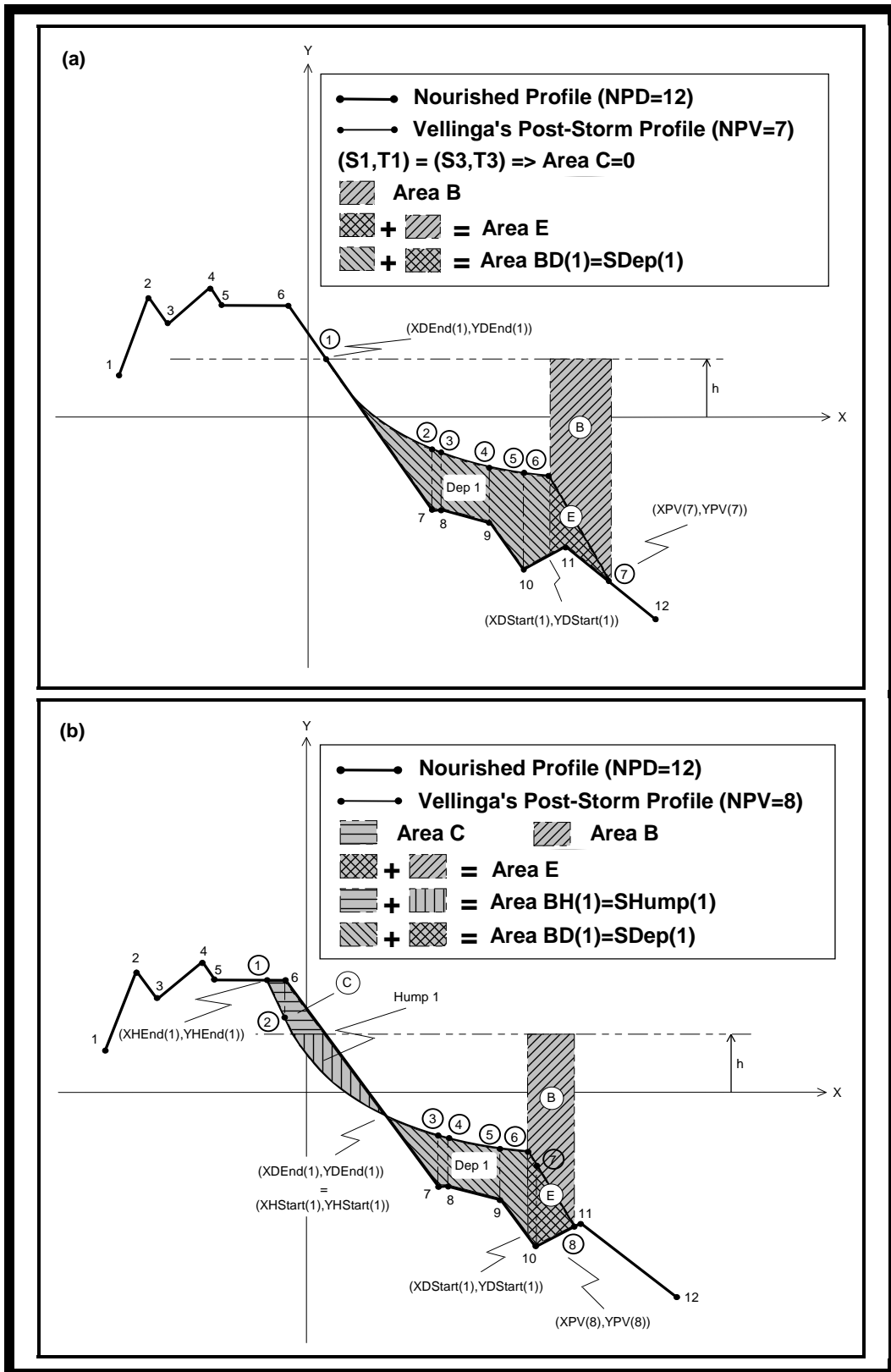
Figure C5.1: Notation used for sand balance when movements of sand are allowed only seaward - example 1.
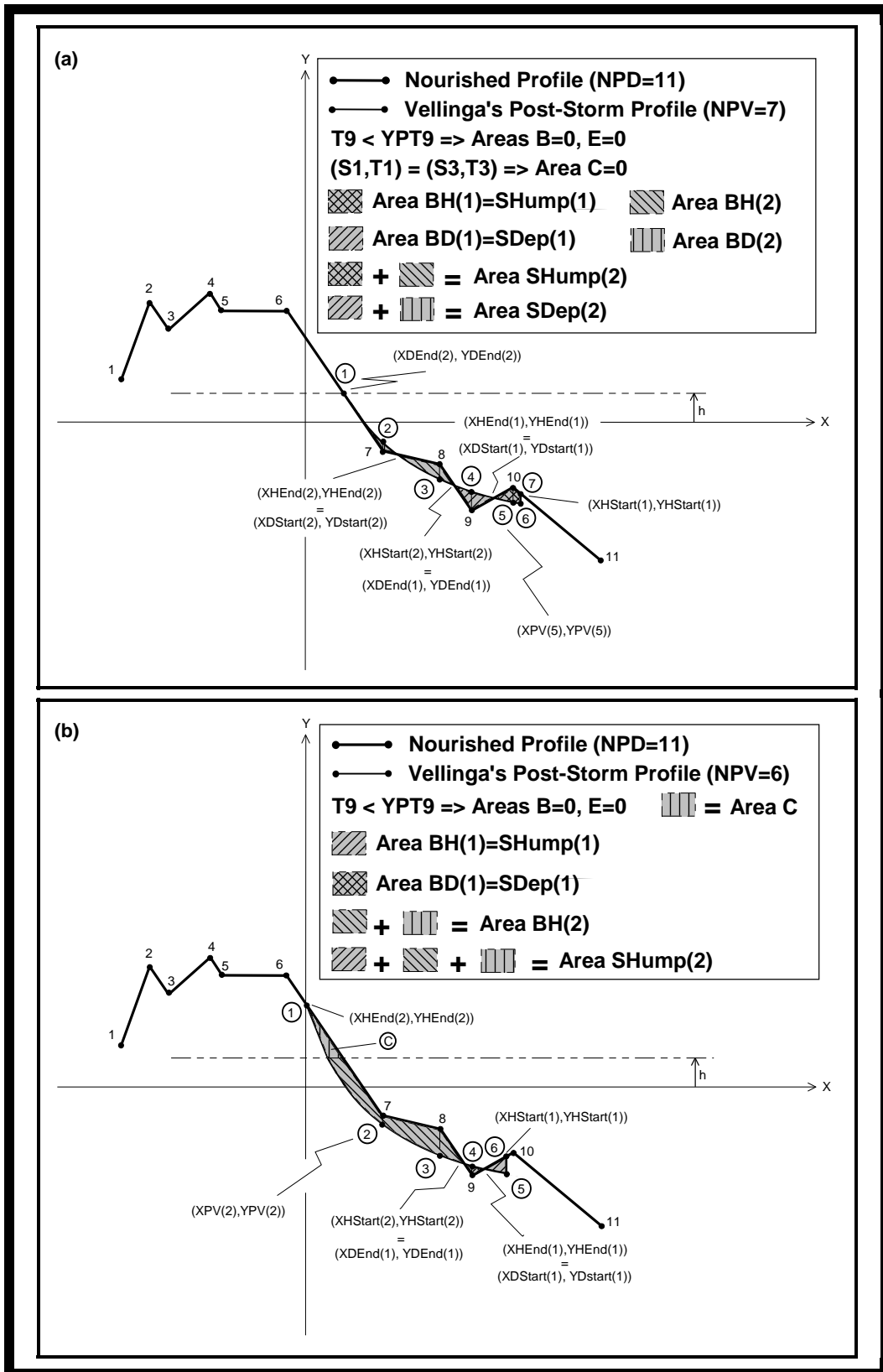
Figure C5.2: Notation used for sand balance when movements of sand are allowed only seaward - example 2.
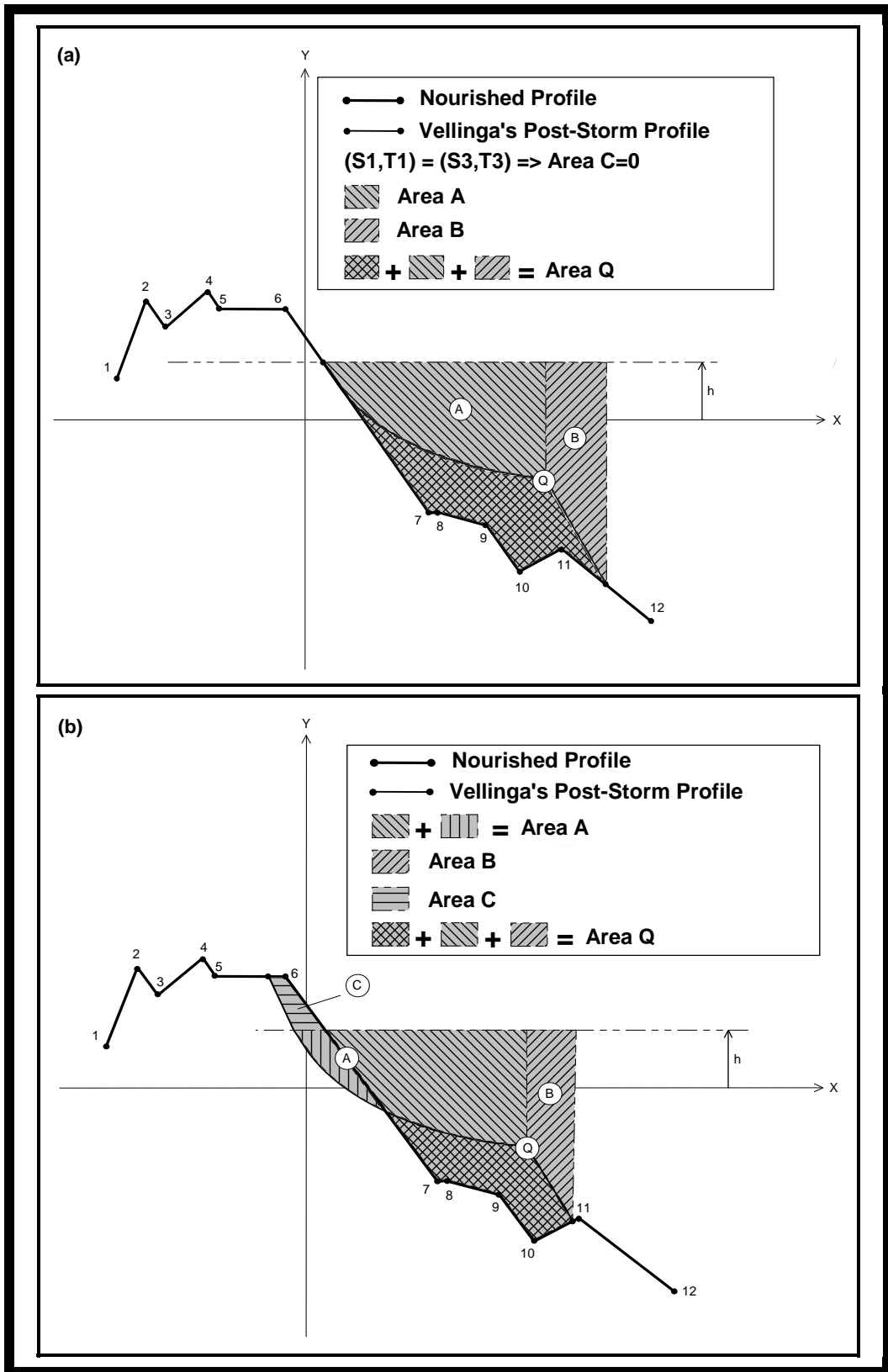
Figure C5.3: Notation used for sand balance when movements of sand are allowed seaward and landward - example 1.
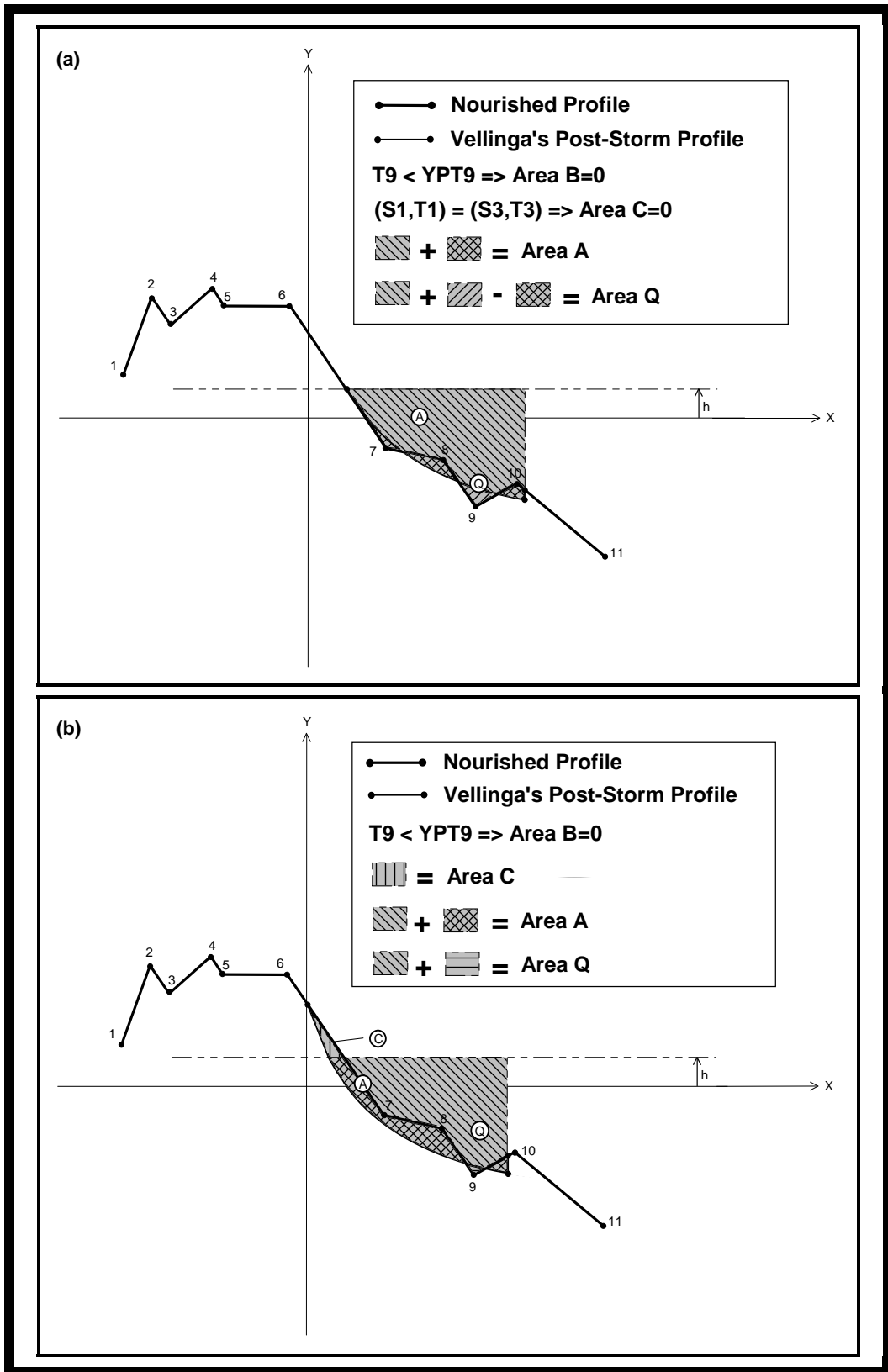
Figure C5.4: Notation used for sand balance when movements of sand are allowed seaward and landward - example 2.
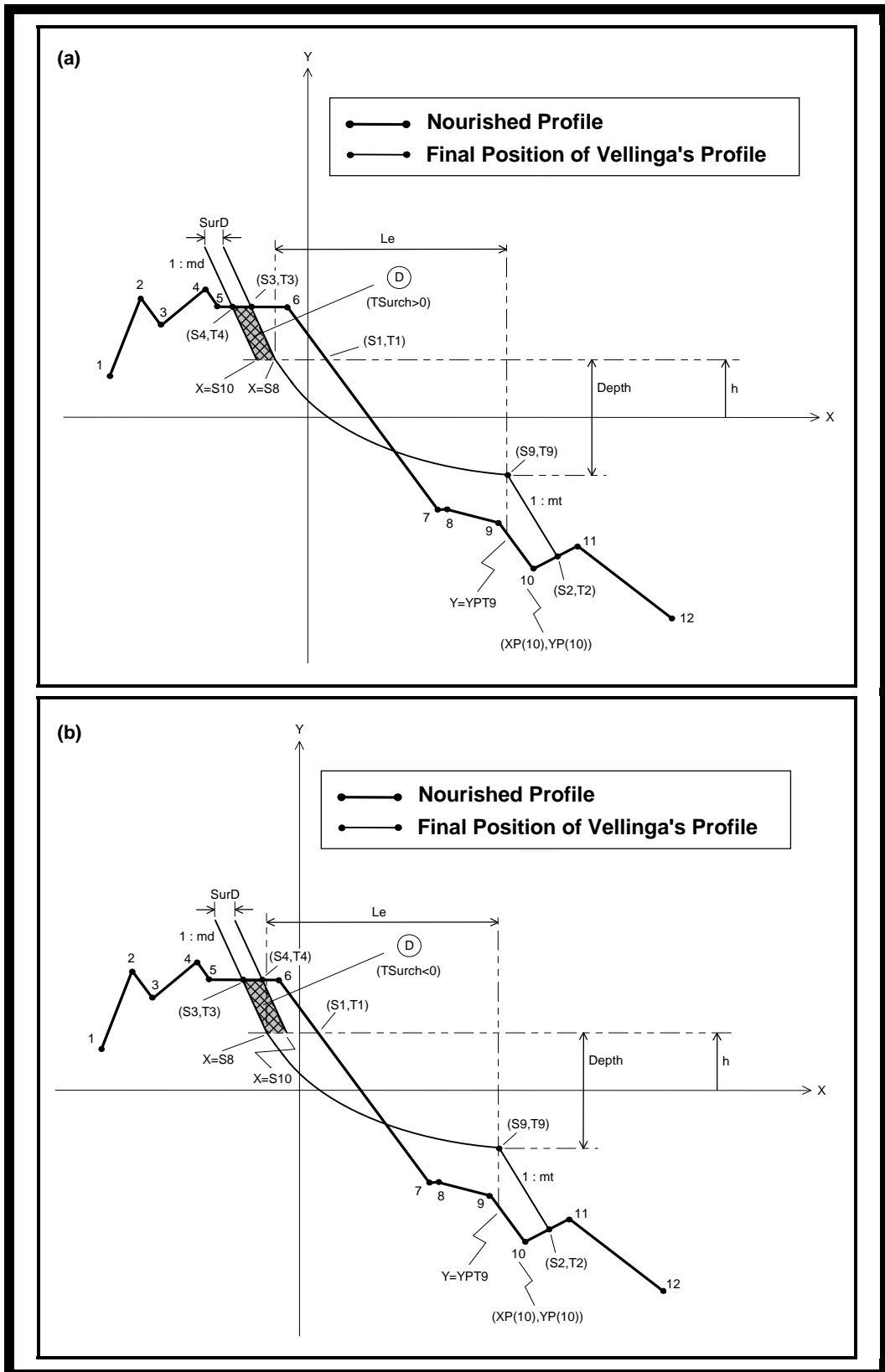
Figure C5.5: Surcharge on erosion area C.

APPENDIX C6 - Contents Of Input And Output Files

In all five input data files required by PARASODE, each piece of information has to be typed on a different line, except if otherwise specified. Data provided on the same line should be separated by a few blanks. The decimal places are separated by "." . In all input files, the character "!" is followed by a comment. It describes the input given on that line and the following lines, until the next comment appears. Examples of input files are given in Appendices D1 and D3.

PARASODE contains pre-set constants. The user must be aware of these values when providing input data. They are tabulated below.

PRE-SET CONSTANTS IN PARASODE	
Definition	**Limit**
Maximum number of FORM calculations	10
Maximum number of iterations in each FORM calculation	200
Maximum number of time-varying actions	5
Maximum number of combinations of actions	16
Maximum number of random variables describing the failure mode	15
Maximum number of points defining the initial dune profile	100
Maximum number of iterations to find the position of Vellinga's profile	999
Maximum number of iterations to find the position of the dune final face	999

Table C6.1: Pre-set constants in PARASODE.

Note that the program execution may be terminated at any time by pressing simultaneously the keys "Ctrl" and "Pause".

general.dad

The file *general.dad* should contain the following information in the order shown below.

- Choose the failure mode to be studied by giving a number as follows:

    1 - Overtopping (H&R)
    2 - Overtopping (Owen)
    3 - Dune Erosion (Vellinga)

- Choose how the still-water-level is defined by giving a number as follows:

    1 - Total level
    2 - Tide + Surge

- If "Overtopping (H&R)" is selected, choose the confidence value of the maximum run-up to be considered by giving a number as follows:

    1 - 37%
    2 - 99%

- If "Overtopping (H&R)" or "Overtopping (Owen)" are selected, choose the method of calculation of the first partial derivatives of the failure function by answering "Y" or "N" to the following question:

    "Are the first derivatives of the failure function supplied (Y/N) ?"

- If "Dune Erosion (Vellinga)" is selected, choose the direction of the sand movements occurring during a storm surge by giving a number as follows:

    1 - Movements of sand in both directions
    2 - Movements of sand only seaward

- Choose the purpose of the analysis by giving a number as follows:

    1 - Reliability analysis for a specified design   (Mode 1)
    2 - Design for a specified reliability level   (Mode 2)

- Input the design life of the structure (in years).

- If Mode 1 has been selected, choose whether combinations of actions are to be considered or not by answering "Y" or "N" to the following question:

    "Would you like to consider combinations of actions (Y/N) ?"

If "Y" is selected:

- Choose the number of combinations of actions to be considered by giving a number as follows:

  1 - The number of time-varying actions (k)
  2 - 2^{k-1}

- Choose which distributions should be provided for each combination of actions by giving a number as follows:

  1 - The basic distributions
  2 - The modified distributions

- Give the number of time-varying actions (the maximum number allowed by the program is 5).

- If the distributions provided are the basic distributions, give the number of time-varying actions and corresponding repetitions in the design life of the structure. For each time-varying action, these two pieces of data are given on the same line. The number of lines taken is equal to the number of time-varying actions and they are presented in increasing order of the number of repetitions.

- If the program is to be run for Mode 1, give the value of the design parameter for which the failure probability is to be found.

- If the program is to be run for Mode 2, give:

  - A starting value of the design parameter.

  - The target probability of failure.

form.dad

The file *form.dad* should contain the specifications required for a FORM calculation:

- If no combinations of actions are considered, choose the starting point for the FORM calculations by giving a number as follows:

  1 - Default values (mean values)
  2 - User specified values

  If "User specified values" are chosen, give the starting values of the random variables (deterministic variables should be ignored).

- If combinations of actions are considered and the distributions provided are the modified distributions, then either "Default values (mean values)" or "User specified values" should be supplied for each combination of actions.

- If no combinations of actions are considered, choose the limiting values for the variables by giving a number as follows:

  1 - Default values (+/- 1E25)
  2 - User specified values

  If "User specified values" are chosen, give, in the same line, the minimum value and the maximum value for each random variable (deterministic variables should be ignored).

- If combinations of actions are considered and the distributions specified are the basic ones, give the same information as above. The program assumes, subsequently, that the limiting values are the same for all combinations of actions.

- If combinations of actions are considered and the distributions specified are the modified ones, give the same information as above as many times as the number of combinations of actions considered.

- Give the maximum number of iterations allowed for each FORM calculation (the maximum number allowed by the program is 200).

- Give the number of FORM calculations to be performed (the maximum number allowed by the program is 10).

- Give the target values for the FORM calculations.

- Choose the required relative accuracy of the reliability index by giving a number as follows:

  1 - Default value (1%)
  2 - User specified value

  If "User specified value" is chosen, give a value within the range [0,1].

- Choose the required smoothing coefficient for the iteration process by giving a number as follows:

  1 - Default value (0)
  2 - User specified value

  If "User specified value" is chosen, give a value within the range [0,1].

- Choose the required accuracy of the failure function by giving a number as follows:

  1 - Default value (1%)
  2 - User specified value

  If "User specified value" is chosen, give a value within the range [0,1].

meandev.dad

The file *meandev.dad* should contain the characteristics of the random variables. For each variable, the following input is required:

- The type of distribution - Specify a number as follows:

  <u>Pre-Defined Distributions</u>

0	-	Deterministic	"Det"
1	-	Normal	"Nor"
2	-	Log-Normal	"LgN"
3	-	Maxima Type I (Gumbel)	"Gum"
4	-	Rectangular (Uniform)	"Uni"
5	-	Gamma	"Gam"
6	-	Beta	"Bet"
7	-	Maxima Type II (Frechet)	"Fre"
8	-	Exponential	"Exp"
9	-	Rayleigh	"Ray"
10	-	Minima Type III (Weibull)	"Wei"

  <u>User-Defined Distributions</u>

11	-	Water Levels	"UD1"
12	-	Extreme Water Levels	"UD2"
13	-	Tide Levels	"UD3"

  The abbreviations used in the program are given above in inverted commas.

- If the variable is deterministic, give its constant value.

- If the variable has a user-defined distribution, the mean and the standard deviation are required in the same line. Note that in this case, an ASCII data file has to be prepared containing the values of the variable, X_i, and the corresponding values of the PDF and the CDF. Intermediate values are obtained by linear interpolation.

- If the variable is not deterministic or it does not have a user-defined distribution, choose the type of truncation, by giving a number as follows:

0	-	Not truncated
1	-	Truncated for X above Xo
2	-	Truncated for X below Xo

- For the failure mode overtopping, if the variable is the significant wave height, the seawall toe level, TL, has to be given. In this case, if the distribution is truncated on the right side, Xo is calculated from eq. (5.4). If the distribution is truncated on the left side, give the value of Xo.

- For other truncated distributions, give the value of Xo.

- • If the variable is described by a Weibull distribution then the mean, the standard deviation and the lower limit have to be specified in this order, in the same line.

- • If the variable is described by a Beta distribution then the mean, the standard deviation, the lower limit and the upper limit have to be specified in this order, in the same line.

- • If the variable is not Weibull or Beta distributed, only the mean and the standard deviation are required, in the same line.

If combinations of actions are considered with modified distributions, then the input data for all the variables have to be repeated for each combination of actions. Alternatively, if basic distributions are given, the input data for all the variables are required only once. This is because the characteristics of the variables are the same for all combinations of actions; what differs in each combination is the power to which each distribution is raised (see Section 2.3.3.3).

coefcor.dad

The file *coefcor.dad* should contain the correlation coefficients, ρ_{ij}, of the random variables. For example, for i=1,...,3 and j=1,...,3, they should be listed in the following order:

ρ_{11}
ρ_{12}
ρ_{13}
ρ_{21}
ρ_{22}
ρ_{23}
ρ_{31}
ρ_{32}
ρ_{33}

perfil.dad

The file *perfil.dad* should contain the following dune erosion parameters:

- Coastal curvature in degrees per 1000m.

- Number of points defining the initial profile (the maximum number allowed by the program is 100).

- Beach profile coordinates, (XP,YP). The coordinates of each point should be provided in the same line, with the Y-coordinate following the X-coordinate. The coordinates should be given, starting at the most landward location and moving seaward. Note that the X values are positive seaward and the Y values are positive upward, with respect to the origin of coordinates.

- Number of points to be changed in the initial profile.

- Number of the first point to be changed.

- Gradient of the eroded dune face, 1:md (md should be provided).

- Gradient of the toe of the post-storm profile, 1:mt (mt should be provided).

- Nourishment top level.

- Gradient of the nourished face, 1:mnour (mnour should be provided).

```
C      ################################################################
C
C      PARASODE (Probabilistic Assessment of Risks Associated with
C               Seawall Overtopping & Dune Erosion)
C
C      ################################################################
C
C      PARASODE is a Level II FORTRAN 77 program which uses the First
C      Order Reliability Method (FORM) for assessing the safety of
C      coastal structures. In particular, it concentrates on the
C      potential failure mechanisms associated with wave overtopping of
C      seawalls & dune erosion. The amount of wave overtopping is
C      calculated by both the H&R equation & Owen's formula; dune
C      erosion is calculated using Vellinga's model. Although the
C      program incorporates these two specific failure mechanisms, the
C      majority of the code is generic & can be adapted to other types
C      of failure without undue difficulty.
C      PARASODE incorporates routines for transforming the correlated
C      variables to a set of non-correlated variables & for mapping
C      non-Normal distributions to equivalent Normal distributions.
C      PARASODE operates in two modes: 1) Mode 1, the analysis mode,
C      in which the failure probability is calculated for a given value
C      of the design parameter; 2) Mode 2: the design mode, in which
C      the value of a specific design parameter is calculated for a
C      target probability of failure. Mode 1 allows for combinations of
C      time-varying actions using the method of Ferry Borges &
C      Castanheta (1983).
C      SI units are used within the program, except if otherwise
C      specified.
C      The program execution may be terminated at any time by pressing,
C      simultaneously, the keys "Ctrl" and "Pause".
C
C      ################################################################
C
C      INPUT VARIABLES:
C      Optio - Data source
C      Opti - End the run of the program or restart calculations
C
C      MODELING VARIABLES:
C      Opt - Failure mode
C      TL - Seawall toe level
C      FDer - Method of calculation of the first partial derivatives of
C             the failure function for overtopping
C      DSWL - Definition of the SWL
C      Mode - Purpose of the analysis
C      Comb - Consideration or not of combination of actions
C      NCombAc, CombAc - Number of combinations of actions
C      Q - Maximum number of combinations of actions allowed by the
C          program
C      NR - Power to which each distribution is raised for each
C           combination of actions
C      Distr - Distributions provided for the combination of actions
C      N - Number of variables
C      L - Maximum number of variables allowed by the program
C      Ext - Abbreviation of the name of the variable
C      ExtExt - Description of the variable
C      ParamDesc - Description of the design parameter
C      Rho - Correlation coefficient
C      Mux - Mean of X
C      Sigmax - Standard deviation of X
C      VarDis - Type of distribution
C      Abrev - Abbreviation of the name of the distribution
C      Trunc - Type of truncation
C      Xo - Point of truncation (if the distribution is truncated)
C      XMin - Minimum value of X
C      XMax - Maximum value of X
C      StartPt - Starting value of the variables
C      MaxIter - Maximum number of iterations
C      NumCalc - Number of FORM calculations
C      L2 - Maximum number of FORM calculations allowed by the program
C      ReqBetaAcc - Required relative accuracy of the reliability index
C      Smooth - Smoothing coefficient for the iteration process
C      ReqOBJFAcc- Required accuracy of the failure function
C      Life - Design life of the structure
```

```
C        StartParam - For Mode=1, it is the prescribed value of the
C                      design parameter; For Mode=2, it is the starting
C                      value of the design parameter (from which the
C                      program iterates to find the required value of the
C                      design parameter)
C        Pf - Design target failure probability
C        RelInd - Reliability index which corresponds to Pf
C        Prob - Probability of failure for a specific FORM calculation
C        Pro - Probability of failure for a specific FORM calculation &
C             for a specific combination of actions
C        j, k0, Ex, Aux - Auxiliary variables
C        TR, Zeta, Lamda, Eta, x1, x2, NPD, t, NPch, NPDOld, MuxN, XP,
C        YP, md, mt, mnour, nourtlev, ctcurv, XPOld, YPOld, Ac, GB, SD,
C        It, AuxRstar, C, C1,
C        T3, OptC, OptD - Variables mentioned in the Common statements
C                         but not used here
C
C        OUTPUT VARIABLES:
C        PFTotal - Total probability of failure considering all the
C                  combinations of actions
C        Reliab - Reliability associated with PFTotal
C
C        ################################################################
C
         Implicit None
         Integer*4 j,k0,Q,L,L2,N,NPD,Mode,Opt,Optio,FDer,Life,t,NPch,
      1            NPDOld,OptD,Comb,CombAc,NCombAc,Distr,MaxIter,
      1            NumCalc,Aux,It,AuxRstar,Ex,DSWL
         Parameter (L=15)
         Parameter (L2=10)
         Parameter (Q=16)
         Character*1 Opti
         Character*3 Abrev(Q,L),Ext(L)
         Character*17 ExtExt(L)
         Character*19 ParamDesc
         Integer*4 VarDis(Q,L),Trunc(Q,L)
         Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L),Eta(Q,L),
      1        x1(Q,L),x2(Q,L),Rho(L,L),Prob(Q),PFTotal(L2),Reliab(L2),
      1        Pro(Q,L2),StartParam,Pf,RelInd,MuxN,XP,YP,md,mt,mnour,
      1        XPOld,YPOld,nourtlev,Ac,GB,SD,C,T3,ctcurv,TR(L2),
      1        StartPt(Q,L),Xo(Q,L),XMax(Q,L),XMin(Q,L),ReqBetaAcc,
      1        Smooth,ReqOBJFAcc,NR(Q,L),TL,C1
         Common/BLOCK1/NPD,XP(100),YP(100)
         Common/BLOCK2/t,NPch,XPOld(100),YPOld(100),NPDOld
         Common/BLOCK3/MuxN(15),C,T3
         Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
         Common/BLOCK5/Ac,GB,SD
         Common/BLOCK6/OptD
         Common/BLOCK7/TR
         Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
         Common/BLOCK9/k0,It,AuxRstar
         Common/BLOCK10/C1
         Common/BLOCK11/DSWL,TL

         Open(Unit=40, File='summary.dat', Status='unknown')
         Open(Unit=50, File='results.dat', Status='unknown')

C        ==================================================================
C        Definition of the data source.
C        ==================================================================
 80      Write(*,21)
         Write(40,21)
         Write(50,21)
 21      Format(/ 3X,'WHAT IS THE DATA SOURCE ? ' // 11X,
      1         'The Screen ..... [ 1 ]' / 11X,
      1         'A Datafile ..... [ 2 ]'/)
 878     Write(*,99)
 99      Format(3X,'Select Option: ',$)
         Read(*,*) Optio
         If (Optio.NE.1.AND.Optio.NE.2) goto 878
         Write(40,2387) Optio
         Write(50,2387) Optio
 2387    Format(3X,'Select Option: ',I1)

C        ==================================================================
C        Definition of the input data.
C        ==================================================================
         If (Optio.EQ.1) then
           Call Dadscreen(NCombAc,Distr,Comb,Mode,Opt,N,Mux,Sigmax,
      1                   VarDis,Abrev,Ext,ParamDesc,Trunc,Xo,MaxIter,
      1                   NumCalc,StartPt,XMax,XMin,ReqBetaAcc,Smooth,
      1                   ReqOBJFAcc,NR,Rho,Life,CombAc,StartParam,Pf,
      1                   RelInd,ExtExt,FDer)
```

```
         else
            Call Dadfile(NCombAc,Distr,Comb,Mode,Opt,N,Mux,Sigmax,VarDis,
     1                   Abrev,Ext,ParamDesc,Trunc,Xo,MaxIter,NumCalc,
     1                   StartPt,XMax,XMin,ReqBetaAcc,Smooth,ReqOBJFAcc,
     1                   NR,Rho,Life,CombAc,StartParam,Pf,RelInd,
     1                   ExtExt,FDer)
         Endif

C        ================================================================
C        Level II calculations for Mode 1 & Mode 2.
C        ================================================================
         If (Mode.EQ.1) then
            Do 3010 j=1,NCombAc
               If ((NCombAc).GT.1) then
                  Write(*,713) j
                  Write(40,713) j
                  Write(50,713) j
 713              Format(//// 3X,'COMBINATION No.',I2)
               Endif
               Do 5950 k0=1,NumCalc
                  Ex=0
                  Call D1Point(j,N,Opt,Mux,Sigmax,VarDis,Abrev,Rho,FDer,
     1                         NR,StartParam,Prob,Ext,ExtExt,ParamDesc,
     1                         Trunc,Xo,MaxIter,StartPt,XMax,XMin,
     1                         ReqBetaAcc,Smooth,ReqOBJFAcc,Comb,Ex)
                  If (Ex.EQ.0) then
                    Pro(j,k0)=Prob(j)
                    PAUSE
                  Endif
 5950          continue
 3010       continue
C           ----------
C           Calculation of the total probability of failure, PFTotal,
C           considering all the combinations of actions.
C           ----------
            If (Ex.EQ.0) then
               Do 6400 k0=1,NumCalc
                  PFTotal(k0)=0.
                  Do 6868 j=1,NCombAc
                   PFTotal(k0)=PFTotal(k0)+Pro(j,k0)
 6868             continue
                  Reliab(k0)=1.-PFTotal(k0)
                  Aux=1
                  Write(*,133)
                  Write(40,133)
                  Write(50,133)
 133              Format(/)
                  Call Allowed(Opt,Aux)
                  Write(*,68) Life,(100.*PFTotal(k0)),Reliab(k0)
                  Write(40,68) Life,(100.*PFTotal(k0)),Reliab(k0)
                  Write(50,68) Life,(100.*PFTotal(k0)),Reliab(k0)
 68               Format(3X,'DESIGN LIFE OF THE STRUCTURE = ',I3
     1                  / 3X,'TOTAL PROBABILITY OF FAILURE (%) = ',F10.6 /
     1                    3X,'RELIABILITY = ',E17.10)
 6400          continue
            Endif
         else
            Do 5960 k0=1,NumCalc
               j=1
               Call D2Point(j,N,Opt,Mux,Sigmax,VarDis,Abrev,Rho,FDer,
     1                      NR,StartParam,Pf,Ext,ExtExt,ParamDesc,Trunc,
     1                      Xo,RelInd,MaxIter,StartPt,XMax,XMin,Smooth,
     1                      ReqOBJFAcc,Comb)
               PAUSE
 5960       continue
         Endif

C        ================================================================
C        End of the main program.
C        ================================================================
 1122 Write(*,14)
 14   Format(// 3X,'WOULD YOU LIKE TO RESTART (Y/N) ? ',$)
      Read(*,2244) Opti
 2244 Format(A1)
      If (Opti.NE.'Y'.AND.Opti.NE.'y'.AND.
     1    Opti.NE.'N'.AND.Opti.NE.'n') goto 1122
      Write(*,5677)
 5677 Format(///)
      Write(40,256) Opti
      Write(50,256) Opti
 256  Format(/// 3X,'WOULD YOU LIKE TO RESTART (Y/N) ? ',A1 ///)
      If (Opti.EQ.'Y'.OR.Opti.EQ.'y') goto 80
      End
```

```
C        ###############################################################
C
         Subroutine Dadscreen(NCombAc,Distr,Comb,Mode,Opt,N,Mux,Sigmax,
     1                        VarDis,Abrev,Ext,ParamDesc,Trunc,Xo,
     1                        MaxIter,NumCalc,StartPt,XMax,XMin,
     1                        ReqBetaAcc,Smooth,ReqOBJFAcc,NR,Rho,Life,
     1                        CombAc,StartParam,Pf,RelInd,ExtExt,FDer)
C
C        ###############################################################
C
C        Reads the required input data from the screen.
C
C        ###############################################################
C
C        MODELING VARIABLES:
C        OptC - Confidence value of the maximum run-up
C        Der - Method of calculation of the first partial derivatives of
C              the failure function for overtopping
C        CAcc - Consideration or not of combination of actions
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        NumTVAc - Number of time-varying actions
C        r - Repetitions of each action in the design life
C        TVAc - Number of the time-varying actions in increasing order of
C               the number of repetitions
C        L - Maximum number of variables allowed by the program
C        Carac - Name of the distribution
C        Def1 - Definition of the limiting values of X
C        Def - Definition of the starting point for the FORM calculations
C        L2 - Maximum number of FORM calculations allowed by the program
C        Def2 - Definition of the required relative accuracy of the
C               reliability index
C        Def3 - Definition of the required smoothing coefficient for the
C               iteration process
C        Def4 - Definition of the required accuracy of the failure
C               function
C        i, j, k, k0, Aux - Auxiliary variables
C        It, AuxRstar - Variables mentioned in the Common statements but
C                       not used here
C
C        OUTPUT VARIABLES:
C        Opt - Failure mode
C        DSWL - Definition of the SWL
C        OptD - Direction of the sand movements in dune erosion
C        C1 - Parameter used in the H&R model to calculate C; it depends
C             on the confidence value assigned to the maximum run-up
C        TL - Seawall toe level
C        Mode - Purpose of the analysis
C        FDer - Method of calculation of the first partial derivatives of
C               the failure function for overtopping
C        Comb - Consideration or not of combination of actions
C        NCombAc, CombAc - Number of combinations of actions
C        NR - Power to which each distribution is raised for each
C             combination of actions
C        Distr - Distributions provided for the combination of actions
C        N - Number of variables
C        Ext - Abbreviation of the name of the variable
C        ExtExt - Description of the variable
C        ParamDesc - Description of the design parameter
C        Rho - Correlation coefficient
C        Mux - Mean of X
C        Sigmax - Standard deviation of X
C        VarDis - Type of distribution
C        Abrev - Abbreviation of the name of the distribution
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        XMin - Minimum value of X
C        XMax - Maximum value of X
C        Zeta, Lamda, Eta - Parameters of a distribution
C        x1 - Lower limit on X for a Beta distribution
C        x2 - Upper limit on X for a Beta distribution
C        StartPt - Starting value of the variables
C        MaxIter - Maximum number of iterations
C        NumCalc - Number of FORM calculations
C        TR - Target values for each FORM calculation
C        ReqBetaAcc - Required relative accuracy of the reliability index
C        Smooth - Smoothing coefficient for the iteration process
C        ReqOBJFAcc- Required accuracy of the failure function
C        Life - Design life of the structure
C        StartParam - For Mode=1, it is the prescribed value of the
```

```
C                      design parameter; For Mode=2, it is the starting
C                      value of the design parameter (from which the
C                      program iterates to find the required value of the
C                      design parameter)
C       Pf - Design target failure probability
C       RelInd - Reliability index which corresponds to Pf
C       ctcurv - Coastal curvature in degrees per 1000m
C       NPD, NPDOld - Number of points defining the initial profile
C       (XP,YP), (XPOld,YPOld) - Coordinates of the points defining the
C                                initial profile
C       NPch - Number of points to be changed in the initial profile
C       t - First point to be changed in the initial profile, point no.t
C       1:md - Gradient of the eroded dune face
C       1:mt - Gradient of the toe of the post-storm profile
C       nourtlev - Nourishment top level
C       1:mnour - Gradient of the nourished face
C
C       ################################################################
C
        Integer*4 i,j,k,k0,L,N,NPD,Mode,Opt,NCombAc,Q,t,NPch,NPDOld,Aux,
     1            MaxIter,NumCalc,Def,Def1,Def2,Def3,Def4,Comb,Distr,L2,
     1            Life,CombAc,NumTVAc,It,AuxRstar,OptC,OptD,FDer,DSWL
        Parameter (L=15)
        Parameter (L2=10)
        Parameter (Q=16)
        Character*3 Abrev(Q,L),Ext(L)
        Character*1 Der,CAcc
        Character*17 ExtExt(L)
        Character*30 Carac(Q,L)
        Character*19 ParamDesc
        Integer*4 VarDis(Q,L),Trunc(Q,L),TVAC(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L),Eta(Q,L),
     1         x1(Q,L),x2(Q,L),XP,YP,md,mt,mnour,XPOld,YPOld,nourtlev,
     1         ctcurv,TR(L2),StartPt(Q,L),Xo(Q,L),XMax(Q,L),XMin(Q,L),
     1         ReqBetaAcc,Smooth,ReqOBJFAcc,NR(Q,L),Rho(L,L),StartParam,
     1         Pf,RelInd,TL,r(L),C1
        Common/BLOCK1/NPD,XP(100),YP(100)
        Common/BLOCK2/t,NPch,XPOld(100),YPOld(100),NPDOld
        Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
        Common/BLOCK6/OptD
        Common/BLOCK7/TR
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
        Common/BLOCK9/k0,It,AuxRstar
        Common/BLOCK10/C1
        Common/BLOCK11/DSWL,TL

        Aux=0

C       ==================================================================
C       Definition of the failure mode to be studied.
C       ==================================================================
        Write(*,2)
        Write(40,2)
        Write(50,2)
 2      Format(// 3X,'WHAT IS THE FAILURE MODE TO BE STUDIED: '//
     1         11X, 'Overtopping (H&R) ................... [ 1 ]' /
     1         11X, 'Overtopping (Owen) ................. [ 2 ]' /
     1         11X, 'Dune Erosion (Vellinga) ............ [ 3 ]' /)
 9778   Write(*,97)
 97     Format(3X,'Select Option: ',$)
        Read(*,*) Opt
        If (Opt.NE.1.AND.Opt.NE.2.AND.Opt.NE.3) goto 9778
        Write(40,9779) Opt
        Write(50,9779) Opt
 9779   Format(3X,'Select Option: ',I2)


C       ----------
C       Definition of the still-water-level.
C       ----------
        If (Opt.EQ.1.OR.Opt.EQ.2.OR.Opt.EQ.3) then
          Write(*,9320)
          Write(40,9320)
          Write(50,9320)
 9320     Format(// 3X,'HOW IS THE STILL-WATER-LEVEL DEFINED ?'
     1           // 11X, 'Total Level .... [ 1 ]' /
     1              11X, 'Tide + Surge ... [ 2 ]' /)
 3925     Write(*,9473)
 9473     Format(3X,'Select Option: ',$)
          Read(*,*) DSWL
          If (DSWL.NE.1.AND.DSWL.NE.2) goto 3925
          Write(40,1437) DSWL
          Write(50,1437) DSWL
 1437     Format(3X,'Select Option: ',I1)
```

```
        Endif

C       ----------
C       Definition of the confidence value of the maximum run-up to be
C       considered.
C       ----------
        If (Opt.EQ.1) then
          Write(*,1324)
          Write(40,1324)
          Write(50,1324)
 1324     Format(// 3X,'WHAT IS THE CONFIDENCE VALUE OF THE MAXIMUM',
     1            1X,'RUN-UP' / 3X,'THAT YOU WOULD LIKE TO CONSIDER ?'
     1            // 11X, '37 % ... [ 1 ]' /
     1               11X, '99 % ... [ 2 ]' /)
 3225     Write(*,9373)
 9373     Format(3X,'Select Option: ',$)
          Read(*,*) OptC
          If (OptC.NE.1.AND.OptC.NE.2) goto 3225
          Write(40,1037) OptC
          Write(50,1037) OptC
 1037     Format(3X,'Select Option: ',I1)
          If (OptC.EQ.1) then
            C1=1.52
          else
            C1=2.15
          Endif
        Endif

C       ----------
C       Definition of the method of calculation of the first partial
C       derivatives of the failure function for overtopping.
C       ----------
 2255   If (Opt.NE.3) then
          Write(*,333)
 333      Format(// 3X,'ARE THE FIRST DERIVATIVES OF THE FAILURE',1X,
     1            'FUNCTION SUPPLIED (Y/N) ? ',$)
          Read(*,3377) Der
 3377     Format(A1)
          If (Der.NE.'Y'.AND.Der.NE.'y'.AND.Der.NE.'N'.AND.Der.NE.'n')
     1        goto 2255
          Write(40,6681) Der
          Write(50,6681) Der
 6681     Format(// 3X,'ARE THE FIRST DERIVATIVES OF THE FAILURE',1X,
     1            'FUNCTION SUPPLIED (Y/N) ? ',A1)
          If ((Der.EQ.'Y').OR.(Der.EQ.'y')) FDer=1
          If ((Der.EQ.'N').OR.(Der.EQ.'n')) FDer=2
        else
          FDer=2
        Endif

C       ----------
C       Definition of the direction of the sand movements occurring
C       during a storm surge.
C       ----------
        OptD=0
        If (Opt.EQ.3) then
          Write(*,1329)
          Write(40,1329)
          Write(50,1329)
 1329     Format(// 3X,'DURING A STORM SURGE, WOULD YOU LIKE TO TAKE',
     1            1X,'INTO ACCOUNT: '//
     1            11X, 'Movements of Sand in Both Directions ?',
     1            1X,'... [ 1 ]' / 11X, 'Movements of Sand',
     1            1X,'only Seaward ? ........ [ 2 ]' /)
 3115     Write(*,937)
 937      Format(3X,'Select Option: ',$)
          Read(*,*) OptD
          If (OptD.NE.1.AND.OptD.NE.2) goto 3115
          Write(40,1937) OptD
          Write(50,1937) OptD
 1937     Format(3X,'Select Option: ',I1)
        Endif

C       ----------
C       If the failure mode is dune erosion the following quantities
C       have to be read:
C       - coastal curvature in degrees per 1000m, ctcurv
C       - number of points defining the initial profile, NPD
C       - beach profile coordinates, (XP,YP)
C       - number of points to be changed in the initial profile, NPch
C       - first point to be changed, point no. t
C       - gradient of the eroded dune face, 1:md
C       - gradient of the toe of the post-storm profile, 1:mt
```

```
C       - nourishment top level, nourtlev
C       - gradient of the nourished face, 1:mnour
C       ----------
        If (Opt.EQ.3) then
 8667     Write(*,7668)
 7668     Format(// 3X,'DUNE EROSION' // 11X,
     1          'Coastal Curvature (Deg/1000m) = ',$)
          Read(*,*) ctcurv
          If (ctcurv.LT.0.OR.ctcurv.GT.24) then
            Write(*,3278)
 3278       Format(// 3X,'ERROR: The Coastal Curvature is not',1X,
     1             'Within [0,24] !')
            goto 8667
          Endif
          Write(40,7669) ctcurv
          Write(50,7669) ctcurv
 7669     Format(// 3X,'DUNE EROSION' // 11X,
     1          'Coastal Curvature (Deg/1000m) = ',E17.10)
 7070     Write(*,7700)
 7700     Format(/ 11X,'Number of Points Defining the Initial',1X,
     1          'Profile (Max=100) = ',$)
          Read(*,*) NPD
          If (NPD.GT.100.OR.NPD.LE.0) then
            Write(*,3251)
 3251       Format(// 3X,
     1            'ERROR: The Maximum Number of Points Defining the' /
     1            3X,'        Initial Profile is not Within ]0,100] !')
            goto 7070
          Endif
          Write(40,7701) NPD
          Write(50,7701) NPD
 7701     Format(/ 11X,'Number of Points Defining the Initial',1X,
     1          'Profile (Max=100) = ',I3)
          Write(*,0211)
 0211     Format(/ 23X, 'Initial Profile' /)
          Write(40,0511)
          Write(50,0511)
 0511     Format(/ 23X, 'Initial Profile' // 19X,'X',18X,'Y' /)
          Do 3120 i=1,NPD
 9090       Write(*,0311) i
 0311       Format(26X,'X(',I3,') = ',$)
            Read(*,*) XP(i)
            If (i.GE.2) then
              If (XP(i).LT.XP(i-1)) then
                Write(*,1466) i,(i-1)
 1466           Format(// 3X,'ERROR: XP(',I3,') < XP(',I3,') !' /)
                goto 9090
              Endif
            Endif
            Write(*,0317) i
 0317       Format(26X,'Y(',I3,') = ',$)
            Read(*,*) YP(i)
            Write(40,0011) XP(i),YP(i)
            Write(50,0011) XP(i),YP(i)
 0011       Format(11X,E17.10,2X,E17.10)
            XPOld(i)=XP(i)
            YPOld(i)=YP(i)
 3120     continue
          NPDOld=NPD

 9072     Write(*,5325)
 5325     Format(/ 11X,'Number of Points to be Changed in the Initial',
     1          1X,'Profile ? ',$)
          Read(*,*) NPch
          If (NPch.LT.1.OR.NPch.GT.NPD) then
            Write(*,1366) NPD
 1366       Format(// 3X,'ERROR: The Number of Points to be Changed',
     1            1X,'in the'/10X,'Initial Profile is not Within [1,',
     1            I3,'] !')
            goto 9072
          Endif
 6305     Write(*,5036)
 5036     Format(/ 11X,'First Point to be Changed ? Point No.',$)
          Read(*,*) t
          If (t.LT.1.OR.t.GT.NPD) then
            Write(*,1322) NPD
 1322       Format(// 3X,'ERROR: The First Point to be Changed in the' /
     1            3X,'        Initial Profile is not Within [1,',
     1            I3,'] !')
            goto 6305
          Endif
          Write(40,7917) NPch,t
          Write(50,7917) NPch,t
```

```
 7917     Format(/ 11X,'Number of Points to be Changed in the Initial',
    1              1X,'Profile = ',I3 // 11X,'First Point to be Changed',
    1              1X,'= Point No.',I3)

 4546      Write(*,3007)
 3007      Format(/ 11X,'Gradient of the Eroded Dune Face = 1:',$)
           Read(*,*) md
           If (md.LE.0) then
             Write(*,1311)
 1311       Format(// 3X,'ERROR: The Gradient of the Eroded Dune',
    1              1X,'Face is <= 1:0 !')
             goto 4546
           Endif

 4547      Write(*,4396)
 4396      Format(/ 11X,'Gradient of the Toe of the Post-Storm Profile',
    1              1X,'= 1:',$)
           Read(*,*) mt
           If (mt.LE.0) then
             Write(*,1151)
 1151       Format(// 3X,'ERROR: The Gradient of the Toe of the',
    1              1X,'Post-Storm Profile is <= 1:0 !')
             goto 4547
           Endif

           Write(*,3407)
 3407      Format(/ 11X,'Nourishment Top Level = ',$)
           Read(*,*) nourtlev

 5599      Write(*,4346)
 4346      Format(/ 11X,'Gradient of the Nourished Face = 1:',$)
           Read(*,*) mnour
           If (mnour.LE.0) then
             Write(*,7103)
 7103       Format(// 3X,'ERROR: The Gradient of the Nourished',
    1              1X,'Face is <= 1:0 !')
             goto 5599
           Endif

           Write(40,7933) md,mt,nourtlev,mnour
           Write(50,7933) md,mt,nourtlev,mnour
 7933      Format(/ 11X,'Gradient of the Eroded Dune Face = 1:',F4.1
    1              // 11X,'Gradient of the Toe of the Post-Storm Profile',
    1              1X,'= 1:',F4.1 // 11X,'Nourishment Top Level = ',E17.10
    1              // 11X,'Gradient of the Nourished Face = 1:',F4.1)
         Endif

C        ================================================================
C        Definition of the number, N, of variables for each failure mode.
C        ================================================================
         If (Opt.EQ.2.OR.Opt.EQ.3) then
           N=7
         else
           N=8
         Endif

         If (DSWL.EQ.2) then
           If (Opt.EQ.2.OR.Opt.EQ.3) N=8
           If (Opt.EQ.1) N=9
         Endif

C        ================================================================
C        Description of each variable for the failure mode chosen.
C        ================================================================
         Call VarExt(N,Opt,Ext,ExtExt,ParamDesc)
         Write(*,1009)
         Write(40,1009)
         Write(50,1009)
 1009    Format(// 3X,'DESCRIPTION OF THE VARIABLES' /)
         Do 1603 i=1,N
           Write(*,1018) i,Ext(i),ExtExt(i)
           Write(40,1018) i,Ext(i),ExtExt(i)
           Write(50,1018) i,Ext(i),ExtExt(i)
 1018      Format(11X,'X(',I3,') = ',A3,' = ',A17)
 1603    continue

C        ================================================================
C        Definition of the purpose of the analysis.
C        ================================================================
         Write(*,1017)
         Write(40,1017)
         Write(50,1017)
 1017    Format(// 3X,'WHAT IS THE PURPOSE OF THE ANALYSIS ? ' // 11X,
```

```
      1          'Reliability Analysis for a Specified Design ... [ 1 ]'
      1           / 11X,
      1          'Design for a Specified Reliability Level ...... [ 2 ]' /)
 7101    Write(*,96)
  96      Format(3X,'Select Option: ',$)
         Read(*,*) Mode
         If (Mode.NE.1.AND.Mode.NE.2) goto 7101
         Write(40,9696) Mode
         Write(50,9696) Mode
 9696    Format(3X,'Select Option: ',I1)

C        ================================================================
C        Reads the design life of the structure, Life.
C        ================================================================
  888    Write(*,091)
  091    Format(// 3X,'DESIGN LIFE OF THE STRUCTURE = ',$)
         Read(*,*) Life
         If (Life.LE.0) then
           Write(*,7882)
 7882      Format(// 3X,'ERROR: The Design Life of the Structure <= 0 !')
           goto 888
         Endif
         Write(40,791) Life
         Write(50,791) Life
  791    Format(// 3X,'DESIGN LIFE OF THE STRUCTURE = ',I3)

C        ================================================================
C        Definition of the combination of actions for Mode 1.
C        ================================================================
         If (Mode.EQ.1) then
 4901      Write(*,3334)
 3334      Format(// 3X,'WOULD YOU LIKE TO CONSIDER',1X,
      1            'COMBINATION OF ACTIONS (Y/N) ? ',$)
           Read(*,3335) CAcc
 3335      Format(A1)
           If (CAcc.NE.'Y'.AND.CAcc.NE.'y'.AND.
      1        CAcc.NE.'N'.AND.CAcc.NE.'n') goto 4901
           Write(40,6622) CAcc
           Write(50,6622) CAcc
 6622      Format(// 3X,'WOULD YOU LIKE TO CONSIDER',1X,
      1            'COMBINATION OF ACTIONS (Y/N) ? ',A1)
           If ((CAcc.EQ.'Y').OR.(CAcc.EQ.'y')) Comb=1
           If ((CAcc.EQ.'N').OR.(CAcc.EQ.'n')) Comb=2
         else
           Comb=2
         Endif

C         ----------
C         Definition of the number of combination of actions, NCombAc.
C         ----------
         If (Comb.EQ.2) then
           NCombAc=1
           Do 2388 i=1,N
             NR(1,i)=1
 2388      continue
         else
           Write(*,1320)
           Write(40,1320)
           Write(50,1320)
 1320      Format(// 3X,'HOW MANY COMBINATIONS WOULD YOU LIKE TO',
      1            1X,'CONSIDER ? ' // 11X,
      1            'The Number of Time-Varying Actions (k) ... [ 1 ]' /
      1            11X, '2^(k-1) ................................ [ 2 ]' /)
 8832      Write(*,9370)
 9370      Format(3X,'Select Option: ',$)
           Read(*,*) CombAc
           If (CombAc.NE.1.AND.CombAc.NE.2) goto 8832
           Write(40,1007) CombAc
           Write(50,1007) CombAc
 1007      Format(3X,'Select Option: ',I1)

C         ----------
C         Definition of the distributions provided for each combination
C          of actions.
C         ----------
           Write(*,1550)
           Write(40,1550)
           Write(50,1550)
 1550      Format(// 3X,'WHICH DISTRIBUTIONS WOULD YOU LIKE TO',
      1            1X,'PROVIDE ? ' // 11X,
      1            'The Basic Distributions ...... [ 1 ]' / 11X,
      1            'The Modified Distributions ... [ 2 ]' /)
  551      Write(*,9550)
```

```
 9550     Format(3X,'Select Option: ',$)
          Read(*,*) Distr
          If (Distr.NE.1.AND.Distr.NE.2) goto 551
          Write(40,1557) Distr
          Write(50,1557) Distr
 1557     Format(3X,'Select Option: ',I1)
        Endif

        If (Comb.EQ.1) then
C         ----------
C         Reads the number of time-varying actions, NumTVAc (maximum=5)
C         & calculates the number of combinations of actions, NCombAc.
C         ----------
 290      Write(*,092)
 092      Format(/ 3X,'NUMBER OF TIME-VARYING ACTIONS (Max=5) = ',$)
          Read(*,*) NumTVAc
          If (NumTVAC.GT.5.OR.NumTVAc.LE.0) then
            Write(*,7887)
 7887       Format(// 3X,'ERROR: The Maximum Number of Time Varying' /
     1              3X,'        Actions is not Within ]0,5] !')
            goto 290
          Endif
          Write(40,7991) NumTVAc
          Write(50,7991) NumTVAc
 7991     Format(// 3X,'NUMBER OF TIME-VARYING ACTIONS (Max=5) = ',I1)
          If (CombAc.EQ.1) then
            NCombAc=NumTVAc
          else
            NCombAc=2**(NumTVAc-1)
          Endif

          If (Distr.EQ.1) then
C           ----------
C           Reads the number of the time-varying actions, TVAC, in
C           increasing order of the number of repetitions, r. Reads the
C           number of repetitions of each action in the design life of
C           the structure.
C           ----------
            Write(*,7833)
 7833       Format(// 3X,'NUMBER OF THE TIME-VARYING ACTIONS IN' / 3X,
     1             'INCREASING ORDER OF THE NUMBER OF REPETITIONS' /)
            Do 140 i=1,NumTVAc
 2377         Write(*,7832) i
 7832         Format(11X,'Action (',I2,') = ',$)
              Read(*,*) TVAc(i)
              If (TVAc(i).GT.N.OR.TVAc(i).LE.0) then
                Write(*,7697) N
 7697           Format(//3X,'ERROR: The Value of the Number of the Time'
     1                 /3X,'        Varying Action is not Within ]0,',
     1                 I2,'] !' /)
                goto 2377
              Endif
 140        continue
            Write(*,782)
            Write(40,782)
            Write(50,782)
 782        Format(//3X,'REPETITIONS OF EACH ACTION IN THE DESIGN LIFE',
     1            /)
 3390       Do 1409 i=1,NumTVAc
 2379         Write(*,7834) TVAc(i)
 7834         Format(11X,'r (',I2,') = ',$)
              Read(*,*) r(TVAc(i))
              If (r(TVAc(i)).LE.0) then
                Write(*,6884)
 6884           Format(//3X,'ERROR: The Number of Repetitions',1X,
     1                 'is <= 0 !' /)
                goto 2379
              Endif
              If (r(TVAc(i)).LT.r(TVAC(i-1))) then
                Write(*,6334)
 6334           Format(// 3X,
     1                 'ERROR: The Actions are not Listed in Increasing'
     1                 /3X,'        Order of the Number of Repetitions !'
     1                 /)
                goto 3390
              Endif
 1409       continue
            Do 4000 i=1,NumTVAc
              Write(40,788) Ext(TVAc(i)),r(TVAc(i))
              Write(50,788) Ext(TVAc(i)),r(TVAc(i))
 788          Format(11X,'r(',A3,') = ',F8.0)
 4000       continue
            Call Combination(N,NumTVAc,CombAc,NCombAc,r,TVAc,NR)
```

```
              Do 8711 i=1,NCombAc
                Do 8712 j=1,N
                  Write(50,*) 'NR(i,j)=',i,j,NR(i,j)
 8712           continue
 8711         continue
            else
              Do 87 i=1,NCombAc
                Do 88 j=1,N
                  NR(i,j)=1
 88             continue
 87           continue
            Endif
          Endif

          If (Mode.EQ.1) then
C         ============================================================
C         Reads the value of the design parameter, StartParam, for which
C         the failure probability (or reliability) is to be found.
C         ============================================================
 2009     Write(*,9002) ParamDesc
 9002     Format(// 3X,'PRESCRIBED VALUE OF THE DESIGN PARAMETER' //
     1           11X,A19,' = ',$)
          Read(*,*) StartParam
          If (StartParam.LT.0) then
            Write(*,6554)
 6554       Format(// 3X,'ERROR: The Prescribed Value of the',
     1              1X,'Design Parameter < 0 !')
            goto 2009
          Endif
          Write(40,7891) ParamDesc,StartParam
          Write(50,7891) ParamDesc,StartParam
 7891     Format(// 3X,'PRESCRIBED VALUE OF THE DESIGN PARAMETER' //
     1          11X,A19,' = ',E17.10 /)
        else
C         ============================================================
C         Reads the design target failure probability, Pf, & starting
C         value of the design parameter, StartParam, from which the
C         program iterates to find the required value of the design
C         parameter for the given failure probability (or reliability).
C         ============================================================
 2101     Write(*,1044)
 1044     Format(// 3X, 'DESIGN TARGET FAILURE PROBABILITY [0,1] = ',$)
          Read(*,*) Pf
          If (Pf.LT.0.OR.Pf.GT.1) then
            Write(*,6234)
 6234       Format(//3X,'ERROR: The Value of the Design Target Failure',
     1              /3X,'        Probability is not Within [0,1] !')
            goto 2101
          Endif
 2099     Write(*,9022) ParamDesc
 9022     Format(// 3X, 'STARTING VALUE OF THE DESIGN PARAMETER - ',
     1           A19 / 3X,'(to start iteration) = ',$)
          Read(*,*) StartParam
          If (StartParam.LT.0) then
            Write(*,6224)
 6224       Format(// 3X,'ERROR: The Prescribed Value of the',
     1              1X,'Design Parameter < 0 !')
            goto 2099
          Endif
          If (StartParam.EQ.0.) then
            Write(*,874)
 874        Format(// 5X,
     1              'NOTE: In design for a specified reliability level,',
     1              1X,'the starting' / 11X,'value of the design',1X,
     1              'parameter should not be set to zero !' / 11X,
     1              'So, the program assumes a starting value of 5m.')
          Write(40,7791) (100.*Pf),ParamDesc,StartParam
          Write(50,7791) (100.*Pf),ParamDesc,StartParam
 7791       Format(// 3X,'DESIGN TARGET FAILURE PROBABILITY (%) = ',
     1              F10.6 // 3X,
     1              'STARTING VALUE OF THE DESIGN PARAMETER - ',A19 /
     1              3X, '(to start iteration) = ',E17.10 // 5X,
     1              'NOTE: In design for a specified reliability level,',
     1              1X,'the starting' / 11X,'value of the design',1X,
     1              'parameter should not be set to zero !' / 11X,
     1              'So, the program assumes a starting value of 5m.')
            StartParam=5.
          else
          Write(40,7751) (100.*Pf),ParamDesc,StartParam
          Write(50,7751) (100.*Pf),ParamDesc,StartParam
 7751       Format(// 3X,'DESIGN TARGET FAILURE PROBABILITY (%) = ',
     1              F10.6 // 3X,
     1              'STARTING VALUE OF THE DESIGN PARAMETER - ',A19
```

```
     1              / 3X,'(to start iteration) = ',E17.10)
          Endif

C        ================================================================
C        Calculation of the reliability index, RelInd, which
C        corresponds to the target failure probability, Pf.
C        ================================================================
          Call InvNormal(Pf,RelInd)
          Write(40,8792) RelInd
          Write(50,8792) RelInd
 8792     Format(/ 3X,'Reliability Index = ',E17.10 /)
        Endif

C        ================================================================
C        Reads the characteristics of the variables:
C        - Type of distribution, VarDist
C        - Type of truncation, Trunc
C        - Point of truncation, Xo (if the distribution is truncated)
C        - Mean, Mux, standard deviation, Sigmax, & lower limit, Zeta
C        ================================================================
        Do 6633 j=1,NCombAc
          If ((NCombAC.GT.1).AND.(Distr.EQ.2)) then
            Write(*,7073) j
            Write(40,7073) j
            Write(50,7073) j
 7073       Format(/// 3X,'COMBINATION No.',I2)
          Endif
          Write(*,8777)
          Write(40,8777)
          Write(50,8777)
 8777     Format(// 3X, 'CHARACTERISTICS OF THE VARIABLES ')
          Write(40,8222)
          Write(50,8222)
 8222     Format(/)
          Do 18 i=1,N
            Write(*,717) Ext(i)
 717        Format(// 11X,'What is the Probability Distribution of ',
     1             A3,' ? ' // 19X,
     1             'Pre-Defined Distributions' / 27X,
     1             'Deterministic .............. [ 0 ]' / 27X,
     1             'Normal ..................... [ 1 ]' / 27X,
     1             'Log-Normal ................. [ 2 ]' / 27X,
     1             'Maxima Type I (Gumbel) ..... [ 3 ]' / 27X,
     1             'Rectangular (Uniform) ...... [ 4 ]' / 27X,
     1             'Gamma ...................... [ 5 ]' / 27X,
     1             'Beta ....................... [ 6 ]' / 27X,
     1             'Maxima Type II (Frechet) ... [ 7 ]' / 27X,
     1             'Exponential ................ [ 8 ]' / 27X,
     1             'Rayleigh ................... [ 9 ]' / 27X,
     1             'Minima Type III (Weibull) .. [ 10]' // 19X,
     1             'User-Defined Distributions' / 27X,
     1             'Water Levels ............... [ 11]' / 27X,
     1             'Extreme Water Levels ....... [ 12]' / 27X,
     1             'Tide Levels ................ [ 13]' /)
 697        Write(*,796)
 796        Format(11X,'Select Option: ',$)
            Read(*,*) VarDis(j,i)
            If (VarDis(j,i).NE.0.AND.VarDis(j,i).NE.1.AND.
     1          VarDis(j,i).NE.2.AND.VarDis(j,i).NE.3.AND.
     1          VarDis(j,i).NE.4.AND.VarDis(j,i).NE.5.AND.
     1          VarDis(j,i).NE.6.AND.VarDis(j,i).NE.7.AND.
     1          VarDis(j,i).NE.8.AND.VarDis(j,i).NE.9.AND.
     1          VarDis(j,i).NE.10.AND.VarDis(j,i).NE.11.AND.
     1          VarDis(j,i).NE.12.AND.VarDis(j,i).NE.13) goto 697
            If (VarDis(j,i).NE.0.AND.VarDis(j,i).LE.10) then
              Write(*,275) Ext(i)
 275          Format(/ 11X,'Is the Distribution of ',A3,' Truncated ? '/
     1               19X,'Not Truncated ................. [ 0 ]' /
     1               19X,'Truncated for Xi above Xo ..... [ 1 ]' /
     1               19X,'Truncated for Xi below Xo ..... [ 2 ]')
 6971         Write(*,1796)
 1796         Format(11X,'Select Option: ',$)
              Read(*,*) Trunc(j,i)
              If (Trunc(j,i).NE.0.AND.Trunc(j,i).NE.1.AND.
     1            Trunc(j,i).NE.2) goto 6971
            Endif
            Xo(j,i)=0.
            If (Trunc(j,i).NE.0) then
C             ----------
C             For the failure mode of overtopping, if the variable is
C             the significant wave height, Hs, & if Hs is limited by the
C             available water depth, then the point of truncation is
C             Xo=0.6(SWL-TL) or Xo=0.6(Tide+Surge-TL), where TL is the
```

```
C                seawall toe level.
C                ----------
                 If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(i.EQ.2).AND.
      1             (Trunc(j,i).EQ.1)) then
                  Write(*,7511)
 7511              Format(/ 11X,'Seawall Toe Level (TL) = ',$)
                  Read(*,*) TL
                  goto 5555
                 Endif
                 Write(*,751) i
  751             Format(/ 11X,'Value of X(',I2,
      1                 ') at which the distribution is to be truncated = ',
      1                 $)
                 Read(*,*) Xo(j,i)
               Endif
               If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(i.EQ.2)) then
                 Write(*,7561)
 7561             Format(/ 11X,'Seawall Toe Level (TL) = ',$)
                 Read(*,*) TL
                 Write(40,7881) TL
                 Write(50,7881) TL
 7881             Format(11X,'Seawall Toe Level (TL) = ',E17.10)
               Endif
 5555          If (VarDis(j,i).NE.0) then
                 Write(*,75) Ext(i)
  75              Format(/ 11X,'Mean Value of ',A3,' = ',$)
                 Read(*,*) Mux(j,i)
                 Write(*,775) Ext(i)
  775             Format(11X,'Standard Deviation of ',A3,' = ',$)
                 Read(*,*) Sigmax(j,i)
               else
                 Write(*,7533) Ext(i)
 7533             Format(/ 11X,'Value of ',A3,' = ',$)
                 Read(*,*) Mux(j,i)
                 Sigmax(j,i)=0
               Endif
               If (VarDis(j,i).EQ.6) then
 1313            Write(*,9952) Ext(i)
 9952             Format(11X,'Insert Limit a (a <= ',A3,' <= b): ',$)
                 Read(*,*) x1(j,i)
                 Write(*,9953) Ext(i)
 9953             Format(11X,'Insert Limit b (a <= ',A3,' <= b): ',$)
                 Read(*,*) x2(j,i)
                 If (x1(j,i).GE.x2(j,i)) then
                   Write(*,8703)
 8703               Format(// 11X,'ERROR: a >= b !' /)
                   goto 1313
                 Endif
               Endif
               If (VarDis(j,i).EQ.10) then
                 Write(*,7501) Ext(i)
 7501             Format(11X,'Lower Limit on ',A3,' = ',$)
                 Read(*,*) Zeta(j,i)
               Endif
               Call WhatDist(i,j,VarDis,Abrev,Carac)
               Call WriCharVar(i,j,VarDis,Trunc,Ext,Carac,Xo,Mux,Sigmax,
      1                     Opt)
  18      continue

C         ==============================================================
C         Calculation of the distribution's parameters, Zeta, Lamda &
C         Eta, for each variable.
C         ==============================================================
          Write(50,4495)
 4495      Format(// 6X,'DISTRIBUTION`S PARAMETERS')
          Do 6070 i=1,N
            Call Parameters(i,j,Mux,Sigmax,VarDis,Ext)
 6070      continue
          If (Distr.EQ.1) then
            If (NCombAc.GT.1) then
              Call EqCharac(N,NCombAc,Abrev,Carac,VarDis,Trunc,Xo,Mux,
      1                   Sigmax)
              Write(*,9911)
              Write(40,9911)
              Write(50,9911)
 9911          Format(// 11X,'THE CHARACTERISTICS OF THE VARIABLES ARE ',
      1               / 11X,'THE SAME FOR ALL COMBINATIONS OF ACTIONS.',
      1               //)
            Endif
            goto 1314
          Endif
 6633   continue
```

```
C       ================================================================
C       Reads the correlation coefficients, Rho, of the variables.
C       ================================================================
 1314   Write(*,971)
        Write(40,971)
        Write(50,971)
 971    Format(// 3X, 'CORRELATION COEFFICIENTS ' /)
        Do 38 i=1,N
          Do 37 j=1,N
            Write(*,74) Ext(i),Ext(j)
 74         Format(11X,'(',A3,',',A3,') = ',$)
            Read(*,*) Rho(i,j)
            Write(40,536) Ext(i),Ext(j),Rho(i,j)
            Write(50,536) Ext(i),Ext(j),Rho(i,j)
 536        Format(11X,'(',A3,',',A3,') = ',E17.10)
 37       continue
 38     continue
 9988   Do 4005 i=1,N
          Do 4050 j=1,N
            If ((Rho(i,j).NE.Rho(j,i)).OR.(ABS(Rho(i,j)).GT.1)) then
              If (Rho(i,j).NE.Rho(j,i)) then
                Write(*,7801) i,j,j,i
                Write(50,7801) i,j,j,i
 7801           Format(// 11X,'ERROR: Rho(',I2,',',I2,
     1                  ') is not equal to Rho(',I2,',',I2,') !' //)
                Write(*,7444) i,j
 7444           Format(11X,'Rho(',I2,',',I2,') = ',$)
                Read(*,*) Rho(i,j)
                Write(*,7445) j,i
 7445           Format(11X,'Rho(',I2,',',I2,') = ',$)
                Read(*,*) Rho(j,i)
                Write(40,36) Ext(i),Ext(j),Rho(i,j),
     1                       Ext(j),Ext(i),Rho(j,i)
                Write(50,36) Ext(i),Ext(j),Rho(i,j),
     1                       Ext(j),Ext(i),Rho(j,i)
 36             Format(11X,'(',A3,',',A3,') = ',E17.10 /
     1                 11X,'(',A3,',',A3,') = ',E17.10)
              else
                Write(*,7901) i,j
                Write(50,7901) i,j
 7901           Format(// 11X,'ERROR: |Rho(',I2,',',I2,')| > 1 !' //)
                Write(*,7455) i,j
 7455           Format(11X,'Rho(',I2,',',I2,') = ',$)
                Read(*,*) Rho(i,j)
                Write(40,362) Ext(i),Ext(j),Rho(i,j)
                Write(50,362) Ext(i),Ext(j),Rho(i,j)
 362            Format(11X,'(',A3,',',A3,') = ',E17.10)
              Endif
              goto 9988
            Endif
 4050     continue
 4005   continue


C       ================================================================
C       Reads the characteristics of the FORM calculations:
C       - starting value of the variables, StartPt
C       - minimum value, XMin, & maximum value, XMax, of the variables
C       - maximum number of iterations, MaxIter
C       - number of FORM calculations, NumCalc
C       - target values for each FORM calculation, TR
C       - required accuracy of the reliability index, ReqBetaAcc
C       - smoothing coefficient for the iteration process, Smooth
C       - required accuracy of the failure function, ReqOBJFAcc
C       ================================================================

C       ----------
C       Starting value of the variables, StartPt.
C       ----------
        Do 78 j=1,NCombAC
          If (NCombAC.GT.1) then
            If (Distr.EQ.2) Write(*,7079) j
            Write(40,7079) j
            Write(50,7079) j
 7079       Format(/// 3X,'COMBINATION No.',I2)
          Endif
 4455     If ((Distr.EQ.2).OR.(Comb.EQ.2)) then
            Write(*,1328)
            Write(40,1328)
            Write(50,1328)
 1328       Format(/ 3X,'STARTING POINT FOR THE FORM CALCULATIONS: '//
     1             11X, 'Default Values (mean values) ... [ 1 ]' /
     1             11X, 'User Specified Values .......... [ 2 ]' /)
 739          Write(*,9477)
```

```
9477        Format(3X,'Select Option: ',$)
            Read(*,*) Def
            If (Def.NE.1.AND.Def.NE.2) goto 739
            Write(40,8833) Def
            Write(50,8833) Def
8833        Format(3X,'Select Option: ',I1)
            Write(*,5671)
            Write(40,5671)
            Write(50,5671)
5671        Format(/ 3X,'STARTING POINT' /)
            If (Def.EQ.1) then
              Do 2156 i=1,N
                StartPt(j,i)=Mux(j,i)
                Write(*,5670) Ext(i),StartPt(j,i)
                Write(40,5670) Ext(i),StartPt(j,i)
                Write(50,5670) Ext(i),StartPt(j,i)
5670              Format(11X,A3,' = ',E17.10)
2156          continue
            else
              Do 2155 i=1,N
                If (VarDis(j,i).NE.0) then
                  Write(*,1132) Ext(i)
1132              Format(11X,A3,' = ',$)
                  Read(*,*) StartPt(j,i)
                  Write(40,5680) Ext(i),StartPt(j,i)
                  Write(50,5680) Ext(i),StartPt(j,i)
5680              Format(11X,A3,' = ',E17.10)
                else
                  StartPt(j,i)=Mux(j,i)
                Endif
2155          continue
            Endif
          Endif
          If (Distr.EQ.1) then
            Write(40,5636)
            Write(50,5636)
5636        Format(/ 3X,'STARTING POINT' /)
            Do 2956 i=1,N
              If (NR(j,i).NE.1.) then
                Call Inverse(j,i,Mux,NR,VarDis,StartPt)
              else
                StartPt(j,i)=Mux(j,i)
              Endif
              Write(40,5629) Ext(i),StartPt(j,i)
              Write(50,5629) Ext(i),StartPt(j,i)
5629          Format(11X,A3,' = ',E17.10)
2956        continue
          Endif
          If (Aux.EQ.1) goto 115
78      continue

C       ----------
C       Minimum value, XMin, & maximum value, XMax, of the variables.
C       ----------
115     Do 789 j=1,NCombAc
          If ((NCombAC.GT.1).AND.(Distr.EQ.2)) then
            Write(*,7074) j
            Write(40,7074) j
            Write(50,7074) j
7074        Format(/// 3X,'COMBINATION No.',I2)
          Endif
          Write(*,1327)
          Write(40,1327)
          Write(50,1327)
1327      Format(/ 3X,'LIMITING VALUES FOR THE VARIABLES: '//
     1           11X, 'Default Values (+/- 1E25) ... [ 1 ]' /
     1           11X, 'User Specified Values ....... [ 2 ]' /)
7397      Write(*,7937)
7937      Format(3X,'Select Option: ',$)
          Read(*,*) Def1
          If (Def1.NE.1.AND.Def1.NE.2) goto 7397
          Write(40,9375) Def1
          Write(50,9375) Def1
9375      Format(3X,'Select Option: ',I1)
          If (Def1.EQ.1) then
            Do 2176 i=1,N
              If (VarDis(j,i).NE.0) then
                XMin(j,i)=-1E25
                XMax(j,i)=+1E25
                If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(DSWL.EQ.1)) then
                    XMin(j,5)=TL
                    XMax(j,5)=StartParam
                    If (XMin(j,5).GT.XMax(j,5)) then
```

```
                        Write(*,8653)
                        Write(50,8653)
8653                     Format(// 11X,'ERROR: The Toe Level is Above the',
    1                              1X,'Seawall Crest Level !')
                          STOP
                        Endif
                  Endif
                else
                  XMin(j,i)=Mux(j,i)
                  XMax(j,i)=Mux(j,i)
                Endif
2176         continue
         else
           Write(*,7974)
7974       Format(/ 3X,'LIMITING VALUES FOR THE VARIABLES')
           Do 2175 i=1,N
             If (VarDis(j,i).NE.0) then
1310           Write(*,7874) Ext(i)
7874           Format(/ 11X,'XMin(',A3,') = ',$)
               Read(*,*) XMin(j,i)
               Write(*,7978) Ext(i)
7978           Format(11X,'XMax(',A3,') = ',$)
               Read(*,*) XMax(j,i)
               If (XMin(j,i).GT.XMax(j,i)) then
                 Write(*,8733) i,i
8733             Format(// 11X,'ERROR: XMin(',I2,') > XMax(',I2,') !')
                 goto 1310
               Endif
               If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(DSWL.EQ.1)) then
                 XMin(j,5)=TL
                 XMax(j,5)=StartParam
                 If (XMin(j,5).GT.XMax(j,5)) then
                   Write(*,8153)
                   Write(50,8153)
8153               Format(// 11X,'ERROR: The Toe Level is Above the',
    1                              1X,'Seawall Crest Level !')
                   STOP
                 Endif
               Endif
             else
               XMin(j,i)=Mux(j,i)
               XMax(j,i)=Mux(j,i)
             Endif
2175         continue
         Endif
         Write(40,5678)
         Write(50,5678)
5678     Format(/ 3X,'LIMITING VALUES FOR THE VARIABLES' /)
         Do 2788 i=1,N
           Call MinMax(i,j,Opt,VarDis,Ext,Trunc,Xo,Mux,Sigmax,XMin,
    1                  XMax)
           If (StartPt(j,i).LT.XMin(j,i)) then
             Write(*,8977) Ext(i),StartPt(j,i),Ext(i),XMin(j,i)
8977         Format(// 3X,'ERROR: Starting Value of ',A3,' = ',E17.10
    1                 / 10X,'< Minimum of ',A3,' = ',E17.10,' !' /)
             Aux=1
             If (Distr.EQ.1) then
               StartPt(j,i)=XMin(j,i)
             else
               goto 4455
             Endif
           Endif
           If (StartPt(j,i).GT.XMax(j,i)) then
             Write(*,8477) Ext(i),StartPt(j,i),Ext(i),XMax(j,i)
8477         Format(// 3X,'ERROR: Starting Value of ',A3,' = ',E17.10
    1                 / 10X,'> Maximum of ',A3,' = ',E17.10,' !' /)
             Aux=1
             If (Distr.EQ.1) then
               StartPt(j,i)=XMax(j,i)
             else
               goto 4455
             Endif
           Endif
2788       continue
         If (Distr.EQ.1) then
           If (NCombAc.GT.1) then
             Do 8001 k=2,NCombAc
               Do 7001 i=1,N
                 XMin(k,i)=XMin(1,i)
                 XMax(k,i)=XMax(1,i)
7001           continue
8001         continue
             Write(40,9111)
```

```
                       Write(50,9111)
  9111            Format(// 11X,
      1                      'THE MINIMUM AND MAXIMUM VALUES OF THE VARIABLES' /
      1                      11X,'ARE THE SAME FOR ALL COMBINATIONS OF ACTIONS.'
      1                      //)
                 Endif
                 goto 1315
               Endif
   789     continue

C          ----------
C          Maximum number of iterations, MaxIter.
C          ----------
  1315     Write(*,5925)
  5925     Format(// 3X,'MAXIMUM NUMBER OF ITERATIONS (Max=200) = ',$)
           Read(*,*) MaxIter
           If (MaxIter.GT.200.OR.MaxIter.LE.0) then
             Write(*,7851)
  7851       Format(// 3X,'ERROR: The Maximum Number of Iterations' /
      1               3X,'        is not Within ]0,200] !')
             goto 1315
           Endif
           Write(40,5905) MaxIter
           Write(50,5905) MaxIter
  5905     Format(// 3X,'MAXIMUM NUMBER OF ITERATIONS (Max=200) = ',I3)

C          ----------
C          Number of FORM calculations, NumCalc.
C          ----------
  7979     Write(*,5095)
  5095     Format(/ 3X,'NUMBER OF FORM CALCULATIONS (Max=10) = ',$)
           Read(*,*) NumCalc
           If (NumCalc.GT.10.OR.NumCalc.LE.0) then
             Write(*,3279)
  3279       Format(// 3X,'ERROR: The Maximum Number of FORM Calculations'
      1             / 3X,'        is not Within ]0,10] !')
             goto 7979
           Endif
           Write(40,5005) NumCalc
           Write(50,5005) NumCalc
  5005     Format(/ 3X,'NUMBER OF FORM CALCULATIONS (Max=10) = ',I2 /)

C          ----------
C          Target values for each FORM calculation, TR.
C          ----------
           If (Opt.NE.3) then
             If ((Opt.EQ.1).OR.(Opt.EQ.2)) then
               Write(*,5023)
               Write(40,5023)
               Write(50,5023)
  5023         Format(/ 3X,'ALLOWABLE DISCHARGE (m3/s/m)' /)
               Do 2007 k0=1,NumCalc
  8799           Write(*,5021) k0
  5021           Format(11X,'Qa(',I2,') = ',$)
                 Read(*,*) TR(k0)
                 If (TR(k0).LT.0) then
                   Write(*,3271)
  3271             Format(// 3X,'ERROR: The Allowable Discharge is < 0 !'/)
                   goto 8799
                 Endif
                 Write(40,6023) k0,TR(k0)
                 Write(50,6023) k0,TR(k0)
  6023           Format(11X,'Qa(',I2,') = ',E17.10)
  2007         continue
             else
               Write(*,5055)
               Write(40,5055)
               Write(50,5055)
  5055         Format(/ 3X,'ALLOWABLE TARGET' /)
               Do 2055 k0=1,NumCalc
                 Write(*,5033) k0
  5033           Format(11X,'Target(',I2,') = ',$)
                 Read(*,*) TR(k0)
                 Write(40,6055) k0,TR(k0)
                 Write(50,6055) k0,TR(k0)
  6055           Format(11X,'Target(',I2,') = ',E17.10)
  2055         continue
             Endif
           else
             Write(*,5022)
             Write(40,5022)
             Write(50,5022)
  5022       Format(/ 3X,'ALLOWABLE EROSION DISTANCE (m)' /)
```

```
          Do 2006 k0=1,NumCalc
 9975       Write(*,6022) k0
 6022        Format(11X,'Eda(',I2,') = ',$)
            Read(*,*) TR(k0)
            If (TR(k0).LT.0) then
              Write(*,3244)
 3244          Format(// 3X,
    1                    'ERROR: The Allowable Erosion Distance is < 0 !' /)
              goto 9975
            Endif
            Write(40,6062) k0,TR(k0)
            Write(50,6062) k0,TR(k0)
 6062        Format(11X,'Eda (',I2,') = ',E17.10)
 2006      continue
        Endif

C        ----------
C        Required accuracy of the reliability index, ReqBetaAcc.
C        ----------
        Write(*,1997)
        Write(40,1997)
        Write(50,1997)
 1997   Format(// 3X,
    1            'REQUIRED RELATIVE ACCURACY OF THE RELIABILITY INDEX: '//
    1            11X, 'Default Value (1%) ......... [ 1 ]' /
    1            11X, 'User Specified Value ....... [ 2 ]' /)
 9000   Write(*,7977)
 7977   Format(3X,'Select Option: ',$)
        Read(*,*) Def2
        If (Def2.NE.1.AND.Def2.NE.2) goto 9000
        Write(40,9399) Def2
        Write(50,9399) Def2
 9399   Format(3X,'Select Option: ',I1)
        If (Def2.EQ.1) then
          ReqBetaAcc=1.
        else
 4788      Write(*,8874)
 8874      Format(// 3X,'REQUIRED RELATIVE ACCURACY OF THE RELIABILITY',
    1               1X,'INDEX [0,1] = ',$)
          Read(*,*) ReqBetaAcc
          If ((ReqBetaAcc.LT.(0.)).OR.(ReqBetaAcc.GT.1.)) then
            Write(*,5577)
 5577        Format(// 3X,'ERROR: The Required Relative Accuracy of the',
    1               / 10X,'Reliability Index is not Within [0,1] ! ')
            goto 4788
          Endif
          Write(40,9971) ReqBetaAcc
          Write(50,9971) ReqBetaAcc
 9971      Format(// 3X,'Required Relative Accuracy of the' /
    1               3X,'Reliability Index [0,1] = ',E17.10 /)
        Endif

C        ----------
C        Smoothing of the iteration process (0<=Smooth<=1).
C        If Smooth=0 then there is no smoothing; if Smooth=0.5 then there
C        is averaging between the last two calculated values of X.
C        ----------
        Write(*,1227)
        Write(40,1227)
        Write(50,1227)
 1227   Format(// 3X,'REQUIRED SMOOTHING COEFFICIENT FOR THE',1X,
    1            'ITERATION PROCESS: '//
    1            11X, 'Default Value (0) .......... [ 1 ]' /
    1            11X, 'User Specified Value ....... [ 2 ]' /)
 2792   Write(*,2972)
 2972   Format(3X,'Select Option: ',$)
        Read(*,*) Def3
        If (Def3.NE.1.AND.Def3.NE.2) goto 2792
        Write(40,9229) Def3
        Write(50,9229) Def3
 9229   Format(3X,'Select Option: ',I1)
        If (Def3.EQ.1) then
          Smooth=0.
        else
 2288      Write(*,8822)
 8822      Format(// 3X,'REQUIRED SMOOTHING COEFFICIENT FOR',
    1               1X,'THE ITERATION PROCESS [0,1] = ',$)
          Read(*,*) Smooth
          If ((Smooth.LT.(0.)).OR.(Smooth.GT.1.)) then
            Write(*,1177)
 1177        Format(//3X,'ERROR: The Required Smoothing Coefficient for',
    1               1X,'the' / 10X,
    1               'Iteration Process is not Within [0,1] !')
```

```
            goto 2288
          else
            Write(40,1178) Smooth
            Write(50,1178) Smooth
 1178        Format(// 3X,'Required Smoothing Coefficient for' /
    1                   3X,'the Iteration Process [0,1] = ',E17.10 /)
          Endif
        Endif

C       ----------
C       Required accuracy of the failure function OBJF, ReqOBJFAcc.
C       If ReqOBJFAcc=1% then ABS(OBJF)<0.01SigmaOBJF, where SigmaOBJF
C       is the standard deviation of the failure function.
C       ----------
        Write(*,3397)
        Write(40,3397)
        Write(50,3397)
 3397   Format(// 3X,'REQUIRED ACCURACY OF THE FAILURE FUNCTION: '//
    1           11X, 'Default Value (1%) ......... [ 1 ]' /
    1           11X, 'User Specified Value ....... [ 2 ]' /)
 2744   Write(*,4472)
 4472   Format(3X,'Select Option: ',$)
        Read(*,*) Def4
        If (Def4.NE.1.AND.Def4.NE.2) goto 2744
        Write(40,4499) Def4
        Write(50,4499) Def4
 4499   Format(3X,'Select Option: ',I1)
        If (Def4.EQ.1) then
          ReqOBJFAcc=1.
        else
 5588     Write(*,8855)
 8855     Format(// 3X,'REQUIRED ACCURACY OF THE FAILURE',1X,
    1             'FUNCTION [0,1] = ',$)
          Read(*,*) ReqOBJFAcc
          If ((ReqOBJFAcc.LT.(0.)).OR.(ReqOBJFAcc.GT.1.)) then
            Write(*,5544)
 5544        Format(// 3X,'ERROR: The Required Accuracy of the',1X,
    1               'Failure' / 10X,
    1               'Function is not Within [0,1] !')
            goto 5588
          Endif
          Write(40,9448) ReqOBJFAcc
          Write(50,9448) ReqOBJFAcc
 9448      Format(// 3X,'Required Accuracy of the Failure' /
    1               3X,'Function [0,1] = ',E17.10 /)
        Endif

        return
        End




C       ################################################################
C
        Subroutine Dadfile(NCombAc,Distr,Comb,Mode,Opt,N,Mux,Sigmax,
    1                      VarDis,Abrev,Ext,ParamDesc,Trunc,Xo,
    1                      MaxIter,NumCalc,StartPt,XMax,XMin,
    1                      ReqBetaAcc,Smooth,ReqOBJFAcc,NR,Rho,Life,
    1                      CombAc,StartParam,Pf,RelInd,ExtExt,FDer)
C
C       ################################################################
C
C       Reads the required input data from data files.
C
C       ################################################################
C
C       MODELING VARIABLES:
C       OptC - Confidence value of the maximum run-up
C       CAcc - Consideration or not of combination of actions
C       Q - Maximum number of combinations of actions allowed by the
C            program
C       NumTVAc - Number of time-varying actions
C       r - Repetitions of each action in the design life
C       TVAc - Number of the time-varying actions in increasing order of
C              the number of repetitions
C       L - Maximum number of variables allowed by the program
C       Der - Method of calculation of the first partial derivatives of
C             the failure function for overtopping
C       Carac - Name of the distribution
C       Def1 - Definition of the limiting values of X
C       Def - Definition of the starting point for the FORM calculations
C       L2 - Maximum number of FORM calculations allowed by the program
C       Def2 - Definition of the required relative accuracy of the
```

```
C              reliability index
C         Def3 - Definition of the required smoothing coefficient for the
C               iteration process
C         Def4 - Definition of the required accuracy of the failure
C               function
C         i, j, k, k0, Aux - Auxiliary variables
C         It, AuxRstar - Variables mentioned in the Common statements but
C                        not used here
C
C         OUTPUT VARIABLES:
C         Opt - Failure mode
C         DSWL - Definition of the SWL
C         TL - Seawall toe level
C         OptD - Direction of the sand movements in dune erosion
C         C1 - Parameter used in the H&R model to calculate C; it depends
C              on the confidence value assigned to the maximum run-up
C         Mode - Purpose of the analysis
C         Comb - Consideration or not of combination of actions
C         NCombAc, CombAc - Number of combinations of actions
C         NR - Power to which each distribution is raised for each
C              combination of actions
C         Distr - Distributions provided for the combination of actions
C         N - Number of variables
C         FDer - Method of calculation of the first partial derivatives of
C                the failure function for overtopping
C         Ext - Abbreviation of the name of the variable
C         ExtExt - Description of the variable
C         ParamDesc - Description of the design parameter
C         Rho - Correlation coefficient
C         Mux - Mean of X
C         Sigmax - Standard deviation of X
C         VarDis - Type of distribution
C         Abrev - Abbreviation of the name of the distribution
C         Trunc - Type of truncation
C         Xo - Point of truncation (if the distribution is truncated)
C         XMin - Minimum value of X
C         XMax - Maximum value of X
C         Zeta, Lamda, Eta - Parameters of a distribution
C         x1 - Lower limit on X for a Beta distribution
C         x2 - Upper limit on X for a Beta distribution
C         StartPt - Starting value of the variables
C         MaxIter - Maximum number of iterations
C         NumCalc - Number of FORM calculations
C         TR - Target values for each FORM calculation
C         ReqBetaAcc - Required relative accuracy of the reliability index
C         Smooth - Smoothing coefficient for the iteration process
C         ReqOBJFAcc- Required accuracy of the failure function
C         Life - Design life of the structure
C         StartParam - For Mode=1, it is the prescribed value of the
C                      design parameter; For Mode=2, it is the starting
C                      value of the design parameter (from which the
C                      program iterates to find the required value of the
C                      design parameter)
C         Pf - Design target failure probability
C         RelInd - Reliability index which corresponds to Pf
C         ctcurv - Coastal curvature in degrees per 1000m
C         NPD, NPDOld - Number of points defining the initial profile
C         (XP,YP), (XPOld,YPOld) - Coordinates of the points defining the
C                                  initial profile
C         NPch - Number of points to be changed in the initial profile
C         t - First point to be changed in the initial profile, point no.t
C         1:md - Gradient of the eroded dune face
C         1:mt - Gradient of the toe of the post-storm profile
C         nourtlev - Nourishment top level
C         1:mnour - Gradient of the nourished face
C
C         ################################################################
C
          Integer*4 i,j,k,k0,L,L2,N,NPD,Mode,Opt,NCombAc,Q,t,NPch,NPDOld,
     1              Life,MaxIter,NumCalc,Def,Def1,Def2,Def3,Def4,Comb,
     1              Distr,Aux,CombAc,NumTVAc,It,AuxRstar,OptC,OptD,FDer,
     1              DSWL
          Parameter (L=15)
          Parameter (L2=10)
          Parameter (Q=16)
          Character*1 Der,CAcc
          Character*3 Abrev(Q,L),Ext(L)
          Character*17 ExtExt(L)
          Character*30 Carac(Q,L)
          Character*19 ParamDesc
          Integer*4 VarDis(Q,L),Trunc(Q,L),TVAC(L)
          Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L),Eta(Q,L),
     1          x1(Q,L),x2(Q,L),XP,YP,md,mt,mnour,XPOld,YPOld,nourtlev,
```

```
     1          ctcurv,Xo(Q,L),XMax(Q,L),XMin(Q,L),TR(L2),ReqBetaAcc,
     1          Smooth,ReqOBJFAcc,NR(Q,L),StartPt(Q,L),Rho(L,L),
     1          StartParam,Pf,RelInd,r(L),TL,C1
        Common/BLOCK1/NPD,XP(100),YP(100)
        Common/BLOCK2/t,NPch,XPOld(100),YPOld(100),NPDOld
        Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
        Common/BLOCK6/OptD
        Common/BLOCK7/TR
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
        Common/BLOCK9/k0,It,AuxRstar
        Common/BLOCK10/C1
        Common/BLOCK11/DSWL,TL

        Open(Unit=15, File='general.dad', Status='Old')
        Open(Unit=30, File='meandev.dad', Status='Old')
        Open(Unit=35, File='coefcor.dad', Status='Old')
        Open(Unit=45, File='perfil.dad', Status='Old')
        Open(Unit=65, File='form.dad', Status='Old')

C       =================================================================
C       Definition of the failure mode to be studied.
C       =================================================================
        Read(15,*) Opt
        If (Opt.NE.1.AND.Opt.NE.2.AND.Opt.NE.3) then
          Write(*,3370)
          Write(50,3370)
 3370     Format(// 3X,'ERROR: Wrong Value for the Failure',1X,
     1          'Mode to be Studied !')
          STOP
        Endif
        Write(40,2)
        Write(50,2)
 2      Format(// 3X,'WHAT IS THE FAILURE MODE TO BE STUDIED: '//
     1          11X, 'Overtopping (H&R) .................... [ 1 ]' /
     1          11X, 'Overtopping (Owen) ................... [ 2 ]' /
     1          11X, 'Dune Erosion (Vellinga) ............. [ 3 ]' /)
        Write(40,9779) Opt
        Write(50,9779) Opt
 9779   Format(3X,'Select Option: ',I2)

C       ----------
C       Definition of the still-water-level.
C       ----------
        If (Opt.EQ.1.OR.Opt.EQ.2.OR.Opt.EQ.3) then
          Read(15,*) DSWL
          If (DSWL.NE.1.AND.DSWL.NE.2) then
            Write(*,3017)
            Write(50,3017)
 3017       Format(// 3X,'ERROR: Wrong Value for the Definition of' /
     1            10X,'the Still-Water-Level !')
            STOP
          Endif
          Write(40,9320)
          Write(50,9320)
 9320     Format(// 3X,'HOW IS THE STILL-WATER-LEVEL DEFINED ?'
     1          // 11X, 'Total Level .... [ 1 ]' /
     1            11X, 'Tide + Surge ... [ 2 ]' /)
          Write(40,9473) DSWL
          Write(50,9473) DSWL
 9473     Format(3X,'Select Option: ',I1)
        Endif

C       ----------
C       Definition of the confidence value of the maximum run-up to be
C       considered.
C       ----------
        If (Opt.EQ.1) then
          Read(15,*) OptC
          If (OptC.NE.1.AND.OptC.NE.2) then
            Write(*,3317)
            Write(50,3317)
 3317       Format(// 3X,'ERROR: Wrong Value for the Confidence Value' /
     1            10X,'of the Maximum Run-Up !')
            STOP
          Endif
          Write(40,1324)
          Write(50,1324)
 1324     Format(// 3X,'WHAT IS THE CONFIDENCE VALUE OF THE MAXIMUM',
     1            1X,'RUN-UP' / 3X,'THAT YOU WOULD LIKE TO CONSIDER ?'
     1            // 11X, '37 % ... [ 1 ]' /
     1              11X, '99 % ... [ 2 ]' /)
          Write(40,1037) OptC
          Write(50,1037) OptC
```

```
 1037      Format(3X,'Select Option: ',I1)
           If (OptC.EQ.1) then
             C1=1.52
           else
             C1=2.15
           Endif
         Endif

C        ----------
C        Definition of the method of calculation of the first partial
C        derivatives of the failure function for overtopping.
C        ----------
         If (Opt.NE.3) then
           Read(15,3377) Der
 3377      Format(A1)
           If (Der.NE.'Y'.AND.Der.NE.'y'.AND.Der.NE.'N'.AND.Der.NE.'n')
     1           then
             Write(*,3373)
             Write(50,3373)
 3373        Format(// 3X,
     1               'ERROR: Wrong Value for the Method of Calculation' /
     1               10X,'of the First Partial Derivatives of the' /
     1               10X,'Failure Function for Overtopping !')
             STOP
           Endif
           Write(40,6681) Der
           Write(50,6681) Der
 6681      Format(// 3X,'ARE THE FIRST DERIVATIVES OF THE FAILURE',1X,
     1               'FUNCTION SUPPLIED (Y/N) ? ',A1)
           If ((Der.EQ.'Y').OR.(Der.EQ.'y')) FDer=1
           If ((Der.EQ.'N').OR.(Der.EQ.'n')) FDer=2
         else
           FDer=2
         Endif

C        ----------
C        Definition of the direction of the sand movements occurring
C        during a storm surge.
C        ----------
         OptD=0
         If (Opt.EQ.3) then
           Read(15,*) OptD
           If (OptD.NE.1.AND.OptD.NE.2) then
             Write(*,3372)
             Write(50,3372)
 3372        Format(// 3X,'ERROR: Wrong Value for the Direction' /
     1               10X,'of the Sand Movements !')
             STOP
           Endif
           Write(40,1302)
           Write(50,1302)
 1302      Format(// 3X,'DURING A STORM SURGE, WOULD YOU LIKE TO TAKE',
     1               1X,'INTO ACCOUNT: '//
     1               11X, 'Movements of Sand in Both Directions ?',
     1               1X,'... [ 1 ]' / 11X, 'Movements of Sand',
     1               1X,'only Seaward ? ........ [ 2 ]' /)
           Write(40,1937) OptD
           Write(50,1937) OptD
 1937      Format(3X,'Select Option: ',I1)
         Endif

C        ----------
C        If the failure mode is dune erosion the following quantities
C        have to be read:
C        - coastal curvature in degrees per 1000m, ctcurv
C        - number of points defining the initial profile, NPD
C        - beach profile coordinates, (XP,YP)
C        - number of points to be changed in the initial profile, NPch
C        - first point to be changed, point no. t
C        - gradient of the eroded dune face, 1:md
C        - gradient of the toe of the post-storm profile, 1:mt
C        - nourishment top level, nourtlev
C        - gradient of the nourished face, 1:mnour
C        ----------
         If (Opt.EQ.3) then
           Read(45,*) ctcurv
           If (ctcurv.LT.0.OR.ctcurv.GT.24) then
             Write(*,3278)
             Write(50,3278)
 3278        Format(// 3X,'ERROR: The Coastal Curvature is not',1X,
     1               'Within [0,24] !')
             STOP
           Endif
```

```
          Write(40,7522) ctcurv
          Write(50,7522) ctcurv
7522     Format(/ 3X,'DUNE EROSION' // 11X,
    1            'Coastal Curvature (Deg/1000m) = ',E17.10 /)

          Read(45,*) NPD
          If (NPD.GT.100.OR.NPD.LE.0) then
            Write(*,3251)
            Write(50,3251)
3251       Format(// 3X,
    1               'ERROR: The Maximum Number of Points Defining the' /
    1                3X,'         Initial Profile is not Within ]0,100] !')
            STOP
          Endif
          Write(40,6011) NPD
          Write(50,6011) NPD
6011     Format(11X,'Number of Points Defining the Initial Profile',1X,
    1            '(Max=100) = ',I3 // 23X, 'Initial Profile ' // 19X,
    1            'X', 18X, 'Y')
          Do 3120 i=1,NPD
            Read(45,*) XP(i),YP(i)
            If (i.GE.2) then
              If (XP(i).LT.XP(i-1)) then
                Write(*,1466) i,(i-1)
                Write(50,1466) i,(i-1)
1466           Format(// 3X,'ERROR: XP(',I3,') < XP(',I3,') !')
                STOP
              Endif
            Endif
            Write(40,9011) XP(i),YP(i)
            Write(50,9011) XP(i),YP(i)
9011       Format(11X,E17.10,2X,E17.10)
            XPOld(i)=XP(i)
            YPOld(i)=YP(i)
3120      continue
          NPDOld=NPD

          Read(45,*) NPch
          If (NPch.LT.1.OR.NPch.GT.NPD) then
            Write(*,1366) NPD
            Write(50,1366) NPD
1366       Format(// 3X,'ERROR: The Number of Points to be Changed',
    1               1X,'in the'/10X,'Initial Profile is not Within [1,',
    1               I3,'] !')
            STOP
          Endif
          Read(45,*) t
          If (t.LT.1.OR.t.GT.NPD) then
            Write(*,1322) NPD
            Write(50,1322) NPD
1322       Format(// 3X,'ERROR: The First Point to be Changed in the' /
    1                3X,'         Initial Profile is not Within [1,',
    1               I3,'] !')
            STOP
          Endif
          Write(40,7517) NPch,t
          Write(50,7517) NPch,t
7517     Format(/ 11X,'Number of Points to be Changed in the Initial',
    1            1X,'Profile = ',I3 // 11X,'First Point to be Changed',
    1            1X,'= Point No.',I3)

          Read(45,*) md
          If (md.LE.0) then
            Write(*,1311)
            Write(50,1311)
1311       Format(// 3X,'ERROR: The Gradient of the Eroded Dune',
    1               1X,'Face is <= 1:0 !')
            STOP
          Endif

          Read(45,*) mt
          If (mt.LE.0) then
            Write(*,1151)
            Write(50,1151)
1151       Format(// 3X,'ERROR: The Gradient of the Toe of the',
    1               1X,'Post-Storm Profile is <= 1:0 !')
            STOP
          Endif

          Read(45,*) nourtlev

          Read(45,*) mnour
          If (mnour.LE.0) then
```

```
          Write(*,7103)
          Write(50,7103)
 7103      Format(// 3X,'ERROR: The Gradient of the Nourished',
     1            1X,'Face is <= 1:0 !')
            STOP
          Endif

          Write(40,7913) md,mt,nourtlev,mnour
          Write(50,7913) md,mt,nourtlev,mnour
 7913      Format(/ 11X,'Gradient of the Eroded Dune Face = 1:',F4.1 //
     1          11X,'Gradient of the Toe of the Post-Storm Profile = 1:',
     1          F4.1 // 11X,'Nourishment Top Level = ',E17.10 // 11X,
     1           'Gradient of the Nourished Face = 1:',F4.1)
        Endif

C       =================================================================
C       Definition of the number, N, of variables for each failure mode.
C       =================================================================
        If (Opt.EQ.1) then
          N=8
        else
          N=7
        Endif

        If (DSWL.EQ.2) then
          If (Opt.EQ.2.OR.Opt.EQ.3) N=8
          If (Opt.EQ.1) N=9
        Endif

C       =================================================================
C       Description of each variable for the failure mode chosen.
C       =================================================================
        Call VarExt(N,Opt,Ext,ExtExt,ParamDesc)
        Write(40,1009)
        Write(50,1009)
 1009  Format(// 3X,'DESCRIPTION OF THE VARIABLES' /)
        Do 1603 i=1,N
          Write(40,1018) i,Ext(i),ExtExt(i)
          Write(50,1018) i,Ext(i),ExtExt(i)
 1018     Format(11X,'X(',I3,') = ',A3,' = ',A17)
 1603   continue

C       =================================================================
C       Definition of the purpose of the analysis.
C       =================================================================
        Read(15,*) Mode
        If (Mode.NE.1.AND.Mode.NE.2) then
          Write(*,3374)
          Write(50,3374)
 3374      Format(// 3X,'ERROR: Wrong Value for the Definition' /
     1             10X,'of the Purpose of the Analysis !')
          STOP
        Endif
        Write(40,1017)
        Write(50,1017)
 1017  Format(// 3X,'WHAT IS THE PURPOSE OF THE ANALYSIS ? ' // 11X,
     1        'Reliability Analysis for a Specified Design ... [ 1 ]'
     1        / 11X,
     1        'Design for a Specified Reliability Level ...... [ 2 ]' /)
        Write(40,9696) Mode
        Write(50,9696) Mode
 9696  Format(3X,'Select Option: ',I1)

C       =================================================================
C       Reads the design life of the structure, Life.
C       =================================================================
        Read(15,*) Life
        If (Life.LE.0) then
          Write(*,7882)
          Write(50,7882)
 7882      Format(// 3X,
     1            'ERROR: The Design Life of the Structure is <= 0 !')
          STOP
        Endif
        Write(40,791) Life
        Write(50,791) Life
 791   Format(// 3X,'DESIGN LIFE OF THE STRUCTURE = ',I3)

C       =================================================================
C       Definition of the combination of actions for Mode 1.
C       =================================================================
        If (Mode.EQ.1) then
          Read(15,3335) CAcc
```

```
 3335     Format(A1)
          If (CAcc.NE.'Y'.AND.CAcc.NE.'y'.AND.
     1        CAcc.NE.'N'.AND.CAcc.NE.'n') then
            Write(*,3357)
            Write(50,3357)
 3357      Format(// 3X,'ERROR: Wrong Value for the Definition of the'/
     1              10X,'Combination of Actions for Mode 1 !')
            STOP
          Endif
          Write(40,6622) CAcc
          Write(50,6622) CAcc
 6622     Format(// 3X,'WOULD YOU LIKE TO CONSIDER',1X,
     1           'COMBINATION OF ACTIONS (Y/N) ? ',A1)
          If ((CAcc.EQ.'Y').OR.(CAcc.EQ.'y')) Comb=1
          If ((CAcc.EQ.'N').OR.(CAcc.EQ.'n')) Comb=2
        else
          Comb=2
        Endif

C       ----------
C        Definition of the number of combination of actions, NCombAc.
C       ----------
        If (Comb.EQ.2) then
          NCombAc=1
          Do 2388 i=1,N
            NR(1,i)=1
 2388      continue
        else
          Read(15,*) CombAc
          If (CombAc.NE.1.AND.CombAc.NE.2) then
            Write(*,3367)
            Write(50,3367)
 3367      Format(// 3X,'ERROR: Wrong Value for the Definition of the'/
     1              10X,'Number of Combination of Actions !')
            STOP
          Endif
          Write(40,1320)
          Write(50,1320)
 1320     Format(// 3X,'HOW MANY COMBINATIONS WOULD YOU LIKE TO',
     1           1X,'CONSIDER ? ' // 11X,
     1           'The Number of Time-Varying Actions (k) ... [ 1 ]' /
     1         11X, '2^(k-1) .................................. [ 2 ]' /)
          Write(40,1007) CombAc
          Write(50,1007) CombAc
 1007      Format(3X,'Select Option: ',I1)

C         ----------
C          Definition of the distributions provided for each combination
C          of actions.
C         ----------
          Read(15,*) Distr
          If (Distr.NE.1.AND.Distr.NE.2) then
            Write(*,9977)
            Write(50,9977)
 9977      Format(// 3X,'ERROR: Wrong Value for the Definition of the'/
     1              10X,'Distributions Provided for Each' /
     1              10X,'Combination of Actions !')
            STOP
          Endif
          Write(40,1550)
          Write(50,1550)
 1550     Format(// 3X,'WHICH DISTRIBUTIONS WOULD YOU LIKE TO',
     1           1X,'PROVIDE ? ' // 11X,
     1           'The Basic Distributions ...... [ 1 ]' / 11X,
     1           'The Modified Distributions ... [ 2 ]' /)
          Write(40,1557) Distr
          Write(50,1557) Distr
 1557      Format(3X,'Select Option: ',I1)
        Endif

        If (Comb.EQ.1) then
C         ----------
C          Reads the number of time-varying actions, NumTVAc (maximum=5)
C          & calculates the number of combinations of actions, NCombAc.
C         ----------
          Read(15,*) NumTVAc
          If (NumTVAc.GT.5.OR.NumTVAc.LE.0) then
            Write(*,7887)
            Write(50,7887)
 7887      Format(// 3X,'ERROR: The Maximum Number of Time Varying' /
     1              3X,'        Actions is not Within ]0,5] !')
            STOP
          Endif
```

```
            Write(40,7911) NumTVAc
            Write(50,7911) NumTVAc
 7911        Format(// 3X, 'NUMBER OF TIME-VARYING ACTIONS (Max=5) = ',I1)
            If (CombAc.EQ.1) then
              NCombAc=NumTVAc
            else
              NCombAc=2**(NumTVAc-1)
            Endif

            If (Distr.EQ.1) then
C            ----------
C            Reads the number of the time-varying actions, TVAC, in
C            increasing order of the number of repetitions, r. Reads the
C            number of repetitions of each action in the design life of
C            the structure.
C            ----------
              Write(40,782)
              Write(50,782)
 782          Format(//3X,'REPETITIONS OF EACH ACTION IN THE DESIGN LIFE',
     1          /)
              Do 140 i=1,NumTVAc
                Read(15,*) TVAc(i),r(TVAc(i))
                If (TVAc(i).GT.N.OR.TVAc(i).LE.0) then
                  Write(*,4886) N
                  Write(50,4886) N
 4886              Format(//3X,'ERROR: The Value of the Number of the Time'
     1                  /3X,'      Varying Action is not Within ]0,',
     1                  I3,'] !')
                  STOP
                Endif
                If (r(TVAc(i)).LE.0) then
                  Write(*,6884)
                  Write(50,6884)
 6884              Format(// 3X,'ERROR: The Number of Repetitions is',1X,
     1                  '<= 0 !')
                  STOP
                Endif
                If (r(TVAc(i)).LT.r(TVAC(i-1))) then
                  Write(*,6334)
                  Write(50,6334)
 6334              Format(// 3X,
     1                  'ERROR: The Actions are not Listed in Increasing'
     1                  /3X,'      Order of the Number of Repetitions !')
                  STOP
                Endif
                Write(40,788) Ext(TVAc(i)),r(TVAc(i))
                Write(50,788) Ext(TVAc(i)),r(TVAc(i))
 788            Format(11X,'r(',A3,') = ',F8.0)
 140          continue
              Call Combination(N,NumTVAc,CombAc,NCombAc,r,TVAc,NR)
              Do 8711 i=1,NCombAc
                Do 8712 j=1,N
                  Write(50,*) 'NR(i,j)=',i,j,NR(i,j)
 8712            continue
 8711          continue
            else
              Do 87 i=1,NCombAc
                Do 88 j=1,N
                  NR(i,j)=1
 88             continue
 87            continue
            Endif
          Endif

          If (Mode.EQ.1) then
C          ============================================================
C          Reads the value of the design parameter, StartParam, for which
C          the failure probability (or reliability) is to be found.
C          ============================================================
            Read(15,*) StartParam
            If (StartParam.LT.0) then
              Write(*,6554)
              Write(50,6554)
 6554          Format(// 3X,'ERROR: The Prescribed Value of the',
     1              1X,'Design Parameter is < 0 !')
              STOP
            Endif
            Write(40,7781) ParamDesc,StartParam
            Write(50,7781) ParamDesc,StartParam
 7781        Format(// 3X,'PRESCRIBED VALUE OF THE DESIGN PARAMETER' //
     1          11X,A19,' = ',E17.10)
          else
C          ============================================================
```

```
C          Reads the design target failure probability, Pf, & starting
C          value of the design parameter, StartParam, from which the
C          program iterates to find the required value of the design
C          parameter for the given failure probability (or reliability).
C          ================================================================
           Read(15,*) StartParam
           If (StartParam.LT.0) then
             Write(*,6224)
             Write(50,6224)
 6224        Format(// 3X,'ERROR: The Prescribed Value of the',
     1                  1X,'Design Parameter is < 0 !')
             STOP
           Endif
           Read(15,*) Pf
           If (Pf.LT.0.OR.Pf.GT.1) then
             Write(*,6234)
             Write(50,6234)
 6234        Format(//3X,'ERROR: The Value of the Design Target Failure',
     1                  /3X,'        Probability is not Within [0,1] !')
             STOP
           Endif
           If (StartParam.EQ.0.) then
             Write(40,7191) (100.*Pf),ParamDesc,StartParam
             Write(50,7191) (100.*Pf),ParamDesc,StartParam
 7191        Format(// 3X,'DESIGN TARGET FAILURE PROBABILITY (%) = ',
     1                  F10.6 // 3X,
     1                  'STARTING VALUE OF THE DESIGN PARAMETER - ',A19/
     1                     3X, '(to start iteration) = ',E17.10 // 5X,
     1                  'NOTE: In design for a specified reliability level,',
     1                  1X,'the starting' / 11X,'value of the design',1X,
     1                  'parameter should not be set to zero !' / 11X,
     1                  'So, the program assumes a starting value of 5m.')
             StartParam=5.
           else
             Write(40,7791) (100.*Pf),ParamDesc,StartParam
             Write(50,7791) (100.*Pf),ParamDesc,StartParam
 7791        Format(// 3X,'DESIGN TARGET FAILURE PROBABILITY (%) = ',
     1                  F10.6 // 3X,
     1                  'STARTING VALUE OF THE DESIGN PARAMETER - ',A19 /
     1                     3X, '(to start iteration) = ',E17.10)
           Endif

C          ================================================================
C          Calculation of the reliability index, RelInd, which
C          corresponds to the target failure probability, Pf.
C          ================================================================
           Call InvNormal(Pf,RelInd)
           Write(40,8792) RelInd
           Write(50,8792) RelInd
 8792      Format(/ 3X,'Reliability Index = ',E17.10 /)
         Endif

C          ================================================================
C          Reads the characteristics of the variables:
C          - Type of distribution, VarDist
C          - Type of truncation, Trunc
C          - Point of truncation, Xo (if the distribution is truncated)
C          - Mean, Mux, standard deviation, Sigmax, & lower limit, Zeta
C          ================================================================
         Do 6633 j=1,NCombAc
           If ((NCombAC.GT.1).AND.(Distr.EQ.2)) then
             Write(40,7073) j
             Write(50,7073) j
 7073        Format(/// 3X,'COMBINATION No.',I2)
           Endif
           Write(40,8777)
           Write(50,8777)
 8777      Format(// 3X, 'CHARACTERISTICS OF THE VARIABLES ' /)
           Do 13 i=1,N
             Read(30,*) VarDis(j,i)
             If (VarDis(j,i).NE.0.AND.VarDis(j,i).NE.1.AND.
     1          VarDis(j,i).NE.2.AND.VarDis(j,i).NE.3.AND.
     1          VarDis(j,i).NE.4.AND.VarDis(j,i).NE.5.AND.
     1          VarDis(j,i).NE.6.AND.VarDis(j,i).NE.7.AND.
     1          VarDis(j,i).NE.8.AND.VarDis(j,i).NE.9.AND.
     1          VarDis(j,i).NE.10.AND.VarDis(j,i).NE.11.AND.
     1          VarDis(j,i).NE.12.AND.VarDis(j,i).NE.13) then
               Write(*,3375) Ext(i)
               Write(50,3375) Ext(i)
 3375          Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
     1                  'Distribution for ',A3,' !')
               STOP
             Endif
```

```
               Trunc(j,i)=0
               If (VarDis(j,i).NE.0.AND.VarDis(j,i).LE.10)
        1          Read(30,*) Trunc(j,i)
               If (Trunc(j,i).NE.0.AND.Trunc(j,i).NE.1.AND.
        1          Trunc(j,i).NE.2) then
                 Write(*,3376) Ext(i)
                 Write(50,3376) Ext(i)
  3376           Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
        1                'Truncation for ',A3,' !')
                 STOP
               Endif
               Xo(j,i)=0.
               If (Trunc(j,i).NE.0) then
C              ----------
C               For the failure mode of overtopping, if the variable is
C               the significant wave height, Hs, & if Hs is limited by the
C               available water depth, then the point of truncation is
C               Xo=0.6(SWL-TL) or Xo=0.6(Tide+Surge-TL), where TL is the
C               seawall toe level.
C              ----------
               If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(i.EQ.2).AND.
        1          (Trunc(j,i).EQ.1)) then
                 Read(30,*) TL
                 goto 5553
               Endif
               Read(30,*) Xo(j,i)
               Endif
               If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(i.EQ.2)) then
                 Read(30,*) TL
                 Write(40,7881) TL
                 Write(50,7881) TL
  7881           Format(11X,'Seawall Toe Level (TL) = ',E17.10)
               Endif
  5553         If (VarDis(j,i).NE.0) then
                 If (VarDis(j,i).EQ.6) then
                   Read(30,*) Mux(j,i),Sigmax(j,i),x1(j,i),x2(j,i)
                   If (x1(j,i).GE.x2(j,i)) then
                     Write(*,8703)
                     Write(50,8703)
  8703               Format(// 11X,'ERROR: a >= b !')
                     STOP
                   Endif
                   goto 5555
                 Endif
                 If (VarDis(j,i).EQ.10) then
                   Read(30,*) Mux(j,i),Sigmax(j,i),Zeta(j,i)
                   goto 5555
                 Endif
                 Read(30,*) Mux(j,i),Sigmax(j,i)
               else
                 Read(30,*) Mux(j,i)
               Endif
  5555         Call WhatDist(i,j,VarDis,Abrev,Carac)
               Call WriCharVar(i,j,VarDis,Trunc,Ext,Carac,Xo,Mux,Sigmax,
        1                      Opt)
  13        continue

C           ================================================================
C           Calculation of the distribution's parameters, Zeta, Lamda &
C           Eta, for each variable.
C           ================================================================
            Write(50,4495)
  4495      Format(// 6X,'DISTRIBUTION`S PARAMETERS')
            Do 6070 i=1,N
              Call Parameters(i,j,Mux,Sigmax,VarDis,Ext)
  6070      continue
            If (Distr.EQ.1) then
              If (NCombAc.GT.1) then
                Call EqCharac(N,NCombAc,Abrev,Carac,VarDis,Trunc,Xo,Mux,
        1                     Sigmax)
                Write(40,9911)
                Write(50,9911)
  9911          Format(// 11X,'THE CHARACTERISTICS OF THE VARIABLES ARE ',
        1                / 11X,'THE SAME FOR ALL COMBINATIONS OF ACTIONS.',
        1                //)
              Endif
              goto 1314
            Endif
  6633    continue

C         ================================================================
C         Reads the correlation coefficients, Rho, of the variables.
C         ================================================================
```

```
1314   Write(40,870)
       Write(50,870)
870     Format(// 3X, 'CORRELATION COEFFICIENTS ' /)
       Do 40 i=1,N
         Do 45 j=1,N
           Read(35,*) Rho(i,j)
           Write(40,78) Ext(i),Ext(j),Rho(i,j)
           Write(50,78) Ext(i),Ext(j),Rho(i,j)
78          Format(11X,'(',A3,',',A3,') = ',E17.10)
45        continue
40       continue
       Do 4005 i=1,N
         Do 4050 j=1,N
           If ((Rho(i,j).NE.Rho(j,i)).OR.(ABS(Rho(i,j)).GT.1)) then
             If (Rho(i,j).NE.Rho(j,i)) then
               Write(*,7801) i,j,j,i
               Write(50,7801) i,j,j,i
7801            Format(// 11X,'ERROR: Rho(',I2,',',I2,
     1                  ') is not equal to Rho(',I2,',',I2,') !')
             else
               Write(*,7901) i,j
               Write(50,7901) i,j
7901            Format(// 11X,'ERROR: |Rho(',I2,',',I2,')| > 1 !')
             Endif
             STOP
           Endif
4050      continue
4005     continue
       Write(40,*) ' '
       Write(50,*) ' '

C       ==================================================================
C       Reads the characteristics of the FORM calculations:
C       - starting value of the variables, StartPt
C       - minimum value, XMin, & maximum value, XMax, of the variables
C       - maximum number of iterations, MaxIter
C       - number of FORM calculations, NumCalc
C       - target values for each FORM calculation, TR
C       - required accuracy of the reliability index, ReqBetaAcc
C       - smoothing coefficient for the iteration process, Smooth
C       - required accuracy of the failure function, ReqOBJFAcc
C       ==================================================================

C       ----------
C       Starting value of the variables, StartPt.
C       ----------
       Do 678 j=1,NCombAC
         If (NCombAC.GT.1) then
           Write(40,7079) j
           Write(50,7079) j
7079        Format(/// 3X,'COMBINATION No.',I2)
         Endif
         If ((Distr.EQ.2).OR.(Comb.EQ.2)) then
           Write(40,132)
           Write(50,132)
132         Format(/ 3X,'STARTING POINT FOR THE FORM CALCULATIONS: '//
     1          11X, 'Default Values (mean values) ... [ 1 ]' /
     1          11X, 'User Specified Values .......... [ 2 ]' /)
           Read(65,*) Def
           If (Def.NE.1.AND.Def.NE.2) then
             Write(*,3307)
             Write(50,3307)
3307          Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
     1             'Starting Point !')
             STOP
           Endif
           Write(40,937) Def
           Write(50,937) Def
937         Format(3X,'Select Option: ',I1)
           Write(40,5666)
           Write(50,5666)
5666        Format(/ 3X,'STARTING POINT' /)
           If (Def.EQ.1) then
             Do 2156 i=1,N
               StartPt(j,i)=Mux(j,i)
               Write(40,5620) Ext(i),StartPt(j,i)
               Write(50,5620) Ext(i),StartPt(j,i)
5620            Format(11X,A3,' = ',E17.10)
2156          continue
           else
             Do 2155 i=1,N
               If (VarDis(j,i).NE.0) then
                 Read(65,*) StartPt(j,i)
```

```
                  Write(40,5600) Ext(i),StartPt(j,i)
                  Write(50,5600) Ext(i),StartPt(j,i)
5600               Format(11X,A3,' = ',E17.10)
                else
                  StartPt(j,i)=Mux(j,i)
                Endif
2155         continue
            Endif
          Endif
          If (Distr.EQ.1) then
            Write(40,5636)
            Write(50,5636)
5636         Format(/ 3X,'STARTING POINT' /)
            Do 2956 i=1,N
              If (NR(j,i).NE.1.) then
                Call Inverse(j,i,Mux,NR,VarDis,StartPt)
              else
                StartPt(j,i)=Mux(j,i)
              Endif
              Write(40,5629) Ext(i),StartPt(j,i)
              Write(50,5629) Ext(i),StartPt(j,i)
5629           Format(11X,A3,' = ',E17.10)
2956         continue
          Endif
678     continue

C       ----------
C       Minimum value, XMin, & maximum value, XMax, of the variables.
C       ----------
        Do 789 j=1,NCombAC
          If ((NCombAC.GT.1).AND.(Distr.EQ.2)) then
            Write(40,7074) j
            Write(50,7074) j
7074         Format(/// 3X,'COMBINATION No.',I2)
          Endif
          Write(40,1327)
          Write(50,1327)
1327       Format(/ 3X,'LIMITING VALUES FOR THE VARIABLES: '//
     1            11X, 'Default Values (+/- 1E25) ... [ 1 ]' /
     1            11X, 'User Specified Values ....... [ 2 ]' /)
          Read(65,*) Def1
          If (Def1.NE.1.AND.Def1.NE.2) then
            Write(*,3371)
            Write(50,3371)
3371         Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
     1              'Limiting Values for the Variables !')
            STOP
          Endif
          Write(40,9377) Def1
          Write(50,9377) Def1
9377       Format(3X,'Select Option: ',I1)
          If (Def1.EQ.1) then
            Do 2176 i=1,N
              If (VarDis(j,i).NE.0) then
                XMin(j,i)=-1E25
                XMax(j,i)=+1E25
                If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(DSWL.EQ.1)) then
                  XMin(j,5)=TL
                  XMax(j,5)=StartParam
                  If (XMin(j,5).GT.XMax(j,5)) then
                    Write(*,8653)
                    Write(50,8653)
8653                 Format(// 11X,'ERROR: The Toe Level is Above the',
     1                     1X,'Seawall Crest Level !')
                    STOP
                  Endif
                Endif
              else
                XMin(j,i)=Mux(j,i)
                XMax(j,i)=Mux(j,i)
              Endif
2176         continue
          else
            Do 2175 i=1,N
              If (VarDis(j,i).NE.0) then
                Read(65,*) XMin(j,i),XMax(j,i)
                If (XMin(j,i).GT.XMax(j,i)) then
                  Write(*,8733) i,i
                  Write(50,8733) i,i
8733               Format(// 11X,'ERROR: XMin(',I2,') > XMax(',I2,') !')
                  STOP
                Endif
                If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(DSWL.EQ.1)) then
```

```
                    XMin(j,5)=TL
                    XMax(j,5)=StartParam
                    If (XMin(j,5).GT.XMax(j,5)) then
                      Write(*,8153)
                      Write(50,8153)
 8153                  Format(// 11X,'ERROR: The Toe Level is Above the',
     1                           1X,'Seawall Crest Level !')
                      STOP
                    Endif
                  Endif
                else
                  XMin(j,i)=Mux(j,i)
                  XMax(j,i)=Mux(j,i)
                Endif
 2175         continue
          Endif
          Write(40,5678)
          Write(50,5678)
 5678      Format(/ 3X,'LIMITING VALUES FOR THE VARIABLES' /)
          Do 2788 i=1,N
            Call MinMax(i,j,Opt,VarDis,Ext,Trunc,Xo,Mux,Sigmax,XMin,
     1                  XMax)
            If (StartPt(j,i).LT.XMin(j,i)) then
              Write(*,8477) Ext(i),StartPt(j,i),Ext(i),XMin(j,i)
              Write(50,8477) Ext(i),StartPt(j,i),Ext(i),XMin(j,i)
 8477          Format(// 3X,'ERROR: Starting Value of ',A3,' = ',E17.10
     1                 / 10X,'< Minimum of ',A3,' = ',E17.10,' !')
              STOP
            Endif
            If (StartPt(j,i).GT.XMax(j,i)) then
              Write(*,8977) Ext(i),StartPt(j,i),Ext(i),XMax(j,i)
              Write(50,8977) Ext(i),StartPt(j,i),Ext(i),XMax(j,i)
 8977          Format(// 3X,'ERROR: Starting Value of ',A3,' = ',E17.10
     1                  / 10X,'> Maximum of ',A3,' = ',E17.10,' !')
              STOP
            Endif
 2788     continue
          If (Distr.EQ.1) then
            If (NCombAc.GT.1) then
              Do 8001 k=2,NCombAc
                Do 7001 i=1,N
                  XMin(k,i)=XMin(1,i)
                  XMax(k,i)=XMax(1,i)
 7001           continue
 8001         continue
              Write(40,9111)
              Write(50,9111)
 9111          Format(// 11X,
     1                    'THE MINIMUM AND MAXIMUM VALUES OF THE VARIABLES' /
     1                    11X,'ARE THE SAME FOR ALL COMBINATIONS OF ACTIONS.'
     1                    //)
            Endif
            goto 1315
          Endif
 789    continue

C       ----------
C       Maximum number of iterations, MaxIter.
C       ----------
 1315  Read(65,*) MaxIter
       If (MaxIter.GT.200.OR.MaxIter.LE.0) then
         Write(*,7851)
         Write(50,7851)
 7851     Format(// 3X,'ERROR: The Maximum Number of Iterations' /
     1             3X,'       is not Within ]0,200] !')
         STOP
       Endif
       Write(40,5995) MaxIter
       Write(50,5995) MaxIter
 5995  Format(// 3X,'MAXIMUM NUMBER OF ITERATIONS (Max=200) = ',I3)

C       ----------
C       Number of FORM calculations, NumCalc.
C       ----------
       Read(65,*) NumCalc
       If (NumCalc.GT.10.OR.NumCalc.LE.0) then
         Write(*,3279)
         Write(50,3279)
 3279     Format(// 3X,'ERROR: The Maximum Number of FORM Calculations'
     1             / 3X,'       is not Within ]0,10] !')
         STOP
       Endif
       Write(40,5005) NumCalc
```

```
          Write(50,5005) NumCalc
  5005    Format(/ 3X,'NUMBER OF FORM CALCULATIONS (Max=10) = ',I2 /)


C         ----------
C         Target values for each FORM calculation, TR.
C         ----------
          Aux=2
          Do 2007 k0=1,NumCalc
            Read(65,*) TR(k0)
            If (TR(k0).LT.0) then
              Write(*,3244)
              Write(50,3244)
  3244       Format(// 3X,
      1               'ERROR: The Allowable Target Value is < 0 !')
              STOP
            Endif
            Call Allowed(Opt,Aux)
  2007    continue

C         ----------
C         Required accuracy of the reliability index, ReqBetaAcc.
C         ----------
          Write(40,1997)
          Write(50,1997)
  1997    Format(// 3X,
      1            'REQUIRED RELATIVE ACCURACY OF THE RELIABILITY INDEX: '//
      1            11X, 'Default Value (1%) ......... [ 1 ]' /
      1            11X, 'User Specified Value ....... [ 2 ]' /)
          Read(65,*) Def2
          If (Def2.NE.1.AND.Def2.NE.2) then
            Write(*,3461)
            Write(50,3461)
  3461     Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
      1            'Accuracy of the Reliability Index ! ')
            STOP
          Endif
          Write(40,9399) Def2
          Write(50,9399) Def2
  9399    Format(3X,'Select Option: ',I1)
          If (Def2.EQ.1) then
            ReqBetaAcc=1.
          else
            Read(65,*) ReqBetaAcc
            If ((ReqBetaAcc.LT.(0.)).OR.(ReqBetaAcc.GT.1.)) then
              Write(*,6677)
              Write(50,6677)
  6677       Format(// 3X,'ERROR: The Required Relative Accuracy of the',
      1              / 10X,'Reliability Index is not Within [0,1] % !')
              STOP
            Endif
            Write(40,9978) ReqBetaAcc
            Write(50,9978) ReqBetaAcc
  9978     Format(// 3X,'Required Relative Accuracy of the',1X,
      1            'Reliability Index [0,1] = ',E17.10 /)
          Endif


C         ----------
C         Smoothing of the iteration process (0<=Smooth<=1).
C         If Smooth=0 then there is no smoothing; if Smooth=0.5 then there
C         is averaging between the last two calculated values of X.
C         ----------
          Write(40,1197)
          Write(50,1197)
  1197    Format(// 3X,'REQUIRED SMOOTHING COEFFICIENT FOR THE',1X,
      1            'ITERATION PROCESS: '//
      1            11X, 'Default Value (0) .......... [ 1 ]' /
      1            11X, 'User Specified Value ....... [ 2 ]' /)
          Read(65,*) Def3
          If (Def3.NE.1.AND.Def3.NE.2) then
            Write(*,3462)
            Write(50,3462)
  3462     Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
      1            'Smoothing Coefficient' / 10X,
      1            'for the Iteration Process ! ')
            STOP
          Endif
          Write(40,9799) Def3
          Write(50,9799) Def3
  9799    Format(3X,'Select Option: ',I1)
          If (Def3.EQ.1) then
            Smooth=0.
          else
            Read(65,*) Smooth
```

```
          If ((Smooth.LT.(0.)).OR.(Smooth.GT.1.)) then
            Write(*,1177)
            Write(50,1177)
 1177      Format(// 3X,'ERROR: The Required Smoothing',1X,
     1            'Coefficient for the' / 10X,
     1            'Iteration Process is not Within [0,1] !')
            STOP
          Endif
          Write(40,1178) Smooth
          Write(50,1178) Smooth
 1178      Format(// 3X,'Required Smoothing Coefficient for',1X,
     1            'the Iteration Process [0,1] = ',E17.10 /)
        Endif

C        ----------
C        Required accuracy of the failure function OBJF, ReqOBJFAcc.
C        If ReqOBJFAcc=1% then ABS(OBJF)<0.01SigmaOBJF, where SigmaOBJF
C        is the standard deviation of the failure function.
C        ----------
        Write(40,3397)
        Write(50,3397)
 3397  Format(// 3X,'REQUIRED ACCURACY OF THE FAILURE FUNCTION: '//
     1          11X, 'Default Value (1%) ......... [ 1 ]' /
     1          11X, 'User Specified Value ....... [ 2 ]' /)
        Read(65,*) Def4
        If (Def4.NE.1.AND.Def4.NE.2) then
          Write(*,3463)
          Write(50,3463)
 3463      Format(// 3X,'ERROR: Wrong Value for the Type of',1X,
     1            'Accuracy of the Failure Function ! ')
          STOP
        Endif
        Write(40,4499) Def4
        Write(50,4499) Def4
 4499  Format(3X,'Select Option: ',I1)
        If (Def4.EQ.1) then
          ReqOBJFAcc=1.
        else
          Read(65,*) ReqOBJFAcc
          If ((ReqOBJFAcc.LT.(0.)).OR.(ReqOBJFAcc.GT.1.)) then
            Write(*,5544)
            Write(50,5544)
 5544      Format(// 3X,'ERROR: The Required Accuracy of the',1X,
     1            'Failure' / 10X,
     1            'Function is not Within [0,1] % !')
            STOP
          Endif
          Write(40,9448) ReqOBJFAcc
          Write(50,9448) ReqOBJFAcc
 9448      Format(// 3X,'Required Accuracy of the Failure',1X,
     1            'Function [0,1] = ',E17.10 /)
        Endif

        Close (Unit=35)
        Close (Unit=15)
        Close (Unit=30)
        Close (Unit=45)
        Close (Unit=65)
        return
        End



C        ###############################################################
C
        Subroutine D1Point(j,N,Opt,Mux,Sigmax,VarDis,Abrev,Rho,FDer,
     1                     NR,StartParam,Prob,Ext,ExtExt,ParamDesc,
     1                     Trunc,Xo,MaxIter,StartPt,XMax,XMin,
     1                     ReqBetaAcc,Smooth,ReqOBJFAcc,Comb,Ex)
C

C        ###############################################################
C
C        Returns the failure probability, Prob, for a given value of the
C        design parameter, StartParam - Mode 1 (analysis mode).
C
C        ###############################################################
C
C        INPUT VARIABLES:
C        Opt - Failure mode
C        DSWL - Definition of the SWL
C        TL - Seawall toe level
C        Comb - Consideration or not of combination of actions
```

```
C          j - Number of the combination of actions
C          NR - Power to which each distribution is raised for each
C               combination of actions
C          N - Number of variables
C          Ext - Abbreviation of the name of the variable
C          ExtExt - Description of the variable
C          ParamDesc - Description of the design parameter
C          Rho - Correlation coefficient
C          Mux - Mean of X
C          Sigmax - Standard deviation of X
C          VarDis - Type of distribution
C          Abrev - Abbreviation of the name of the distribution
C          Trunc - Type of truncation
C          Xo - Point of truncation (if the distribution is truncated)
C          XMin - Minimum value of X
C          XMax - Maximum value of X
C          Zeta - Parameter of a distribution
C          StartPt - Starting value of the variables
C          FDer - Method of calculation of the first partial derivatives of
C                  the failure function for overtopping
C          MaxIter - Maximum number of iterations
C          k0 - Number of the FORM calculation
C          TR - Target values for each FORM calculation
C          ReqBetaAcc - Required relative accuracy of the reliability index
C          Smooth - Smoothing coefficient for the iteration process
C          ReqOBJFAcc- Required accuracy of the failure function
C          StartParam - Prescribed value of the design parameter
C
C          MODELING VARIABLES:
C          Q - Maximum number of combinations of actions allowed by the
C              program
C          L - Maximum number of variables allowed by the program
C          L1 - Maximum number of iterations allowed by the program
C          X - Variables of the failure mode
C          XOld - X in the previous iteration
C          MuxN - Mean of the equivalent Normal distribution of X
C          SigmaxN - Standard deviation of the equivalent Normal
C                    distribution of X
C          Y - Non-correlated, Normal transformed variables
C          YOld - Y in the previous iteration
C          Muy - Mean of Y
C          Sigmay - Standard deviation of Y
C          V - Matrix of eigenvectors
C          Vt - Transpose of V
C          OBJF - Failure function
C          MuOBJF - Mean of OBJF
C          SigmaOBJF - Standard deviation of OBJF
C          VarOBJF - Variance of OBJF
C          OBJGRD - First partial derivatives of OBJF
C          NSTATE - Variable used by subroutine EO4XAF (for more details
C                   see NAG, 1993)
C          It - Iteration number
C          L2 - Maximum number of FORM calculations allowed by the program
C          BetaAcc -  Relative accuracy of the reliability index
C          Inf - Influence of each variable on the reliability index
C          Alpha - Sensitivity factors
C          Step - Change in X if Smooth=0 & if the iteration process is in
C                  a loop
C          k1 - Number of times that X is changed if Smooth=0 & if the
C               iteration process is in a loop
C          OBJFAcc - Accuracy of the failure function
C          Count - Number of variables for which OBJGRD=0
C          RelInd - Reliability index
C          Prob2 - Probability of failure which corresponds to RelInd
C          RelOld - Reliabiliy index in the previous iteration
C          Prob1 - Probability of failure which corresponds to RelOld
C          DifProb - Difference in Prob between the last two iterations
C          nourtlev - Nourishment top level
C          1:mnour - Gradient of the nourished face
C          Pi - 3.14159...
C          i, ii, iii, k, Aux, Aux1, AuxRstar,
C          Out, OBJGRDPrev, Mu1OBJF, Sin - Auxiliary variables
C          ctcurv, md, mt, C, T3,
C          Lamda, Eta, x1, x2 - Variables mentioned in the Common
C                               statements but not used here
C
C          OUTPUT VARIABLES:
C          Prob - Probability of failure which corresponds to RelInd
C          Ex - Auxiliary variable
C
C          ############################################################
C
           Integer*4 i,j,k,k0,L,L2,N,Opt,L1,It,FDer,NSTATE,Q,MaxIter,Count,
```

```
     1              Aux,Comb,Aux1,AuxRstar,Ex,DSWL,ii,iii
        Parameter (L=15)
        Parameter (L2=10)
        Parameter (Q=16)
        Parameter (L1=200)
        Character*3 Abrev(Q,L),Ext(L)
        Character*17 ExtExt(L)
        Character*19 ParamDesc
        Integer*4 VarDis(Q,L),Trunc(Q,L),Out(L),k1(L),Sin(L1,L)
        Real*8 X(L),XOld(L,L1),Mux(Q,L),MuOBJF,Mu1OBJF,Sigmax(Q,L),
     1        Xo(Q,L),SigmaOBJF,Prob(Q),OBJGRD(L),OBJF,VarOBJF,
     1        Rho(L,L),Alpha(L),V(L,L),Vt(L,L),Sigmay(L),Muy(L),Y(L),
     1        Inf(L), OBJGRDPrev(L),Zeta(Q,L),Lamda(Q,L),Pi,SigmaxN(L),
     1        YOld(L,L1),x1(Q,L),x2(Q,L),Eta(Q,L),RelInd,Startparam,
     1        MuxN,TL,C,T3,md,mt,mnour,nourtlev,ctcurv,TR(L2),Step(L),
     1        StartPt(Q,L),XMax(Q,L),XMin(Q,L),BetaAcc,ReqBetaAcc,
     1        RelOld,Smooth,Prob1,Prob2,DifProb,ReqOBJFAcc,OBJFAcc,
     1        NR(Q,L)
        Common/BLOCK3/MuxN(15),C,T3
        Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
        Common/BLOCK7/TR
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
        Common/BLOCK9/k0,It,AuxRstar
        Common/BLOCK11/DSWL,TL

        Pi=4.*ATAN(1.)
        RelInd=0.
        Do 3325 i=1,N
          Sin(1,i)=0
          Step(i)=0.001
          Out(i)=0
          k1(i)=1
 3325   continue

        Write(*,25)
        Write(40,25)
        Write(50,25)
 25     Format(/)
        Aux1=1
        Call Allowed(Opt,Aux1)

        Do 7855 i=1,N
          X(i)=StartPt(j,i)
 7855   continue

C       =================================================================
C       Definition of the iteration number, It.
C       =================================================================
        It=1
 53     Write(*,86) It
        Write(50,86) It
 86     Format(/ 6X,'ITERATION No.',I3)
 54     It=It+1

C       =================================================================
C       For the failure mode of overtopping, if the variable is the
C       significant wave height, Hs=X(2), & if Hs is limited by the
C       available water depth, then the point of truncation is
C       Xo=0.6(SWL-TL) or Xo=0.6(Tide+Surge-TL).
C       =================================================================
        If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(Trunc(j,2).EQ.1)) then
          If (DSWL.EQ.1) then
            Xo(j,2)=0.6*(X(5)-TL)
          else
            Xo(j,2)=0.6*(X(5)+X(6)-TL)
          Endif
          Write(50,4444) Ext(2),Xo(j,2)
 4444     Format(// 16X,'Xo(',A3,') = ',E17.10)
          If (XMax(j,2).NE.Xo(j,2)) then
            XMax(j,2)=Xo(j,2)
            Write(50,5971) Ext(2),XMax(j,2)
 5971       Format(16X,'XMax(',A3,') = ',E17.10)
            If (XMin(j,2).GT.XMax(j,2)) then
              Write(*,8703)
              Write(50,8703)
 8703         Format(// 11X,'ERROR: XMin(2) > XMax(2) !')
              STOP
            Endif
            If (X(2).GT.XMax(j,2)) then
              Write(*,8367) Ext(2),X(2),XMax(j,2)
              Write(50,8367) Ext(2),X(2),XMax(j,2)
 8367         Format(/ 11X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
              X(2)=XMax(j,2)
```

```
               Write(*,8471) Ext(2),X(2)
               Write(50,8471) Ext(2),X(2)
 8471           Format(11X,A3,' = ',E17.10 /)
             Endif
           Endif
         Endif

C         =================================================================
C         Transformation of non-Normal correlated variables (X with
C         mean=Mux & standard deviation=Sigmax) to Normal correlated
C         variables (X with mean=MuxN & standard deviation=SigmaxN).
C         =================================================================
         Write(50,4395)
 4395    Format(// 11X,'EQUIVALENT NORMAL DISTRIBUTION`S PARAMETERS')
         Do 7870 i=1,N
           Call EqNorDis(i,j,N,X,VarDis,NR,Ext,MuxN,SigmaxN,Trunc,Xo)
 7870    continue
         If (Opt.EQ.23) SigmaxN(5)=0.11*(X(4)-2.25)

C         =================================================================
C         Transformation of correlated Normal variables (X with mean=MuxN
C         & standard deviation=SigmaxN) to non-correlated Normal variables
C         (Y with mean=Muy & standard deviation=Sigmay).
C         =================================================================
         Call Correlated(N,Ext,MuxN,SigmaxN,Rho,Sigmay,Muy,V,Vt)
         Write(50,*) ' '
         Do 635 i=1,N
           Y(i)=0.
           Do 637 k=1,N
             Y(i)=Vt(i,k)*X(k)+Y(i)
 637        continue
           Write(50,1998) Ext(i),Y(i)
 1998       Format(11X,'Y(',A3,') = ',E17.10)
 635      continue
         Write(50,*) ' '

C         =================================================================
C         Calculation of the failure function, OBJF, & its derivatives,
C         OBJGRD, at the design point.
C         =================================================================
         If (FDer.EQ.2) then
           Call Derivadas(StartParam,N,X,Ext,Opt,FDer,OBJF,OBJGRD)
         else
           Call OBJFUN(FDer,N,X,OBJF,OBJGRD,NSTATE,Opt,StartParam)
           Do 5690 i=1,N
             Write(50,9995) Ext(i),OBJGRD(i)
 9995         Format(11X,'dZ/d',A3,' = ',E17.10)
 5690       continue
         Endif

         If (AuxRstar.EQ.1) then
           If (It.NE.2) X(1)=(X(1)+XOld(1,It-1))/2.
           If ((X(1).LT.XMin(j,1)).OR.(X(1).GT.XMax(j,1))) then
             If (X(1).LT.XMin(j,1)) then
               Write(*,8976) Ext(1),X(1),XMin(j,1)
               Write(50,8976) Ext(1),X(1),XMin(j,1)
 8976           Format(/ 11X,A3,' = ',E17.10,' < ',E17.10, ' ! ' /)
               X(1)=1.001*XMin(j,1)
C               X(1)=XMin(j,1)+(XOld(1,It-1)-XMin(j,1))/2.
             else
               Write(*,8461) Ext(1),X(1),XMax(j,1)
               Write(50,8461) Ext(1),X(1),XMax(j,1)
 8461           Format(/ 3X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
               X(1)=1.001*XMax(j,1)
C               X(1)=XOld(1,It-1)+(XMax(j,1)-XOld(1,It-1))/2.
             Endif
             Write(50,8109) Ext(1),X(1)
 8109         Format(/ 11X,A3,' = ',E17.10 /)
             goto 53
           Endif
           Write(50,8277) Ext(1),X(1)
 8277       Format(/ 11X,A3,' = ',E17.10 /)
           goto 53
         Endif

         Write(50,9312) OBJF
 9312    Format(/ 11X,'Z = ',E17.10 /)
         Count=0
         Do 409 i=1,N
           OBJGRDPrev(i)=OBJGRD(i)
           If (ABS(OBJGRDPrev(i)).LT.(1E-25)) Count=Count+1
 409      continue
         If (Count.EQ.N) then
```

```
          Write(*,99)
          Write(50,99)
 99        Format(// 11X,'ERROR: The derivatives are zero !')
          STOP
        Endif
        Do 402 i=1,N
          OBJGRD(i)=0.
          Do 450 k=1,N
            OBJGRD(i)=OBJGRDPrev(k)*V(k,i)+OBJGRD(i)
 450       continue
          Write(50,810) i,OBJGRD(i)
 810       Format(11X, 'dZ/dY(',I2,') = ',E17.10)
 402     continue

C        =================================================================
C        Calculation of the mean value, MuOBJF, & the standard deviation,
C        SigmaOBJF, of the failure function. Calculation of the
C        reliability index, RelInd.
C        =================================================================
        Mu1OBJF=0
        VarOBJF=0
        Do 84 i=1,N
          Mu1OBJF=(Muy(i)-Y(i))*OBJGRD(i)+Mu1OBJF
          VarOBJF=((OBJGRD(i)*Sigmay(i))**2)+VarOBJF
 84      continue
        MuOBJF=OBJF+Mu1OBJF
        SigmaOBJF=SQRT(VarOBJF)
        RelOld=RelInd
        RelInd=MuOBJF/SigmaOBJF
        Write(50,33) MuOBJF,SigmaOBJF,RelInd,RelOld
 33      Format(/ 11X,'Mean Value of Z = ',E17.10 / 11X,
     1            'Standard Deviation of Z = ',E17.10 / 11X,
     1            'New Reliability Index = ',E17.10 / 11X,
     1            'Old Reliability Index = ',E17.10 / )

C        =================================================================
C        Calculation of the sensitivity factors, Alpha, & the new design
C        point, Y (& consequently X).
C        =================================================================
        Do 9009 i=1,N
          Alpha(i)=(Sigmay(i)/SigmaOBJF)*OBJGRD(i)
          YOld(i,It-1)=Y(i)
          Y(i)=Muy(i)-Alpha(i)*RelInd*Sigmay(i)
 9009    continue

        Do 1635 i=1,N
          XOld(i,It-1)=X(i)
          X(i)=0.
          Do 1637 k=1,N
            X(i)=V(i,k)*Y(k)+X(i)
 1637      continue
          Write(50,1187) Ext(i),XOld(i,It-1),Ext(i),X(i)
 1187      Format(11X,'XOld(',A3,') = ',E17.10,3X,A3,' = ',E17.10)
 1635    continue
        Write(50,*) ' '
        Do 88 i=1,N
          Write(50,87) i,Alpha(i)
 87        Format(11X,'Alpha(Y',I2,') = ',E17.10)
 88      continue

C        =================================================================
C        If the new calculated design point lies outside [XMin,XMax], the
C        program continues calculations using a new design point between
C        the last computed design point & the boundary which was
C        exceeded.
C        =================================================================
        Do 9895 i=1,N
          If ((X(i).LT.XMin(j,i)).OR.(X(i).GT.XMax(j,i))) then
            If (X(i).LT.XMin(j,i)) then
              Write(*,8977) Ext(i),X(i),XMin(j,i)
              Write(50,8977) Ext(i),X(i),XMin(j,i)
 8977          Format(/ 11X,A3,' = ',E17.10,' < ',E17.10, ' ! ' /)
              X(i)=XMin(j,i)+(XOld(i,It-1)-XMin(j,i))/2.
            else
              Write(*,8467) Ext(i),X(i),XMax(j,i)
              Write(50,8467) Ext(i),X(i),XMax(j,i)
 8467          Format(/ 3X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
              X(i)=XOld(i,It-1)+(XMax(j,i)-XOld(i,It-1))/2.
            Endif
            Write(50,8166) Ext(i),X(i)
 8166        Format(/ 11X,A3,' = ',E17.10 /)
            Aux=1
          else
```

```
           Aux=0
         Endif
 9895   continue
        If (Aux.NE.0) goto 53

C       ================================================================
C       Check convergence at the design point.
C       ================================================================
        Do 85 i=1,N
          If (((ABS(XOld(i,It-1)/X(i)-1).GT.0.0001).AND.(X(i).NE.0.))
     1       .OR.((X(i).EQ.0.).AND.(X(i)-XOld(i,It-1).GT.0.00001))) then
            If (It.GT.MaxIter) then
              Write(*,9049)
              Write(40,9049)
              Write(50,9049)
 9049         Format(/ 11X,'Not converging at the design point !'///
     1                6X,'CONVERGENCE NOT FOUND WITHIN THE MAXIMUM',
     1                1X,'NUMBER OF ITERATIONS !')
              goto 9999
            else
              goto 13
            Endif
          Endif
 85     continue
        goto 23

C       ================================================================
C       Smoothing of the iteration process (0<=Smooth<=1).
C       ================================================================
 13     Write(50,*) ' '
        Do 4488 i=1,N
          If (Smooth.EQ.0.) then
            If (It.GT.3) then
              If (((ABS(XOld(i,It-1)/X(i)-1).GT.0.0001).AND.
     1           (X(i).NE.0.)).OR.((X(i).EQ.0.).AND.
     1           (X(i)-XOld(i,It-1).GT.0.00001)).OR.(Out(i).EQ.1)) then
                If (((ABS(XOld(i,It-2)/X(i)-1).LE.0.0001).AND.
     1           (X(i).NE.0.)).OR.((X(i).EQ.0.).AND.
     1           (X(i)-XOld(i,It-2).LE.0.00001)).OR.
     1           ((ABS(XOld(i,It-3)/X(i)-1).LE.0.0001)
     1           .AND.(X(i).NE.0.)).OR.((X(i).EQ.0.).AND.
     1           (X(i)-XOld(i,It-3).LE.0.00001)).OR.(Out(i).EQ.1)) then
                  Write(*,*) '*****'
                  Write(50,*) '*****'
                  k1(i)=k1(i)+1
                  If (X(i).GT.XOld(i,It-1)) then
                    Sin(k1(i),i)=4
                    If (((Sin(k1(i)-1,i)-Sin(k1(i),i)).NE.0).AND.
     1               (k1(i).GT.2)) Step(i)=Step(i)/10.
                    X(i)=XOld(i,It-1)+Step(i)*ABS(XOld(i,It-1)-X(i))
                  Endif
                  If (X(i).LT.XOld(i,It-1)) then
                    Sin(k1(i),i)=1
                    If (((Sin(k1(i)-1,i)-Sin(k1(i),i)).NE.0).AND.
     1               (k1(i).GT.2)) Step(i)=Step(i)/10.
                    X(i)=X(i)+Step(i)*ABS(XOld(i,It-1)-X(i))
                  Endif
                  Write(50,2237) Ext(i),X(i)
 2237             Format(11X,A3,' = ',E17.10)
                  Out(i)=1
                Endif
              Endif
            Endif
          else
            X(i)=(1.-Smooth)*X(i)+Smooth*XOld(i,It-1)
            Write(50,2287) Ext(i),X(i)
 2287       Format(11X,A3,' = ',E17.10)
          Endif
 4488   continue
        If (Smooth.EQ.0.) then
          Do 8509 i=1,N
            If (((ABS(XOld(i,It-1)/X(i)-1).GT.0.0001).AND.(X(i).NE.0.))
     1         .OR.((X(i).EQ.0.).AND.(X(i)-XOld(i,It-1).GT.0.00001))) then
              Write(*,949) It
              Write(50,949) It
 949          Format(/ 11X,'Not converging at the design point !' ///
     1                6X,'ITERATION No.',I3)
              goto 54
            Endif
 8509     continue
        else
          Write(*,9491) It
          Write(50,9491) It
```

```
 9491      Format(/ 11X,'Not converging at the design point !' ///
      1            6X,'ITERATION No.',I3)
           goto 54
         Endif

C       ================================================================
C       Check value of the failure function at the design point.
C       ================================================================
  23     OBJFAcc=ReqOBJFAcc*SigmaOBJF/100.
         If (ABS(OBJF).GT.OBJFAcc) then
           If (It.GT.MaxIter) then
             Write(*,9339)
             Write(40,9339)
             Write(50,9339)
 9339      Format(/ 11X,'The accuracy of the failure function'
      1            / 11X,'is less than the required value !'
      1            /// 6X,'CONVERGENCE NOT FOUND WITHIN THE MAXIMUM',
      1            1X,'NUMBER OF ITERATIONS !')
             goto 9999
           Endif
           Write(*,3394) It
           Write(50,3394) It
 3394      Format(/ 11X,'The accuracy of the failure function'
      1            / 11X,'is less than the required value !'
      1            /// 6X,'ITERATION No.',I3)
           goto 54
         Endif

C       ================================================================
C       Check relative accuracy of the reliability index.
C       ================================================================
         BetaAcc=100.*ABS((RelInd-RelOld)/RelInd)
         If (BetaAcc.GT.ReqBetaAcc) then
           If (It.GT.MaxIter) then
             Write(*,9419)
             Write(40,9419)
             Write(50,9419)
 9419      Format(/11X,'The relative accuracy of the reliability index'
      1            / 11X,'is less than the required value !'
      1            /// 6X,'CONVERGENCE NOT FOUND WITHIN THE MAXIMUM',
      1            1X,'NUMBER OF ITERATIONS !')
             goto 9999
           Endif
           Write(*,2394) It
           Write(50,2394) It
 2394      Format(/11X,'The relative accuracy of the reliability index'
      1            / 11X,'is less than the required value !'
      1            /// 6X,'ITERATION No.',I3)
           goto 54
         Endif

C       ================================================================
C       Calculation of the probability of failure & of the difference in
C       Pf between the last 2 iterations.
C       ================================================================
         Call NormalDist(j,RelOld,Prob)
         Prob1=Prob(j)
         Call NormalDist(j,RelInd,Prob)
         Prob2=Prob(j)
         DifProb=ABS(Prob1-Prob2)
         Write(50,1009) (100.*Prob(j)),DifProb
 1009  Format(// 11X,'Probability of Failure (%) = ', E17.10 /
      1        11X,'Difference in Pf Between the Last 2 Iterations = ',
      1        E17.10 /)

C       ================================================================
C       Print the final results.
C       ================================================================
         It=It-1
         Write(40,8860)
         Write(50,8860)
 8860  Format(// 3X,'FINAL RESULTS' //)
         Write(40,100) It,OBJF,MuOBJF,SigmaOBJF,RelInd,BetaAcc,
      1                (100.*Prob(j)),DifProb
         Write(50,100) It,OBJF,MuOBJF,SigmaOBJF,RelInd,BetaAcc,
      1                (100.*Prob(j)),DifProb
 100   Format(11X,'Total Number of Iterations = ',I3 / 11X,
      1        'Failure Function Z (X) = ',E17.10 / 11X,
      1        'Mean Value of Z = ', E17.10 / 11X,
      1        'Standard Deviation of Z = ', E17.10 / 11X,
      1        'Reliability Index = ', E17.10 / 11X,
      1        'Relative Accuracy of the Reliability Index (%) = ',
      1         E17.10 / 11X,'Probability of Failure (%) = ', F10.6 /
```

```
      1          11X,'Difference in Pf Between the Last 2 Iterations = ',
      1             E17.10)
        Write(40,922)
        Write(50,922)
 922    Format(// 11X, 'DESIGN POINT COORDINATES' /)
        Do 539 i=1,N
          Write(40,726) Ext(i),X(i)
          Write(50,726) Ext(i),X(i)
 726      Format(19X,A3,' = ',E17.10)
 539    continue
        Write(40,1677)
        Write(50,1677)
 1677   Format(/)
        iii=0
        Do 444 i=1,N
          Inf(i)=((Alpha(i))**2)*100
          Do 3359 ii=1,N
            If (i.NE.ii) then
              If (Rho(i,ii).NE.0.) iii=1
            Endif
 3359      continue
          If (iii.EQ.1) then
            Write(40,459) i,Alpha(i),Ext(i),Inf(i)
            Write(50,459) i,Alpha(i),Ext(i),Inf(i)
 459        Format(11X,'Alpha(Y',I2,') = ',E17.10 / 11X,
      1             'Influence of Y(',A3,') on the Reliability',1X,
      1             'Index = ',E14.7 /)
            iii=0
          else
            Write(40,449) Ext(i),Alpha(i),Ext(i),Inf(i)
            Write(50,449) Ext(i),Alpha(i),Ext(i),Inf(i)
 449        Format(11X,'Alpha(',A3,') = ',E17.10 / 11X,
      1             'Influence of ',A3,' on the Reliability',1X,
      1             'Index = ',E14.7 /)
          Endif
 444    continue
 9999   Ex=1
        return
        End




C       ################################################################
C
        Subroutine Allowed(Opt,Aux)
C
C       ################################################################
C
C       Writes the target value, TR, for each failure mode in the files
C       summary.dat & results.dat .
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Opt - Failure mode
C       TR - Target values for each FORM calculation
C       k0 - Number of the FORM calculation
C       Aux - Auxiliary variable
C
C       MODELING VARIABLES:
C       L2 - Maximum number of FORM calculations allowed by the program
C       It, AuxRstar - Variables mentioned in the Common statements but
C                      not used here
C
C       ################################################################
C
        Integer*4 Opt,L2,k0,Aux,It,AuxRstar
        Parameter (L2=10)
        Real*8 TR(L2)
        Common/BLOCK7/TR
        Common/BLOCK9/k0,It,AuxRstar
        If (Opt.NE.3) then
          If ((Opt.EQ.1).OR.(Opt.EQ.2)) then
            If (Aux.EQ.1) Write(*,5023) k0,TR(K0)
            Write(40,5023) k0,TR(K0)
            Write(50,5023) k0,TR(K0)
 5023       Format(3X,'ALLOWABLE DISCHARGE - m3/s/m (',I2,') = ',
      1             E10.3)
          else
            If (Aux.EQ.1) Write(*,5077) k0,TR(K0)
            Write(40,5077) k0,TR(K0)
            Write(50,5077) k0,TR(K0)
 5077       Format(3X,'ALLOWABLE TARGET (',I2,') = ',E10.3)
```

```
          Endif
        else
          If (Aux.EQ.1) Write(*,5022) k0,TR(K0)
          Write(40,5022) k0,TR(K0)
          Write(50,5022) k0,TR(K0)
 5022      Format(3X,'ALLOWABLE EROSION DISTANCE - m (',I2,') = ',F6.2)
        Endif
        return
        End




C        #################################################################
C
         Subroutine D2Point(j,N,Opt,Mux,Sigmax,VarDis,Abrev,Rho,FDer,
     1                     NR,StartParam,Pf,Ext,ExtExt,ParamDesc,Trunc,
     1                     Xo,RelInd,MaxIter,StartPt,XMax,XMin,Smooth,
     1                     ReqOBJFAcc,Comb)
C
C        #################################################################
C
C        Calculates the value of the design parameter, Param, for a
C        target value of the failure probability, Pf - Mode 2
C        (design mode).
C
C        #################################################################
C
C        INPUT VARIABLES:
C        Opt - Failure mode
C        DSWL - Definition of the SWL
C        TL - Seawall toe level
C        Comb - Consideration or not of combination of actions
C        j - Number of the combination of actions
C        NR - Power to which each distribution is raised for each
C             combination of actions
C        N - Number of variables
C        Ext - Abbreviation of the name of the variable
C        ExtExt - Description of the variable
C        ParamDesc - Description of the design parameter
C        Rho - Correlation coefficient
C        Mux - Mean of X
C        Sigmax - Standard deviation of X
C        VarDis - Type of distribution
C        Abrev - Abbreviation of the name of the distribution
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        XMin - Minimum value of X
C        XMax - Maximum value of X
C        Zeta - Parameter of a distribution
C        StartPt - Starting value of the variables
C        FDer - Method of calculation of the first partial derivatives of
C               the failure function for overtopping
C        MaxIter - Maximum number of iterations
C        TR - Target values for each FORM calculation
C        k0 - Number of the FORM calculation
C        Smooth - Smoothing coefficient for the iteration process
C        ReqOBJFAcc- Required accuracy of the failure function
C        Pf - Design target failure probability
C        RelInd - Reliability index which corresponds to Pf
C        StartParam - Starting value of the design parameter (to start
C                     iteration)
C
C        MODELING VARIABLES:
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        L - Maximum number of variables allowed by the program
C        L1 - Maximum number of iterations allowed by the program
C        X - Variables of the failure mode
C        XOld - X in the previous iteration
C        MuxN - Mean of the equivalent Normal distribution of X
C        SigmaxN - Standard deviation of the equivalent Normal
C                  distribution of X
C        Y - Non-correlated, Normal transformed variables
C        YOld - Y in the previous iteration
C        Muy - Mean of Y
C        Sigmay - Standard deviation of Y
C        V - Matrix of eigenvectors
C        Vt - Transpose of V
C        OBJF, OBJFOld - Failure function
C        MuOBJF - Mean of OBJF
C        SigmaOBJF - Standard deviation of OBJF
C        VarOBJF - Variance of OBJF
C        OBJGRD - First partial derivatives of OBJF
```

```
C         NSTATE - Variable used by subroutine EO4XAF (for more details
C                  see NAG, 1993)
C         ItInt - Total number of iterations
C         It - Iteration number for each value of Param
C         L2 - Maximum number of FORM calculations allowed by the program
C         Inf - Influence of each variable on the reliability index
C         Alpha - Sensitivity factors
C         Step - Change in X if Smooth=0 & if the iteration process is in
C                  a loop
C         k1 - Number of times that X is changed if Smooth=0 & if the
C                  iteration process is in a loop
C         OBJFAcc - Accuracy of the failure function
C         Count - Number of variables for which OBJGRD=0
C         Param - Value of the design parameter which corresponds to Pf
C         dparamdOBJF - Inverse of the first partial derivative of OBJF
C                        with respect to Param
C         nourtlev - Nourishment top level
C         1:mnour - Gradient of the nourished face
C         Pi - 3.14159...
C         i, ii, iii, k, Aux, Aux1, Out, OBJGRDPrev,
C         Mu1OBJF, Sin, AuxRstar - Auxiliary variables
C         ctcurv, md, mt, C, T3, Lamda,
C         Eta, x1, x2 - Variables mentioned in the Common statements but
C                        not used here
C
C         ############################################################
C
          Integer*4 i,j,k,k0,L,L1,L2,N,Opt,It,FDer,NSTATE,Q,MaxIter,ItInt,
     1             Count,Aux,Comb,Aux1,AuxRstar,DSWL,ii,iii
          Parameter (L=15)
          Parameter (L2=10)
          Parameter (L1=200)
          Parameter (Q=16)
          Character*3 Ext(L),Abrev(Q,L)
          Character*17 ExtExt(L)
          Character*19 ParamDesc
          Integer*4 VarDis(Q,L),Trunc(Q,L),Sin(L1,L),Out(L),k1(L)
          Real*8 X(L),XOld(L,L1),Mux(Q,L),MuOBJF,Mu1OBJF,Sigmax(Q,L),
     1         SigmaOBJF,VarOBJF,Rho(L,L),Alpha(L),Inf(L),V(L,L),
     1         Vt(L,L),Sigmay(L),Muy(L),Y(L),YOld(L,L1),OBJGRD(L),
     1         OBJGRDPrev(L),Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),
     1         x2(Q,L),Pi,OBJF,Param,Pf,StartParam,RelInd,MuxN,
     1         SigmaxN(L),dparamdOBJF,OBJFOld,TL,Xo(Q,L),TR(L2),
     1         StartPt(Q,L),XMax(Q,L),XMin(Q,L),Smooth,ReqOBJFAcc,
     1         OBJFAcc,Step(L),NR(Q,L),md,mt,mnour,nourtlev,C,T3
          Common/BLOCK3/MuxN(15),C,T3
          Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
          Common/BLOCK7/TR
          Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
          Common/BLOCK9/k0,It,AuxRstar
          Common/Block11/DSWL,TL

          Pi=4.*ATAN(1.)
          ItInt=0
          Do 3325 i=1,N
            Sin(1,i)=0
            Step(i)=0.001
            Out(i)=0
            k1(i)=1
 3325     continue
          Param=StartParam

          Write(*,8877)
          Write(40,8877)
          Write(50,8877)
 8877     Format(/)
          Aux1=1
          Call Allowed(Opt,Aux1)

          Do 7855 i=1,N
            X(i)=StartPt(j,i)
 7855     continue

C         ================================================================
C         Definition of the iteration number, It.
C         ================================================================
          It=1
 5399     Write(*,860) It
          Write(50,860) It
 860      Format(/ 6X,'ITERATION No.',I3)
 5499     It=It+1
          XMax(j,5)=Param
          If (Opt.EQ.1.OR.Opt.EQ.2) then
```

```
            If (VarDis(j,5).EQ.11) XMax(j,5)=5.77
            If (VarDis(j,5).EQ.12) XMax(j,5)=6.995
            If (VarDis(j,5).EQ.13) XMax(j,5)=5.57
          Endif
          If (Opt.EQ.3) then
            If (VarDis(j,1).EQ.11) XMax(j,1)=5.77
            If (VarDis(j,1).EQ.12) XMax(j,1)=6.995
            If (VarDis(j,1).EQ.13) XMax(j,1)=5.57
          Endif

C       ==================================================================
C       For the failure mode of overtopping, if the variable is the
C       significant wave height, Hs=X(2), & if Hs is limited by the
C       available water depth, then the point of truncation is
C       Xo=0.6(SWL-TL) or Xo=0.6(Tide+Surge-TL).
C       ==================================================================
          If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(Trunc(j,2).EQ.1)) then
            If (DSWL.EQ.1) then
              Xo(j,2)=0.6*(X(5)-TL)
            else
              Xo(j,2)=0.6*(X(5)+X(6)-TL)
            Endif
            Write(50,4444) Ext(2),Xo(j,2)
 4444       Format(// 16X,'Xo(',A3,') = ',E17.10)
            If (XMax(j,2).NE.Xo(j,2)) then
              XMax(j,2)=Xo(j,2)
              Write(50,5971) Ext(2),XMax(j,2)
 5971         Format(16X,'XMax(',A3,') = ',E17.10)
              If (XMin(j,2).GT.XMax(j,2)) then
                Write(*,8703)
                Write(50,8703)
 8703           Format(// 11X,'ERROR: XMin(2) > XMax(2) !')
                STOP
              Endif
              If (X(2).GT.XMax(j,2)) then
                Write(*,8367) Ext(2),X(2),XMax(j,2)
                Write(50,8367) Ext(2),X(2),XMax(j,2)
 8367           Format(/ 11X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
                X(2)=XMax(j,2)
                Write(*,8471) Ext(2),X(2)
                Write(50,8471) Ext(2),X(2)
 8471           Format(11X,A3,' = ',E17.10 /)
              Endif
            Endif
          Endif

C       ==================================================================
C       Transformation of non-Normal correlated variables (X with
C       mean=Mux & standard deviation=Sigmax) to Normal correlated
C       variables (X with mean=MuxN & standard deviation=SigmaxN).
C       ==================================================================
          Write(50,4395)
 4395     Format(// 11X,'EQUIVALENT NORMAL DISTRIBUTION`S PARAMETERS')
          Do 7870 i=1,N
            Call EqNorDis(i,j,N,X,VarDis,NR,Ext,MuxN,SigmaxN,Trunc,Xo)
 7870     continue

C       ==================================================================
C       Transformation of correlated Normal variables (X with mean=MuxN
C       & standard deviation=SigmaxN) to non-correlated Normal variables
C       (Y with mean=Muy & standard deviation=Sigmay).
C       ==================================================================
          Call Correlated(N,Ext,MuxN,SigmaxN,Rho,Sigmay,Muy,V,Vt)
          Write(50,*) ' '
          Do 6351 i=1,N
            Y(i)=0.
            Do 6371 k=1,N
              Y(i)=Vt(i,k)*X(k)+Y(i)
 6371       continue
            Write(50,2998) Ext(i),Y(i)
 2998       Format(11X,'Y(',A3,') = ',E17.10)
 6351     continue

C       ==================================================================
C       Calculation of the failure function, OBJF, & its derivatives,
C       OBJGRD, at the design point.
C       ==================================================================
          If (FDer.EQ.2) then
            Call Derivadas(Param,N,X,Ext,Opt,FDer,OBJF,OBJGRD)
          else
            Call OBJFUN(FDer,N,X,OBJF,OBJGRD,NSTATE,Opt,Param)
            Do 5691 i=1,N
              Write(50,9905) Ext(i),OBJGRD(i)
```

```
 9905        Format(11X,'dZ/d',A3,' = ',E17.10)
 5691     continue
        Endif

        If (AuxRstar.EQ.1) then
          If (It.NE.2) X(1)=(X(1)+XOld(1,It-1))/2.
          If ((X(1).LT.XMin(j,1)).OR.(X(1).GT.XMax(j,1))) then
            If (X(1).LT.XMin(j,1)) then
              Write(*,1076) Ext(1),X(1),XMin(j,1)
              Write(50,1076) Ext(1),X(1),XMin(j,1)
 1076          Format(/ 11X,A3,' = ',E17.10,' < ',E17.10, ' ! ' /)
              X(1)=XMin(j,1)+(XOld(1,It-1)-XMin(j,1))/2.
            else
              Write(*,8401) Ext(1),X(1),XMax(j,1)
              Write(50,8401) Ext(1),X(1),XMax(j,1)
 8401          Format(/ 3X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
              X(1)=XOld(1,It-1)+(XMax(j,1)-XOld(1,It-1))/2.
            Endif
            Write(50,8100) Ext(1),X(1)
 8100        Format(/ 11X,A3,' = ',E17.10 /)
            goto 5399
          Endif
          Write(50,8244) Ext(1),X(1)
 8244      Format(/ 11X,A3,' = ',E17.10 /)
          goto 5399
        Endif

        Write(50,9312) OBJF
 9312  Format(/ 11X,'Z=',E17.10 /)
        OBJFOld=OBJF
C        Write(50,8822) OBJFOld
C 8822   Format(11X,'ZOld=Z=',E17.10)
        Count=0
        Do 1409 i=1,N
          OBJGRDPrev(i)=OBJGRD(i)
          If (ABS(OBJGRDPrev(i)).LT.(1E-25)) Count=Count+1
 1409   continue
        If (Count.EQ.N) then
          Write(*,99)
          Write(50,99)
 99        Format(// 11X,'ERROR: The derivatives are zero !')
          STOP
        Endif
        Do 4012 i=1,N
          OBJGRD(i)=0.
          Do 4501 k=1,N
            OBJGRD(i)=OBJGRDPrev(k)*V(k,i)+OBJGRD(i)
 4501      continue
          Write(50,8108) i,OBJGRD(i)
 8108      Format(11X,'dZ/dY(',I2,') = ',E17.10)
 4012   continue
        Write(50,*) ' '

C        ==================================================================
C        Calculation of the mean value, MuOBJF, & the standard deviation,
C        SigmaOBJF, of the failure function.
C        ==================================================================
        Mu1OBJF=0
        VarOBJF=0
        Do 844 i=1,N
          Mu1OBJF=(Muy(i)-Y(i))*OBJGRD(i)+Mu1OBJF
          VarOBJF=((OBJGRD(i)*Sigmay(i))**2)+VarOBJF
 844    continue
        MuOBJF=OBJFOld+Mu1OBJF
        SigmaOBJF=SQRT(VarOBJF)

C        ==================================================================
C        Calculation of the sensitivity factors, Alpha, & the new design
C        point, Y (& consequently X).
C        ==================================================================
        Do 9019 i=1,N
          Alpha(i)=(Sigmay(i)/SigmaOBJF)*OBJGRD(i)
          YOld(i,It-1)=Y(i)
          Y(i)=Muy(i)-Alpha(i)*RelInd*Sigmay(i)
 9019   continue

        Do 2635 i=1,N
          XOld(i,It-1)=X(i)
          X(i)=0.
          Do 1657 k=1,N
            X(i)=V(i,k)*Y(k)+X(i)
 1657      continue
          Write(50,1987) Ext(i),XOld(i,It-1),Ext(i),X(i)
```

```
 1987      Format(11X,'XOld(',A3,') = ',E17.10,3X,A3,' = ',E17.10)
 2635   continue
        Write(50,*) ' '
        Do 889 i=1,N
          Write(50,879) i,Alpha(i)
  879     Format(11X,'Alpha(Y',I2,') = ',E17.10)
  889   continue

C       ==================================================================
C       If the new calculated design point lies outside [XMin,XMax], the
C       program continues calculations using a new design point between
C       the last computed design point & the boundary which was
C       exceeded.
C       ==================================================================
        Do 9895 i=1,N
          If ((X(i).LT.XMin(j,i)).OR.(X(i).GT.XMax(j,i))) then
            If (X(i).LT.XMin(j,i)) then
              Write(*,8977) Ext(i),X(i),XMin(j,i)
              Write(50,8977) Ext(i),X(i),XMin(j,i)
 8977          Format(/ 11X,A3,' = ',E17.10,' < ',E17.10, ' ! ' /)
              X(i)=XMin(j,i)+(XOld(i,It-1)-XMin(j,i))/2.
              Write(50,8177) Ext(i),X(i)
 8177          Format(/ 11X,A3,' = ',E17.10 /)
            else
              Write(*,8477) Ext(i),X(i),XMax(j,i)
              Write(50,8477) Ext(i),X(i),XMax(j,i)
 8477          Format(/ 3X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
              X(i)=XOld(i,It-1)+(XMax(j,i)-XOld(i,It-1))/2.
              Write(50,8166) Ext(i),X(i)
 8166          Format(/ 11X,A3,' = ',E17.10 /)
            Endif
            Aux=1
          else
            Aux=0
          Endif
 9895   continue
        If (Aux.NE.0) goto 5399

C       ==================================================================
C       Check convergence at the design point.
C       ==================================================================
        Do 859 i=1,N
          If (((ABS(XOld(i,It-1)/X(i)-1).GT.0.0001).AND.(X(i).NE.0.))
     1       .OR.((X(i).EQ.0.).AND.(X(i)-XOld(i,It-1).GT.0.00001))) then
            If (It.GT.MaxIter) then
              Write(*,9049)
              Write(40,9049)
              Write(50,9049)
 9049          Format(/ 11X,'Not converging at the design point !'///
     1                6X,'CONVERGENCE NOT FOUND WITHIN THE MAXIMUM',
     1                1X,'NUMBER OF ITERATIONS !')
              goto 9999
            else
              goto 13
            Endif
          Endif
  859   continue
        goto 23

C       ==================================================================
C       Smoothing of the iteration process (0<=Smooth<=1).
C       ==================================================================
 13     Write(50,*) ' '
        Do 4488 i=1,N
          If (Smooth.EQ.0.) then
            If (It.GT.3) then
              If (((ABS(XOld(i,It-1)/X(i)-1).GT.0.0001).AND.
     1           (X(i).NE.0.)).OR.((X(i).EQ.0.).AND.
     1           (X(i)-XOld(i,It-1).GT.0.00001)).OR.(Out(i).EQ.1)) then
                If (((ABS(XOld(i,It-2)/X(i)-1).LE.0.0001).AND.
     1             (X(i).NE.0.)).OR.((X(i).EQ.0.).AND.
     1             (X(i)-XOld(i,It-2).LE.0.00001)).OR.
     1             ((ABS(XOld(i,It-3)/X(i)-1).LE.0.0001
     1             .AND.(X(i).NE.0.)).OR.((X(i).EQ.0.).AND.
     1             (X(i)-XOld(i,It-3).LE.0.00001)).OR.(Out(i).EQ.1)) then
                  Write(*,*) '*****'
                  Write(50,*) '*****'
                  k1(i)=k1(i)+1
                  If (X(i).GT.XOld(i,It-1)) then
                    Sin(k1(i),i)=4
                    If (((Sin(k1(i)-1,i)-Sin(k1(i),i)).NE.0).AND.
     1                 (k1(i).GT.3)) Step(i)=Step(i)/10.
                    X(i)=XOld(i,It-1)+Step(i)*ABS(XOld(i,It-1)-X(i))
```

```
                    Endif
                    If (X(i).LT.XOld(i,It-1)) then
                      Sin(k1(i),i)=1
                      If (((Sin(k1(i)-1,i)-Sin(k1(i),i)).NE.0).AND.
     1                   (k1(i).GT.3)) Step(i)=Step(i)/10.
                      X(i)=X(i)+Step(i)*ABS(XOld(i,It-1)-X(i))
                    Endif
                    Write(50,2237) Ext(i),X(i)
 2237                 Format(11X,A3,' = ',E17.10)
                    Out(i)=1
                  Endif
                Endif
              Endif
            else
              X(i)=(1.-Smooth)*X(i)+Smooth*XOld(i,It-1)
              Write(50,2287) Ext(i),X(i)
 2287          Format(11X,A3,' = ',E17.10)
            Endif
 4488     continue
          If (Smooth.EQ.0.) then
            Do 8509 i=1,N
              If (((ABS(XOld(i,It-1)/X(i)-1).GT.0.0001).AND.(X(i).NE.0.))
     1         .OR.((X(i).EQ.0.).AND.(X(i)-XOld(i,It-1).GT.0.00001))) then
                Write(*,949) It
                Write(50,949) It
 949             Format(/ 11X,'Not converging at the design point !' ///
     1                   6X,'ITERATION No.',I3)
                goto 5499
              Endif
 8509       continue
          else
            Write(*,9491) It
            Write(50,9491) It
 9491        Format(/ 11X,'Not converging at the design point !' ///
     1               6X,'ITERATION No.',I3)
            goto 5499
          Endif

C       ================================================================
C       Check value of the failure function at the design point. If
C       ABS(OBJFOld)>OBJFAcc, calculate a better estimate of the design
C       parameter, Param.
C       ================================================================
 23     ItInt=ItInt+(It-1)
        OBJFAcc=ReqOBJFAcc*SigmaOBJF/100.
        If (ABS(OBJFOld).GT.OBJFAcc) then
          If (It.GT.MaxIter) then
            Write(*,9419)
            Write(40,9419)
            Write(50,9419)
 9419        Format(/ 11X,'The accuracy of the failure function'
     1               / 11X,'is less than the required value !'
     1               /// 6X,'CONVERGENCE NOT FOUND WITHIN THE MAXIMUM',
     1               1X,'NUMBER OF ITERATIONS !')
            goto 9999
          Endif
          It=1
          Do 3295 i=1,N
            Step(i)=0.001
            Out(i)=0
            k1(i)=1
 3295     continue
          Write(*,2394) It
          Write(50,2394) It
 2394      Format(/ 11X,'The accuracy of the failure function'
     1             / 11X,'is less than the required value !'
     1             /// 6X,'ITERATION No.',I3)

C         ----------
C         Calculation of dparam/dOBJF.
C         ----------
          N=N+1
          X(N)=Param
          If (FDer.EQ.2) then
            Call Derivadas(Param,N,X,Ext,Opt,FDer,OBJF,OBJGRD)
            Write(50,9995) N,OBJGRD(N)
 9995        Format(// 11X,'dZ/dX(',I2,') = ',E17.10)
          else
            Call OBJFUN(FDer,N,X,OBJF,OBJGRD,NSTATE,Opt,Param)
            Do 4690 i=1,N
              Write(50,9095) Ext(i),OBJGRD(i)
 9095          Format(11X,'dZ/d',A3,' = ',E17.10)
 4690       continue
```

```
               Endif

               If (AuxRstar.EQ.1) then
                 If (It.NE.2) X(1)=(X(1)+XOld(1,It-1))/2.
                 If ((X(1).LT.XMin(j,1)).OR.(X(1).GT.XMax(j,1))) then
                   If (X(1).LT.XMin(j,1)) then
                     Write(*,5975) Ext(1),X(1),XMin(j,1)
                     Write(50,5975) Ext(1),X(1),XMin(j,1)
 5975                 Format(/ 11X,A3,' = ',E17.10,' < ',E17.10, ' ! ' /)
                     X(1)=XMin(j,1)+(XOld(1,It-1)-XMin(j,1))/2.
                   else
                     Write(*,8551) Ext(1),X(1),XMax(j,1)
                     Write(50,8551) Ext(1),X(1),XMax(j,1)
 8551                 Format(/ 3X,A3,' = ',E17.10,' > ',E17.10, ' ! ' /)
                     X(1)=XOld(1,It-1)+(XMax(j,1)-XOld(1,It-1))/2.
                   Endif
                   Write(50,8116) Ext(1),X(1)
 8116              Format(/ 11X,A3,' = ',E17.10 /)
                   goto 5399
                 Endif
                 Write(50,8233) Ext(1),X(1)
 8233            Format(/ 11X,A3,' = ',E17.10 /)
                 goto 5399
               Endif

               dparamdOBJF=1./OBJGRD(N)
               Write(50,9772) OBJFOld
 9772          Format(/ 11X,'Z=',E17.10)
               Write(50,9395) dparamdOBJF
 9395          Format(11X,'dparamdOBJF = ',E17.10)

C           ----------
C            New estimate of the value of the design parameter, param.
C           ----------
               Param=Param-OBJFOld*dparamdOBJF
C              If (Param.LT.0) Param=0
               If (Param.LT.0) then
                 If (Opt.EQ.3) then
                   Write(*,4007)
                   Write(50,4007)
 4007              Format(// 11X,'ERROR: Nourishment Width < 0 !')
                   STOP
                 else
                   Write(*,4037)
                   Write(50,4037)
 4037              Format(// 11X,'ERROR: Crest Level < 0 !')
                   STOP
                 Endif
               Endif
               Write(50,5599) Param
 5599          Format(/ 11X,'New Value of the Design Parameter = ',E17.10)
               N=N-1
               goto 5499
             Endif

C        ================================================================
C        Print the final results.
C        ================================================================
         It=ItInt
         Write(40,8760)
         Write(50,8760)
 8760    Format(// 3X,'FINAL RESULTS' //)
         Write(40,100) It,OBJFOLD,MuOBJF,SigmaOBJF,RelInd,(100.*Pf),Param
         Write(50,100) It,OBJFOLD,MuOBJF,SigmaOBJF,RelInd,(100.*Pf),Param
 100     Format(11X,'Total Number of Iterations = ',I3 / 11X,
     1            'Failure Function Z (X) = ',E17.10 / 11X,
     1            'Mean Value of Z = ', E17.10 / 11X,
     1            'Standard Deviation of Z = ', E17.10 / 11X,
     1            'Reliability Index = ', E17.10 / 11X,
     1            'Target Probability of Failure (%) = ', F10.6 / 11X,
     1            'Design Parameter = ', E17.10)
         Write(40,922)
         Write(50,922)
 922     Format(// 11X, 'DESIGN POINT COORDINATES' /)
         Do 539 i=1,N
           Write(40,726) Ext(i),X(i)
           Write(50,726) Ext(i),X(i)
 726        Format(19X,A3,' = ',E17.10)
 539      continue
         Write(40,1677)
         Write(50,1677)
 1677    Format(/)
         iii=0
```

```
         Do 444 i=1,N
           Inf(i)=((Alpha(i))**2)*100
           Do 3359 ii=1,N
             If (i.NE.ii) then
               If (Rho(i,ii).NE.0.) iii=1
             Endif
 3359      continue
           If (iii.EQ.1) then
             Write(40,459) i,Alpha(i),Ext(i),Inf(i)
             Write(50,459) i,Alpha(i),Ext(i),Inf(i)
 459        Format(11X,'Alpha(Y',I2,') = ',E17.10 / 11X,
      1             'Influence of Y(',A3,') on the Reliability',1X,
      1             'Index = ',E14.7 /)
             iii=0
           else
             Write(40,449) Ext(i),Alpha(i),Ext(i),Inf(i)
             Write(50,449) Ext(i),Alpha(i),Ext(i),Inf(i)
 449        Format(11X,'Alpha(',A3,') = ',E17.10 / 11X,
      1             'Influence of ',A3,' on the Reliability',1X,
      1             'Index = ',E14.7 /)
           Endif
 444      continue
 9999     return
          End




C        ##################################################################
C
         Subroutine VarExt(N,Opt,Ext,ExtExt,ParamDesc)
C
C        ##################################################################
C
C        Defines the variables specific to each failure mode, ExtExt,
C        their abbreviation, Ext, & the design parameter, ParamDesc.
C
C        ##################################################################
C
C        INPUT VARIABLES:
C        Opt - Failure mode
C        N - Number of variables
C        DSWL - Definition of the SWL
C
C        MODELING VARIABLE:
C        L - Maximum number of variables allowed by the program
C        TL - Variable mentioned in the Common statements but not used
C             here
C
C        OUTPUT VARIABLES:
C        Ext - Abbreviation of the name of the variable
C        ExtExt - Description of the variable
C        ParamDesc - Description of the design parameter
C
C        ##################################################################
C
         Integer*4 L,N,Opt,DSWL
         Character*3 Ext(L)
         Character*17 ExtExt(L)
         Character*19 ParamDesc
         Parameter (L=15)
         Real*8 TL
         Common/BLOCK11/DSWL,TL

         If (Opt.EQ.1) then
           Ext(1)='Tp '
           Ext(2)='Hs '
           Ext(3)='A  '
           Ext(4)='B  '
           Ext(5)='SWL'
           Ext(6)='TAl'
           Ext(7)='r  '
           Ext(8)='eB '
           ExtExt(1)='Peak Wave Period '
           ExtExt(2)='Wave Height      '
           ExtExt(3)='H&R Parameter    '
           ExtExt(4)='H&R Parameter    '
           ExtExt(5)='Still-Water-Level'
           ExtExt(6)='Seawall Slope    '
           ExtExt(7)='Roughness        '
           ExtExt(8)='Model Parameter  '
           If (DSWL.EQ.2) then
             Ext(5)='Tid'
             Ext(6)='Sur'
```

```
                    Ext(7)='TAl'
                    Ext(8)='r  '
                    Ext(9)='eB '
                    ExtExt(5)='Tide Level      '
                    ExtExt(6)='Surge           '
                    ExtExt(7)='Seawall Slope   '
                    ExtExt(8)='Roughness       '
                    ExtExt(9)='Model Parameter '
                Endif
                ParamDesc='Seawall Crest Level'
              elseif (Opt.EQ.2) then
                Ext(1)='Tm '
                Ext(2)='Hs '
                Ext(3)='A  '
                Ext(4)='B  '
                Ext(5)='SWL'
                Ext(6)='r  '
                Ext(7)='eB '
                ExtExt(1)='Mean Wave Period '
                ExtExt(2)='Wave Height      '
                ExtExt(3)='Owen Parameter   '
                ExtExt(4)='Owen Parameter   '
                ExtExt(5)='Still-Water-Level'
                ExtExt(6)='Roughness        '
                ExtExt(7)='Model Parameter  '
                If (DSWL.EQ.2) then
                    Ext(5)='Tid'
                    Ext(6)='Sur'
                    Ext(7)='r  '
                    Ext(8)='eB '
                    ExtExt(5)='Tide Level      '
                    ExtExt(6)='Surge           '
                    ExtExt(7)='Roughness       '
                    ExtExt(8)='Model Parameter '
                Endif
                ParamDesc='Seawall Crest Level'
              elseif (Opt.EQ.3) then
                Ext(1)='Hs '
                Ext(2)='D50'
                Ext(3)='DP '
                Ext(4)='SD '
                Ext(5)='GB '
                Ext(6)='Ac '
                Ext(7)='h  '
                ExtExt(1)='Wave Height     '
                ExtExt(2)='Particle Size   '
                ExtExt(3)='Initial Profile '
                ExtExt(4)='Surge Duration  '
                ExtExt(5)='Gust Bumps      '
                ExtExt(6)='Accuracy Comput.'
                ExtExt(7)='Surge Level     '
                If (DSWL.EQ.2) then
c               Ext(1)='Tid'
c               Ext(2)='Sur'
c               Ext(3)='Hs '
c               Ext(4)='D50'
c               Ext(5)='DP '
c               Ext(6)='SD '
c               Ext(7)='GB '
c               Ext(8)='Ac '
c               ExtExt(1)='Tide Level      '
c               ExtExt(2)='Surge           '
c               ExtExt(3)='Wave Height     '
c               ExtExt(4)='Particle Size   '
c               ExtExt(5)='Initial Profile '
c               ExtExt(6)='Surge Duration  '
c               ExtExt(7)='Gust Bumps      '
c               ExtExt(8)='Accuracy Comput.'

                    Ext(7)='Tid'
                    Ext(8)='Sur'
                    ExtExt(7)='Tide Level      '
                    ExtExt(8)='Surge           '
                Endif
                ParamDesc='Nourishment Width'
              else
                ParamDesc='Nothing'
                If (N.GE.1) then
                    Ext(1)='X1 '
                    ExtExt(1)='X(1)'
                    If (N.GE.2) then
                        Ext(2)='X2 '
                        ExtExt(2)='X(2)'
```

```
        If (N.GE.3) then
          Ext(3)='X3 '
          ExtExt(3)='X(3)'
          If (N.GE.4) then
            Ext(4)='X4 '
            ExtExt(4)='X(4)'
            If (N.GE.5) then
              Ext(5)='X5 '
              ExtExt(5)='X(5)'
              If (N.GE.6) then
                Ext(6)='X6 '
                ExtExt(6)='X(6)'
                If (N.GE.7) then
                  Ext(7)='X7 '
                  ExtExt(7)='X(7)'
                  If (N.GE.8) then
                    Ext(8)='X8 '
                    ExtExt(8)='X(8)'
                    If (N.GE.9) then
                      Ext(9)='X9 '
                      ExtExt(9)='X(9)'
                      If (N.GE.10) then
                        Ext(10)='X10'
                        ExtExt(10)='X(10)'
                        If (N.GE.11) then
                          Ext(11)='X11'
                          ExtExt(11)='X(11)'
                          If (N.GE.12) then
                            Ext(12)='X12'
                            ExtExt(12)='X(12)'
                            If (N.GE.13) then
                              Ext(13)='X13'
                              ExtExt(13)='X(13)'
                              If (N.GE.14) then
                                Ext(14)='X14'
                                ExtExt(14)='X(14)'
                                If (N.GE.15) then
                                  Ext(15)='X15'
                                  ExtExt(15)='X(15)'
                                Endif
                              Endif
                            Endif
                          Endif
                        Endif
                      Endif
                    Endif
                  Endif
                Endif
              Endif
            Endif
          Endif
        Endif
      Endif
    return
    End




C     ################################################################
C
      Subroutine Combination(N,NumTVAc,CombAc,NCombAc,r,TVAc,NR)
C
C     ################################################################
C
C     Returns, for each combination of actions, the power, NR, to
C     which each distribution is raised.
C
C     ################################################################
C
C     INPUT VARIABLES:
C     N - Number of variables
C     NCombAc, CombAc - Number of combinations of actions
C     NumTVAc - Number of time-varying actions
C     r - Repetitions of each action in the design life
C     TVAc - Number of the time-varying actions in increasing order of
C             the number of repetitions
C
C     MODELING VARIABLES:
C     Q - Maximum number of combinations of actions allowed by the
C         program
C     L - Maximum number of variables allowed by the program
```

```
C        i, j, Aux - Auxiliary variables
C
C         OUTPUT VARIABLE:
C        NR - Power to which each distribution is raised for each
C             combination of actions
C
C         ##############################################################
C
         Integer*4 j,i,L,Q,Aux,NumTVAc,NCombAc,CombAc,N
         Parameter (L=15)
         Parameter (Q=16)
         Integer*4 TVAc(L)
         Real*8 NR(Q,L),r(L)
         Aux=-1
         If (CombAc.EQ.2) then
           Do 144 i=1,N
             Do 199 j=1,NCombAc
               NR(j,i)=1
 199         continue
 144       continue
           Do 14 i=1,NumTVAc
             Do 13 j=1,NCombAc
               If (i.EQ.1) then
                 Aux=Aux*(-1)
                 If (Aux.LT.0) then
                   NR(j,TVac(1))=1
                 else
                   NR(j,TVac(1))=r(TVAc(1))
                 Endif
               elseif (i.EQ.2) then
                 If (j.EQ.1.OR.j.EQ.5.OR.j.EQ.9.OR.j.EQ.13) then
                   NR(j,TVac(2))=r(TVAc(2))/r(TVAc(1))
                 elseif (j.EQ.2.OR.j.EQ.6.OR.j.EQ.10.OR.j.EQ.14) then
                   NR(j,TVac(2))=r(TVAc(2))
                 else
                   NR(j,TVac(2))=1
                 Endif
               elseif (i.EQ.3) then
                 If (j.EQ.1.OR.j.EQ.2.OR.j.EQ.9.OR.j.EQ.10) then
                   NR(j,TVac(3))=r(TVAc(3))/r(TVAc(2))
                 elseif (j.EQ.3.OR.j.EQ.11) then
                   NR(j,TVac(3))=r(TVAc(3))/r(TVAc(1))
                 elseif (j.EQ.4.OR.j.EQ.12) then
                   NR(j,TVac(3))=r(TVAc(3))
                 else
                   NR(j,TVac(3))=1
                 Endif
               elseif (i.EQ.4) then
                 If (j.EQ.1.OR.j.EQ.2.OR.j.EQ.3.OR.j.EQ.4) then
                   NR(j,TVac(4))=r(TVAc(4))/r(TVAc(3))
                 elseif (j.EQ.5.OR.j.EQ.6) then
                   NR(j,TVac(4))=r(TVAc(4))/r(TVAc(2))
                 elseif (j.EQ.7) then
                   NR(j,TVac(4))=r(TVAc(4))/r(TVAc(1))
                 elseif (j.EQ.8) then
                   NR(j,TVac(4))=r(TVAc(4))
                 else
                   NR(j,TVac(4))=1
                 Endif
               else
                 If (j.EQ.9.OR.j.EQ.10.OR.j.EQ.11.OR.j.EQ.12) then
                   NR(j,TVac(5))=r(TVAc(5))/r(TVAc(3))
                 elseif (j.EQ.13.OR.j.EQ.14) then
                   NR(j,TVac(5))=r(TVAc(5))/r(TVAc(2))
                 elseif (j.EQ.15) then
                   NR(j,TVac(5))=r(TVAc(5))/r(TVAc(1))
                 elseif (j.EQ.16) then
                   NR(j,TVac(5))=r(TVAc(5))
                 else
                   NR(j,TVac(5))=r(TVAc(5))/r(TVAc(4))
                 Endif
               Endif
 13          continue
 14        continue
         else
           Do 15 i=1,NCombAc
             Do 17 j=1,N
               NR(i,j)=1
 17          continue
             Do 16 j=1,NumTVAc
               If (i.EQ.j) then
                 NR(i,TVAc(j))=r(TVAc(j))
               elseif (i.GT.j) then
```

```
              NR(i,TVAc(j))=1
            else
              NR(i,TVAc(j))=(r(TVAc(j)))/(r(TVAc(j-1)))
            Endif
 16         continue
 15       continue
        Endif
        return
        End




C       ################################################################
C
        Subroutine InvNormal(C,Rel)
C
C       ################################################################
C
C       Returns the reliability level, Rel, for a specific value, C, of
C       the exceedance cumulative distribution function of the standard
C       Normal distribution (tabulated in statistical books, e.g.
C       Abramowitz & Stegun, 1964). It uses the data file distnorm.dad
C       which contains the tabulated values.
C
C       ################################################################
C
C       INPUT VARIABLE:
C       C - Value of the exceedance cumulative distribution function of
C           the standard Normal distribution
C
C       MODELING VARIABLES:
C       M - Number of points of the standard Normal distribution
C           tabulated in file distnorm.dad
C       Phi - Value tabulated in file distnorm.dad of the exceedance
C             cumulative distribution function of the standard Normal
C             distribution
C       Beta - Standard Normal variable, tabulated in file distnorm.dad,
C              which corresponds to Phi
C       k - Auxiliary variable
C
C       OUTPUT VARIABLE:
C       Rel - Standard Normal variable which corresponds to C
C
C       ################################################################
C
        Integer*4 k,M
        Parameter (M=453)
        Real*8 C,Rel,Beta(M),Phi(M)
        Open(Unit=20, File='distnorm.dad', Status='Old')
        Do 910 k=1,M
          Read(20,*) Beta(k), Phi(k)
 910    continue
        Do 1132 k=1,M
          If (C.LE.0.5) then
            If (Phi(k).LE.C) then
              If (Phi(k).LT.C) then
              Rel=((C-Phi(k))*((Beta(k+1))-Beta(k)))/
     1            ((Phi(k+1))-Phi(k))+Beta(k)
              goto 1150
              else
               Rel=Beta(k)
               goto 1150
              Endif
            Endif
          else
            If (Phi(k).LE.(1.-C)) then
              If (Phi(k).LT.(1.-C)) then
              Rel=-(((((1.-C)-Phi(k))*((Beta(k+1))-Beta(k)))/
     1            ((Phi(k+1))-Phi(k))+Beta(k))
              goto 1150
              else
               Rel=-Beta(k)
               goto 1150
              Endif
            Endif
          Endif
 1132   continue
 1150   Close (Unit=20)
        return
        End
```

```
C        ###############################################################
C
         Subroutine WhatDist(i,j,VarDis,Abrev,Carac)
C
C        ###############################################################
C
C        Returns the name, Carac, & the abbreviation, Abrev, of the
C        chosen statistical distribution, VarDis.
C
C        ###############################################################
C
C        INPUT VARIABLES:
C        i - Number of the variable
C        j - Number of the combination of actions
C        VarDis - Type of distribution
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C
C        OUTPUT VARIABLES:
C        Carac - Name of the distribution
C        Abrev - Abbreviation of the name of the distribution
C
C        ###############################################################
C
         Integer*4 L,Q,i,j
         Parameter (L=15)
         Parameter (Q=16)
         Character*30 Carac(Q,L)
         Character*3 Abrev(Q,L)
         Integer*4 VarDis(Q,L)
         If (VarDis(j,i).EQ.0) then
           Carac(j,i)='Deterministic'
           Abrev(j,i)='Det'
         elseif (VarDis(j,i).EQ.1) then
           Carac(j,i)='Normal (Gaussian)'
           Abrev(j,i)='Nor'
         elseif (VarDis(j,i).EQ.2) then
           Carac(j,i)='Log-Normal'
           Abrev(j,i)='LgN'
         elseif (VarDis(j,i).EQ.3) then
           Carac(j,i)='Maxima Type I (Gumbel)'
           Abrev(j,i)='Gum'
         elseif (VarDis(j,i).EQ.4) then
           Carac(j,i)='Rectangular (Uniform)'
           Abrev(j,i)='Uni'
         elseif (VarDis(j,i).EQ.5) then
           Carac(j,i)='Gamma'
           Abrev(j,i)='Gam'
         elseif (VarDis(j,i).EQ.6) then
           Carac(j,i)='Beta'
           Abrev(j,i)='Bet'
         elseif (VarDis(j,i).EQ.7) then
           Carac(j,i)='Maxima Type II (Frechet)'
           Abrev(j,i)='Fre'
         elseif (VarDis(j,i).EQ.8) then
           Carac(j,i)='Exponential'
           Abrev(j,i)='Exp'
         elseif (VarDis(j,i).EQ.9) then
           Carac(j,i)='Rayleigh'
           Abrev(j,i)='Ray'
         elseif (VarDis(j,i).EQ.10) then
           Carac(j,i)='Minima Type III (Weibull)'
           Abrev(j,i)='Wei'
         elseif (VarDis(j,i).EQ.11) then
           Carac(j,i)='User-Defined Distribution'
           Abrev(j,i)='UD1'
         elseif (VarDis(j,i).EQ.12) then
           Carac(j,i)='User-Defined Distribution'
           Abrev(j,i)='UD2'
         else
           Carac(j,i)='User-Defined Distribution'
           Abrev(j,i)='UD3'
         Endif
         return
         End


C        ###############################################################
C
```

```
          Subroutine WriCharVar(i,j,VarDis,Trunc,Ext,Carac,Xo,Mux,Sigmax,
     1                    Opt)
C
C        ################################################################
C
C        Writes the following characteristics of the variables in the
C        files summary.dat & results.dat :
C        - Type of distribution
C        - Type of truncation & point of truncation, Xo (if the
C          distribution is truncated)
C        - Mean, Mux, standard deviation, Sigmax, & lower limit, Zeta
C
C        ################################################################
C
C        INPUT VARIABLES:
C        Opt - Failure mode
C        TL - Seawall toe level
C        DSWL -  Definition of the SWL
C        i - Number of the variable
C        j - Number of the combination of actions
C        Ext - Abbreviation of the name of the variable
C        Mux - Mean of X
C        Sigmax - Standard deviation of X
C        VarDis - Type of distribution
C        Carac - Name of the distribution
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        Zeta - Parameter of a distribution
C        x1 - Lower limit on X for a Beta distribution
C        x2 - Upper limit on X for a Beta distribution
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C             program
C        Lamda, Eta - Variables mentioned in the Common statement but not
C                     used here
C
C        ################################################################
C
          Integer*4 i,j,L,Q,Opt,DSWL
          Parameter (L=15)
          Parameter (Q=16)
          Character*3 Ext(L)
          Character*30 Carac(Q,L)
          Integer*4 VarDis(Q,L),Trunc(Q,L)
          Real*8 Xo(Q,L),Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L),
     1         Eta(Q,L),x1(Q,L),x2(Q,L),TL
          Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
          Common/BLOCK11/DSWL,TL

          Write(40,8020) Ext(i),Carac(j,i)
          Write(50,8020) Ext(i),Carac(j,i)
 8020   Format(11X,'Probability Distribution of ',A3,' = ',A30)

          If (Trunc(j,i).NE.0) then
            If (Trunc(j,i).EQ.1) then
              If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(i.EQ.2)) then
                If (DSWL.EQ.1) then
                  Write(40,802) Ext(i),TL
                  Write(50,802) Ext(i),TL
 802                Format(11X,'The Distribution of ',A3,
     1                    ' is truncated above Xo = 0.6(SWL-TL)' / 11X,
     1                    'Seawall Toe Level (TL) = ',E17.10)
                else
                  Write(40,8029) Ext(i),TL
                  Write(50,8029) Ext(i),TL
 8029               Format(11X,'The Distribution of ',A3,
     1                    ' is truncated above Xo = 0.6(Tide+Surge-TL)' /
     1                    11X,'Seawall Toe Level (TL) = ',E17.10)
                Endif
              else
                Write(40,82) Ext(i),Xo(j,i)
                Write(50,82) Ext(i),Xo(j,i)
 82                 Format(11X,'The Distribution of ',A3,
     1                    ' is truncated above Xo = ',E17.10)
              Endif
            else
              Write(40,829) Ext(i),Xo(j,i)
              Write(50,829) Ext(i),Xo(j,i)
 829              Format(11X,'The Distribution of ',A3,
     1                  ' is truncated below Xo = ',E17.10)
            Endif
```

```
          Endif

          Write(40,8299) Ext(i),Mux(j,i),Ext(i),Sigmax(j,i)
          Write(50,8299) Ext(i),Mux(j,i),Ext(i),Sigmax(j,i)
 8299   Format(11X,'Mean Value of ',A3,' = ',E17.10 / 11X,
     1          'Standard Deviation of ',A3,' = ',E17.10)

          If (VarDis(j,i).EQ.6) then
            Write(40,2198) Ext(i),x1(j,i),x2(j,i)
            Write(50,2198) Ext(i),x1(j,i),x2(j,i)
 2198      Format(11X,'Limits a and b of ',A3,' = [',
     1            E17.10,',',E17.10,']')
          Endif

          If (VarDis(j,i).EQ.10) then
            Write(40,4198) Ext(i),Zeta(j,i)
            Write(50,4198) Ext(i),Zeta(j,i)
 4198      Format(11X,'Lower Limit on ',A3,' = ',E17.10)
          Endif

          Write(40,4598)
          Write(50,4598)
 4598   Format(/)

          return
          End



C        ################################################################
C
          Subroutine Parameters(i,j,Mux,Sigmax,VarDis,Ext)
C
C        ################################################################
C
C        Calls the subroutines required to calculate the distribution's
C        parameters, Zeta, Lamda & Eta, for each variable.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        i - Number of the variable
C        j - Number of the combination of actions
C        Ext - Abbreviation of the name of the variable
C        Mux - Mean of X
C        Sigmax - Standard deviation of X
C        VarDis - Type of distribution
C        x1 - Lower limit on X for a Beta distribution
C        x2 - Upper limit on X for a Beta distribution
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C             program
C
C        OUTPUT VARIABLES:
C        Zeta, Lamda, Eta - Parameters of a distribution
C
C        ################################################################
C
          Integer*4 i,j,L,Q
          Parameter (L=15)
          Parameter (Q=16)
          Character*3 Ext(L)
          Integer*4 VarDis(Q,L)
          Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L),Eta(Q,L),
     1        x1(Q,L),x2(Q,L)
          Common/BLOCK8/Zeta,Lamda,Eta,x1,x2

          If (VarDis(j,i).EQ.0.OR.VarDis(j,i).GT.10) then
            Zeta(j,i)=Mux(j,i)
            Lamda(j,i)=Sigmax(j,i)
            Write(50,2245) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 2245      Format(/ 16X,'Mean (',A3,') = ',E17.10 /
     1            16X,'Standard Deviation (',A3,') = ',E17.10)
          elseif (VarDis(j,i).EQ.1) then
            Call PNormal(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
          elseif (VarDis(j,i).EQ.2) then
            Call PLogNormal(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
          elseif (VarDis(j,i).EQ.3) then
            Call PGumbel(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
          elseif (VarDis(j,i).EQ.4) then
            Call PRectangular(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
```

```
          elseif (VarDis(j,i).EQ.5) then
            Call PGamma(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
          elseif (VarDis(j,i).EQ.6) then
            Call PBeta(j,i,Ext,Mux,Sigmax,Zeta,Lamda,x1,x2)
          elseif (VarDis(j,i).EQ.7) then
            Call PFrechet(j,i,Ext,Mux,Sigmax)
          elseif (VarDis(j,i).EQ.8) then
            Call PExponential(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
          elseif (VarDis(j,i).EQ.9) then
            Call PRayleigh(j,i,Ext,Mux,Sigmax,Zeta)
          else
            Call PWeibull(j,i,Ext,Mux,Sigmax)
          Endif
          return
          End




C       ################################################################
C
          Subroutine EqCharac(N,NCombAc,Abrev,Carac,VarDis,Trunc,Xo,Mux,
     1                        Sigmax)
C
C       ################################################################
C
C       If the distributions provided for the variables are the basic
C       distributions, then the characteristics of the variables are the
C       same for all the combinations considered. What differs in each
C       combination is the power to which each distribution is raised.
C       This subroutine gives to the variables in all combinations of
C       actions the same characteristics which the variables have in
C       combination 1.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       N - Number of variables
C       NCombAc - Number of combinations of actions
C
C       INPUT/OUTPUT VARIABLES:
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       VarDis - Type of distribution
C       Carac - Name of the distribution
C       Abrev - Abbreviation of the name of the distribution
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       Zeta, Lamda, Eta - Parameters of a distribution
C       x1 - Lower limit on X for a Beta distribution
C       x2 - Upper limit on X for a Beta distribution
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       i, k - Auxiliary variables
C
C       ################################################################
C
          Integer*4 i,k,L,N,Q,NCombAc
          Parameter (L=15)
          Parameter (Q=16)
          Character*3 Abrev(Q,L)
          Character*30 Carac(Q,L)
          Integer*4 VarDis(Q,L),Trunc(Q,L)
          Real*8 Xo(Q,L),Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L),
     1          Eta(Q,L),x1(Q,L),x2(Q,L)
          Common/BLOCK8/Zeta,Lamda,Eta,x1,x2

          Do 8000 k=2,NCombAc
            Do 7000 i=1,N
              Abrev(k,i)=Abrev(1,i)
              Carac(k,i)=Carac(1,i)
              VarDis(k,i)=VarDis(1,i)
              Trunc(k,i)=Trunc(1,i)
              Xo(k,i)=Xo(1,i)
              Mux(k,i)=Mux(1,i)
              Sigmax(k,i)=Sigmax(1,i)
              Zeta(k,i)=Zeta(1,i)
              If (VarDis(1,i).NE.9) Lamda(k,i)=Lamda(1,i)
              If (VarDis(1,i).EQ.10) Eta(k,i)=Eta(1,i)
              If (VarDis(1,i).EQ.6) then
                x1(k,i)=x1(1,i)
```

```
              x2(k,i)=x2(1,i)
            Endif
 7000     continue
 8000   continue
        return
        End




C       ##############################################################
C
        Subroutine Inverse(j,i,Mux,NR,VarDis,StartPt)
C
C       ##############################################################
C
C       Returns the argument, StartPt, corresponding to a value of 0.5
C       of a chosen cumulative distribution function, i.e.
C       CDF(StartPt)=0.5 .
C
C       ##############################################################
C
C       INPUT VARIABLES:
C       i - Number of the variable
C       j - Number of the combination of actions
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Mux - Mean of X
C       VarDis - Type of distribution
C       Zeta, Lamda, Eta - Parameters of a distribution
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C            program
C       CDF - Value of the cumulative distribution function
C       C - Value of the exceedance cumulative distribution function
C       Rel - Standard Normal variable which corresponds to C
C       x1, x2 - Variables mentioned in the Common statement but not
C                used here
C
C       OUTPUT VARIABLE:
C       StartPt - Starting value of the variables
C
C       ##############################################################
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Integer*4 VarDis(Q,L)
        Real*8 Mux(Q,L),Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L),
     1        Rel,CDF,C,StartPt(Q,L),NR(Q,L)
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2

        CDF=0.5**(1./NR(j,i))
        If (VarDis(j,i).EQ.0) then
          StartPt(j,i)=Mux(j,i)
        elseif (VarDis(j,i).EQ.1) then
          C=1.-CDF
          Call InvNormal(C,Rel)
          StartPt(j,i)=Rel*Lamda(j,i)+Zeta(j,i)
        elseif (VarDis(j,i).EQ.2) then
          C=1.-CDF
          Call InvNormal(C,Rel)
          StartPt(j,i)=EXP(Rel*Lamda(j,i)+Zeta(j,i))
        elseif (VarDis(j,i).EQ.3) then
          StartPt(j,i)=Lamda(j,i)-(LOG(-LOG(CDF)))/(Zeta(j,i))
        elseif (VarDis(j,i).EQ.4) then
          StartPt(j,i)=Zeta(j,i)+(Lamda(j,i)-Zeta(j,i))*CDF
        elseif (VarDis(j,i).EQ.5) then
          Call InvGamma(j,i,CDF,StartPt)
        elseif (VarDis(j,i).EQ.6) then
          Call InvBeta(j,i,CDF,StartPt)
        elseif (VarDis(j,i).EQ.7) then
          StartPt(j,i)=Lamda(j,i)*((-LOG(CDF))**(-1./Zeta(j,i)))
        elseif (VarDis(j,i).EQ.8) then
          StartPt(j,i)=Zeta(j,i)-Lamda(j,i)*LOG(1.-CDF)
        elseif (VarDis(j,i).EQ.9) then
          StartPt(j,i)=Zeta(j,i)*SQRT(-2.*LOG(1.-CDF))
        elseif (VarDis(j,i).EQ.10) then
          StartPt(j,i)=Zeta(j,i)+Lamda(j,i)*((-LOG(1.-CDF))**
     1                (1./Eta(j,i)))
        elseif (VarDis(j,i).EQ.11) then
          Call InvUser1(j,i,CDF,StartPt)
```

```
            elseif (VarDis(j,i).EQ.12) then
              Call InvUser2(j,i,CDF,StartPt)
            else
              Call InvUser3(j,i,CDF,StartPt)
            Endif
            return
            End




C        ################################################################
C
            Subroutine MinMax(i,j,Opt,VarDis,Ext,Trunc,Xo,Mux,Sigmax,XMin,
     1                        XMax)
C
C        ################################################################
C
C        Defines the minimum value, XMin, & the maximum value, XMax, for
C        each variable depending on the type of distribution & on whether
C        or not the distribution is truncated.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        Opt - Failure mode
C        i - Number of the variable
C        j - Number of the combination of actions
C        Ext - Abbreviation of the name of the variable
C        Mux - Mean of X
C        Sigmax - Standard deviation of X
C        VarDis - Type of distribution
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        Zeta, Lamda - Parameters of a distribution
C        x1 - Lower limit on X for a Beta distribution
C        x2 - Upper limit on X for a Beta distribution
C
C        INPUT/OUTPUT VARIABLES:
C        XMin - Minimum value of X
C        XMax - Maximum value of X
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        Eta - Variable mentioned in the Common statement but not used
C             here
C
C        ################################################################
C
            Integer*4 i,j,L,Q,Opt
            Parameter (L=15)
            Parameter (Q=16)
            Character*3 Ext(L)
            Integer*4 VarDis(Q,L),Trunc(Q,L)
            Real*8 Xo(Q,L),XMin(Q,L),XMax(Q,L),Mux(Q,L),Sigmax(Q,L),
     1          Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L)
            Common/BLOCK8/Zeta,Lamda,Eta,x1,x2

            If (Trunc(j,i).NE.0) then
              If ((Trunc(j,i).EQ.1).AND.(XMax(j,i).NE.Xo(j,i))) then
                If ((Opt.EQ.1.OR.Opt.EQ.2).AND.(i.EQ.2)) goto 11
                XMax(j,i)=Xo(j,i)
              Endif
 11           If ((Trunc(j,i).EQ.2).AND.(XMin(j,i).NE.Xo(j,i)))
     1          XMin(j,i)=Xo(j,i)
            Endif
            If ((VarDis(j,i).EQ.2).AND.(XMin(j,i).LE.0.)) XMin(j,i)=1E-25
            If (VarDis(j,i).EQ.4) then
              If (XMin(j,i).LT.Zeta(j,i)) XMin(j,i)=Zeta(j,i)
              If (XMax(j,i).GT.Lamda(j,i)) XMax(j,i)=Lamda(j,i)
            Endif
            If ((VarDis(j,i).EQ.5).AND.(XMin(j,i).LT.0.)) XMin(j,i)=0.
            If (VarDis(j,i).EQ.6) then
              If (XMin(j,i).LT.x1(j,i)) XMin(j,i)=x1(j,i)
              If (XMax(j,i).GT.x2(j,i)) XMax(j,i)=x2(j,i)
            Endif
            If ((VarDis(j,i).EQ.7).AND.(XMin(j,i).LE.0.)) XMin(j,i)=1E-25
            If ((VarDis(j,i).EQ.8).AND.(XMin(j,i).LT.Zeta(j,i)))
     1        XMin(j,i)=Zeta(j,i)
            If ((VarDis(j,i).EQ.9).AND.(XMin(j,i).LT.0.)) XMin(j,i)=0.
            If ((VarDis(j,i).EQ.10).AND.(XMin(j,i).LT.Zeta(j,i)))
     1          XMin(j,i)=Zeta(j,i)
```

```
        If (VarDis(j,i).EQ.11) then
          If (XMin(j,i).LT.(-5.53)) XMin(j,i)=-5.53
          If (XMax(j,i).GT.(5.77)) XMax(j,i)=5.77
        Endif
        If (VarDis(j,i).EQ.12) then
          If (XMin(j,i).LT.(5.45)) XMin(j,i)=5.45
          If (XMax(j,i).GT.(6.995)) XMax(j,i)=6.995
        Endif
        If (VarDis(j,i).EQ.13) then
          If (XMin(j,i).LT.(-5.33)) XMin(j,i)=-5.33
          If (XMax(j,i).GT.(5.57)) XMax(j,i)=5.57
        Endif
        Write(40,5971) Ext(i),XMin(j,i),Ext(i),XMax(j,i)
        Write(50,5971) Ext(i),XMin(j,i),Ext(i),XMax(j,i)
 5971   Format(11X,'XMin(',A3,') = ',E17.10,3X,'XMax(',A3,') = ',
     1          E17.10)
        return
        End




C       ################################################################
C
        Subroutine EqNorDis(i,j,N,X,VarDis,NR,Ext,MuxN,SigmaxN,Trunc,Xo)
C
C       ################################################################
C
C       Calls the subroutines required to calculate the mean, MuxN, &
C       the standard deviation, SigmaxN, of the equivalent Normal
C       distribution for each variable.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       j - Number of the combination of actions
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       N - Number of variables
C       i - Number of the variable
C       Ext - Abbreviation of the name of the variable
C       X - Variables of the failure mode
C       VarDis - Type of distribution
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       Zeta, Lamda, Eta - Parameters of a distribution
C       x1 - Lower limit on X for a Beta distribution
C       x2 - Upper limit on X for a Beta distribution
C
C       MODELING VARIABLES:
C       Q - Maximum number of combinations of actions allowed by the
C            program
C       L - Maximum number of variables allowed by the program
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
        Integer*4 i,j,L,N,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 VarDis(Q,L),Trunc(Q,L)
        Real*8 X(L),MuxN(L),SigmaxN(L),Xo(Q,L),x1(Q,L),x2(Q,L),
     1         Zeta(Q,L),Lamda(Q,L),Eta(Q,L),NR(Q,L)
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2

        If (VarDis(j,i).EQ.0) then
          MuxN(i)=Zeta(j,i)
          SigmaxN(i)=Lamda(j,i)
          Call NorWrite(i,Ext,MuxN,SigmaxN)
        elseif (VarDis(j,i).EQ.1) then
          Call NormalD(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,
     1            NR)
        elseif (VarDis(j,i).EQ.2) then
          Call LogNormal(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,
     1               NR)
        elseif (VarDis(j,i).EQ.3) then
          Call Gumbel(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,NR)
        elseif (VarDis(j,i).EQ.4) then
          Call Rectangular(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,
```

```
      1                   NR)
         elseif (VarDis(j,i).EQ.5) then
           Call Gamma(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,NR)
         elseif (VarDis(j,i).EQ.6) then
           Call BetaDis(i,j,Ext,X,Zeta,Lamda,x1,x2,MuxN,SigmaxN,Trunc,
      1                Xo,NR)
         elseif (VarDis(j,i).EQ.7) then
           Call Frechet(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,NR)
         elseif (VarDis(j,i).EQ.8) then
           Call Exponential(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,
      1                NR)
         elseif (VarDis(j,i).EQ.9) then
           Call Rayleigh(i,j,Ext,X,Zeta,MuxN,SigmaxN,Trunc,Xo,NR)
         elseif (VarDis(j,i).EQ.10) then
           Call Weibull(i,j,Ext,X,Zeta,Lamda,Eta,MuxN,SigmaxN,Trunc,
      1                Xo,NR)
         elseif (VarDis(j,i).EQ.11) then
           Call User1(i,j,Ext,X,MuxN,SigmaxN,NR)
         elseif (VarDis(j,i).EQ.12) then
           Call User2(i,j,Ext,X,MuxN,SigmaxN,NR)
         else
           Call User3(i,j,Ext,X,MuxN,SigmaxN,NR)
         Endif
         return
         End




C      ################################################################
C
         Subroutine Correlated(N,Ext,MuxN,SigmaxN,Rho,Sigmay,Muy,V,Vt)
C
C      ################################################################
C
C      Returns the means, Muy, & the standard deviations, Sigmay, of
C      the non-correlated Normal variables, Y. It also returns the
C      matrix V & its transpose, Vt, required to transform the
C      variables X into Y & vice versa.
C
C      ################################################################
C
C      INPUT VARIABLES:
C      N - Number of variables
C      Ext - Abbreviation of the name of the variable
C      Rho - Correlation coefficient
C      MuxN - Mean of the equivalent Normal distribution of X
C      SigmaxN - Standard deviation of the equivalent Normal
C               distribution of X
C
C      MODELING VARIABLES:
C      L - Maximum number of variables allowed by the program
C      Covx - Covariance matrix
C      D - Vector of eigenvalues
C      i, k - Auxiliary variables
C
C      OUTPUT VARIABLES:
C      Muy - Mean of Y (non-correlated, Normal transformed variables)
C      Sigmay - Standard deviation of Y
C      V - Matrix of eigenvectors
C      Vt - Transpose of V
C
C      ################################################################
C
         Integer*4 i,k,L,N
         Parameter (L=15)
         Character*3 Ext(L)
         Real*8 MuxN(L),SigmaxN(L),Rho(L,L),Covx(L,L),D(L),V(L,L),
      1        Sigmay(L),Vt(L,L),Muy(L)

         Do 701 i=1,N
           Do 702 k=1,N
             Covx(i,k)=Rho(i,k)*SigmaxN(i)*SigmaxN(k)
 702       continue
 701     continue
         Call Algebra(N,Covx,D,V)
         Write(50,400)
 400     Format(/ 11X,'EIGENVALUES' /)
         Do 93 i=1,N
           Write(50,35) Ext(i),D(i)
 35        Format(11X,'D(',A3,') = ',E17.10)
 93      continue
         Write(50,500)
 500     Format(/ 11X,'EIGENVECTORS' /)
```

```
          Write(50,*) ' '
          Do 138 k=1,N
            Do 139 i=1,N
              Write(50,392) Ext(i),Ext(k),V(i,k)
 392          Format(11X,'V(',A3,',',A3,') = ',E17.10)
 139        continue
 138      continue
          Write(50,*) ' '
          Do 492 i=1,N
            Do 551 k=1,N
              Vt(i,k)=V(k,i)
              Write(50,9891) Ext(i),Ext(k),Vt(i,k)
 9891         Format(11X,'Vt(',A3,',',A3,') = ',E17.10)
 551        continue
 492      continue
          Do 235 i=1,N
            Muy(i)=0.
            Do 237 k=1,N
              Muy(i)=Vt(i,k)*MuxN(k)+Muy(i)
 237        continue
 235      continue
          Write(50,*) ' '
          Do 333 i=1,N
            Sigmay(i)=SQRT(D(i))
            Write(50,992) Ext(i),Muy(i),Ext(i),Sigmay(i)
 992        Format(11X,'Muy(',A3,') = ',E17.10,3X,'Sigmay(',A3,') = ',
     1           E17.10)
 333      continue
          return
          End




C         ################################################################
C
          Subroutine Derivadas(Par,N,X,Ext,Opt,FDer,OBJF,OBJGRD)
C
C         ################################################################
C
C         Returns the value of the failure function, OBJF, & the values of
C         the first partial derivatives, OBJGRD. If the expressions for
C         the first partial derivatives are given (FDer=1), they are used
C         to calculate OBJGRD; subroutine OBJFUN is called for this
C         purpose. Oherwise (FDer=2), the derivatives have to be
C         calculated using the subroutine E04XAF. E04XAF is a NAG Fortran
C         subroutine.
C         ----------
C         For more details see NAG (1993).
C
C         ################################################################
C
C         INPUT VARIABLES:
C         Opt - Failure mode
C         Ext - Abbreviation of the name of the variable
C         N - Number of variables
C         X - Variables of the failure mode
C         FDer - Method of calculation of the first partial derivatives of
C                the failure function for overtopping
C         Par - For Mode=1 it is the prescribed value of the design
C                parameter (Param); for Mode=2 it is the starting value of
C                the design parameter (StartParam)
C
C         MODELING VARIABLES:
C         IUSER - Failure mode
C         L - Maximum number of variables allowed by the program
C         LHES, NSTATE, IFAIL, IWARN, MSGLVL, INFO, EPSRF, HCNTRL, HESIAN,
C         HFORW, WORK - Variables used by subroutine EO4XAF (for more
C                       details see NAG, 1993)
C         E04XAF, OBJFUN - External subroutines
C         USER - For Mode=1 it is the prescribed value of the design
C                parameter (Param); for Mode=2 it is the starting value of
C                the design parameter (StartParam)
C         i - Auxiliary variable
C
C         OUTPUT VARIABLES:
C         OBJF - Failure function
C         OBJGRD - First partial derivatives of OBJF
C
C         ################################################################
C
          Integer*4 N,L,i
          Parameter (L=15)
          Character*3 Ext(L)
```

```
        Integer*4 LHES,IFAIL,IWARN,FDer,MSGLVL,NSTATE,
     1           INFO(L),IUSER(1),Opt
        Real*8 EPSRF,OBJF,HCNTRL(L),HESIAN(L,L),HFORW(L),OBJGRD(L),
     1         USER(1),WORK(L*L+L),X(L),Par
        External E04XAF,OBJFUN

        LHES=N
        MSGLVL=1
        EPSRF=-1.
        Do 20 i=1,N
          HFORW(i)=-1.
 20     continue
        IFAIL=1
        IUSER(1)=Opt
        USER(1)=Par

        If (FDer.EQ.2) then
          Call E04XAF(MSGLVL,N,EPSRF,X,FDer,OBJFUN,LHES,HFORW,OBJF,
     1                OBJGRD,HCNTRL,HESIAN,IWARN,WORK,IUSER,
     1                USER,INFO,IFAIL)
          If ((IFAIL.EQ.0).OR.(IFAIL.EQ.2)) then
          else
            Write(*,9997) IFAIL
            Write(50,9997) IFAIL
 9997       Format(/ 11X,'On exit from E04XAF IFAIL = ',I2)
            PAUSE
          Endif
        else
          Call OBJFUN(FDer,N,X,OBJF,OBJGRD,NSTATE,IUSER,USER)
          Do 5691 i=1,N
            Write(50,9095) Ext(i),OBJGRD(i)
 9095       Format(11X,'dZ/d',A3,' = ',E17.10)
 5691     continue
        Endif
        return
        End




C       ###############################################################
C
        Subroutine OBJFUN(FDer,N,X,OBJF,OBJGRD,NSTATE,IUSER,USER)
C
C       ###############################################################
C
C       Calls the subroutines which contain the failure function. For
C       the failure mode of overtopping, they also contain the first
C       partial derivatives of the failure function.
C
C       ###############################################################
C
C       INPUT VARIABLES:
C       IUSER - Failure mode
C       OptD - Direction of the sand movements in dune erosion
C       N - Number of variables
C       X - Variables of the failure mode
C       FDer - Method of calculation of the first partial derivatives of
C              the failure function for overtopping
C       USER - For Mode=1 it is the value of the prescribed design
C              parameter (Param); for Mode=2 it is the starting value of
C              the design parameter (StartParam)
C
C       MODELING VARIABLES:
C       Opt - Failure mode
C       NSTATE - Variable used by subroutine EO4XAF (for more details
C                see NAG, 1993)
C
C       OUTPUT VARIABLES:
C       OBJF - Failure function
C       OBJGRD - First partial derivatives of OBJF
C
C       ###############################################################
C
        Integer*4 FDer,N,NSTATE,IUSER(1),Opt,OptD
        Real*8 OBJF,OBJGRD(N),USER(1),X(N)
        Common/BLOCK6/OptD

        Opt=IUSER(1)
        If (Opt.EQ.1) then
          Call HandR(FDer,N,X,OBJF,OBJGRD,USER)
        elseif (Opt.EQ.2) then
          Call Owen(FDer,N,X,OBJF,OBJGRD,USER)
        else
```

```
            If (OptD.EQ.2) then
              Call SDunes(N,X,OBJF,USER)
            else
              Call LDunes(N,X,OBJF,USER)
            Endif
          Endif
          return
          End




C        ################################################################
C
         Subroutine NormalDist(j,RelInd,Prob)
C
C        ################################################################
C
C        Returns the value of the exceedance cumulative distribution
C        function of the standard Normal distribution, Prob (tabulated in
C        statistical books, e.g. Abramowitz & Stegun, 1964), for a given
C        RelInd value. It uses the data file distnorm.dad which contains
C        the tabulated values.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        RelInd - Standard Normal variable which corresponds to Prob
C        j - Number of the combination of actions
C
C        MODELING VARIABLES:
C        M - Number of points of the standard Normal distribution
C            tabulated in file distnorm.dad
C        Phi - Value tabulated in file distnorm.dad of the exceedance
C              cumulative distribution function of the standard Normal
C              distribution
C        Beta - Standard Normal variable, tabulated in file distnorm.dad,
C               which corresponds to Phi
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        Pi - 3.14159...
C        i - Auxiliary variable
C
C        OUTPUT VARIABLES:
C        Prob - Value of the exceedance cumulative distribution function
C               of the standard Normal distribution
C
C        ################################################################
C
         Integer*4 j,i,M,Q
         Parameter (M=453)
         Parameter (Q=16)
         Real*8 RelInd,Prob(Q),Beta(M),Phi(M),Pi
         Open(Unit=20, File='distnorm.dad', Status='Old')
         Pi=4.*ATAN(1.)
         Do 10 i=1,M
           Read(20,*) Beta(i), Phi(i)
 10      continue
         Do 32 i=1,M
           If (Beta(i).GE.ABS(RelInd)) then
             If (Beta(i).GT.ABS(RelInd)) then
               If (RelInd.GE.0.) then
                 Prob(j)=
     1               ((Beta(i)-RelInd)/(Beta(i)-Beta(i-1)))*Phi(i-1)+
     1               ((RelInd-Beta(i-1))/(Beta(i)-Beta(i-1)))*Phi(i)
                 goto 50
               else
                 Prob(j)=
     1               1.-(((Beta(i)-ABS(RelInd))/(Beta(i)-Beta(i-1)))
     1               *Phi(i-1)+((ABS(RelInd)-Beta(i-1))/
     1               (Beta(i)-Beta(i-1)))*Phi(i))
                 goto 50
               Endif
             else
               If (RelInd.GE.0.) then
                 Prob(j)=Phi(i)
                 goto 50
               else
                 Prob(j)=1.-Phi(i)
                 goto 50
               Endif
             Endif
           Endif
 32      continue
```

```
 50       Close (Unit=20)
          return
          End




C         ################################################################
C
          Subroutine PNormal(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
C
C         ################################################################
C
C         Returns the parameters, Zeta & Lamda, of a Normal distribution.
C
C         ################################################################
C
C         INPUT VARIABLES:
C         Ext - Abbreviation of the name of the variable
C         Mux - Mean of X
C         Sigmax - Standard deviation of X
C         i - Number of the variable
C         j - Number of the combination of actions
C
C         MODELING VARIABLES:
C         L - Maximum number of variables allowed by the program
C         Q - Maximum number of combinations of actions allowed by the
C             program
C
C         OUTPUT VARIABLES:
C         Zeta, Lamda - Parameters of the Normal distribution
C
C         ################################################################
C
          Integer*4 j,i,L,Q
          Parameter (L=15)
          Parameter (Q=16)
          Character*3 Ext(L)
          Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L)
          Zeta(j,i)=Mux(j,i)
          Lamda(j,i)=Sigmax(j,i)
          If (Lamda(j,i).GT.0.) then
            Write(50,2245) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 2245     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1               16X,'Lamda(',A3,') = ',E17.10)
          else
            Write(*,5777) Ext(i)
            Write(50,5777) Ext(i)
 5777     Format(// 16X,'ERROR - Normal distribution'/ 19X,
     1             'Lamda(',A3,') <= 0 !' /)
            STOP
          Endif
          return
          End




C         ################################################################
C
          Subroutine PLogNormal(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
C
C         ################################################################
C
C         Returns the parameters, Zeta & Lamda, of a Log-Normal
C         distribution.
C
C         ################################################################
C
C         INPUT VARIABLES:
C         Ext - Abbreviation of the name of the variable
C         Mux - Mean of X
C         Sigmax - Standard deviation of X
C         i - Number of the variable
C         j - Number of the combination of actions
C
C         MODELING VARIABLES:
C         L - Maximum number of variables allowed by the program
C         Q - Maximum number of combinations of actions allowed by the
C             program
C
C         OUTPUT VARIABLES:
C         Zeta, Lamda - Parameters of the Log-Normal distribution
C
C         ################################################################
```

```
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L)
        Lamda(j,i)=SQRT(LOG(1.+((Sigmax(j,i)**2)/(Mux(j,i)**2))))
        Zeta(j,i)=LOG(Mux(j,i))-((Lamda(j,i)**2)/2.)
        If (Lamda(j,i).GT.0.) then
          Write(50,2245) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 2245     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1             16X,'Lamda(',A3,') = ',E17.10)
        else
          Write(*,5202) Ext(i)
          Write(50,5202) Ext(i)
 5202     Format(// 16X,'ERROR - Log-Normal distribution'/ 19X,
     1           'Lamda(',A3,') <= 0 !' /)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine PGumbel(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
C
C       ################################################################
C
C       Returns the parameters, Zeta & Lamda, of a Gumbel distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       Pi - 3.14159...
C
C       OUTPUT VARIABLES:
C       Zeta, Lamda - Parameters of the Gumbel distribution
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Pi,Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L)
        Pi=4.*ATAN(1.)
        Zeta(j,i)=Pi/((SQRT(6.))*Sigmax(j,i))
        Lamda(j,i)=Mux(j,i)-0.57722/(Zeta(j,i))
        If (Zeta(j,i).GT.0.) then
          Write(50,6890) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 6890     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1             16X,'Lamda(',A3,') = ',E17.10)
        else
          Write(*,5202) Ext(i)
          Write(50,5202) Ext(i)
 5202     Format(// 16X,'ERROR - Gumbel distribution'/ 19X,
     1           'Zeta(',A3,') <= 0 !' /)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine PRectangular(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
C
C       ################################################################
C
C       Returns the parameters, Zeta & Lamda, of a Rectangular
```

```
C       distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       Zeta, Lamda - Parameters of the Rectangular distribution
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L)
        Zeta(j,i)=Mux(j,i)-SQRT(3.)*Sigmax(j,i)
        Lamda(j,i)=Mux(j,i)+SQRT(3.)*Sigmax(j,i)
        If (Zeta(j,i).LT.Lamda(j,i)) then
          Write(50,6871) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 6871     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1             16X,'Lamda(',A3,') = ',E17.10)
        else
          Write(*,5202) Ext(i),Ext(i)
          Write(50,5202) Ext(i),Ext(i)
 5202     Format(// 16X,'ERROR - Rectangular distribution'/ 19X,
     1          'Zeta(',A3,') >= Lamda(',A3,') !' /)
          STOP
        Endif
        return
        End


C       ################################################################
C
        Subroutine PGamma(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
C
C       ################################################################
C
C       Returns the parameters, Zeta & Lamda, of a Gamma distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       Zeta, Lamda - Parameters of the Gamma distribution
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L)
        Zeta(j,i)=(Mux(j,i)/Sigmax(j,i))**2
        Lamda(j,i)=Mux(j,i)/(Sigmax(j,i)**2)
        If ((Zeta(j,i).GT.0.0).AND.(Lamda(j,i).GT.0.0)) then
          Write(50,6820) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 6820     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1             16X,'Lamda(',A3,') = ',E17.10)
        else
```

```
          If (Zeta(j,i).EQ.0.0) then
            Write(*,5670) Ext(i)
            Write(50,5670) Ext(i)
 5670       Format(/ 16X,'ERROR - Gamma distribution' / 19X,
     1               'Zeta(',A3,') = 0 !' /)
          Endif
          If (Lamda(j,i).LE.0.0) then
            Write(*,5689) Ext(i)
            Write(50,5689) Ext(i)
 5689       Format(// 16X,'ERROR - Gamma distribution' / 19X,
     1               'Lamda(',A3,') <= 0 !' /)
          Endif
          STOP
        Endif
        return
        End




C       ##################################################################
C
        Subroutine PBeta(j,i,Ext,Mux,Sigmax,Zeta,Lamda,x1,x2)
C
C       ##################################################################
C
C       Returns the parameters, Zeta & Lamda, of a Beta distribution.
C
C       ##################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       x1 - Lower limit on X
C       x2 - Upper limit on X
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C            program
C
C       OUTPUT VARIABLES:
C       Zeta, Lamda - Parameters of the Beta distribution
C
C       ##################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),x1(Q,L),x2(Q,L),Zeta(Q,L),Lamda(Q,L)
        Zeta(j,i)=(x1(j,i)-Mux(j,i))/(x2(j,i)-x1(j,i))+(x2(j,i)-
     1             Mux(j,i))*((Mux(j,i)-x1(j,i))**2)/((Sigmax(j,i)**2)*
     1             (x2(j,i)-x1(j,i)))
        Lamda(j,i)=((x2(j,i)-x1(j,i))*Zeta(j,i)/(Mux(j,i)-x1(j,i)))-
     1             Zeta(j,i)
        If ((Zeta(j,i).GT.0.0).AND.(Lamda(j,i).GT.0.0)) then
          Write(50,6861) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 6861     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1             16X,'Lamda(',A3,') = ',E17.10)
        else
          If (Zeta(j,i).LE.0.0) then
            Write(*,5673) Ext(i)
            Write(50,5673) Ext(i)
 5673       Format(/ 16X,'ERROR - Beta distribution'/ 19X,
     1               'Zeta(',A3,') <= 0 !' /)
          Endif
          If (Lamda(j,i).LE.0.0) then
            Write(*,5682) Ext(i)
            Write(50,5682) Ext(i)
 5682       Format(// 16X,'ERROR - Beta distribution'/ 19X,
     1               'Lamda(',A3,') <= 0 !' /)
          Endif
          STOP
        Endif
        return
        End




C       ##################################################################
```

```
C
         Subroutine PFrechet(j,i,Ext,Mux,Sigmax)
C
C        #############################################################
C
C        Returns the parameters, Zeta & Lamda, of a Frechet distribution.
C        Since there is no explicit form for calculating Zeta, it uses
C        subroutine Zbrent to evaluate it.
C
C        #############################################################
C
C        INPUT VARIABLES:
C        Ext - Abbreviation of the name of the variable
C        Mux - Mean of X
C        Sigmax - Standard deviation of X
C        i - Number of the variable
C        j - Number of the combination of actions
C
C        MODELING VARIABLES:
C        VarDis - Type of distribution
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        Zbrent, Gammln - External functions
C        a1 - Lower limit on Zbrent
C        a2 - Upper limit on Zbrent
C        Tol - Accuracy of Zbrent
C        T, Aux1, Aux2 - Auxiliary variables
C        Eta, x1, x2 - Variables mentioned in the Common statement but
C                      not used here
C
C        OUTPUT VARIABLES:
C        Zeta, Lamda - Parameters of the Frechet distribution
C
C        #############################################################
C
         Integer*4 j,i,L,Q
         Parameter (L=15)
         Parameter (Q=16)
         Character*3 Ext(L)
         Real*8 Mux(Q,L),Sigmax(Q,L),a1,a2,Tol,Zeta(Q,L),Lamda(Q,L),
     1         Eta(Q,L),x1(Q,L),x2(Q,L),VarDis(Q,L),T,Aux1,Aux2,Zbrent,
     1         Gammln
         Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
         External Zbrent
         External Gammln
         Aux1=Mux(j,i)
         Aux2=Sigmax(j,i)
         a1=2.05
         a2=99999999.0
         Tol=0.00000001
         VarDis(j,i)=7
         Zeta(j,i)=Zbrent(j,i,Aux1,Aux2,a1,a2,Tol,VarDis)
         T=1.-1./Zeta(j,i)
         Lamda(j,i)=Mux(j,i)/EXP(Gammln(T))
         If ((Zeta(j,i).GT.2.0).AND.(Lamda(j,i).GT.0.0)) then
           Write(50,8889) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 8889      Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1              16X,'Lamda(',A3,') = ',E17.10)
         else
           If (Zeta(j,i).LE.2.0) then
             Write(*,5699) Ext(i)
             Write(50,5699) Ext(i)
 5699        Format(/ 16X,'ERROR - Frechet distribution'/ 19X,
     1                'Zeta(',A3,') <= 2 !' /)
           Endif
           If (Lamda(j,i).LE.0.0) then
             Write(*,9989) Ext(i)
             Write(50,9989) Ext(i)
 9989        Format(// 16X,'ERROR - Frechet distribution'/ 19X,
     1                'Lamda(',A3,') <= 0 !')
           Endif
           STOP
         Endif
         return
         End



C        #############################################################
C
         Subroutine PExponential(j,i,Ext,Mux,Sigmax,Zeta,Lamda)
C
```

```
C       ################################################################
C
C       Returns the parameters, Zeta & Lamda, of an Exponential
C       distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       Zeta, Lamda - Parameters of the Exponential distribution
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),Zeta(Q,L),Lamda(Q,L)
        Zeta(j,i)=Mux(j,i)-Sigmax(j,i)
        Lamda(j,i)=Sigmax(j,i)
        If (Lamda(j,i).GT.0.0) then
          Write(50,6869) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i)
 6869     Format(/ 16X,'Zeta(',A3,') = ',E17.10 /
     1             16X,'Lamda(',A3,') = ',E17.10)
        else
          Write(*,9979) Ext(i)
          Write(50,9979) Ext(i)
 9979     Format(// 16X,'ERROR - Exponential distribution'/ 19X,
     1             'Lamda(',A3,') <= 0 !' /)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine PRayleigh(j,i,Ext,Mux,Sigmax,Zeta)
C
C       ################################################################
C
C       Returns the parameter, Zeta, of a Rayleigh distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       Pi - 3.14159...
C
C       OUTPUT VARIABLES:
C       Zeta - Parameter of the Rayleigh distribution
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Pi,Mux(Q,L),Sigmax(Q,L),Zeta(Q,L)
        Pi=4.*ATAN(1.)
        If (ABS(((Sigmax(j,i)/Mux(j,i))**2)-(4./Pi-1.)).LE.
     1     0.0000001) then
```

```
          Zeta(j,i)=Mux(j,i)*SQRT(2.)/SQRT(Pi)
          If (Zeta(j,i).GT.0.) then
            Write(50,6879) Ext(i),Zeta(j,i)
 6879       Format(/ 16X,'Zeta(',A3,') = ',E17.10)
          else
            Write(*,8133) Ext(i)
            Write(50,8133) Ext(i)
 8133       Format(// 16X,'ERROR - Rayleigh distribution'/ 19X,
     1             'Zeta(',A3,') <= 0 !'/)
            STOP
          Endif
        else
          Write(*,8132) Ext(i),Ext(i)
          Write(50,8132) Ext(i),Ext(i)
 8132     Format(// 16X,'ERROR - Rayleigh distribution'/ 19X,
     1           '(Mux(',A3,')/Sigmax(',A3,
     1           ')) not equal to SQRT(4/Pi-1) !'/)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine PWeibull(j,i,Ext,Mux,Sigmax)
C
C       ################################################################
C
C       Returns the parameters, Lamda & Eta, of a Weibull distribution
C       for a given Zeta parameter. Since there is no explicit form for
C       calculating Eta, it uses subroutine Zbrent to evaluate it.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Ext - Abbreviation of the name of the variable
C       Mux - Mean of X
C       Sigmax - Standard deviation of X
C       Zeta - Parameter of the Weibull distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       VarDis - Type of distribution
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       Zbrent, Gammln - External functions
C       a1 - Lower limit on Zbrent
C       a2 - Upper limit on Zbrent
C       Tol - Accuracy of Zbrent
C       T, Aux1, Aux2 - Auxiliary variables
C       x1, x2 - Variables mentioned in the Common statement but not
C                used here
C
C       OUTPUT VARIABLES:
C       Lamda, Eta - Parameters of the Weibull distribution
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Real*8 Mux(Q,L),Sigmax(Q,L),a1,a2,Tol,Zeta(Q,L),Lamda(Q,L),
     1         Eta(Q,L),x1(Q,L),x2(Q,L),VarDis(Q,L),T,Aux1,Aux2,Zbrent,
     1         Gammln
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
        External Zbrent
        External Gammln
        Aux1=Mux(j,i)
        Aux2=Sigmax(j,i)
        a1=0.012
        a2=9999.0
        Tol=0.00000001
        VarDis(j,i)=10
        Eta(j,i)=Zbrent(j,i,Aux1,Aux2,a1,a2,Tol,VarDis)
        T=1.+1./Eta(j,i)
        Lamda(j,i)=(Mux(j,i)-Zeta(j,i))/EXP(Gammln(T))
        If ((Lamda(j,i).GT.0.0).AND.(Eta(j,i).GT.0.0)) then
          Write(50,8880) Ext(i),Zeta(j,i),Ext(i),Lamda(j,i),
```

```
     1                      Ext(i),Eta(j,i)
  8880     Format(/ 16X,'Zeta(',A3,') = ',E17.10
     1            / 16X,'Lamda(',A3,') = ',E17.10
     1            / 16X,'Eta(',A3,') = ',E17.10)
         else
           If (Lamda(j,i).LE.0.0) then
             Write(*,5679) Ext(i)
             Write(50,5679) Ext(i)
  5679       Format(// 16X,'ERROR - Weibull distribution'/ 19X,
     1              'Lamda(',A3,') <= 0 !')
           Endif
           If (Eta(j,i).LE.0.0) then
             Write(*,9239) Ext(i)
             Write(50,9239) Ext(i)
  9239       Format(// 16X,'ERROR - Weibull distribution'/ 19X,
     1              'Eta(',A3,') <= 0 !')
           Endif
           STOP
         Endif
         return
         End




C      #################################################################
C
         Subroutine InvGamma(j,i,CDF,StartPt)
C
C      #################################################################
C
C      Returns StartPt for a specific value CDF of the cumulative
C      distribution function of a Gamma distribution. Since there is no
C      explicit form for the inverse function of a Gamma distribution,
C      it uses subroutine Zbrent to evaluate it.
C
C      #################################################################
C
C      INPUT VARIABLES:
C      Lamda - Parameter of the Gamma distribution
C      CDF - Value of the cumulative distribution function of a Gamma
C            distribution
C      i - Number of the variable
C      j - Number of the combination of actions
C
C      MODELING VARIABLES:
C      VarDis - Type of distribution
C      L - Maximum number of variables allowed by the program
C      Q - Maximum number of combinations of actions allowed by the
C            program
C      Zbrent - External function
C      a1 - Lower limit on Zbrent
C      a2 - Upper limit on Zbrent
C      Tol - Accuracy of Zbrent
C      Aux1, Aux2 - Auxiliary variables
C      Zeta, Eta, x1, x2 - Variables mentioned in the Common statement
C                          but not used here
C
C      OUTPUT VARIABLE:
C      StartPt - Value of the inverse function of a Gamma distribution
C
C      #################################################################
C
         Integer*4 j,i,L,Q
         Parameter (L=15)
         Parameter (Q=16)
         Real*8 a1,a2,Tol,Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L),
     1         CDF,StartPt(Q,L),VarDis(Q,L),Aux1,Aux2,Zbrent
         Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
         External Zbrent
         Aux1=CDF
         Aux2=0.
         a1=0.0
         a2=1E25
         Tol=0.00000001
         VarDis(j,i)=5
         StartPt(j,i)=(Zbrent(j,i,Aux1,Aux2,a1,a2,Tol,VarDis))/Lamda(j,i)
         return
         End




C      #################################################################
C
```

```
          Subroutine InvBeta(j,i,CDF,StartPt)
C
C          ###################################################################
C
C          Returns StartPt for a specific value CDF of the cumulative
C          distribution function of a Beta distribution. Since there is no
C          explicit form for the inverse function of a Beta distribution,
C          it uses subroutine Zbrent to evaluate it.
C
C          ###################################################################
C
C          INPUT VARIABLES:
C          x1 - Lower limit on X
C          x2 - Upper limit on X
C          CDF - Value of the cumulative distribution function of a Beta
C                distribution
C          i - Number of the variable
C          j - Number of the combination of actions
C
C          MODELING VARIABLES:
C          VarDis - Type of distribution
C          Zbrent - External function
C          a1 - Lower limit on Zbrent
C          a2 - Upper limit on Zbrent
C          Tol - Accuracy of Zbrent
C          L - Maximum number of variables allowed by the program
C          Q - Maximum number of combinations of actions allowed by the
C                program
C          Aux1, Aux2 - Auxiliary variables
C          Zeta, Lamda, Eta - Variables mentioned in the Common statement
C                             but not used here
C
C          OUTPUT VARIABLE:
C          StartPt - Value of the inverse function of a Beta distribution
C
C          ###################################################################
C
          Integer*4 j,i,L,Q
          Parameter (L=15)
          Parameter (Q=16)
          Real*8 a1,a2,Tol,Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L),
     1          StartPt(Q,L),VarDis(Q,L),CDF,Zbrent,Aux1,Aux2
          Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
          External Zbrent
          Aux1=CDF
          Aux2=0.
          a1=0
          a2=1
          Tol=0.00000001
          VarDis(j,i)=6
          StartPt(j,i)=x1(j,i)+(x2(j,i)-x1(j,i))*
     1                 Zbrent(j,i,Aux1,Aux2,a1,a2,Tol,VarDis)
          return
          End




C          ###################################################################
C
          Subroutine InvUser1(j,i,CDF1,StartPt)
C
C          ###################################################################
C
C          Returns the value of the water level, StartPt, for a specific
C          value, CDF1, of the user-defined distribution of water levels.
C          It uses the data file wldata.dad which contains the tabulated
C          values.
C
C          ###################################################################
C
C          INPUT VARIABLE:
C          i - Number of the variable
C          j - Number of the combination of actions
C          CDF1 - Value of the cumulative distribution function of the
C                 user-defined distribution of water levels
C
C          MODELING VARIABLES:
C          M - Number of points of the user-defined distribution tabulated
C                in file wldata.dad
C          WL - Value tabulated in file wldata.dad of the water level of
C                the user-defined distribution
C          PDF - Value tabulated in file wldata.dad of the probability
C                 density function of the user-defined distribution
```

```
C        CDF - Value tabulated in file wldata.dad of the cumulative
C            distribution function of the user-defined distribution
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        k - Auxiliary variable
C
C        OUTPUT VARIABLE:
C        StartPt - Value of the inverse function of the user-defined
C                distribution of water levels which corresponds to CDF1
C
C        ###############################################################
C
         Integer*4 i,j,k,M,Q,L
         Parameter (M=126)
         Parameter (Q=16)
         Parameter (L=15)
         Real*8 WL(M),PDF(M),CDF(M),CDF1,StartPt(Q,L)
         Open(Unit=70, File='wldata.dad', Status='Old')
         Do 10 k=1,M
           Read(70,*) WL(k),PDF(k),CDF(k)
  10     continue
         Do 32 k=1,M
           If (CDF(k).LE.CDF1) then
             If (CDF(k).LT.CDF1) then
               StartPt(j,i)=((CDF1-CDF(k))*((WL(k+1))-WL(k)))/
     1              ((CDF(k+1))-CDF(k))+WL(k)
               goto 1150
             else
               StartPt(j,i)=WL(k)
               goto 1150
             Endif
           Endif
  32     continue
 1150    Close (Unit=70)
         return
         End


C        ###############################################################
C
         Subroutine InvUser2(j,i,CDF1,StartPt)
C
C        ###############################################################
C
C        Returns the value of the extreme water level, StartPt, for a
C        specific value, CDF1, of the user-defined distribution of
C        extreme water levels. It uses the data file extwldat.dad which
C        contains the tabulated values.
C
C        ###############################################################
C
C        INPUT VARIABLE:
C        i - Number of the variable
C        j - Number of the combination of actions
C        CDF1 - Value of the cumulative distribution function of the
C            user-defined distribution of extreme water levels
C
C        MODELING VARIABLES:
C        M - Number of points of the user-defined distribution tabulated
C            in file extwldat.dad
C        ExtWL - Value tabulated in file extwldat.dad of the extreme
C              water level of the user-defined distribution
C        PDF - Value tabulated in file extwldat.dad of the probability
C            density function of the user-defined distribution
C        CDF - Value tabulated in file extwldat.dad of the cumulative
C            distribution function of the user-defined distribution
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        k - Auxiliary variable
C
C        OUTPUT VARIABLE:
C        StartPt - Value of the inverse function of the user-defined
C                distribution of extreme water levels which corresponds
C                to CDF1
C
C        ###############################################################
C
         Integer*4 i,j,k,M,Q,L
         Parameter (M=17)
         Parameter (Q=16)
```

```
          Parameter (L=15)
          Real*8 ExtWL(M),PDF(M),CDF(M),CDF1,StartPt(Q,L)
          Open(Unit=75, File='extwldat.dad', Status='Old')
          Do 10 k=1,M
            Read(75,*) ExtWL(k),PDF(k),CDF(k)
 10       continue
          Do 32 k=1,M
            If (CDF(k).LE.CDF1) then
              If (CDF(k).LT.CDF1) then
                StartPt(j,i)=((CDF1-CDF(k))*((ExtWL(k+1))-ExtWL(k)))/
      1            ((CDF(k+1))-CDF(k))+ExtWL(k)
                goto 1150
              else
                StartPt(j,i)=ExtWL(k)
                goto 1150
              Endif
            Endif
 32       continue
 1150     Close (Unit=75)
          return
          End




 C       ################################################################
 C
          Subroutine InvUser3(j,i,CDF1,StartPt)
 C
 C       ################################################################
 C
 C       Returns the value of the tide level, StartPt, for a specific
 C       value, CDF1, of the user-defined distribution of tide levels.
 C       It uses the data file tide.dad which contains the tabulated
 C       values.
 C
 C       ################################################################
 C
 C       INPUT VARIABLE:
 C       i - Number of the variable
 C       j - Number of the combination of actions
 C       CDF1 - Value of the cumulative distribution function of the
 C              user-defined distribution of tide levels
 C
 C       MODELING VARIABLES:
 C       M - Number of points of the user-defined distribution tabulated
 C           in file tide.dad
 C       Tide - Value tabulated in file tide.dad of the tide level of
 C              the user-defined distribution
 C       PDF - Value tabulated in file tide.dad of the probability
 C             density function of the user-defined distribution
 C       CDF - Value tabulated in file tide.dad of the cumulative
 C             distribution function of the user-defined distribution
 C       L - Maximum number of variables allowed by the program
 C       Q - Maximum number of combinations of actions allowed by the
 C           program
 C       k - Auxiliary variable
 C
 C       OUTPUT VARIABLE:
 C       StartPt - Value of the inverse function of the user-defined
 C                 distribution of tide levels which corresponds to CDF1
 C
 C       ################################################################
 C
          Integer*4 i,j,k,M,Q,L
          Parameter (M=122)
          Parameter (Q=16)
          Parameter (L=15)
          Real*8 Tide(M),PDF(M),CDF(M),CDF1,StartPt(Q,L)
          Open(Unit=80, File='tide.dad', Status='Old')
          Do 10 k=1,M
            Read(80,*) Tide(k),PDF(k),CDF(k)
 10       continue
          Do 32 k=1,M
            If (CDF(k).LE.CDF1) then
              If (CDF(k).LT.CDF1) then
                StartPt(j,i)=((CDF1-CDF(k))*((Tide(k+1))-Tide(k)))/
      1            ((CDF(k+1))-CDF(k))+Tide(k)
                goto 1150
              else
                StartPt(j,i)=Tide(k)
                goto 1150
              Endif
            Endif
```

```
 32      continue
1150     Close (Unit=80)
         return
         End




C        ################################################################
C
         Subroutine NorWrite(i,Ext,MuxN,SigmaxN)
C
C        ################################################################
C
C        Writes the mean, MuxN, & the standard deviation, SigmaxN, of the
C        equivalent Normal distribution of X in the file results.dat .
C
C        ################################################################
C
C        INPUT VARIABLES:
C        Ext - Abbreviation of the name of the variable
C        MuxN - Mean of the equivalent Normal distribution of X
C        SigmaxN - Standard deviation of the equivalent Normal
C                distribution of X
C        i - Number of the variable
C
C        MODELING VARIABLE:
C        L - Maximum number of variables allowed by the program
C
C        ################################################################
C
         Integer*4 i,L
         Parameter (L=15)
         Character*3 Ext(L)
         Real*8 MuxN(L),SigmaxN(L)

         Write(50,8709) Ext(i),MuxN(i),Ext(i),SigmaxN(i)
 8709    Format(/ 16X,'Mean Value N(',A3,') = ',E17.10 /
     1           16X,'Standard Deviation N(',A3,') = ',E17.10)
         return
         End




C        ################################################################
C
         Subroutine NormalD(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,
     1                      Xo,NR)
C
C        ################################################################
C
C        Calculates the value of the probability density function, PDFx,
C        at X & the value of the cumulative distribution function, CDFx,
C        at X of a Normal distribution (CDFx is calculated using
C        subroutine NormalDist). If the Normal distribution is truncated
C        at X=Xo, it also calculates the value of the cumulative
C        distribution function at X=X0, CDFx0. These values are used to
C        return the mean, MuxN, & the standard deviation, SigmaxN, of the
C        equivalent Normal distribution through subroutine Truncation.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        X - Variable
C        Ext - Abbreviation of the name of the variable
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        NR - Power to which each distribution is raised for each
C             combination of actions
C        Zeta, Lamda - Parameters of the Normal distribution
C        i - Number of the variable
C        j - Number of the combination of actions
C
C        MODELING VARIABLES:
C        PDFx - Value of the probability density function of a Normal
C               distribution at X
C        CDFx - Value of the cumulative distribution function of a
C               Normal distribution at X
C        CDFx0 - Value of the cumulative distribution function of a
C                Normal distribution at X=X0
C        SNX - Standard Normal variable
C        Prob - Value of the exceedance cumulative distribution function
C               of the standard Normal distribution which corresponds to
C               SNX
```

```
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C          program
C       Pi - 3.14159...
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 Trunc(Q,L)
        Real*8 X(L),MuxN(L),SigmaxN(L),Zeta(Q,L),Lamda(Q,L),CDFx(L),
     1         CDFx0(L),PDFx(L),Xo(Q,L),Pi,Prob(Q),SNX,NR(Q,L)

        Pi=4.*ATAN(1.)
        SNX=(X(i)-Zeta(j,i))/Lamda(j,i)
        Call NormalDist(j,SNX,Prob)
        CDFx(i)=1.-Prob(j)
        PDFx(i)=(EXP(-(((X(i)-Zeta(j,i))**2))/
     1          (2.*(Lamda(j,i)**2))))/
     1          (Lamda(j,i)*SQRT(2.*Pi))
C         Write(50,9824) SNX,Prob(j),CDFx(i),PDFx(i)
C 9824   Format(16X,'Normd X=',E17.10,3X,'Prob=',E17.10 /
C     1          16X,'CDFX=',E17.10,3X,'PDFX=',E17.10 /)
        If (Trunc(j,i).NE.0) then
          SNX=(Xo(j,i)-Zeta(j,i))/Lamda(j,i)
          Call NormalDist(j,SNX,Prob)
          CDFx0(i)=1.-Prob(j)
C          Write(50,9724) SNX,Prob(j),CDFx0(i)
C 9724    Format(16X,'Normd X=',E17.10,3X,'Prob=',E17.10 /
C     1          16X,'CDFX0=',E17.10 /)
        Endif
        Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                  PDFx,NR)
        If ((Trunc(j,i).EQ.0).AND.(NR(j,i).EQ.1)) then
          MuxN(i)=Zeta(j,i)
          SigmaxN(i)=Lamda(j,i)
          Call NorWrite(i,Ext,MuxN,SigmaxN)
        Endif
        return
        End




C       ################################################################
C
        Subroutine LogNormal(i,j,Ext,X,Zeta,Lamda,MuxN,
     1                      SigmaxN,Trunc,Xo,NR)
C
C       ################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of a Log-Normal distribution (CDFx is calculated using
C       subroutine NormalDist). If the Log-Normal distribution is
C       truncated at X=Xo, it also calculates the value of the
C       cumulative distribution function at X=X0, CDFx0. It uses these
C       values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine Truncation.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C           combination of actions
C       Zeta, Lamda - Parameters of the Log-Normal distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       PDFx - Value of the probability density function of a Log-Normal
C              distribution at X
```

```
C       CDFx - Value of the cumulative distribution function of a
C              Log-Normal distribution at X
C       CDFx0 - Value of the cumulative distribution function of a
C               Log-Normal distribution at X=X0
C       SNX - Standard Normal Variable
C       Prob - Value of the exceedance cumulative distribution function
C              of the standard Normal distribution which corresponds to
C              SNX
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       Pi - 3.14159...
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ##################################################################
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 Trunc(Q,L)
        Real*8 X(L),MuxN(L),SigmaxN(L),Zeta(Q,L),Lamda(Q,L),CDFx(L),
     1         CDFx0(L),PDFx(L),Prob(Q),Pi,SNX,Xo(Q,L),NR(Q,L)
        Pi=4.*ATAN(1.)
        If (X(i).GT.0.) then
          SNX=(LOG(X(i))-Zeta(j,i))/(Lamda(j,i))
          Call NormalDist(j,SNX,Prob)
          CDFx(i)=1.-Prob(j)
          PDFx(i)=(1./(X(i)*Lamda(j,i)))*(EXP(-(SNX**2)/2.))/SQRT(2.*Pi)
          If (Trunc(j,i).NE.0) then
            SNX=(LOG(Xo(j,i))-Zeta(j,i))/(Lamda(j,i))
            Call NormalDist(j,SNX,Prob)
            CDFx0(i)=1.-Prob(j)
          Endif
          Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,
     1                    CDFx0,PDFx,NR)
          If ((Trunc(j,i).EQ.0).AND.(NR(j,i).EQ.1)) then
            MuxN(i)=X(i)*(1.-LOG(X(i))+Zeta(j,i))
            SigmaxN(i)=X(i)*Lamda(j,i)
            Call NorWrite(i,Ext,MuxN,SigmaxN)
          Endif
        else
          Write(*,5209) Ext(i)
          Write(50,5209) Ext(i)
 5209     Format(// 16X,'ERROR - Log-Normal distribution'/ 19X,
     1           A3,' <= 0 !' /)
          STOP
        Endif
        return
        End




C       ##################################################################
C
        Subroutine Gumbel(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,NR)
C
C       ##################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of a Gumbel distribution. If the Gumbel distribution is
C       truncated at X=Xo, it also calculates the value of the
C       cumulative distribution function at X=X0, CDFx0. It uses these
C       values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine Truncation.
C
C       ##################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Zeta, Lamda - Parameters of the Gumbel distribution
C       i - Number of the variable
```

```
C         j - Number of the combination of actions
C
C         MODELING VARIABLES:
C         PDFx - Value of the probability density function of a Gumbel
C               distribution at X
C         CDFx - Value of the cumulative distribution function of a Gumbel
C               distribution at X
C         CDFx0 - Value of the cumulative distribution function of a
C                Gumbel distribution at X=X0
C         L - Maximum number of variables allowed by the program
C         Q - Maximum number of combinations of actions allowed by the
C             program
C
C         OUTPUT VARIABLES:
C         MuxN - Mean of the equivalent Normal distribution of X
C         SigmaxN - Standard deviation of the equivalent Normal
C                  distribution of X
C
C         ################################################################
C
          Integer*4 i,j,L,Q
          Parameter (L=15)
          Parameter (Q=16)
          Character*3 Ext(L)
          Integer*4 Trunc(Q,L)
          Real*8 X(L),MuxN(L),SigmaxN(L),Zeta(Q,L),Lamda(Q,L),
      1         CDFx(L),CDFx0(L),PDFx(L),Xo(Q,L),NR(Q,L)
          CDFx(i)=EXP(-EXP(-Zeta(j,i)*(X(i)-Lamda(j,i))))
          PDFx(i)=Zeta(j,i)*EXP(-Zeta(j,i)*(X(i)-Lamda(j,i)))*CDFx(i)
          If (Trunc(j,i).NE.0) CDFx0(i)=EXP(-EXP(-Zeta(j,i)*
      1                              (Xo(j,i)-Lamda(j,i))))
          Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
      1               PDFx,NR)
          return
          End




C         ################################################################
C
          Subroutine Rectangular(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,
      1                    Xo,NR)
C
C         ################################################################
C
C         Calculates the value of the probability density function, PDFx,
C         at X & the value of the cumulative distribution function, CDFx,
C         at X of a Rectangular distribution. If the Rectangular
C         distribution is truncated at X=Xo, it also calculates the value
C         of the cumulative distribution function at X=X0, CDFx0. It uses
C         these values to return the mean, MuxN, & the standard deviation,
C         SigmaxN, of the equivalent Normal distribution through
C         subroutine Truncation.
C
C         ################################################################
C
C         INPUT VARIABLES:
C         X - Variable
C         Ext - Abbreviation of the name of the variable
C         Trunc - Type of truncation
C         Xo - Point of truncation (if the distribution is truncated)
C         NR - Power to which each distribution is raised for each
C             combination of actions
C         Zeta, Lamda - Parameters of the Rectangular distribution
C         i - Number of the variable
C         j - Number of the combination of actions
C
C         MODELING VARIABLES:
C         PDFx - Value of the probability density function of a
C               Rectangular distribution at X
C         CDFx - Value of the cumulative distribution function of a
C               Rectangular distribution at X
C         CDFx0 - Value of the cumulative distribution function of a
C                Rectangular distribution at X=X0
C         L - Maximum number of variables allowed by the program
C         Q - Maximum number of combinations of actions allowed by the
C             program
C
C         OUTPUT VARIABLES:
C         MuxN - Mean of the equivalent Normal distribution of X
C         SigmaxN - Standard deviation of the equivalent Normal
C                  distribution of X
C
```

```
C        ##############################################################
C
         Integer*4 i,j,L,Q
         Parameter (L=15)
         Parameter (Q=16)
         Character*3 Ext(L)
         Integer*4 Trunc(Q,L)
         Real*8 X(L),MuxN(L),SigmaxN(L),Zeta(Q,L),Lamda(Q,L),
     1          CDFx(L),CDFx0(L),PDFx(L),Xo(Q,L),NR(Q,L)
         If ((X(i).LE.Lamda(j,i)).AND.(X(i).GE.Zeta(j,i)).AND.
     1      (Zeta(j,i).LT.Lamda(j,i))) then
           CDFx(i)=(X(i)-Zeta(j,i))/(Lamda(j,i)-Zeta(j,i))
           PDFx(i)=1./(Lamda(j,i)-Zeta(j,i))
           If (Trunc(j,i).NE.0) CDFx0(i)=(Xo(j,i)-Zeta(j,i))/
     1                                   (Lamda(j,i)-Zeta(j,i))
           Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                     PDFx,NR)
         else
           Write(*,5679) Ext(i),Zeta(j,i),Lamda(j,i)
           Write(50,5679) Ext(i),Zeta(j,i),Lamda(j,i)
 5679     Format(// 16X,'ERROR - Rectangular distribution'/ 19X,
     1            A3,' is not in the required range [',E17.10,',',
     1            E17.10,'] !' /)
           STOP
         Endif
         return
         End




C        ##############################################################
C
         Subroutine Gamma(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,NR)
C
C        ##############################################################
C
C        Calculates the value of the probability density function, PDFx,
C        at X & the value of the cumulative distribution function, CDFx,
C        at X of a Gamma distribution (it uses the functions Gammln &
C        Gammp to calculate, respectively, the Gamma function & the
C        incomplete Gamma function). If the Gamma distribution is
C        truncated at X=Xo, it also calculates the value of the
C        cumulative distribution function at X=X0, CDFx0. It uses these
C        values to return the mean, MuxN, & the standard deviation,
C        SigmaxN, of the equivalent Normal distribution through
C        subroutine Truncation.
C
C        ##############################################################
C
C        INPUT VARIABLES:
C        X - Variable
C        Ext - Abbreviation of the name of the variable
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        NR - Power to which each distribution is raised for each
C             combination of actions
C        Zeta, Lamda - Parameters of the Gamma distribution
C        i - Number of the variable
C        j - Number of the combination of actions
C
C        MODELING VARIABLES:
C        PDFx - Value of the probability density function of a Gamma
C               distribution at X
C        CDFx - Value of the cumulative distribution function of a Gamma
C               distribution at X
C        CDFx0 - Value of the cumulative distribution function of a Gamma
C                distribution at X=X0
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        Gam - Gamma function
C        Gammln, Gammp - External functions
C        XX, T, T1 - Auxiliary variables
C
C        OUTPUT VARIABLES:
C        MuxN - Mean of the equivalent Normal distribution of X
C        SigmaxN - Standard deviation of the equivalent Normal
C                  distribution of X
C
C        ##############################################################
C
         Integer*4 i,j,L,Q
         Parameter (L=15)
```

```
         Parameter (Q=16)
         Character*3 Ext(L)
         Integer*4 Trunc(Q,L)
         Real*8 Zeta(Q,L),X(L),Lamda(Q,L),T,T1,Gam,MuxN(L),SigmaxN(L),
     1          CDFx(L),CDFx0(L),PDFx(L),Xo(Q,L),Gammln,Gammp,NR(Q,L),XX
         External Gammln
         External Gammp
         If (X(i).GE.0.0) then
           XX=Zeta(j,i)
           Gam=EXP(Gammln(XX))
           T=Lamda(j,i)*X(i)
           Write(50,9230) Zeta(j,i),Gam
 9230      Format(16X,'Gamma(',E17.10,') = ',E17.10)
           PDFx(i)=(Lamda(j,i)*(T**(Zeta(j,i)-1.))*(EXP(-T)))/Gam
           CDFx(i)=Gammp(XX,T)
           If (Trunc(j,i).NE.0) then
             T1=Lamda(j,i)*Xo(j,i)
             CDFx0(i)=Gammp(XX,T1)
           Endif
           Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                     PDFx,NR)
         else
           Write(*,5679) Ext(i)
           Write(50,5679) Ext(i)
 5679      Format(// 16X,'ERROR - Gamma distribution'/ 19X,A3,' < 0 !' /)
           STOP
         Endif
         return
         End




C       ###################################################################
C
        Subroutine BetaDis(i,j,Ext,X,Zeta,Lamda,x1,x2,MuxN,SigmaxN,
     1                     Trunc,Xo,NR)
C
C       ###################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of a Beta distribution (it uses the functions Beta & Betai
C       to calculate, respectively, the Beta function & the incomplete
C       Beta function). If the Beta distribution is truncated at X=Xo,
C       it also calculates the value of the cumulative distribution
C       function at X=X0, CDFx0. It uses these values to return the
C       mean, MuxN, & the standard deviation, SigmaxN, of the equivalent
C       Normal distribution through subroutine Truncation.
C
C       ###################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Zeta, Lamda - Parameters of the Beta distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C       x1 - Lower limit on X
C       x2 - Upper limit on X
C
C       MODELING VARIABLES:
C       PDFx - Value of the probability density function of a Beta
C              distribution at X
C       CDFx - Value of the cumulative distribution function of a Beta
C              distribution at X
C       CDFx0 - Value of the cumulative distribution function of a Beta
C               distribution at X=X0
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       Beta, Betai - External functions
C       XX, YY, T, BetaAux - Auxiliary variables
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ###################################################################
```

```
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 Trunc(Q,L)
        Real*8 Zeta(Q,L),X(L),Lamda(Q,L),x1(Q,L),x2(Q,L),BetaAux(Q,L),T,
     1         MuxN(L),SigmaxN(L),CDFx(L),CDFx0(L),PDFx(L),Xo(Q,L),
     1         Betai,Beta,NR(Q,L),XX,YY
        External Beta
        External Betai
        If ((X(i).LE.x2(j,i)).AND.(X(i).GE.x1(j,i)).AND.
     1      (x1(j,i).LT.x2(j,i))) then
          XX=Zeta(j,i)
          YY=Lamda(j,i)
          BetaAux(j,i)=Beta(XX,YY)
          PDFx(i)=((X(i)-x1(j,i))**(Zeta(j,i)-1.))*((x2(j,i)-X(i))**
     1           (Lamda(j,i)-1.))/(BetaAux(j,i)*((x2(j,i)-x1(j,i))**
     1           (Zeta(j,i)+Lamda(j,i)-1.)))
          T=(X(i)-x1(j,i))/(x2(j,i)-x1(j,i))
          CDFx(i)=Betai(XX,YY,T)
          If (Trunc(j,i).NE.0) then
            T=(Xo(j,i)-x1(j,i))/(x2(j,i)-x1(j,i))
            CDFx0(i)=Betai(XX,YY,T)
          Endif
          Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                   PDFx,NR)
        else
          Write(*,5579) Ext(i),x1(j,i),x2(j,i)
          Write(50,5579) Ext(i),x1(j,i),x2(j,i)
 5579     Format(// 16X,'ERROR - Beta distribution'/ 19X,
     1           A3,' is not in the specified range '/ 19X,
     1           '[',E17.10,',',E17.10,'] !' /)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine Frechet(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,Xo,
     1                   NR)
C
C       ################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of a Frechet distribution. If the Frechet distribution is
C       truncated at X=Xo, it also calculates the value of the
C       cumulative distribution function at X=X0, CDFx0. It uses these
C       values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine Truncation.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Zeta, Lamda - Parameters of the Frechet distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       PDFx - Value of the probability density function of a Frechet
C             distribution at X
C       CDFx - Value of the cumulative distribution function of a
C             Frechet distribution at X
C       CDFx0 - Value of the cumulative distribution function of a
C              Frechet distribution at X=X0
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
```

```
C               distribution of X
C
C       ################################################################
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 Trunc(Q,L)
        Real*8 Zeta(Q,L),X(L),Lamda(Q,L),MuxN(L),SigmaxN(L),CDFx(L),
     1         CDFx0(L),PDFx(L),Xo(Q,L),NR(Q,L)
        If (X(i).GT.0.0) then
          CDFx(i)=EXP(-((Lamda(j,i)/X(i))**Zeta(j,i)))
          PDFx(i)=(Zeta(j,i)/Lamda(j,i))*((Lamda(j,i)/X(i))**
     1          (Zeta(j,i)+1.))*CDFx(i)
          If (Trunc(j,i).NE.0) CDFx0(i)=EXP(-((Lamda(j,i)/
     1                             Xo(j,i))**Zeta(j,i)))
          Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                  PDFx,NR)
        else
          Write(*,9979) Ext(i)
          Write(50,9979) Ext(i)
 9979     Format(// 16X,'ERROR - Frechet distribution'/ 19X,
     1           A3,' <= 0 !' /)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine Exponential(i,j,Ext,X,Zeta,Lamda,MuxN,SigmaxN,Trunc,
     1                   Xo,NR)
C
C       ################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of an Exponential distribution. If the Exponential
C       distribution is truncated at X=Xo, it also calculates the value
C       of the cumulative distribution function at X=X0, CDFx0. It uses
C       these values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine Truncation.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Zeta, Lamda - Parameters of the Exponential distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       PDFx - Value of the probability density function of an
C              Exponential distribution at X
C       CDFx - Value of the cumulative distribution function of an
C              Exponential distribution at X
C       CDFx0 - Value of the cumulative distribution function of an
C               Exponential distribution at X=X0
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 Trunc(Q,L)
```

```
          Real*8 X(L),Zeta(Q,L),MuxN(L),SigmaxN(L),CDFx(L),CDFx0(L),
     1           PDFx(L),Xo(Q,L),Lamda(Q,L),NR(Q,L)
          If (X(i).GE.Zeta(j,i)) then
            CDFx(i)=1.-EXP(-((X(i)-Zeta(j,i))/Lamda(j,i)))
            PDFx(i)=(1./Lamda(j,i))*(1.-CDFx(i))
            If (Trunc(j,i).NE.0) CDFx0(i)=1.-EXP(-((Xo(j,i)-Zeta(j,i))/
     1                                   Lamda(j,i)))
            Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                   PDFx,NR)
          else
            Write(*,8889) Ext(i),Zeta(j,i)
            Write(50,8889) Ext(i),Zeta(j,i)
 8889     Format(// 16X,'ERROR - Exponential distribution'/ 19X,
     1           A3,' < ',E17.10,' !' /)
            STOP
          Endif
          return
          End



C       ################################################################
C
          Subroutine Rayleigh(i,j,Ext,X,Zeta,MuxN,SigmaxN,Trunc,Xo,NR)
C
C       ################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of a Rayleigh distribution. If the Rayleigh distribution is
C       truncated at X=Xo, it also calculates the value of the
C       cumulative distribution function at X=X0, CDFx0. It uses these
C       values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine Truncation.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Zeta - Parameter of the Rayleigh distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       PDFx - Value of the probability density function of a Rayleigh
C              distribution at X
C       CDFx - Value of the cumulative distribution function of a
C              Rayleigh distribution at X
C       CDFx0 - Value of the cumulative distribution function of a
C               Rayleigh distribution at X=X0
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
          Integer*4 i,j,L,Q
          Parameter (L=15)
          Parameter (Q=16)
          Character*3 Ext(L)
          Integer*4 Trunc(Q,L)
          Real*8 X(L),Zeta(Q,L),MuxN(L),SigmaxN(L),CDFx(L),CDFx0(L),
     1           PDFx(L),Xo(Q,L),NR(Q,L)
          If (X(i).GE.0.0) then
            CDFx(i)=1.-EXP(-(X(i)**2)/(2.*(Zeta(j,i)**2)))
            PDFx(i)=(X(i)/(Zeta(j,i)**2))*EXP(-(X(i)**2)/
     1           (2.*(Zeta(j,i)**2)))
            If (Trunc(j,i).NE.0) CDFx0(i)=1.-EXP(-(Xo(j,i)**2)/
     1                                   (2.*(Zeta(j,i)**2)))
            Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,CDFx0,
     1                   PDFx,NR)
          else
```

```
          Write(*,8779) Ext(i)
          Write(50,8779) Ext(i)
 8779     Format(// 16X,'ERROR - Rayleigh distribution'/ 19X,
     1            A3,' < 0 !' /)
          STOP
        Endif
        return
        End




C       ################################################################
C
        Subroutine Weibull(i,j,Ext,X,Zeta,Lamda,Eta,
     1                  MuxN,SigmaxN,Trunc,Xo,NR)
C
C       ################################################################
C
C       Calculates the value of the probability density function, PDFx,
C       at X & the value of the cumulative distribution function, CDFx,
C       at X of a Weibull distribution. If the Weibull distribution is
C       truncated at X=Xo, it also calculates the value of the
C       cumulative distribution function at X=X0, CDFx0. It uses these
C       values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine Truncation.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       Trunc - Type of truncation
C       Xo - Point of truncation (if the distribution is truncated)
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       Zeta, Lamda, Eta - Parameters of the Weibull distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       PDFx - Value of the probability density function of a Weibull
C              distribution at X
C       CDFx - Value of the cumulative distribution function of a
C              Weibull distribution at X
C       CDFx0 - Value of the cumulative distribution function of a
C               Weibull distribution at X=X0
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
        Integer*4 i,j,L,Q
        Parameter (L=15)
        Parameter (Q=16)
        Character*3 Ext(L)
        Integer*4 Trunc(Q,L)
        Real*8 Zeta(Q,L),X(L),Lamda(Q,L),Eta(Q,L),MuxN(L),SigmaxN(L),
     1         CDFx(L),CDFx0(L),PDFx(L),Xo(Q,L),NR(Q,L)
        If (X(i).GE.Zeta(j,i)) then
          CDFx(i)=1.-EXP(-(((X(i)-Zeta(j,i))/Lamda(j,i))**Eta(j,i)))
          PDFx(i)=(Eta(j,i)/Lamda(j,i))*(1.-CDFx(i))*
     1            (((X(i)-Zeta(j,i))/Lamda(j,i))**(Eta(j,i)-1))
          If (Trunc(j,i).NE.0) CDFx0(i)=1.-EXP(-(((Xo(j,i)-Zeta(j,i))
     1                                  /Lamda(j,i))**Eta(j,i)))
          Call Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,
     1                  CDFx0,PDFx,NR)
        else
          Write(*,9979) Ext(i)
          Write(50,9979) Ext(i)
 9979     Format(// 16X,'ERROR - Weibull distribution'/ 19X,
     1            A3,' < Zeta !' /)
          STOP
        Endif
        return
        End
```

```
C      ################################################################
C
       Subroutine User1(i,j,Ext,X,MuxN,SigmaxN,NR)
C
C      ################################################################
C
C      Returns the value of the probability density function, PDFx,
C      & the value of the cumulative distribution function, CDFx, of
C      the user-defined distribution of water levels, for a given value
C      of the water level, X. It uses the data file wldata.dad which
C      contains the tabulated values. It uses these values to return
C      the mean, MuxN, & the standard deviation, SigmaxN, of the
C      equivalent Normal distribution through subroutine PCDF.
C
C      ################################################################
C
C      INPUT VARIABLES:
C      X - Variable
C      Ext - Abbreviation of the name of the variable
C      NR - Power to which each distribution is raised for each
C           combination of actions
C      i - Number of the variable
C      j - Number of the combination of actions
C
C      MODELING VARIABLES:
C      M - Number of points of the user-defined distribution tabulated
C          in file wldata.dad
C      WL - Value tabulated in file wldata.dad of the water level of
C           the user-defined distribution
C      PDF - Value tabulated in file wldata.dad of the probability
C            density function of the user-defined distribution
C      CDF - Value tabulated in file wldata.dad of the cumulative
C            distribution function of the user-defined distribution
C      L - Maximum number of variables allowed by the program
C      Q - Maximum number of combinations of actions allowed by the
C          program
C      PDFx - Value of the probability density function of the
C             user-defined distribution at X
C      CDFx - Value of the cumulative distribution function of the
C             user-defined distribution at X
C      k - Auxiliary variable
C
C      OUTPUT VARIABLES:
C      MuxN - Mean of the equivalent Normal distribution of X
C      SigmaxN - Standard deviation of the equivalent Normal
C                distribution of X
C
C      ################################################################
C
       Integer*4 i,j,k,M,Q,L
       Parameter (M=126)
       Parameter (Q=16)
       Parameter (L=15)
       Character*3 Ext(L)
       Real*8 WL(M),PDF(M),CDF(M),X(L),PDFx(L),CDFx(L),MuxN(L),
     1        SigmaxN(L),NR(Q,L)
       Open(Unit=70, File='wldata.dad', Status='Old')
       Do 10 k=1,M
         Read(70,*) WL(k),PDF(k),CDF(k)
 10    continue
       Do 32 k=1,M
         If (WL(k).GE.X(i)) then
           If (WL(k).GT.X(i)) then
             PDFx(i)=((WL(k)-X(i))/(WL(k)-WL(k-1)))*PDF(k-1)+
     1              ((X(i)-WL(k-1))/(WL(k)-WL(k-1)))*PDF(k)
             CDFx(i)=((WL(k)-X(i))/(WL(k)-WL(k-1)))*CDF(k-1)+
     1              ((X(i)-WL(k-1))/(WL(k)-WL(k-1)))*CDF(k)
             goto 50
           else
             PDFx(i)=PDF(k)
             CDFx(i)=CDF(k)
             goto 50
           Endif
         Endif
 32    continue
 50    Call PCDF(i,j,Ext,X,MuxN,SigmaxN,PDFx,CDFx,NR)
       Close (Unit=70)
       return
       End
```

```
C       ################################################################
C
        Subroutine User2(i,j,Ext,X,MuxN,SigmaxN,NR)
C
C       ################################################################
C
C       Returns the value of the probability density function, PDFx,
C       & the value of the cumulative distribution function, CDFx, of
C       the user-defined distribution of extreme water levels, for a
C       given value of the extreme water level, X. It uses the data file
C       extwldat.dad which contains the tabulated values. It uses these
C       values to return the mean, MuxN, & the standard deviation,
C       SigmaxN, of the equivalent Normal distribution through
C       subroutine PCDF.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       M - Number of points of the user-defined distribution tabulated
C           in file extwldat.dad
C       ExtWL - Value tabulated in file extwldat.dad of the extreme
C               water level of the user-defined distribution
C       PDF - Value tabulated in file extwldat.dad of the probability
C             density function of the user-defined distribution
C       CDF - Value tabulated in file extwldat.dad of the cumulative
C             distribution function of the user-defined distribution
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       PDFx - Value of the probability density function of the
C              user-defined distribution at X
C       CDFx - Value of the cumulative distribution function of the
C              user-defined distribution at X
C       k - Auxiliary variable
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
        Integer*4 i,j,k,M,Q,L
        Parameter (M=17)
        Parameter (Q=16)
        Parameter (L=15)
        Character*3 Ext(L)
        Real*8 ExtWL(M),PDF(M),CDF(M),X(L),PDFx(L),CDFx(L),MuxN(L),
     1        SigmaxN(L),NR(Q,L)
        Open(Unit=75, File='extwldat.dad', Status='Old')
        Do 10 k=1,M
          Read(75,*) ExtWL(k),PDF(k),CDF(k)
 10     continue
        Do 32 k=1,M
          If (ExtWL(k).GE.X(i)) then
            If (ExtWL(k).GT.X(i)) then
              PDFx(i)=((ExtWL(k)-X(i))/(ExtWL(k)-ExtWL(k-1)))*PDF(k-1)+
     1                ((X(i)-ExtWL(k-1))/(ExtWL(k)-ExtWL(k-1)))*PDF(k)
              CDFx(i)=((ExtWL(k)-X(i))/(ExtWL(k)-ExtWL(k-1)))*CDF(k-1)+
     1                ((X(i)-ExtWL(k-1))/(ExtWL(k)-ExtWL(k-1)))*CDF(k)
              goto 50
            else
              PDFx(i)=PDF(k)
              CDFx(i)=CDF(k)
              goto 50
            Endif
          Endif
 32     continue
 50     Call PCDF(i,j,Ext,X,MuxN,SigmaxN,PDFx,CDFx,NR)
        Close (Unit=75)
        return
        End
```

```
C       ################################################################
C
        Subroutine User3(i,j,Ext,X,MuxN,SigmaxN,NR)
C
C       ################################################################
C
C       Returns the value of the probability density function, PDFx,
C       & the value of the cumulative distribution function, CDFx, of
C       the user-defined distribution of tide levels, for a given value
C       of the tide level, X. It uses the data file tide.dad which
C       contains the tabulated values. It uses these values to return
C       the mean, MuxN, & the standard deviation, SigmaxN, of the
C       equivalent Normal distribution through subroutine PCDF.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       X - Variable
C       Ext - Abbreviation of the name of the variable
C       NR - Power to which each distribution is raised for each
C            combination of actions
C       i - Number of the variable
C       j - Number of the combination of actions
C
C       MODELING VARIABLES:
C       M - Number of points of the user-defined distribution tabulated
C           in file tide.dad
C       Tide - Value tabulated in file tide.dad of the tide level of
C              the user-defined distribution
C       PDF - Value tabulated in file tide.dad of the probability
C             density function of the user-defined distribution
C       CDF - Value tabulated in file tide.dad of the cumulative
C             distribution function of the user-defined distribution
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       PDFx - Value of the probability density function of the
C              user-defined distribution at X
C       CDFx - Value of the cumulative distribution function of the
C              user-defined distribution at X
C       k - Auxiliary variable
C
C       OUTPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       SigmaxN - Standard deviation of the equivalent Normal
C                 distribution of X
C
C       ################################################################
C
        Integer*4 i,j,k,M,Q,L
        Parameter (M=122)
        Parameter (Q=16)
        Parameter (L=15)
        Character*3 Ext(L)
        Real*8 Tide(M),PDF(M),CDF(M),X(L),PDFx(L),CDFx(L),MuxN(L),
     1         SigmaxN(L),NR(Q,L)
        Open(Unit=80, File='tide.dad', Status='Old')
        Do 10 k=1,M
          Read(80,*) Tide(k),PDF(k),CDF(k)
 10     continue
        Do 32 k=1,M
          If (Tide(k).GE.X(i)) then
            If (Tide(k).GT.X(i)) then
              PDFx(i)=((Tide(k)-X(i))/(Tide(k)-Tide(k-1)))*PDF(k-1)+
     1                ((X(i)-Tide(k-1))/(Tide(k)-Tide(k-1)))*PDF(k)
              CDFx(i)=((Tide(k)-X(i))/(Tide(k)-Tide(k-1)))*CDF(k-1)+
     1                ((X(i)-Tide(k-1))/(Tide(k)-Tide(k-1)))*CDF(k)
              goto 50
            else
              PDFx(i)=PDF(k)
              CDFx(i)=CDF(k)
              goto 50
            Endif
          Endif
 32     continue
 50     Call PCDF(i,j,Ext,X,MuxN,SigmaxN,PDFx,CDFx,NR)
        Close (Unit=80)
        return
        End
```

```
C        ################################################################
C
         Subroutine Algebra(N,Covx,D,V)
C
C        ################################################################
C
C        Returns the eigenvalues, D, & the eigenvectores, V, of the
C        matrix Covx. It uses subroutines Tred2 & TQLI to calculate these
C        values.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        Covx - Covariance matrix
C        N - Number of variables
C
C        MODELING VARIABLES:
C        E - Off-diagonal elements of the orthogonal matrix, Q, obtained
C            from Covx
C        i, k, NP - Auxiliary variables
C
C        OUTPUT VARIABLES:
C        D - Eigenvalues
C        V - Eigenvectors
C
C        ################################################################
C
         Integer*4 i,k,N,NP
         Parameter (NP=15)
         Real*8  D(NP),E(NP),V(NP,NP),Covx(NP,NP)
         Call Tred2(Covx,N,NP,D,E)
         Do 128 k=1,N
           Do 129 i=1,N
             V(i,k)=Covx(i,k)
 129       continue
 128     continue
         Call TQLI(D,E,N,NP,V)
         return
         End




C        ################################################################
C
         Function Zbrent(j,i,Aux1,Aux2,a1,a2,Tol,VarDis)
C
C        ################################################################
C
C        Using Brent`s method, it returns the root of a function (e.g.
C        GG, TT, MM, BB) known to lie between a1 & a2. The root, returned
C        as Zbrent, is refined until its accuracy is within Tol.
C        ----------
C        For more details, see Press et al (1992), pp.354.
C
C        ################################################################
C
         Integer*4 j,i,L,Q,Itmax,Iter
         Parameter (L=15)
         Parameter (Q=16)
         Real*8 Aux1,Aux2,VarDis(Q,L),A,B,C,D,E,a1,a2,FA,FB,
     1        FC,Eps,Tol,Tol1,S,P,T,R,XM,GG,TT,MM,BB,Zbrent
         Parameter (Itmax=100,Eps=3.0E-8)
         External GG
         External TT
         External MM
         External BB
         A=a1
         B=a2
         If (VarDis(j,i).EQ.7) then
           FA=GG(Aux1,Aux2,A)
           FB=GG(Aux1,Aux2,B)
         elseif (VarDis(j,i).EQ.10) then
           FA=TT(i,j,Aux1,Aux2,A)
           FB=TT(i,j,Aux1,Aux2,B)
         elseif (VarDis(j,i).EQ.5) then
           FA=MM(i,j,Aux1,A)
           FB=MM(i,j,Aux1,B)
         elseif (VarDis(j,i).EQ.6) then
           FA=BB(i,j,Aux1,A)
           FB=BB(i,j,Aux1,B)
         Endif
         If (((FA.GT.0.).AND.(FB.GT.0.)).OR.((FA.LT.0.).AND.
     1    (FB.LT.0.))) PAUSE 'Root must be bracketed for Zbrent.'
```

C7-88

```
        C=B
        FC=FB
        Do 11 Iter=1,Itmax
          If (((FB.GT.0.).AND.(FC.GT.0.)).OR.((FB.LT.0.).AND.
   1        (FC.LT.0.))) then
            C=A
            FC=FA
            D=B-A
            E=D
          Endif
          If ((ABS(FC)).LT.(ABS(FB))) then
            A=B
            B=C
            C=A
            FA=FB
            FB=FC
            FC=FA
          Endif
          Tol1=2.*Eps*ABS(B)+0.5*Tol
          XM=0.5*(C-B)
          If (((ABS(XM)).LE.Tol1).OR.(FB.EQ.0.)) then
            Zbrent=B
            return
          Endif
          If (((ABS(E)).GE.Tol1).AND.((ABS(FA)).GT.(ABS(FB)))) then
            S=FB/FA
            If (A.EQ.C) then
              P=2.*XM*S
              T=1.-S
            else
              T=FA/FC
              R=FB/FC
              P=S*(2.*XM*T*(T-R)-(B-A)*(R-1.))
              T=(T-1.)*(R-1.)*(S-1.)
            Endif
            If (P.GT.0.) T=-T
            P=ABS(P)
            If ((2.*P).LT.(DMIN1((3.*XM*T-ABS(Tol1*T)),(ABS(E*T)))))
   1          then
              E=D
              D=P/T
            else
              D=XM
              E=D
            Endif
          else
            D=XM
            E=D
          Endif
          A=B
          FA=FB
          If ((ABS(D)).GT.Tol1) then
            B=B+D
          else
            B=B+SIGN(Tol1,XM)
          Endif
          If (VarDis(j,i).EQ.7) then
            FB=GG(Aux1,Aux2,B)
          elseif (VarDis(j,i).EQ.10) then
            FB=TT(i,j,Aux1,Aux2,B)
          elseif (VarDis(j,i).EQ.5) then
            FB=MM(i,j,Aux1,B)
          elseif (VarDis(j,i).EQ.6) then
            FB=BB(i,j,Aux1,B)
          Endif
 11     continue
        PAUSE 'Zbrent exceeding maximum iterations.'
        Zbrent=B
        return
        End




C       ##############################################################
C
        Function Gammln(XX)
C
C       ##############################################################
C
C       Returns the value ln(Gamma(XX)) for XX>0. Full accuracy is
C       obtained for XX>1. For 0<XX<1, the reflection formula can be
C       used first.
C       ----------
```

```
C        For more details, see Press et al (1992), pp.207.
C
C        ################################################################
C
         Integer*4 k
         Real*8 XX,COF(6),STP,X,TMP,SER,Y,Gammln
         Save COF,STP
         Data   COF,STP/76.18009172947146D0,-86.50532032941677D0,
      1         24.01409824083091D0,-1.231739572450155D0,
      1         0.1208650973866179D-2,-0.5395239384953D-5,
      1         2.5066282746310005D0/
         X=XX
         Y=X
         TMP=X+5.5D0
         TMP=(X+0.5D0)*LOG(TMP)-TMP
         SER=1.000000000190015D0
         Do 11 k=1,6
           Y=Y+1.0D0
           SER=SER+COF(k)/Y
 11      continue
         Gammln=TMP+LOG(STP*SER/X)
         return
         End




C        ################################################################
C
         Subroutine Truncation(i,j,Ext,X,MuxN,SigmaxN,Trunc,Xo,CDFx,
      1                       CDFx0,PDFx,NR)
C
C        ################################################################
C
C        Returns the value of the probability density function, PDFx, at
C        X & the value of the cumulative distribution function, CDFx, at
C        X of a distribution truncated at X=Xo & with a CDFx=CDFx0 at
C        X=X0. It also returns the mean, MuxN, & the standard deviation,
C        SigmaxN, of the equivalent Normal distribution through
C        subroutine PCDF.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        X - Variable
C        Ext - Abbreviation of the name of the variable
C        Trunc - Type of truncation
C        Xo - Point of truncation (if the distribution is truncated)
C        NR - Power to which each distribution is raised for each
C             combination of actions
C        CDFx0 - Value of the cumulative distribution function at X=X0
C        i - Number of the variable
C        j - Number of the combination of actions
C
C        INPUT/OUTPUT VARIABLES:
C        PDFx - Value of the probability density function at X
C        CDFx - Value of the cumulative distribution function at X
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C
C        OUTPUT VARIABLES:
C        MuxN - Mean of the equivalent Normal distribution of X
C        SigmaxN - Standard deviation of the equivalent Normal
C                  distribution of X
C
C        ################################################################
C
         Integer*4 i,j,L,Q
         Parameter (L=15)
         Parameter (Q=16)
         Character*3 Ext(L)
         Integer*4 Trunc(Q,L)
         Real*8 X(L),MuxN(L),SigmaxN(L),CDFx(L),CDFx0(L),PDFx(L),Xo(Q,L),
      1         NR(Q,L)
         If (Trunc(j,i).NE.0) then
           If (Trunc(j,i).EQ.1) then
             If (X(i).GT.Xo(j,i)) then
               PDFx(i)=0.
               CDFx(i)=1.
             else
               PDFx(i)=PDFx(i)/CDFx0(i)
```

```
          CDFx(i)=CDFx(i)/CDFx0(i)
        Endif
      else
        If (X(i).LT.Xo(j,i)) then
          PDFx(i)=0.
          CDFx(i)=0.
        else
          PDFx(i)=PDFx(i)/(1.-CDFx0(i))
          CDFx(i)=(CDFx(i)-CDFx0(i))/(1.-CDFx0(i))
        Endif
      Endif
    Endif
    Call PCDF(i,j,Ext,X,MuxN,SigmaxN,PDFx,CDFx,NR)
    return
    End




C       ################################################################
C
        Function Gammp(XX,X)
C
C       ################################################################
C
C       Returns the incomplete Gamma function, gamma(XX,X)/Gamma(XX).
C       ----------
C       For more details, see Press et al (1992), pp.211.
C
C       ################################################################
C
        Real*8 XX,X,Gamser,Gln,Gammcf,Gammp
        If ((X.LT.0.).OR.(XX.LE.0.)) PAUSE 'Bad arguments in Gammp'
        If (X.LT.(XX+1.)) then
          Call GSER(Gamser,XX,X,Gln)
          Gammp=Gamser
        else
          Call GCF(Gammcf,XX,X,Gln)
          Gammp=1.-Gammcf
        Endif
        return
        End




C       ################################################################
C
        Function Beta(XX,YY)
C
C       ################################################################
C
C       Returns the value of the Beta function, B(XX,YY).
C       ----------
C       For more details, see Press et al (1992), pp.209.
C
C       ################################################################
C
        Real*8 XX,YY,XY,Beta,Gammln
        External Gammln
        XY=XX+YY
        Beta=EXP(Gammln(XX)+Gammln(YY)-Gammln(XY))
        return
        End




C       ################################################################
C
        Function Betai(XX,YY,U)
C
C       ################################################################
C
C       Returns the incomplete Beta function, Bu(XX,YY)/B(XX,YY).
C       ----------
C       For more details, see Press et al (1992), pp.220.
C
C       ################################################################
C
        Real*8 XX,YY,XY,U,BT,Gammln,Betacf,Betai
        External Gammln
        External Betacf
        If ((U.LT.0.).OR.(U.GT.1.)) PAUSE 'Bad argument U in Betai'
        If ((U.EQ.0.).OR.(U.EQ.1.)) then
          BT=0.
```

```
          else
            XY=XX+YY
            BT=EXP(Gammln(XY)-Gammln(XX)-Gammln(YY)+XX
     1         *LOG(U)+YY*LOG(1.-U))
          Endif
          If (U.LT.((XX+1.)/(XX+YY+2.))) then
            Betai=BT*Betacf(XX,YY,U)/XX
            return
          else
            XY=1.-U
            Betai=1.-BT*Betacf(YY,XX,XY)/YY
            return
          Endif
          End




C        ###################################################################
C
          Subroutine Tred2(Covx,N,NP,D,E)
C
C        ###################################################################
C
C        Householder reduction of a real symmetric N by N matrix Covx,
C        stored in an NP by NP physical array. On output, Covx is
C        replaced by the orthogonal matrix Q effecting the
C        transformation. D returns the diagonal elements of the
C        tridiagonal matrix, & E the off-diagonal elements, with
C        E(1)=0.
C        ----------
C        For more details, see Press et al (1992), pp.467.
C
C        ###################################################################
C
          Integer*4 i,t,K,L,N,NP
          Real*8  H,Scale,F,G,HH,Covx(NP,NP),D(NP),E(NP)
          Do 18 i=N,2,-1
            L=i-1
            H=0.
            Scale=0.
            If (L.GT.1) then
              Do 11 K=1,L
                Scale=Scale+DABS(Covx(i,K))
 11           continue
              If (Scale.EQ.0.) then
                E(i)=Covx(i,L)
              else
                Do 12 K=1,L
                  Covx(i,K)=Covx(i,K)/Scale
                  H=H+(Covx(i,K)**2)
 12             continue
                F=Covx(i,L)
                G=-DSIGN(DSQRT(H),F)
                E(i)=Scale*G
                H=H-F*G
                Covx(i,L)=F-G
                F=0.
                Do 15 t=1,L
                  Covx(t,i)=Covx(i,t)/H
                  G=0.
                  Do 13 K=1,t
                    G=G+Covx(t,K)*Covx(i,K)
 13               continue
                  Do 14 K=t+1,L
                    G=G+Covx(K,t)*Covx(i,K)
 14               continue
                  E(t)=G/H
                  F=F+E(t)*Covx(i,t)
 15             continue
                HH=F/(H+H)
                Do 17 t=1,L
                  F=Covx(i,t)
                  G=E(t)-HH*F
                  E(t)=G
                  Do 16 K=1,t
                    Covx(t,K)=Covx(t,K)-F*E(K)-G*Covx(i,K)
 16               continue
 17             continue
              Endif
            else
              E(i)=Covx(i,L)
            Endif
            D(i)=H
```

```
18      continue
        D(1)=0.
        E(1)=0.
        Do 24 i=1,N
          L=i-1
          If (D(i).NE.0.) then
            Do 22 t=1,L
              G=0.
              Do 19 K=1,L
                G=G+Covx(i,K)*Covx(K,t)
19            continue
              Do 21 K=1,L
                Covx(K,t)=Covx(K,t)-G*Covx(K,i)
21            continue
22          continue
          Endif
          D(i)=Covx(i,i)
          Covx(i,i)=1.
          Do 23 t=1,L
            Covx(i,t)=0.
            Covx(t,i)=0.
23        continue
24      continue
        return
        End




C       ################################################################
C
        Subroutine TQLI(D,E,N,NP,V)
C
C       ################################################################
C
C       QL algorithm with implicit shifts, to determine the eingenvalues
C       & eigenvectors of a real, symmetric, tridiagonal matrix, or of
C       a real, symmetric matrix previously reduced by TRED2. D is a
C       vector of length NP. On input, its first N elements are the
C       diagonal elements of the tridiagonal matrix. On output, it
C       returns the eingenvalues. The vector E inputs the subdiagonal
C       elements of the tridiagonal matrix, with E(1) arbitrary. On
C       output E is destroyed. The matrix V(N by N matrix stored in an
C       NP by NP array) is input as the identity matrix.If the
C       eigenvectors of a matrix that has been reduced by TRED2 are
C       required, then V is input as the matrix output by TRED2. In
C       either case, the Kth column of V returns the normalized
C       eigenvector corresponding to D(K).
C       ----------
C       For more details, see Press et al (1992), pp.473.
C
C       ################################################################
C
        Integer*4 i,K,L,M,N,NP,Iter
        Real*8  B,C,DD,F,G,P,R,S,D(NP),E(NP),V(NP,NP),Minha,Pythag
        External Pythag
        Open(Unit=100, File='lixo.dat', Status='unknown')
        Do 11 i=2,N
          E(i-1)=E(i)
11      continue
        E(N)=0.
        Do 15 L=1,N
          Iter=0
1         Do 12 M=L,N-1
            DD=DABS(D(M))+DABS(D(M+1))
            Write(100,*)
            If (DABS(E(M))+DD.EQ.DD) goto 2
12        continue
          M=N
2         If (M.NE.L) then
            If (Iter.EQ.30) Pause 'Too many iterations in TQLI'
            Iter=Iter+1
            G=(D(L+1)-D(L))/(2.*E(L))
            Write(100,*)
            Minha=1.0
            R=Pythag(G,Minha)
            G=D(M)-D(L)+E(L)/(G+DSIGN(R,G))
            S=1.
            C=1.
            P=0.
            Do 14 i=M-1,L,-1
              F=S*E(i)
              B=C*E(i)
              R=Pythag(F,G)
```

```
              E(i+1)=R
              If (R.EQ.0.) then
                D(i+1)=D(i+1)-P
                E(M)=0.
                goto 1
              Endif
              S=F/R
              C=G/R
              G=D(i+1)-P
              R=(D(i)-G)*S+2.*C*B
              P=S*R
              D(i+1)=G+P
              G=C*R-B
              Do 13 K=1,N
                F=V(K,i+1)
                V(K,i+1)=S*V(K,i)+C*F
                V(K,i)=C*V(K,i)-S*F
 13           continue
 14         continue
            D(L)=D(L)-P
            E(L)=G
            E(M)=0.
            goto 1
          Endif
 15     continue
        return
        End




C       ################################################################
C
        Function GG(Aux1,Aux2,AB)
C
C       ################################################################
C
C       GG is a function for which the root is to be calculated using
C       subroutine Zbrent. The root of GG gives the value of the
C       parameter Zeta of a Frechet distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Aux1, Aux2, AB - Auxiliary variables
C
C       MODELING VARIABLES:
C       Gammln - External function
C       AB1, AB2 - Auxiliary variables
C
C       ################################################################
C
        Real*8 Aux1,Aux2,AB,AB1,AB2,Gammln,GG
        External Gammln
        AB1=1.-(1./AB)
        AB2=1.-(2./AB)
        GG=(Aux2**2)+(Aux1**2)-((Aux1**2)*
     1    (EXP(Gammln(AB2))))/((EXP(Gammln(AB1)))**2)
        return
        End




C       ################################################################
C
        Function TT(i,j,Aux1,Aux2,AB)
C
C       ################################################################
C
C       TT is a function for which the root is to be calculated using
C       subroutine Zbrent. The root of TT gives the value of the
C       parameter Eta of a Weibull distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Zeta - Parameter of a Weibull distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C       Aux1, Aux2, AB - Auxiliary variables
C
C       MODELING VARIABLES:
C       Gammln - External function
C       L - Maximum number of variables allowed by the program
```

```
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       AB1, AB2 - Auxiliary variables
C       Lamda, Eta, x1, x2 - Variables mentioned in the Common statement
C                          but not used here
C
C       ################################################################
C
        Integer*4 i,j,Q,L
        Parameter (Q=16)
        Parameter (L=15)
        Real*8 Aux1,Aux2,Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L),
     1         Gammln,TT,AB,AB1,AB2
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
        External Gammln
        AB1=1.+(1./AB)
        AB2=1.+(2./AB)
        TT=Aux2-((Aux1-Zeta(j,i))/EXP(Gammln(AB1)))*
     1     SQRT((EXP(Gammln(AB2)))-((EXP(Gammln(AB1)))**2))
        return
        End




C       ################################################################
C
        Function MM(i,j,Aux1,AB)
C
C       ################################################################
C
C       MM is a function for which the root is to be calculated using
C       subroutine Zbrent. The root of MM gives a value used by
C       subroutine InvGamma to calculate the inverse function of a Gamma
C       distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Zeta - Parameter of a Gamma distribution
C       i - Number of the variable
C       j - Number of the combination of actions
C       Aux1, AB - Auxiliary variables
C
C       MODELING VARIABLES:
C       Gammp - External function
C       L - Maximum number of variables allowed by the program
C       Q - Maximum number of combinations of actions allowed by the
C           program
C       Aux2 - Auxiliary variable
C       Lamda, Eta, x1, x2 - Variables mentioned in the Common statement
C                          but not used here
C
C       ################################################################
C
        Integer*4 j,i,L,Q
        Parameter (Q=16)
        Parameter (L=15)
        Real*8 Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L),
     1         AB,Aux1,Aux2,Gammp,MM
        Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
        External Gammp
        Aux2=Zeta(j,i)
        MM=Aux1-Gammp(Aux2,AB)
        return
        End




C       ################################################################
C
        Function BB(i,j,Aux1,AB)
C
C       ################################################################
C
C       BB is a function for which the root is to be calculated using
C       subroutine Zbrent. The root of BB gives a value used by
C       subroutine InvBeta to calculate the inverse function of a Beta
C       distribution.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       Zeta, Lamda - Parameters of a Beta distribution
```

```
C        i - Number of the variable
C        j - Number of the combination of actions
C        Aux1, AB - Auxiliary variables
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        Betai - External function
C        Aux2, Aux3 - Auxiliary variables
C        Eta, x1, x2 - Variables mentioned in the Common statement but
C                      not used here
C
C        ###############################################################
C
         Integer*4 j,i,L,Q
         Parameter (Q=16)
         Parameter (L=15)
         Real*8 Zeta(Q,L),Lamda(Q,L),Eta(Q,L),x1(Q,L),x2(Q,L),
     1         AB,Aux1,Betai,BB,Aux2,Aux3
         Common/BLOCK8/Zeta,Lamda,Eta,x1,x2
         External Betai
         Aux2=Zeta(j,i)
         Aux3=Lamda(j,i)
         BB=Aux1-Betai(Aux2,Aux3,AB)
         return
         End




C        ###############################################################
C
         Subroutine PCDF(i,j,Ext,X,MuxN,SigmaxN,PDFx,CDFx,NR)
C
C        ###############################################################
C
C        Returns the mean, MuxN, & the standard deviation, SigmaxN,
C        of the equivalent Normal distribution according to the Rackwitz
C        & Fiessler (1978) approximation for non-Normal variables. It
C        writes these values in the file results.dat through subroutine
C        NorWrite.
C
C        ###############################################################
C
C        INPUT VARIABLES:
C        X - Variable
C        Ext - Abbreviation of the name of the variable
C        NR - Power to which each distribution is raised for each
C             combination of actions
C        PDFx - Value of the probability density function at X
C        CDFx - Value of the cumulative distribution function at X
C        i - Number of the variable
C        j - Number of the combination of actions
C
C        MODELING VARIABLES:
C        SNX - Standard Normal variable
C        C - Value of the exceedance cumulative distribution function of
C            the standard Normal distribution which corresponds to SNX
C        C1, C2 - Value of the cumulative distribution function at X
C        L - Maximum number of variables allowed by the program
C        Q - Maximum number of combinations of actions allowed by the
C            program
C        Pi - 3.14159...
C
C        OUTPUT VARIABLES:
C        MuxN - Mean of the equivalent Normal distribution of X
C        SigmaxN - Standard deviation of the equivalent Normal
C                  distribution of X
C
C        ###############################################################
C
         Integer*4 i,j,Q,L
         Parameter (L=15)
         Parameter (Q=16)
         Character*3 Ext(L)
         Real*8 Pi,X(L),CDFx(L),PDFx(L),MuxN(L),SigmaxN(L),C,C1,C2,SNX,
     1         NR(Q,L)
         Pi=4.*ATAN(1.)
         C1=CDFx(i)**NR(j,i)
         C2=CDFx(i)**(NR(j,i)-1)
         C=1.-C1
         Call InvNormal(C,SNX)
         SigmaxN(i)=(EXP(-(SNX**2)/2.))/(NR(j,i)*C2*PDFx(i)*SQRT(2.*Pi))
```

```
          MuxN(i)=X(i)-SigmaxN(i)*SNX
          Call NorWrite(i,Ext,MuxN,SigmaxN)
          return
          End




C        ################################################################
C
          Subroutine GSER(Gamser,XX,X,Gln)
C
C        ################################################################
C
C        Returns the incomplete Gamma function, gamma(XX,X)/Gamma(XX),
C        evaluated by its series representation as Gamser. Also returns
C        ln(Gamma(XX)) as Gln.
C        ----------
C        For more details, see Press et al (1992), pp.212.
C
C        ################################################################
C
          Integer*4 Itmax,N
          Real*8 Eps
          Parameter (Itmax=100,Eps=3.0E-7)
          Real*8 XX,X,Gamser,Gln,AP,SUM,DEL,Gammln
          External Gammln
          Gln=Gammln(XX)
          If (X.LE.0.) then
            If (X.LT.0.) Pause 'x < 0 in GSER'
            Gamser=0.
            return
          Endif
          AP=XX
          SUM=1./XX
          DEL=SUM
          Do 11 N=1,Itmax
            AP=AP+1.
            DEL=DEL*X/AP
            SUM=SUM+DEL
            If ((ABS(DEL)).LT.(ABS(SUM)*Eps)) goto 1
 11       continue
          PAUSE 'XX too large; Itmax too small in GSER'
 1        Gamser=SUM*EXP(-X+XX*LOG(X)-Gln)
          return
          End




C        ################################################################
C
          Subroutine GCF(Gammcf,XX,X,Gln)
C
C        ################################################################
C
C        Returns the incomplete Gamma function, gamma(XX,X)/Gamma(XX),
C        evaluated by its continued fraction representation as Gammcf.
C        Also returns ln(Gamma(XX)) as Gln.
C        ----------
C        For more details, see Press et al (1992), pp.212.
C
C        ################################################################
C
          Integer*4 Itmax,i
          Real*8 Eps,Fpmin
          Parameter (Itmax=100,Eps=3.0E-7,Fpmin=1.0E-30)
          Real*8 XX,X,Gammcf,Gln,an,b,c,d,del,h,Gammln
          External Gammln
          Gln=Gammln(XX)
          b=X+1.-XX
          c=1./Fpmin
          d=1./b
          h=d
          Do 11 i=1,Itmax
            an=-i*(i-XX)
            b=b+2.
            d=an*d+b
            If (ABS(d).LT.Fpmin) d=Fpmin
            c=b+an/c
            If (ABS(c).LT.Fpmin) c=Fpmin
            d=1./d
            del=d*c
            h=h*del
            If (ABS(del-1.).LT.Eps) goto 1
```

```
 11       continue
          PAUSE 'XX too large; Itmax too small in GCF'
  1       Gammcf=EXP(-X+XX*LOG(X)-Gln)*h
          return
          End




C         ################################################################
C
          Function Betacf(ZZ,WW,VV)
C
C         ################################################################
C
C         Returns continued fraction for the incomplete Beta function by
C         modified Lentz's method.
C         ----------
C         For more details, see Press et al (1992), pp.221.
C
C         ################################################################
C
          Integer*4 Maxit,M,M2
          Real*8 Eps,Fpmin
          Parameter (Maxit=100,Eps=3.0E-7,Fpmin=1.0E-30)
          Real*8 ZZ,WW,VV,aa,c,d,del,h,qab,qam,qap,Betacf
          qab=ZZ+WW
          qap=ZZ+1.
          qam=ZZ-1.
          c=1.
          d=1.-qab*VV/qap
          If (ABS(d).LT.Fpmin) d=Fpmin
          d=1./d
          h=d
          Do 11 M=1,Maxit
            M2=2*M
            aa=M*(WW-M)*VV/((qam+M2)*(ZZ+M2))
            d=1.+aa*d
            If (ABS(d).LT.Fpmin) d=Fpmin
            c=1.+aa/c
            If (ABS(c).LT.Fpmin) c=Fpmin
            d=1./d
            h=h*d*c
            aa=-(ZZ+M)*(qab+M)*VV/((ZZ+M2)*(qap+M2))
            d=1.+aa*d
            If (ABS(d).LT.Fpmin) d=Fpmin
            c=1.+aa/c
            If (ABS(c).LT.Fpmin) c=Fpmin
            d=1./d
            del=d*c
            h=h*del
            If ((ABS(del-1.)).LT.Eps) goto 1
 11       continue
          PAUSE 'ZZ or WW too big, or Maxit too small in Betacf'
  1       Betacf=h
          return
          End




C         ################################################################
C
          Function Pythag(a,b)
C
C         ################################################################
C
C         Computes SQRT(a*a+b*b) without destructive underflow or
C         overflow.
C         ----------
C         For more details, see Press et al (1992), pp.62.
C
C         ################################################################
C
          Real*8 a,b,aa,ab,Pythag
          aa=DABS(a)
          ab=DABS(b)
          If (aa.GT.ab) then
            Pythag=aa*DSQRT(1.+(ab/aa)**2)
          else
            If (ab.EQ.0.) then
              Pythag=0.
            else
              Pythag=ab*DSQRT(1.+(aa/ab)**2)
            Endif
```

```
            Endif
            return
            End



C         ################################################################
C
            Subroutine HandR(FDer,N,X,OBJF,OBJGRD,Par)
C
C         ################################################################
C
C         Returns the failure function, OBJF, & its first partial
C         derivatives, OBJGRD, for the failure mode of overtopping using
C         the H&R model.
C
C         ################################################################
C
C         INPUT VARIABLES:
C         It - Iteration number
C         FDer - Method of calculation of the first partial derivatives of
C                the failure function overtopping
C         N - Number of variables
C         DSWL - Definition of the SWL
C         X - Variables of the failure mode
C         Par - For Mode=1, it is the prescribed seawall crest level; for
C               Mode=2, it is the starting seawall crest level (from which
C               the program iterates to find the required value of the
C               seawall crest level)
C         C1 - Parameter used in the H&R model to calculate C; it depends
C               on the confidence value assigned to the maximum run-up
C         TR - Allowable discharge for each FORM calculation
C         k0 - Number of the FORM calculation
C
C         MODELING VARIABLES:
C         L - Maximum number of variables allowed by the program
C         Tp - Wave period of peak spectral density
C         Hs - Significant height of the incident waves
C         A - Model coefficient
C         B - Model coefficient
C         SWL - Still-water-level
C         Tide - Tide level
C         Surge - Surge
C         TangAlpha - Tangent of the seawall slope
C         r - Roughness of the seawall's front slope
C         eB - Model parameter
C         Ep - Surf similarity parameter
C         C - Parameter in the H&R model
C         CL - Seawall crest level
C         TL - Seawall toe level
C         Rc - Seawall freeboard
C         Rstar - Dimensionless freeboard
C         Q - Mean overtopping discharge over unit length of seawall
C         Pi - 3.14159...
C         AuxRstar - Variable mentioned in the Common statements but
C                     not used here
C
C         OUTPUT VARIABLES:
C         OBJF - Failure function
C         OBJGRD - First partial derivatives of OBJF
C
C         ################################################################
C
            Integer*4 N,k0,L,FDer,It,AuxRstar,DSWL
            Parameter (L=15)
            Real*8 X(N),Par,OBJF,OBJGRD(N),TR(L)
            Real*8 Tp,Hs,A,B,C,C1,SWL,Tide,Surge,TangAlpha,r,eB,Q,CL,Rc,
      1           Rstar,Ep,Pi,TL
            Common/BLOCK7/TR
            Common/BLOCK9/k0,It,AuxRstar
            Common/BLOCK10/C1
            Common/BLOCK11/DSWL,TL

            Pi=4.*ATAN(1.)
            Tp=X(1)
            Hs=X(2)
            A=X(3)
            B=X(4)
            If (DSWL.EQ.1) then
              SWL=X(5)
              TangAlpha=X(6)
              r=X(7)
              eB=X(8)
```

```
            else
              Tide=X(5)
              Surge=X(6)
              SWL=Tide+Surge
              If (SWL.LT.TL) then
                Write(*,789) SWL,TL
                Write(50,789) SWL,TL
  789           Format(// 3X,'ERROR: SWL Below Toe Level !'
      1                 / 10X,'SWL=',E17.10,3X,'TL=',E17.10 /)
                STOP
              Endif
              TangAlpha=X(7)
              r=X(8)
              eB=X(9)
            Endif

C       ================================================================
C       Definition of the seawall crest level, CL. CL has the value of
C       the design parameter (in Mode 1: if DSWL=1, X=8; if DSWL=2, X=9)
C       or it is the new variable (in Mode 2: if DSWL=1, X=9; if DSWL=2,
C       X=10).
C       ================================================================

            If ((N.EQ.9.AND.DSWL.EQ.1).OR.(N.EQ.10.AND.DSWL.EQ.2)) then
              CL=X(N)
            else
              CL=Par
            Endif
            If (CL.LT.SWL) then
              Write(*,78) SWL,CL
              Write(50,78) SWL,CL
   78         Format(// 3X,'ERROR: SWL Above Crest Level !'
      1               / 10X,'SWL=',E17.10,3X,'CL=',E17.10 /)
              STOP
            Endif

C       ================================================================
C       Calculation of C.
C       ================================================================
C
C       ----------
C       Calculation of the surf similarity parameter, Ep.
C       ----------
            Ep=TangAlpha/SQRT((2.*Pi*Hs)/(9.81*(Tp**2)))
            If (Ep.LE.2.) then
              C=C1*1.35*Ep
            else
              C=C1*(3.-0.15*Ep)
            Endif

C       ================================================================
C       Calculation of the dimensionless freeboard, Rstar.
C       ================================================================
            Rc=CL-SWL
            Rstar=Rc/(r*C*Hs)

C       ================================================================
C       Calculation, for Mode=2, of the first partial derivative of the
C       failure function in relation to CL=X(N), OBJGRD(N).
C       ================================================================
            If ((N.EQ.9.AND.DSWL.EQ.1).OR.(N.EQ.10.AND.DSWL.EQ.2)) then
              If (FDer.EQ.1) then
                If ((Rstar.LT.1.).AND.(Rstar.GE.0.)) then
                  If (Ep.LE.2) then
                    OBJGRD(N)=((A*eB*B*SQRT(C1*1.35*Tp*TangAlpha)
      1                       *(9.81**(3./4.))*(Hs**(1./4.)))/
      1                       (r*((2.*Pi)**(1./4.))))*((1.-Rstar)**(eB*B-1))
                    goto 15
                  else
                    OBJGRD(N)=(A*eB*B*SQRT(9.81*Hs*C)/r)*((1-Rstar)**
      1                       (eB*B-1))
                    goto 15
                  Endif
                Endif
                If (Rstar.GE.1.) then
                  OBJGRD(N)=0.
                  goto 15
                Endif
                Write(*,8700)
                Write(50,8700)
 8700           Format(// 11X,'ERROR: Mode 2 - R* < 0 !')
                STOP
              Endif
```

```
          Endif

C         ================================================================
C         Calculation of the mean overtopping discharge over unit length
C         of seawall, Q.
C         ================================================================
          If (Rstar.GE.1.) then
            Q=0.
          Endif
          If ((Rstar.LT.1.).AND.(Rstar.GE.0.)) then
            Q=A*SQRT(9.81*((C*Hs)**3))*((1.-Rstar)**(eB*B))
          Endif
          If (Rstar.LT.0.) then
            If (Rc.LT.0.) then
              Write(*,8730)
              Write(50,8730)
 8730         Format(/ 11X,'ERROR: Rc < 0 - R* < 0 !' /)
            Endif
            If (C.LT.0.) then
              Write(*,8720)
              Write(50,8720)
 8720         Format(// 11X,'ERROR: C < 0 - R* < 0 !')
            Endif
            STOP
          Endif

C         ================================================================
C         Calculation of the failure function, OBJF, & of its first
C         partial derivatives, OBJGRD (if required).
C         ================================================================
          OBJF=TR(k0)-Q
          If (FDer.EQ.1) then
            If ((Rstar.LT.1.).AND.(Rstar.GE.0.)) then
              If (Ep.LE.2) then
                OBJGRD(1)=-(((1.5*A*SQRT(Tp*(9.81**(5./2.))*(Hs**(3./2.))*
     1                    (C1**3)*(1.35**3)*(TangAlpha**3)*((2.*Pi)**
     1                    (-3./2.)))))*((1.-Rstar)**(eB*B))+(A*eB*B*
     1                    ((1.-Rstar)**
     1                    (eB*B-1)))*((Rc*(9.81**(3./4.))*(Hs**(1./4.))*
     1                    SQRT(C1*1.35*TangAlpha))/(r*SQRT(Tp)*((2.*Pi)
     1                    **(1./4.)))))
                OBJGRD(2)=-(((3./4.)*A*(Hs**(-1./4.))*SQRT((9.81**(5./2.))
     1                    *(C1**3)*(1.35**3)*(Tp**3)*(TangAlpha**3)*
     1                    ((2.*Pi)**(-3./2.))))*((1.-Rstar)**(eB*B))+
     1                    (A*eB*B*((1.-Rstar)**(eB*B-1)))*
     1                    ((Rc*SQRT(C1*1.35*Tp*TangAlpha)*(9.81**(3./4.)))
     1                    /(2.*r*(Hs**(3./4.))*((2.*Pi)**(1./4.)))))
                OBJGRD(3)=-SQRT((9.81**(5./2.))*(Hs**(3./2.))*(C1**3)*
     1                    (1.35**3)*(TangAlpha**3)*(Tp**3)*((2.*Pi)**
     1                    (-3./2.)))*((1.-Rstar)**(eB*B))
                OBJGRD(4)=-A*eB*SQRT((9.81**(5./2.))*(Hs**(3./2.))*(C1**3)
     1                    *(1.35**3)*(Tp**3)*(TangAlpha**3)*((2.*Pi)**
     1                    (-3./2.)))*((1.-Rstar)**(eB*B))*LOG(1.-Rstar)
                If (DSWL.EQ.1) then
                  OBJGRD(5)=-((A*eB*B*SQRT(C1*1.35*Tp*TangAlpha)*
     1                      (9.81**(3./4.))*(Hs**(1./4.)))/(r*((2.*Pi)**
     1                      (1./4.))))*((1.-Rstar)**(eB*B-1))
                  OBJGRD(6)=-(((3.*A*SQRT((9.81**(5./2.))*(Hs**(3./2.))*
     1                      (C1**3)*(1.35**3)*(Tp**3)*((2.*Pi)**(-3./2.))*
     1                      TangAlpha))/2.)*((1.-Rstar)**(eB*B))+
     1                      ((A*eB*B*Rc*(Hs**(1./4.))*(9.81**(3./4.))*
     1                      SQRT(C1*1.35*Tp))/(r*((2.*Pi)**(1./4.))*
     1                      SQRT(TangAlpha)))*((1.-Rstar)**(eB*B-1)))
                  OBJGRD(7)=-A*eB*B*((1.-Rstar)**(eB*B-1))*((Rc*
     1                      SQRT(C1*1.35*Tp*TangAlpha)*(9.81**(3./4.))
     1                      *(Hs**(1./4.)))/((r**2)*((2.*Pi)**(1./4.))))
                  OBJGRD(8)=-A*B*SQRT((9.81**(5./2.))*(Hs**(3./2.))*
     1                      (C1**3)*(1.35**3)*(Tp**3)*(TangAlpha**3)*
     1                      ((2.*Pi)**(-3./2.)))*((1.-Rstar)**(eB*B))*
     1                      LOG(1.-Rstar)
                else
                  OBJGRD(5)=-((A*eB*B*SQRT(C1*1.35*Tp*TangAlpha)*
     1                      (9.81**(3./4.))*(Hs**(1./4.)))/(r*((2.*Pi)**
     1                      (1./4.))))*((1.-Rstar)**(eB*B-1))
                  OBJGRD(6)=-((A*eB*B*SQRT(C1*1.35*Tp*TangAlpha)*
     1                      (9.81**(3./4.))*(Hs**(1./4.)))/(r*((2.*Pi)**
     1                      (1./4.))))*((1.-Rstar)**(eB*B-1))
                  OBJGRD(7)=-(((3.*A*SQRT((9.81**(5./2.))*(Hs**(3./2.))*
     1                      (C1**3)*(1.35**3)*(Tp**3)*((2.*Pi)**(-3./2.))*
     1                      TangAlpha))/2.)*((1.-Rstar)**(eB*B))+
     1                      ((A*eB*B*Rc*(Hs**(1./4.))*(9.81**(3./4.))*
     1                      SQRT(C1*1.35*Tp))/(r*((2.*Pi)**(1./4.))*
     1                      SQRT(TangAlpha)))*((1.-Rstar)**(eB*B-1)))
```

```
               OBJGRD(8)=-A*eB*B*((1.-Rstar)**(eB*B-1))*((Rc*
     1                   SQRT(C1*1.35*Tp*TangAlpha)*(9.81**(3./4.))
     1                   *(Hs**(1./4.)))/((r**2)*((2.*Pi)**(1./4.))))
               OBJGRD(9)=-A*B*SQRT((9.81**(5./2.))*(Hs**(3./2.))*
     1                   (C1**3)*(1.35**3)*(Tp**3)*(TangAlpha**3)*
     1                   ((2.*Pi)**(-3./2.)))*((1.-Rstar)**(eB*B))*
     1                   LOG(1.-Rstar)
             Endif
           else
             OBJGRD(1)=((3.*A*9.81*C1*0.15*Hs*SQRT(C)*TangAlpha)/
     1                 (2.*SQRT(2.*Pi)))*((1.-Rstar)**(eB*B))+(A*eB*B)*
     1                 ((1.-Rstar)**(eB*B-1))*((Rc*0.15*TangAlpha*9.81*
     1                 C1)/(r*SQRT(2.*Pi*C)))
             OBJGRD(2)=-((3.*A*(C**(3./2.))*SQRT(9.81*Hs))/2.)*((1.-
     1                 Rstar)**(eB*B))-((3.*A*9.81*0.15*Tp*SQRT(C)*C1*
     1                 TangAlpha)/(4.*SQRT(2.*Pi)))*((1.-Rstar)**
     1                 (eB*B))-((A*eB*B*SQRT(9.81*(Hs**3)*(C**3))))*
     1                 ((1.-Rstar)**(eB*B-1))*((Rc*(C+0.15*C1*Tp*
     1                 TangAlpha*SQRT(9.81)/(2.*SQRT(2.*Pi*Hs)))))/
     1                 ((Hs**2)*r*(C**2)))
             OBJGRD(3)=-SQRT(9.81*(Hs**3)*(C**3))*((1.-Rstar)**(eB*B))
             OBJGRD(4)=-A*eB*SQRT(9.81*(Hs**3)*(C**3))*((1.-Rstar)**
     1                 (eB*B))*LOG(1.-Rstar)
             If (DSWL.EQ.1) then
               OBJGRD(5)=-A*eB*B*SQRT(9.81*Hs*C)*((1.-Rstar)**(eB*B-1))
     1                   /r
               OBJGRD(6)=((3.*A*Hs*9.81*0.15*Tp*C1*SQRT(C))/(2.*
     1                   SQRT(2.*Pi)))*((1.-Rstar)**(eB*B))+(A*eB*B)*
     1                   ((1.-Rstar)**(eB*B-1))*((Rc*0.15*Tp*9.81*C1)/
     1                   (r*SQRT(2.*Pi*C)))
               OBJGRD(7)=-A*eB*B*Rc*SQRT(9.81*Hs*C)*((1.-Rstar)**
     1                   (eB*B-1))/(r**2)
               OBJGRD(8)=-A*B*SQRT(9.81*(Hs**3)*(C**3))*((1.-Rstar)**
     1                   (eB*B))*LOG(1.-Rstar)
             else
               OBJGRD(5)=-A*eB*B*SQRT(9.81*Hs*C)*((1.-Rstar)**(eB*B-1))
     1                   /r
               OBJGRD(6)=-A*eB*B*SQRT(9.81*Hs*C)*((1.-Rstar)**(eB*B-1))
     1                   /r
               OBJGRD(7)=((3.*A*Hs*9.81*0.15*Tp*C1*SQRT(C))/(2.*
     1                   SQRT(2.*Pi)))*((1.-Rstar)**(eB*B))+(A*eB*B)*
     1                   ((1.-Rstar)**(eB*B-1))*((Rc*0.15*Tp*9.81*C1)/
     1                   (r*SQRT(2.*Pi*C)))
               OBJGRD(8)=-A*eB*B*Rc*SQRT(9.81*Hs*C)*((1.-Rstar)**
     1                   (eB*B-1))/(r**2)
               OBJGRD(9)=-A*B*SQRT(9.81*(Hs**3)*(C**3))*((1.-Rstar)**
     1                   (eB*B))*LOG(1.-Rstar)
             Endif
           Endif
         Endif
         If (Rstar.GE.1.) then
           Do 2342 i=1,N
             OBJGRD(i)=0.
2342       continue
         Endif
         If (Rstar.LT.0.) then
           Write(*,8701)
           Write(50,8701)
8701       Format(// 11X,'ERROR: R* < 0 !')
           STOP
         Endif
       Endif
15     return
       End




C      ################################################################
C
       Subroutine Owen(FDer,N,X,OBJF,OBJGRD,Par)
C
C      ################################################################
C
C      Returns the failure function, OBJF, & its first partial
C      derivatives, OBJGRD, for the failure mode of overtopping using
C      Owen's model.
C
C      ################################################################
C
C      INPUT VARIABLES:
C      FDer - Method of calculation of the first partial derivatives of
C             the failure function of overtopping
C      N - Number of variables
```

```
C        DSWL - Definition of the SWL
C        X - Variables of the failure mode
C        Par - For Mode=1, it is the prescribed seawall crest level; for
C              Mode=2, it is the starting seawall crest level (from which
C              the program iterates to find the required value of the
C              seawall crest level)
C        TR - Allowable discharge for each FORM calculation
C        k0 - Number of the FORM calculation
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        Tm - Mean zero-crossing wave period
C        Hs - Significant height of the incident waves
C        S - Incident wave steepness
C        A - Model coefficient
C        B - Model coefficient
C        SWL - Still-water-level
C        Tide - Tide level
C        Surge - Surge
C        r - Roughness of the seawall's front slope
C        eB - Model parameter
C        CL - Seawall crest level
C        TL - Seawall toe level
C        Q - Mean overtopping discharge over unit length of seawall
C        Pi - 3.14159...
C        It, AuxRstar - Variables mentioned in the Common statements but
C                       not used here
C
C        OUTPUT VARIABLES:
C        OBJF - Failure function
C        OBJGRD - First partial derivatives of OBJF
C
C        ################################################################
C
         Integer*4 N,L,k0,FDer,It,AuxRstar,DSWL
         Parameter (L=15)
         Real*8 X(N),Par,OBJF,OBJGRD(N),TR(L)
         Real*8 Pi,Tm,Hs,A,B,SWL,Tide,Surge,r,eB,S,CL,Q,TL
         Common/BLOCK7/TR
         Common/BLOCK9/k0,It,AuxRstar
         Common/BLOCK11/DSWL,TL

         Pi=4.*ATAN(1.)
         Tm=X(1)
         Hs=X(2)
         A=X(3)
         B=X(4)
         If (DSWL.EQ.1) then
           SWL=X(5)
           r=X(6)
           eB=X(7)
         else
           Tide=X(5)
           Surge=X(6)
           SWL=Tide+Surge
           If (SWL.LT.TL) then
             Write(*,789) SWL,TL
             Write(50,789) SWL,TL
 789         Format(// 3X,'ERROR: SWL Below Toe Level !'
      1            / 10X,'SWL=',E17.10,3X,'TL=',E17.10 /)
             STOP
           Endif
           r=X(7)
           eB=X(8)
         Endif
         S=(2.*Pi*Hs)/(9.81*(Tm**2))

C        =================================================================
C        Definition of the seawall crest level, CL. CL has the value of
C        the design parameter (in Mode 1: if DSWL=1, X=7; if DSWL=2, X=8)
C        or it is the new variable (in Mode 2: if DSWL=1, X=8; if DSWL=2,
C        X=9). For Mode=2, the first partial derivative of the failure
C        function in relation to CL=X(N), OBJGRD(N), has to be calculated.
C        =================================================================

         If ((N.EQ.8.AND.DSWL.EQ.1).OR.(N.EQ.9.AND.DSWL.EQ.2)) then
           CL=X(N)
           If (FDer.EQ.1) then
             OBJGRD(N)=(A*eB*B*SQRT(9.81*Hs)/(r))*
      1              EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
             goto 15
           Endif
         else
```

```
          CL=Par
        Endif
        If (CL.LT.SWL) then
          Write(*,78) SWL,CL
          Write(50,78) SWL,CL
 78       Format(// 3X,'ERROR: SWL Above Crest Level !' /
      1          10X,'SWL=',E17.10,3X,'CL=',E17.10)
          STOP
        Endif


C       =================================================================
C       Calculation of the mean overtopping discharge over unit length
C       of seawall, Q.
C       =================================================================
        Q=A*SQRT((9.81*(Hs**3))/(S/(2.*Pi)))*
      1   EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))


C       =================================================================
C       Calculation of the failure function, OBJF, & of its first
C       partial derivatives, OBJGRD (if required).
C       =================================================================
        OBJF=TR(k0)-Q
        If (FDer.EQ.1) then
          OBJGRD(1)=((A*SQRT((9.81*2.*Pi*(Hs**3))/(S))/(2.*S))*
      1             EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))+
      1             (A*eB*B*(CL-SWL)*SQRT(9.81*Hs)/(2.*r*S))*
      1             EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi))))*
      1             ((-4.*Pi*Hs)/(9.81*(Tm**3)))
          OBJGRD(2)=-(A*9.81*Tm)*EXP((-eB*B*(CL-SWL))/
      1             (r*Tm*SQRT(9.81*Hs)))
      1             -(A*eB*B*(CL-SWL)*SQRT(9.81)/(2.*r*SQRT(Hs)))*
      1             EXP((-eB*B*(CL-SWL))/(r*Tm*SQRT(9.81*Hs)))
          OBJGRD(3)=-SQRT((9.81*(Hs**3))/(S/(2.*Pi)))*
      1             EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
          OBJGRD(4)=(A*eB*(CL-SWL)*SQRT(9.81*Hs)/r)*
      1             EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
          If (DSWL.EQ.1) then
            OBJGRD(5)=-(A*eB*B*SQRT(9.81*Hs)/r)*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
            OBJGRD(6)=-(A*eB*B*(CL-SWL)*SQRT(9.81*Hs)/(r**2))*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
            OBJGRD(7)=(A*B*(CL-SWL)*SQRT(9.81*Hs)/r)*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
          else
            OBJGRD(5)=-(A*eB*B*SQRT(9.81*Hs)/r)*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
            OBJGRD(6)=-(A*eB*B*SQRT(9.81*Hs)/r)*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
            OBJGRD(7)=-(A*eB*B*(CL-SWL)*SQRT(9.81*Hs)/(r**2))*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
            OBJGRD(8)=(A*B*(CL-SWL)*SQRT(9.81*Hs)/r)*
      1               EXP((-eB*B*SQRT(S)*(CL-SWL))/(r*Hs*SQRT(2.*Pi)))
          Endif
        Endif
 15     return
        End




C       ################################################################
C
        Subroutine SDunes(N,X,OBJF,Par)
C
C       ################################################################
C
C       Returns the failure function, OBJF, for the failure mode of dune
C       erosion using Vellinga's method & allowing for movements of
C       sand only seaward during the storm surge.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       N - Number of variables
C       X - Variables of the failure mode
C       DSWL - Definition of the SWL
C       Par - For Mode=1, it is the prescribed nourishment width; for
C             Mode=2, it is the starting nourishment width (from which
C             the program iterates to find the required value of the
C             nourishment width)
C       TR - Allowable erosion distance for each FORM calculation
C       k0 - Number of the FORM calculation
C       ctcurv - Coastal curvature in degrees per 1000m
C               (0<=ctcurv<=24Deg/1000m)
```

```
C          NPD - Initially, the number of points defining the initial
C                profile; then, the number of points defining the
C                nourished profile
C          NPDOld - Number of points defining the initial profile
C          (XP,YP) - Initially, the coordinates of the points defining the
C                    initial profile; then, the coordinates of the points
C                    defining the changed profile; finally, the coordinates
C                    of the points defining the nourished profile
C          (XPOld,YPOld) - Coordinates of the points defining the initial
C                          profile
C          NPch - Number of points to be changed in the initial profile
C          t - First point to be changed in the initial profile, point no.t
C
C          MODELING VARIABLES:
C          L - Maximum number of variables allowed by the program
C          ka - Maximum number of iterations allowed by the program to find
C               the final position of Vellinga's profile & the final
C               location of the surcharge face
C          je - Number of the iteration to find the final position of
C               Vellinga's profile
C          j - Number of the iteration to find the final location of the
C              surcharge face
C          Er - Maximum number of points (XP,YP) allowed by the program
C          h - Storm surge level
C          Tide - Tide level
C          Surge - Surge
C          Hs - Offshore significant wave height
C          DP - Change in the initial profile
C          D50 - Particle size: D50 of the dune sand
C          Ac - Accuracy of the computation
C          GB - Gust bumps
C          SD - Storm surge duration
C          MuHs - Mean of Hs expressed as a function of h
C          W - Fall velocity of dune sand in seawater
C          Depth - Depth of Vellinga's parabolic post-storm profile
C          Le - Length of Vellinga's parabolic post-storm profile
C          nourwidt - Nourishment width at top level
C          (S1,T1) - Intersection point between the nourished profile & the
C                    surge level
C          (S9,T9) - Point where the parabolic part of Vellinga's profile
C                    finishes
C          YPT9 - Y-coordinate of the point in the nourished profile which
C                 has X=S9
C          (S2,T2) - Intersection point between the nourished profile & the
C                    gradient 1:mt of Vellinga's profile
C          S8 - X-coordinate of the starting point of the parabolic part of
C               Vellinga's post-storm profile
C          (S3,T3) - Intersection point between the nourished profile & the
C                    gradient 1:md of Vellinga's profile
C          NumHump - Number of humps
C          NumDep - Number of depressions
C          BH - Area of a hump
C          SHump - Cumulative area of the humps starting from the seaward
C                  end of the profiles
C          BD - Area of a depression
C          SDep - Cumulative area of the depressions starting from the
C                 seaward end of the profiles
C          C - Area of erosion between the surge level, the nourished
C              profile above surge & the gradient 1:md
C          B - Area between the surge level & the gradient 1:mt of
C              Vellinga's profile
C          E - Area between points (S9,T9) & (S2,T2), between the surge
C              level & the nourished profile
C          (S4,T4) - Intersection point between the nourished profile & the
C                    surcharge gradient, 1:md
C          S10 - X-coordinate of the point of intersection between the
C                surge level & the gradient 1:md of the surcharge
C          SurchEros - Surcharge on erosion area C above surge level to
C                      take into account the effects of the accuracy of
C                      the computation, of the storm surge duration & of
C                      the gust bumps
C          SurchLongT - Surcharge on erosion area C above surge level to
C                       take into account the effect of a gradient in the
C                       longshore transport
C          TSurch - Total surcharge on erosion area C which is the sum of
C                   the surcharges SurchEros plus SurchLongT
C          SurD - Surcharge distance
C          Err - Error in the balance between erosion & accretion (assuming
C                movements of sand only seaward during the storm surge)
C          Mov - Value by which S8 is changed in each iteration performed
C                to find the final position of Vellinga's parabolic profile
C          i, M1, Q1, Q3, LD, BE, AL, Aux1, Sin, Fim - Auxiliary variables
C          It, AuxRstar, MuxN,
```

```
C       md, mt, nourtlev, mnour, TL  - Variables mentioned in the Common
C                                      statements but not used here
C
C       OUTPUT VARIABLE:
C       OBJF - Failure function
C
C       ##############################################################
C
        Integer*4 N,L,NPD,t,k0,ka,je,j,i,M1,Q1,NPch,NPDOld,Aux1,Fim,
     1            NumDep,NumHump,Er,It,AuxRstar,DSWL
        Parameter (L=15)
        Parameter (Er=100)
        Parameter (ka=999)
        Real*8 X(L),DP,h,Tide,Surge,Hs,D50,XP,YP,LD,W,BE,AL,Depth,Le,S1,
     1         T1,S9,T9,B,E,S2,T2,S8(0:Ka),C,S3,T3,SurchEros,OBJF,MuxN,
     1         Par,md,mt,XPOld,YPOld,SurD(0:ka),mnour,nourtlev,SD,Ac,GB,
     1         S4,T4,nourwidt,Err(0:ka),S10(0:ka),Mov,Sin(0:ka),YPT9,
     1         BD(Er),BH(Er),SDep(0:Er),SHump(0:Er),ctcurv,SurchLongT,
     1         TSurch,TR(L),TL,MuHs
        Common/BLOCK1/NPD,XP(100),YP(100)
        Common/BLOCK2/t,NPch,XPOld(100),YPOld(100),NPDOld
        Common/BLOCK3/MuxN(15),C,T3
        Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
        Common/BLOCK5/Ac,GB,SD
        Common/BLOCK7/TR
        Common/BLOCK9/k0,It,AuxRstar
        Common/BLOCK11/DSWL,TL

        Q1=0
        Aux1=0
        Mov=10.
        S8(0)=0.
        Err(0)=1
        Sin(0)=-1.
        je=1
        j=1
        MuHs=0.

        If (DSWL.EQ.1) then
          If (X(7).GE.3.0.AND.X(7).LT.7.0)
     1      MuHs=0.6*X(7)+4.82-0.0063*((7.-X(7))**3.13)
          If (X(7).GT.7.0) MuHs=0.6*X(7)+4.82
          Hs=X(1)+MuHs
          D50=X(2)
          DP=X(3)
          SD=X(4)
          GB=X(5)
          Ac=X(6)
          h=X(7)
        else
          If ((X(7)+X(8)).GE.3.0.AND.(X(7)+X(8)).LT.7.0)
     1      MuHs=0.6*(X(7)+X(8))+4.82-0.0063*((7.-(X(7)+X(8)))**3.13)
          If ((X(7)+X(8)).GT.7.0) MuHs=0.6*(X(7)+X(8))+4.82
          Hs=X(1)+MuHs
          D50=X(2)
          DP=X(3)
          SD=X(4)
          GB=X(5)
          Ac=X(6)
          Tide=X(7)
          Surge=X(8)
          h=Tide+Surge
        Endif

        NPD=NPDOld
        Do 3610 i=1,NPD
          XP(i)=XPOld(i)
          YP(i)=YPOld(i)
 3610   continue

        Call Profile(t,NPch,DP,h,X,S1,T1,N,Par,nourwidt)

C       ==================================================================
C       Calculation of the shape of Vellinga's parabolic post-storm
C       profile & of the position of the offshore point where the
C       parabolic profile terminates: from the starting point of the
C       profile, the length is Le & the depth is Depth.
C       ==================================================================
        LD=LOG10(D50)/LOG10(10.)
        W=1./(10**(0.476*(LD**2)+2.18*LD+3.226))
        BE=Hs/7.6
        AL=((Hs/7.6)**1.28)*((0.0268/W)**0.56)
        Le=250.*AL
```

```
          Depth=5.717*BE

C         ==================================================================
C         Definition of the position of Vellinga's profile relative to the
C         nourished profile in order to obtain a balance between the area
C         of erosion & the area of accretion.
C         ==================================================================
C
          Do 2005 i=NPD,1,-1
            If (YP(i).GE.T1) then
              M1=i
              goto 3069
            Endif
 2005     continue
          Write(*,9929)
          Write(50,9929)
 9929     Format(// 6X,'ERROR: Surge Level Above the Dune Height !')
          STOP
 3069     If (YP(M1).NE.T1) M1=M1+1

C         ----------
C         Definition of S8 which is the X-coordinate of the starting
C         point of the parabolic part of Vellinga's profile. As a first
C         approximation, S8=S1.
C         ----------
          S8(je)=S1
 7557     Call LandProf(je,S8,T1,M1,S1,C,S3,T3)

C         ----------
C         Calculation of the coordinates, (S9,T9), of the point where the
C         parabolic profile finishes.
C         ----------
          S9=S8(je)+Le
          T9=T1-Depth
          Call SeaProf(S9,T9,T1,B,E,S2,T2,YPT9)
          Call HumpDep(je,BE,AL,S1,T1,S2,T2,S3,T3,S8,S9,T9,B,C,E,md,YPT9,
     1               NumDep,NumHump,BD,BH,SDep,SHump)

C         ==================================================================
C         Calculation of the balance of erosion & accretion.
C         ==================================================================
          If (NumDep.GE.1) then
            If (NumDep.GT.NumHump) then
              If (NumHump.EQ.0) then
                Err(je)=BD(NumDep)
                Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
              else
                Call Error(Q1,BH,BD,NumDep,NumHump,SDep,SHump,Fim,je,
     1                     Err,T2,T3,S8,Mov,Sin,Aux1)
              Endif
            Endif
            If (NumDep.EQ.NumHump) then
              Call Error(Q1,BH,BD,NumDep,NumHump,SDep,SHump,Fim,je,Err,
     1                   T2,T3,S8,Mov,Sin,Aux1)
            Endif
            If (Fim.EQ.3) then
              S4=0.
              SurchEros=0.
              SurchLongT=0.
              TSurch=0.
              SurD(j)=0.
              goto 3355
            else
              If (Fim.EQ.1) goto 7557
              If (Fim.EQ.2) goto 999
            Endif
          Endif
          If ((NumDep.EQ.0).AND.(NumHump.EQ.0)) then
            S4=0.
            SurchEros=0.
            SurchLongT=0.
            TSurch=0.
            SurD(j)=0.
            Fim=3
            goto 3355
          Endif

 999      S8(1)=S8(je)

          Call SurchC(ctcurv,BE,W,S8,T1,S3,S4,T4,S10,SurD,SurchEros,
     1                SurchLongT,TSurch)

C         ==================================================================
```

```
C        Calculation of the failure function, OBJF.
C        ================================================================
 3355    OBJF=TR(k0)+S4

         return
         End




C        ################################################################
C
         Subroutine LDunes(N,X,OBJF,Par)
C
C        ################################################################
C
C        Returns the failure function, OBJF, for the failure mode of dune
C        erosion using Vellinga's method & allowing for movements of
C        sand in both directions during the storm surge.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        N - Number of variables
C        X - Variables of the failure mode
C        DSWL - Definition of the SWL
C        Par - For Mode=1, it is the prescribed nourishment width; for
C              Mode=2, it is the starting nourishment width (from which
C              the program iterates to find the required value of the
C              nourishment width)
C        TR - Allowable erosion distance for each FORM calculation
C        k0 - Number of the FORM calculation
C        ctcurv - Coastal curvature in degrees per 1000m
C                 (0<=ctcurv<=24Deg/1000m)
C        NPD - Initially, the number of points defining the initial
C              profile; then, the number of points defining the
C              nourished profile
C        NPDOld - Number of points defining the initial profile
C        (XP,YP) - Initially, the coordinates of the points defining the
C                  initial profile; then, the coordinates of the points
C                  defining the changed profile; finally, the coordinates
C                  of the points defining the nourished profile
C        (XPOld,YPOld) - Coordinates of the points defining the initial
C                        profile
C        NPch - Number of points to be changed in the initial profile
C        t - First point to be changed in the initial profile, point no.t
C
C        MODELING VARIABLES:
C        L - Maximum number of variables allowed by the program
C        ka - Maximum number of iterations allowed by the program to find
C             the final position of Vellinga's profile & the final
C             location of the surcharge face
C        je - Number of the iteration to find the final position of
C             Vellinga's profile
C        j - Number of the iteration to find the final location of the
C            surcharge face
C        h - Storm surge level
C        Tide - Tide level
C        Surge - Surge
C        Hs - Offshore significant wave height
C        DP - Change in the initial profile
C        D50 - Particle size: D50 of the dune sand
C        Ac - Accuracy of the computation
C        GB - Gust bumps
C        SD - Storm surge duration
C        MuHs - Mean of Hs expressed as a function of h
C        W - Fall velocity of dune sand in seawater
C        Depth - Depth of Vellinga's parabolic post-storm profile
C        Le - Length of Vellinga's parabolic post-storm profile
C        nourwidt - Nourishment width at top level
C        (S1,T1) - Intersection point between the nourished profile & the
C                  surge level
C        (S9,T9) - Point where the parabolic part of Vellinga's profile
C                  finishes
C        (S2,T2) - Intersection point between the nourished profile & the
C                  gradient 1:mt of Vellinga's profile
C        S8 - X-coordinate of the starting point of the parabolic part of
C             Vellinga's post-storm profile
C        (S3,T3) - Intersection point between the nourished profile & the
C                  gradient 1:md of Vellinga's profile
C        A - Area between the surge level & the parabolic part of
C            Vellinga's profile
C        C - Area of erosion between the surge level, the nourished
C            profile above surge & the gradient 1:md
```

```
C       B - Area between the surge level & the gradient 1:mt of
C           Vellinga's profile
C       Q - Area between the surge level & the nourished profile below
C           surge
C       (S4,T4) - Intersection point between the nourished profile & the
C                  surcharge gradient, 1:md
C       S10 - X-coordinate of the point of intersection between the
C             surge level & the gradient 1:md of the surcharge
C       SurchEros - Surcharge on erosion area C above surge level to
C                   take into account the effects of the accuracy of
C                   the computation, of the storm surge duration & of
C                   the gust bumps
C       SurchLongT - Surcharge on erosion area C above surge level to
C                    take into account the effect of a gradient in the
C                    longshore transport
C       TSurch - Total surcharge on erosion area C which is the sum of
C                the surcharges SurchEros plus SurchLongT
C       SurD - Surcharge distance
C       Err - Error in the balance between erosion & accretion (assuming
C             movements of sand in both directions during the storm
C             surge)
C       Mov - Value by which S8 is changed in each iteration performed
C             to find the final position of Vellinga's parabolic profile
C       i, M1, Q1, Q3, LD, BE, AL, Aux1,
C       Sin, Fim, DS - Auxiliary variables
C       It, AuxRstar, MuxN,
C       md, mt, nourtlev, mnour, TL  - Variables mentioned in the Common
C                                      statements but not used here
C
C       OUTPUT VARIABLE:
C       OBJF - Failure function
C
C       ################################################################
C
        Integer*4 N,L,NPD,t,ka,k0,j,i,M1,NPch,NPDOld,Aux1,je,Q1,Fim,It,
     1            AuxRstar,DSWL
        Parameter (L=15)
        Parameter (ka=999)
        Real*8 X(L),DP,h,Tide,Surge,Hs,D50,XP,YP,LD,W,BE,AL,Depth,Le,A,
     1         S1,T1,S9,T9,Q,B,S2,T2,S8(0:Ka),DS,C,S3,T3,SurchEros,OBJF,
     1         MuxN,Par,md,mt,XPOld,YPOld,SurD(0:ka),SD,Ac,GB,mnour,
     1         nourtlev,S10(0:ka),S4,T4,nourwidt,Err(0:ka),Mov,
     1         Sin(0:ka),ctcurv,SurchLongT,TSurch,TR(L),TL,MuHs
        Common/BLOCK1/NPD,XP(100),YP(100)
        Common/BLOCK2/t,NPch,XPOld(100),YPOld(100),NPDOld
        Common/BLOCK3/MuxN(15),C,T3
        Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
        Common/BLOCK5/Ac,GB,SD
        Common/BLOCK7/TR
        Common/BLOCK9/k0,It,AuxRstar
        Common/BLOCK11/DSWL,TL

        Fim=0
        Q1=0
        Aux1=0
        Mov=10.
        S8(0)=0.
        Err(0)=1
        Sin(0)=0.
        j=1
        je=1
        MuHs=0.

        If (DSWL.EQ.1) then
          If (X(7).GE.3.0.AND.X(7).LT.7.0)
     1      MuHs=0.6*X(7)+4.82-0.0063*((7.-X(7))**3.13)
          If (X(7).GT.7.0) MuHs=0.6*X(7)+4.82
          Hs=X(1)+MuHs
          D50=X(2)
          DP=X(3)
          SD=X(4)
          GB=X(5)
          Ac=X(6)
          h=X(7)
        else
          If ((X(7)+X(8)).GE.3.0.AND.(X(7)+X(8)).LT.7.0)
     1      MuHs=0.6*(X(7)+X(8))+4.82-0.0063*((7.-(X(7)+X(8)))**3.13)
          If ((X(7)+X(8)).GT.7.0) MuHs=0.6*(X(7)+X(8))+4.82
          Hs=X(1)+MuHs
          D50=X(2)
          DP=X(3)
          SD=X(4)
          GB=X(5)
```

```
          Ac=X(6)
          Tide=X(7)
          Surge=X(8)
          h=Tide+Surge
        Endif

        NPD=NPDOld
        Do 3610 i=1,NPD
          XP(i)=XPOld(i)
          YP(i)=YPOld(i)
 3610   continue

        Call Profile(t,NPch,DP,h,X,S1,T1,N,Par,nourwidt)

C       ===============================================================
C       Calculation of the shape of Vellinga's parabolic post-storm
C       profile & of the position of the offshore point where the
C       parabolic profile terminates: from the starting point of the
C       profile, the length is Le & the depth is Depth. Calculation of
C       the area A between the surge level & the parabolic part of
C       Vellinga's profile.
C       ===============================================================
        LD=LOG10(D50)/LOG10(10.)
        W=1./(10**(0.476*(LD**2)+2.18*LD+3.226))
        BE=Hs/7.6
        AL=((Hs/7.6)**1.28)*((0.0268/W)**0.56)
        Le=250.*AL
        Depth=5.717*BE
        A=AL*BE*854.8

C       ===============================================================
C       Definition of the position of Vellinga's profile relative to the
C       nourished profile in order to obtain a balance between the area
C       of erosion & the area of accretion.
C       ===============================================================
C
        Do 2005 i=NPD,1,-1
          If (YP(i).GE.T1) then
            M1=i
            goto 3069
          Endif
 2005   continue
        Write(*,9929)
        Write(50,9929)
 9929   Format(// 6X,'ERROR: Surge Level Above the Dune Height !')
        STOP
 3069   If (YP(M1).NE.T1) M1=M1+1

C       ----------
C       Definition of S8 which is the X-coordinate of the starting
C       point of the parabolic part of Vellinga's profile. As a first
C       approximation, S8=S1.
C       ----------
        S8(je)=S1
 6556   Call LandProf(je,S8,T1,M1,S1,C,S3,T3)

C       ----------
C       Calculation of the coordinates, (S9,T9), of the point where the
C       parabolic profile finishes.
C       ----------
        S9=S8(je)+Le
        T9=T1-Depth
        Call LSeaProf(S9,T9,S1,M1,T1,B,Q,S2,T2)

C       ===============================================================
C       Calculation of Err. Err is the error in the balance between
C       erosion & accretion (assuming movements of sand in both
C       directions during the storm surge).
C       If |Err|>=1 then the balance between erosion & accretion is
C       not satisfactory. If Err>1, then accretion exceeds erosion
C       (Vellinga's profile has to be moved landward); if Err<-1,
C       then erosion exceeds accretion (Vellinga's profile has to be
C       moved seaward, if S8 not equal to S1; if S8=S1, no erosion is
C       expected). After Vellinga's profile is moved, the new
C       X-coordinate of the starting point of the parabolic part of the
C       profile, S8, is calculated.
C       ===============================================================
        Err(je)=Q-B-A-C
        If ((Err(je).LE.0.).AND.(Q1.EQ.0)) then
          SurchEros=0.
          SurchLongT=0.
          TSurch=0.
          SurD(j)=0.
```

```
                S4=0.
                Fim=3
                goto 3355
             Endif
             If (Err(je).LT.0.) then
                Sin(je)=1.
             else
                Sin(je)=-1.
             Endif
             If (ABS(Err(je)).GE.1.) then
                DS=Err(je)/(T3-T2)
                je=je+1
                If (je.GT.999) then
                   Write(*,9987)
                   Write(50,9987)
  9987             Format(// 6X,
       1                  'ERROR: The maximum number of iterations allowed by' /
       1                  6X,'        the program to find the final position of'/
       1                  6X,'        Vellinga`s profile has been exceeded !')
                   STOP
                Endif
                S8(je)=S8(je-1)-DS
                If ((ABS(S8(je)-S8(je-2)).LT.0.0001).OR.(Aux1.EQ.1)) then
                   If (Sin(je-1).NE.Sin(je-2)) then
                      Mov=Mov/10.
                   Endif
                   S8(je)=S8(je-1)+Sin(je-1)*Mov
                   Aux1=1
                Endif
                Q1=1
                goto 6556
             Endif

          S8(1)=S8(je)

          Call SurchC(ctcurv,BE,W,S8,T1,S3,S4,T4,S10,SurD,SurchEros,
       1              SurchLongT,TSurch)

C        ================================================================
C        Calculation of the failure function OBJF.
C        ================================================================
  3355   OBJF=TR(k0)+S4

          return
          End




C        ################################################################
C
         Subroutine Profile(t,NPch,DP,h,X,S1,T1,N,Par,nourwidt)
C
C        ################################################################
C
C        Returns the coordinates of the nourished profile, which is used
C        for comparison with Vellinga's profile.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        N - Number of variables
C        X - Variables of the failure mode
C        Par - For Mode=1, it is the prescribed nourishment width; for
C              Mode=2, it is the starting nourishment width (from which
C              the program iterates to find the required value of the
C              nourishment width)
C        DSWL - Definition of the SWL
C        DP - Change in the initial profile
C        h - Storm surge level
C        NPch - Number of points to be changed in the initial profile
C        t - First point to be changed in the initial profile, point no.t
C        nourtlev - Nourishment top level
C        1:mnour - Gradient of the nourished face
C
C        INPUT/OUTPUT VARIABLES:
C        NPD - Initially, the number of points defining the initial
C              profile; then, the number of points defining the
C              nourished profile
C        (XP,YP) - Initially, the coordinates of the points defining the
C                  initial profile; then, the coordinates of the points
C                  defining the changed profile; finally, the coordinates
C                  of the points defining the nourished profile
C
```

```
C       MODELING VARIABLES:
C       L - Maximum number of variables allowed by the program
C       (XN,YN) - Intersection point between the changed profile & the
C                 nourishment top level
C       (XM,YM) - The most seaward point at the nourishment top level
C       (XQ,YQ) - Intersection point between the nourishment slope
C                 1:mnour & the changed profile
C       (XB,YB) - Intersection point between the changed profile & the
C                 surge level
C       i, k, M, M1, M2, M3, Q3,
C       Zes, G, XPOld, YPOld - Auxiliary variables
C       md, mt, ctcurv, TL - Variables mentioned in the Common
C                            statements but not used here
C
C       OUTPUT VARIABLES:
C       nourwidt - Nourishment width at top level
C       (S1,T1) - Intersection point between the nourished profile & the
C                 surge level
C
C       ################################################################
C
        Integer*4 M,N,L,NPD,t,i,M1,M2,M3,Q3,NPch,k,DSWL
        Parameter (L=15)
        Parameter (M=100)
        Real*8 X(L),DP,h,XP,YP,S1,T1,Par,XB,YB,md,mt,mnour,nourtlev,
     1         nourwidt,ctcurv,XN,YN,XM,YM,XQ,YQ,Zes,G,TL,XPOld(M),
     1         YPOld(M)
        Common/BLOCK1/NPD,XP(100),YP(100)
        Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv
        Common/BLOCK11/DSWL,TL

        Q3=0

C       ================================================================
C       Change in the Y-coordinate, YP, of NPch points of the initial
C       profile (YP(t),YP(t+1),...,YP(t+NPch-1)). The change is DP & the
C       new profile is referred to as the changed profile.
C       ================================================================
        Do 2291 i=t,(t+NPch-1)
          YP(i)=YP(i)+DP
 2291   continue

        Do 3499 i=1,NPD
          XPOld(i)=XP(i)
          YPOld(i)=YP(i)
 3499   continue

C       ================================================================
C       Calculation of the intersection point, (XB,YB), between the
C       changed profile & the surge level.
C       ================================================================
        Do 2305 i=NPD,1,-1
          If (YP(i).GE.h) then
            M1=i
            goto 2306
          Endif
 2305   continue
        Write(*,589)
        Write(50,589)
 589    Format(// 6X,'ERROR: Surge Level Above the Dune Height !')
        STOP
 2306   XB=XP(M1)+(XP(M1+1)-XP(M1))*(h-YP(M1))/(YP(M1+1)-YP(M1))
        YB=h

C       ================================================================
C       Calculation of a new profile based on nourishment
C       characteristics (gradient of the nourished face, 1:mnour, &
C       nourishment top level, nourtlev), on the given value of the
C       design parameter (nourishment width at top level, nourwidt) & on
C       the mode to be studied. The new profile is referred to as the
C       nourished profile.
C       ================================================================

C       ----------
C       First, the width of the nourishment will have the value of the
C       design parameter (in Mode 1: if DSWL=1, X=7; if DSWL=2, X=8) or
C       it is the new variable (in Mode 2: if DSWL=1, X=8; if DSWL=2,
C       X=9). For Mode=2, the first partial derivative of the failure
C       function in relation to nourwidt=X(N), OBJGRD(N), has to be
C       calculated.
C       ----------
        If ((N.EQ.8.AND.DSWL.EQ.1).OR.(N.EQ.9.AND.DSWL.EQ.2)) then
          nourwidt=X(N)
```

```
            else
              nourwidt=Par
            Endif
            If (nourwidt.EQ.0.) then
              S1=XB
              T1=YB
              return
            Endif
            If (nourwidt.LT.0.) then
              Write(*,2155) nourwidt
              Write(50,2155) nourwidt
 2155         Format(// 11X,'ERROR: Nourishment Width = ',E17.10,' < 0 !')
              STOP
            Endif

C         ----------
C         Second, calculation of the intersection point, (XN,YN), between
C         the changed profile & the nourishment top level.
C         ----------
            Do 2375 i=NPD,1,-1
              If (YP(i).GT.nourtlev) then
                M2=i
                goto 2386
              Endif
              If (YP(i).EQ.nourtlev) then
                M2=i
                Q3=1
                goto 2386
              Endif
 2375       continue
            Write(*,1589)
            Write(50,1589)
 1589       Format(// 6X,'ERROR: Nourishment Top Level Above the',
     1              1X,'Dune Height !')
            STOP
 2386       XN=XP(M2)+(XP(M2+1)-XP(M2))*(nourtlev-YP(M2))/(YP(M2+1)-YP(M2))
            YN=nourtlev

C         ----------
C         Third, extension of the nourishment seaward by distance nourwidt
C         at the nourishment top level gives point (XM,YM).
C         ----------
            XM=XN+nourwidt
            YM=YN

C         ----------
C         Fourth, extension of the nourishment downwards at a slope of
C         1:mnour until it intersects the changed profile gives point
C         (XQ,YQ).
C         ----------
            Do 4499 i=M2,NPD
              If (XP(i).GT.XM) then
                M3=i
                goto 4398
              Endif
 4499       continue
            Write(*,8225)
            Write(50,8225)
 8225       Format(// 11X,'Extend the Initial Profile Seaward !')
            STOP
 4398       Zes=XP(M3)-XM
            G=YM-Zes/mnour
            If (G.GT.YP(M3)) then
              M3=M3+1
              If (M3.GT.NPD) then
                Write(*,9225)
                Write(50,9225)
 9225           Format(// 11X,'The Nourished Face Does Not Intersect the' /
     1                  11X,'Profile! Extend the Initial Profile Seaward !')
                STOP
              Endif
              goto 4398
            Endif
            XQ=XP(M3)-Zes*(YP(M3)-G)/(YM-G-(Zes*(YP(M3-1)-YP(M3))/
     1        (XP(M3)-XP(M3-1))))
            YQ=YP(M3)+(XP(M3)-XQ)*(YP(M3-1)-YP(M3))/(XP(M3)-XP(M3-1))

C         ----------
C         Fifth, if the top level of the nourishment is above the surge
C         level, & the nourished profile intersects the changed profile
C         below surge level, then the new coordinates of the point
C         (XB,YB) in the nourished profile have to be calculated, (S1,T1).
C         ----------
```

```
          If ((nourtlev.GE.h).AND.(YQ.LT.YB)) then
            S1=XM+mnour*(YM-YB)
          else
            S1=XB
          Endif
          T1=YB

C         ----------
C         Finally, definition of the points, (XP,YP), of the whole profile
C         after the nourishment.
C         ----------
          If ((M2-Q3).GE.1) then
            Do 2381 i=1,(M2-Q3)
               XP(i)=XPOld(i)
               YP(i)=YPOld(i)
 2381       continue
          Endif
          XP(M2-Q3+1)=XN
          YP(M2-Q3+1)=YN
          XP(M2-Q3+2)=XM
          YP(M2-Q3+2)=YM
          XP(M2-Q3+3)=XQ
          YP(M2-Q3+3)=YQ

          NPD=(M2-Q3+3)+(NPD-M3+1)
          k=0
          Do 2997 i=(M2-Q3+4),NPD
            XP(i)=XPOld(M3+k)
            YP(i)=YPOld(M3+k)
            k=k+1
 2997     continue

          return
          End



C         ##################################################################
C
          Subroutine LandProf(je,S8,T1,M1,S1,C,S3,T3)
C
C         ##################################################################
C
C         Returns the intersection point, (S3,T3), between the nourished
C         profile & the gradient 1:md of Vellinga's profile. It also
C         returns the area, C, between the surge level, the nourished
C         profile above surge & the gradient 1:md.
C
C         ##################################################################
C
C         INPUT VARIABLES:
C         je - Number of the iteration to find the final position of
C              Vellinga's profile
C         NPD - Number of points defining the nourished profile
C         1:md - Gradient of the eroded dune face
C         (XP,YP) - Coordinates of the points defining the nourished
C                   profile
C         (S1,T1) - Intersection point between the nourished profile & the
C                   surge level
C         M1 - Number of the point, (XP(M1),YP(M1)), from seaward for
C              which YP(M1)<=T1
C         S8 - X-coordinate of the starting point of the parabolic part of
C              Vellinga's post-storm profile
C
C         MODELING VARIABLES:
C         ka - Maximum number of iterations allowed by the program to find
C              the final position of Vellinga's profile & the final
C              location of the surcharge face
C         i, M3, U, V - Auxiliary variables
C         mt, nourtlev, mnour, ctcurv - Variables mentioned in the Common
C                                       statements but not used here
C
C         OUTPUT VARIABLES:
C         (S3,T3) - Intersection point between the nourished profile & the
C                   gradient 1:md of Vellinga's profile
C         C - Area of erosion between the surge level, the nourished
C             profile above surge & the gradient 1:md
C
C         ##################################################################
C
          Integer*4 NPD,i,je,M1,M3,ka
          Parameter (ka=999)
          Real*8 T1,XP,YP,S1,S8(0:ka),C,S3,T3,U,V,md,mt,mnour,nourtlev,
```

```
      1        ctcurv
         Common/BLOCK1/NPD,XP(100),YP(100)
         Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv

C        ===================================================================
C        Calculation of the intersection point, (S3,T3), between the
C        nourished profile & the gradient 1:md of Vellinga's profile.
C        ===================================================================
         If (S8(je).LT.XP(1)) then
           Write(*,8625)
           Write(50,8625)
 8625      Format(// 11X,'Extend the Initial Profile Landward !')
           STOP
         Endif
         Do 0903 i=1,NPD
           If (XP(i).GT.S8(je)) then
             M3=i
             goto 4977
           Endif
 0903    continue
         Write(*,8925)
         Write(50,8925)
 8925    Format(// 11X,'Extend the Initial Profile Seaward !')
         STOP
 4977    U=S8(je)-XP(M3-1)
         V=T1+U/md
         If (V.LT.YP(M3-1)) then
           M3=M3-1
           If (M3.LE.1) then
             Write(*,8225)
             Write(50,8225)
 8225        Format(// 11X,'Extend the Initial Profile Landward !')
             STOP
           Endif
           goto 4977
         Endif
         S3=XP(M3-1)+U*(V-YP(M3-1))/(V-T1-(U*(YP(M3-1)-YP(M3))/
      1     (XP(M3)-XP(M3-1))))
         If (S3.LT.S8(je)) then
           T3=YP(M3-1)-(S3-XP(M3-1))*(YP(M3-1)-YP(M3))/(XP(M3)-XP(M3-1))
         else
           S3=S8(je)
           T3=T1
           C=0.
           goto 2591
         Endif

C        ===================================================================
C        Calculation of area, C, between the surge level, the nourished
C        profile above surge & the gradient 1:md.
C        ===================================================================
         If ((XP(M3).LE.S1).AND.(XP(M1-1).LE.S1)) then
           C=0.5*(XP(M3)-S3)*((T3-T1)+(YP(M3)-T1))+
      1      0.5*(S1-XP(M1-1))*(YP(M1-1)-T1)
         else
           C=0.5*(S1-S3)*(T3-T1)
         Endif
         If ((M1-2).GE.M3) then
           Do 4701 i=M3,(M1-2)
             C=C+0.5*(XP(i+1)-XP(i))*((YP(i+1)-T1)+(YP(i)-T1))
 4701      continue
         Endif
         C=C-0.5*(S8(je)-S3)*(T3-T1)
 2591    return
         End




C        ################################################################
C
         Subroutine SeaProf(S9,T9,T1,B,E,S2,T2,YPT9)
C
C        ################################################################
C
C        Returns the point of intersection, (S2,T2), between the
C        nourished profile & the gradient 1:mt of Vellinga's profile. It
C        also returns the area, B, between the surge level & the gradient
C        1:mt; & the area, E, between points (S9,T9) & (S2,T2), between
C        the surge level & the nourished profile.
C
C        ################################################################
C
C        INPUT VARIABLES:
```

```
C        NPD - Number of points defining the nourished profile
C        (XP,YP) - Coordinates of the points defining the nourished
C                  profile
C        1:mt - Gradient of the toe of the post-storm profile
C        T1 - Y-coordinate of the intersection point between the
C             nourished profile & the surge level
C        (S9,T9) - Point where the parabolic part of Vellinga's profile
C                  finishes
C
C        MODELING VARIABLES:
C        i, M2, R1, R2, Zes, G - Auxiliary variables
C        md, nourtlev, mnour, ctcurv - Variables mentioned in the Common
C                                      statements but not used here
C
C        OUTPUT VARIABLES:
C        YPT9 - Y-coordinate of the point in the nourished profile which
C               has X=S9
C        (S2,T2) - Intersection point between the nourished profile & the
C                  gradient 1:mt of Vellinga's profile
C        B - Area between the surge level & the gradient 1:mt of
C            Vellinga's profile
C        E - Area between points (S9,T9) & (S2,T2), between the surge
C            level & the nourished profile
C
C        #############################################################
C
         Integer*4 NPD,i,M2,R1,R2
         Real*8 XP,YP,T1,S2,T2,S9,T9,B,E,Zes,G,md,mt,mnour,
     1         nourtlev,YPT9,ctcurv
         Common/BLOCK1/NPD,XP(100),YP(100)
         Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv

C        ==================================================================
C        Calculation of the point of intersection, (S2,T2), between the
C        nourished profile & the gradient 1:mt of Vellinga's profile.
C        ==================================================================
         Do 4499 i=1,NPD
           If (XP(i).GT.S9) then
             M2=i
             R1=M2
             YPT9=YP(M2-1)+(YP(M2)-YP(M2-1))*(S9-XP(M2-1))/
     1          (XP(M2)-XP(M2-1))
             If (YPT9.GE.T9) then
               S2=S9
               T2=YPT9
               B=0.
               E=0.
               M2=M2-1
               return
             else
               goto 4398
             Endif
           Endif
 4499    continue
         Write(*,8275)
         Write(50,8275)
 8275    Format(// 11X,'Extend the Initial Profile Seaward !')
         STOP
 4398    Zes=XP(M2)-S9
         G=T9-Zes/mt
         If (G.GT.YP(M2)) then
           M2=M2+1
           If (M2.GT.NPD) then
             Write(*,8225)
             Write(50,8225)
 8225        Format(// 11X,'Extend the Initial Profile Seaward !')
             STOP
           Endif
           goto 4398
         Endif
         S2=XP(M2)-Zes*(YP(M2)-G)/(T9-G-(Zes*(YP(M2-1)-YP(M2))/
     1     (XP(M2)-XP(M2-1))))
         T2=YP(M2)+(XP(M2)-S2)*(YP(M2-1)-YP(M2))/(XP(M2)-XP(M2-1))
         M2=M2-1

C        ==================================================================
C        Calculation of the area, B, between the surge level & the
C        gradient 1:mt.
C        ==================================================================
         B=0.5*(S2-S9)*((T1-T2)+(T1-T9))

C        ==================================================================
C        Calculation of the area, E, between points (S9,T9) & (S2,T2),
```

```
C       between the surge level & the nourished profile.
C       ================================================================
        Do 101 i=1,NPD
          If (XP(i).GT.S2) then
            R2=i-1
            goto 102
          Endif
 101    continue
        Write(*,8925)
        Write(50,8925)
 8925   Format(// 11X,'Extend the Initial Profile Seaward !')
        STOP
 102    If ((R2.GT.S9).AND.(R1.LT.S2)) then
          E=0.5*(XP(R1)-S9)*((T1-YP(R1))+(T1-YPT9))+0.5*(S2-XP(R2))*
     1     ((T1-T2)+(T1-YP(R2)))
        else
          E=0.5*(S2-S9)*((T1-T2)+(T1-YPT9))
        Endif
        If (R2.GT.R1) then
          Do 3781 i=R1,(R2-1)
            E=E+0.5*(XP(i+1)-XP(i))*((T1-YP(i))+(T1-YP(i+1)))
 3781    continue
        Endif
        return
        End




C       ################################################################
C
        Subroutine HumpDep(je,BE,AL,S1,T1,S2,T2,S3,T3,S8,S9,T9,B,C,E,
     1                     md,YPT9,NumDep,NumHump,BD,BH,SDep,SHump)
C
C       ################################################################
C
C       Returns the number of humps, Numhump, the number of depressions,
C       NumDep, & the corresponding areas, BH & BD. SHump & SDep are the
C       corresponding cumulatives areas starting from the seaward end of
C       the profiles.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       je - Number of the iteration to find the final position of
C            Vellinga's profile
C       NPD - Number of points defining the nourished profile
C       (XP,YP) - Coordinates of the points defining the nourished
C                 profile
C       (S1,T1) - Intersection point between the nourished profile & the
C                 surge level
C       (S9,T9) - Point where the parabolic part of Vellinga's profile
C                 finishes
C       YPT9 - Y-coordinate of the point in the nourished profile which
C              has X=S9
C       (S2,T2) - Intersection point between the nourished profile & the
C                 gradient 1:mt of Vellinga's profile
C       S8 - X-coordinate of the starting point of the parabolic part of
C            Vellinga's post-storm profile
C       (S3,T3) - Intersection point between the nourished profile & the
C                 gradient 1:md of Vellinga's profile
C       1:md - Gradient of the eroded dune face
C       C - Area of erosion between the surge level, the nourished
C           profile above surge & the gradient 1:md
C       B - Area between the surge level & the gradient 1:mt of
C           Vellinga's profile
C       E - Area between points (S9,T9) & (S2,T2), between the surge
C           level & the nourished profile
C       BE, AL - Auxiliary variables
C
C       MODELING VARIABLES:
C       ka - Maximum number of iterations allowed by the program to find
C            the final position of Vellinga's profile & the final
C            location of the surcharge face
C       Er - Maximum number of points (XP,YP) allowed by the program
C       (XPOld1,YPOld1) - Coordinates of the points defining the
C                         nourished profile
C       NPV - Number of points defining Vellinga's profile
C       (XPV,YPV) - Coordinates of the points defining Vellinga's
C                   profile
C       (XHStart,YHStart) - Coordinates of the starting point of a hump
C       (XHEnd,YHEnd) - Coordinates of the end point of a hump
C       BH1 - Area between Vellinga's profile & the surge level at the
C             location of a hump. If it is the last hump & the area C>0,
```

```
C              then BH1 is the area as defined previously plus the area C
C         BH2 - Area between the nourished profile & the surge level at
C               the location of a hump
C         (XDStart,YDStart) - Coordinates of the starting point of a
C                         depression
C         (XDEnd,YDEnd) - Coordinates of the end point of a depression
C         BD1 - Area between Vellinga's profile & the surge level at the
C               location of a depression. If it is the first depression &
C               the area B>0, then BD1 is the area as defined previously
C               plus the area B
C         BD2 - Area between the nourished profile & the surge level at
C               the location of a depression. If it is the first
C               depression & the area B>0, then BD2 is the area as defined
C               previously plus the area E
C         (XFUNC,YFUNC) - Coordinates of intersection point between the
C                        nourished profile & Vellinga's profile
C         i, j, k, N1, N2, N3, D0, D1, D2, D3, H0, H1, H2, H3,
C         Aux3, Aux4 - Auxiliary variables
C
C         OUTPUT VARIABLES:
C         NumHump - Number of humps
C         NumDep - Number of depressions
C         BH - Area of a hump
C         SHump - Cumulative area of the humps starting from the seaward
C                  end of the profiles
C         BD - Area of a depression
C         SDep - Cumulative area of the depressions starting from the
C                seaward end of the profiles
C
C         ############################################################
C
          Integer*4 NPD,k,ka,Er,j,je,i,N1,N2,N3,D0,D1,D2,D3,H0,H1,H2,H3,
     1            Aux3,Aux4,NumDep,NumHump,NPV
          Parameter (Er=100)
          Parameter (ka=999)
          Real*8 XP,YP,BE,AL,S1,T1,S2,T2,S3,T3,S8(0:Ka),S9,T9,B,C,E,
     1          md,YPT9,BD(Er),BD1(Er),BD2(Er),BH(Er),BH1(Er),BH2(Er),
     1          SDep(0:Er),SHump(0:Er),XPV(Er),YPV(Er),XHStart(Er),
     1          XHEnd(Er),YHStart(Er),YHEnd(Er),XDStart(Er),XDEnd(Er),
     1          YDStart(Er),YDEnd(Er),XFunc,YFunc,XPOld1(Er),YPOld1(Er)
          Common/BLOCK1/NPD,XP(100),YP(100)

          Do 367 k=1,NPD
            XPOld1(k)=XP(k)
            YPOld1(k)=YP(k)
 367      continue

C         ================================================================
C         Calculation of the Y-coordinate, YPV, of Vellinga's points,
C         (XPV,YPV), corresponding to the X-values of the points in the
C         nourished profile (XP,YP). The total number of points in
C         Vellinga's profile is NPV.
C         ================================================================
          Do 11 i=1,NPD
            If (XP(i).GT.S3) then
              N1=i
              goto 15
            Endif
 11       continue
 15       Do 12 i=NPD,1,-1
            If (XP(i).LT.S9) then
              N2=i
              goto 16
            Endif
 12       continue
 16       Do 13 i=1,NPD
            If (XP(i).GT.S8(je)) then
              N3=i
              goto 17
            Endif
 13       continue
 17       If (N2.GE.N1) then
            NPV=N2-N1+4
          else
            NPV=3
          Endif
          XPV(1)=S3
          YPV(1)=T3
          XPV(NPV)=S2
          YPV(NPV)=T2
          XPV(NPV-1)=S9
          YPV(NPV-1)=T9
          If ((N3-1).GE.N1) then
```

```
        Do 14 i=2,(N3-1-N1+2)
          XPV(i)=XP(N1+i-2)
          YPV(i)=YPV(i-1)-(XPV(i)-XPV(i-1))/md
 14       continue
        Endif
        If ((N3-N1+2).LE.(NPV-2)) then
          Do 18 i=(N3-N1+2),(NPV-2)
            XPV(i)=XP(N1+i-2)
            YPV(i)=T1-(0.4714*Be*SQRT((AL**(-1))*(XPV(i)-S8(je))+18)
     1            -2.*BE)
 18       continue
        Endif

C       ================================================================
C       Determination of the intersection points between the two
C       profiles. Where the nourished profile is above Vellinga's
C       profile it is a hump. A hump starts at (XHStart,YHStart) &
C       finishes landward at (XHEnd,YHEnd). If the nourished profile is
C       below Vellinga's profile, it is a depression. A depression
C       starts at (XDStart,YDStart) & finishes landward at
C       (XDEnd,YDEnd). The total number of humps is Numhump & the total
C       number of depressions is NumDep.
C       ================================================================
        NumDep=0
        NumHump=0
        If (NPV.EQ.3) then
          If (YPT9.GE.T9) then
            NumHump=1
            XHStart(1)=S9
            YHStart(1)=YPT9
            XHEnd(1)=S3
            YHEnd(1)=T3
            Write(*,*) 'Only 1 hump; Nao sei se isto esta bem !'
            Write(50,*) 'Only 1 hump; Nao sei se isto esta bem !'
          else
            NumDep=1
            XDStart(1)=S9
            YDStart(1)=YPT9
            XDEnd(1)=S3
            YDEnd(1)=T3
            Write(*,*) 'Only 1 depression; Nao sei se isto esta bem !'
            Write(50,*) 'Only 1 depression; Nao sei se isto esta bem !'
          Endif
          goto 2950
        Endif
        Do 19 i=(NPV-2),2,-1
          If (
     1        (YPV(i).GT.YP(N1+i-2)).OR.
     1        (
     1        (YPV(i).EQ.YP(N1+i-2)).AND.
     1        ((
     1          (YPV(i+1).GT.YP(N1+i-1)).AND.(i.NE.(NPV-2))).OR.
     1          ((B.GT.0.).AND.(i.EQ.(NPV-2)))
     1        ))
     1        ) then
            If ((((YPV(i+1).LE.YP(N1+i-1)).AND.(i.NE.(NPV-2))).OR.
     1        ((B.EQ.0.).AND.(i.EQ.(NPV-2)))).OR.
     1        ((i+1).EQ.(NPV-1))) then
              NumDep=NumDep+1
              If (((i+1).EQ.(NPV-1)).AND.(B.EQ.0.)) then
                XP(N1+i-1)=S9
                YP(N1+i-1)=YPT9
              Endif
              If (((i+1).EQ.(NPV-1)).AND.(B.GT.0.)) then
                XDStart(NumDep)=S9
                YDStart(NumDep)=YPT9
                goto 20
              Endif
              Aux3=0
              Call XYFunc(je,i,N1,Aux3,BE,AL,T1,S8,XFunc,YFunc)
              XDStart(NumDep)=XFunc
              YDStart(NumDep)=YFunc
              Do 3673 k=1,NPD
                XP(k)=XPOld1(k)
                YP(k)=YPOld1(k)
 3673         continue
            Endif
 20         If ((YPV(i-1).LE.YP(N1+i-3)).AND.((i-1).NE.1)) then
              Aux3=1
              Call XYFunc(je,i,N1,Aux3,BE,AL,T1,S8,XFunc,YFunc)
              XDEnd(NumDep)=XFunc
              YDEnd(NumDep)=YFunc
            Endif
```

```
              If ((i-1).EQ.1) then
                XDEnd(NumDep)=S3
                YDEnd(NumDep)=T3
              Endif
              If (((i+1).EQ.(NPV-1)).AND.(B.EQ.0.)) then
                NumHump=NumHump+1
                XHStart(NumHump)=S2
                YHStart(NumHump)=T2
                XHEnd(NumHump)=XDStart(NumDep)
                YHEnd(NumHump)=YDStart(NumDep)
              Endif
            Endif
            If (
     1          (YPV(i).LT.YP(N1+i-2)).OR.
     1          (
     1           (YPV(i).EQ.YP(N1+i-2)).AND.
     1           ((
     1             (YPV(i+1).LT.YP(N1+i-1)).AND.(i.NE.(NPV-2))).OR.
     1             ((B.EQ.0.).AND.(i.EQ.(NPV-2)))
     1           ))
     1          ) then
              If ((((YPV(i+1).GE.YP(N1+i-1)).AND.(i.NE.(NPV-2))).OR.
     1          ((B.GT.0.).AND.(i.EQ.(NPV-2)))).OR.
     1          ((i+1).EQ.(NPV-1))) then
                NumHump=NumHump+1
                If (((i+1).EQ.(NPV-1)).AND.(B.EQ.0.)) then
                  XHStart(NumHump)=S2
                  YHStart(NumHump)=T2
                  goto 21
                Endif
                If (((i+1).EQ.(NPV-1)).AND.(B.GT.0.)) then
                  XP(N1+i-1)=S9
                  YP(N1+i-1)=YPT9
                Endif
                Aux3=0
                Call XYFunc(je,i,N1,Aux3,BE,AL,T1,S8,XFunc,YFunc)
                XHStart(NumHump)=XFunc
                YHStart(NumHump)=YFunc
                Do 1367 k=1,NPD
                  XP(k)=XPOld1(k)
                  YP(k)=YPOld1(k)
 1367           continue
              Endif
 21           If ((YPV(i-1).GE.YP(N1+i-3)).AND.((i-1).NE.1)) then
                Aux3=1
                Call XYFunc(je,i,N1,Aux3,BE,AL,T1,S8,XFunc,YFunc)
                XHEnd(NumHump)=XFunc
                YHEnd(NumHump)=YFunc
              Endif
              If ((i-1).EQ.1) then
                XHEnd(NumHump)=S3
                YHEnd(NumHump)=T3
              Endif
              If (((i+1).EQ.(NPV-1)).AND.(B.GT.0.)) then
                NumDep=NumDep+1
                XDStart(NumDep)=S9
                YDStart(NumDep)=YPT9
                XDEnd(NumDep)=XHStart(NumHump)
                YDEnd(NumDep)=YHStart(NumHump)
              Endif
            Endif
          Endif
 19     continue

C       ================================================================
C       Calculation of the areas of the humps, BH. SHump is the
C       cumulative area of the humps calculated from the seaward end of
C       the profiles.
C       ================================================================
C
C       ----------
C       Calculation of BH1 & BH2. BH1 is the area between Vellinga's
C       profile & the surge level at the location of a hump. If it is
C       the last hump & the area C>0, then BH1 is the area as defined
C       previously plus the area C. BH2 is the area between the
C       nourished profile & the surge level at the location of a hump.
C       BH is the difference between BH1 & BH2.
C       ----------
 2950   If (NumHump.GE.1) then
          SHump(0)=0.
          Do 27 j=1,NumHump
            If ((C.GT.0.).AND.(j.EQ.NumHump)) then
              BH1(j)=((2.*0.4714*BE*AL/3.)*
     1             ((((XHStart(j)-S8(je))/AL+18)**(3./2.))-
```

```
     1                (18**(3./2.)))-
     1                (2.*BE*(XHStart(j)-S8(je))))+C
              Aux4=0
              H0=0
              Do 32 i=NPD,1,-1
                If (XP(i).LE.XHStart(j)) then
                  H1=i
                  If (XP(i).EQ.XHStart(j)) H0=1
                  goto 33
                Endif
 32           continue
 33           H2=0
              Do 34 i=1,NPD
                If (XP(i).GE.S1) then
                  H3=i
                  If (XP(i).EQ.S1) H2=1
                  goto 35
                Endif
 34           continue
 35           BH2(j)=0.
              If ((H0.NE.1).AND.(XP(H1).GE.S1)) then
                If ((YPT9.GE.T9).AND.(j.EQ.1)) then
                  BH2(j)=BH2(j)+0.5*(XHStart(j)-XP(H1))*
     1                          ((T1-YHStart(j))+(T1-YP(H1)))
                  goto 36
                Endif
                BH2(j)=BH2(j)+0.5*(XHStart(j)-XP(H1))*
     1                          ((T1-YHStart(j))+(T1-YP(H1)))
 36             Aux4=1
              Endif
              If ((H2.NE.1).AND.(XP(H3).LE.XHStart(j))) then
                BH2(j)=BH2(j)+0.5*(XP(H3)-S1)*(T1-YP(H3))
                Aux4=1
              Endif
              BH1(j)=(2.*0.4714*BE*AL/3.)*
     1               ((((XHStart(j)-S8(je))/AL+18)**(3./2.))-
     1               (((XHEnd(j)-S8(je))/AL+18)**(3./2.)))+
     1               2.*BE*(XHEnd(j)-XHStart(j))
              Aux4=1
              H0=0
              Do 28 i=NPD,1,-1
                If (XP(i).LE.XHStart(j)) then
                  H1=i
                  If (XP(i).EQ.XHStart(j)) H0=1
                  goto 29
                Endif
 28           continue
 29           H2=0
              Do 30 i=1,NPD
                If (XP(i).GE.XHEnd(j)) then
                  H3=i
                  If (XP(i).EQ.XHEnd(j)) H2=1
                  goto 31
                Endif
 30           continue
 31           BH2(j)=0.
              If ((H0.NE.1).AND.(H1.GE.H3))
     1          BH2(j)=BH2(j)+0.5*(XHStart(j)-XP(H1))*
     1                          ((T1-YHStart(j))+(T1-YP(H1)))
              If ((H2.NE.1).AND.(H1.GE.H3))
     1          BH2(j)=BH2(j)+0.5*(XP(H3)-XHEnd(j))*
     1                          ((T1-YP(H3))+(T1-YHEnd(j)))
              Endif
              If (H1.GT.H3) then
                Aux4=1
                Do 37 i=H3,(H1-1)
                  BH2(j)=BH2(j)+0.5*(XP(i+1)-XP(i))*
     1                          ((T1-YP(i))+(T1-YP(i+1)))
 37             continue
              Endif
              If (Aux4.EQ.0) BH2(j)=0.5*(XHStart(j)-S1)*(T1-YHStart(j))
              If (Aux4.EQ.1.AND.H1.LT.H3) BH2(j)=0.5*(XHStart(j)-
     1             XHEnd(j))*((T1-YHStart(j))+(T1-YHEnd(j)))
              BH(j)=BH1(j)-BH2(j)
              SHump(j)=SHump(j-1)+BH(j)
 27       continue
        Endif

C       ================================================================
C       Calculation of the areas of the depressions, BD. SDep is the
C       cumulative area of the depressions calculated from the seaward
C       end of the profiles.
C       ================================================================
```

```
C
C         ----------
C         Calculation of BD1 & BD2. BD1 is the area between Vellinga's
C         profile & the surge level at the location of a depression. If it
C         is the first depression & the area B>0, then BD1 is the area as
C         defined previously plus the area B. BD2 is the area between the
C         nourished profile & the surge level at the location of a
C         depression. If it is the first depression & the area B>0, then
C         BD2 is the area as defined previously plus the area E. BD is the
C         difference between BD2 & BD1.
C         ----------
          If (NumDep.GE.1) then
            SDep(0)=0.
            Do 39 j=1,NumDep
              If ((YPT9.LT.T9).AND.(j.EQ.1)) then
                BD1(j)=((2.*0.4714*BE*AL/3.)*
     1                 ((((XDStart(j)-S8(je))/AL+18)**(3./2.))-
     1                 (((XDEnd(j)-S8(je))/AL+18)**(3./2.)))-
     1                 (2.*BE*(XDStart(j)-XDEnd(j))))+B
                Aux4=0
                D0=0
                Do 44 i=NPD,1,-1
                  If (XP(i).LE.XDStart(j)) then
                    D1=i
                    If (XP(i).EQ.XDStart(j)) D0=1
                    goto 45
                  Endif
 44             continue
 45             D2=0
                Do 46 i=1,NPD
                  If (XP(i).GE.XDEnd(j)) then
                    D3=i
                    If (XP(i).EQ.XDEnd(j)) D2=1
                    goto 47
                  Endif
 46             continue
 47             BD2(j)=0.
                If ((D0.NE.1).AND.(D1.GE.D3)) then
                  BD2(j)=BD2(j)+0.5*(XDStart(j)-XP(D1))*
     1                     ((T1-YDStart(j))+(T1-YP(D1)))
                  Aux4=1
                Endif
                If ((D2.NE.1).AND.(D1.GE.D3)) then
                  BD2(j)=BD2(j)+0.5*(XP(D3)-XDEnd(j))*((T1-YP(D3))+
     1                     (T1-YDEnd(j)))
                  Aux4=1
                Endif
                BD2(j)=BD2(j)+E
              else
                BD1(j)=(2.*0.4714*BE*AL/3.)*
     1                 ((((XDStart(j)-S8(je))/AL+18)**(3./2.))-
     1                 (((XDEnd(j)-S8(je))/AL+18)**(3./2.)))-
     1                 (2.*BE*(XDStart(j)-XDEnd(j)))
                Aux4=1
                D0=0
                Do 40 i=NPD,1,-1
                  If (XP(i).LE.XDStart(j)) then
                    D1=i
                    If (XP(i).EQ.XDStart(j)) D0=1
                    goto 41
                  Endif
 40             continue
 41             D2=0
                Do 42 i=1,NPD
                  If (XP(i).GE.XDEnd(j)) then
                    D3=i
                    If (XP(i).EQ.XDEnd(j)) D2=1
                    goto 43
                  Endif
 42             continue
 43             BD2(j)=0.
                If (D0.NE.1) BD2(j)=BD2(j)+0.5*(XDStart(j)-XP(D1))*
     1                     ((T1-YDStart(j))+(T1-YP(D1)))
                If (D2.NE.1) BD2(j)=BD2(j)+0.5*(XP(D3)-XDEnd(j))*
     1                     ((T1-YP(D3))+(T1-YDEnd(j)))
              Endif
              If (D1.GT.D3) then
                Aux4=1
                Do 48 i=D3,(D1-1)
                  BD2(j)=BD2(j)+0.5*(XP(i+1)-XP(i))*
     1                     ((T1-YP(i))+(T1-YP(i+1)))
 48             continue
              Endif
```

```
            If (Aux4.EQ.0) BD2(j)=BD2(j)+0.5*(XDStart(j)-XDEnd(j))*
       1                              ((T1-YDStart(j))+(T1-YDEnd(j)))
            BD(j)=BD2(j)-BD1(j)
            SDep(j)=SDep(j-1)+BD(j)
 39       continue
        Endif

C       ================================================================
C       Since sand movements are only allowed seaward during the storm
C       surge, the first hump & its area have to be neglected for the
C       case B=0 & the number of humps, the areas BH & SHump have to be
C       adjusted.
C       ================================================================
        If (B.EQ.0.) then
          NumHump=NumHump-1
          If (NumHump.GE.1) then
            Do 50 j=1,NumHump
              SHump(j)=SHump(j+1)-BH(1)
 50         continue
            Do 51 j=1,NumHump
              BH(j)=BH(j+1)
 51         continue
          Endif
        Endif
        return
        End




C       ################################################################
C
        Subroutine Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
C
C       ################################################################
C
C       If |Err|>=1 then the balance between erosion & accretion is
C       not satisfactory (assuming movements of sand only seaward
C       during the storm surge).
C       If Err>1, then accretion exceeds erosion (Vellinga's profile
C       has to be moved landward); if Err<-1, then erosion exceeds
C       accretion (Vellinga's profile has to be moved seaward, if S8
C       not equal to S1; if S8=S1, no erosion is expected). If
C       Vellinga's profile is moved, the new X-coordinate of the
C       starting point of the parabolic part of the profile, S8, is
C       returned.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       T2 - Y-coordinate of the intersection point between the
C            nourished profile & the gradient 1:mt of Vellinga's profile
C       T3 - Y-coordinate of the intersection point between the
C            nourished profile & the gradient 1:md of Vellinga's profile
C       Err - Error in the balance between erosion & accretion (assuming
C             movements of sand only seaward during the storm surge)
C
C       INPUT/OUTPUT VARIABLES:
C       je - Number of the iteration to find the final position of
C            Vellinga's profile
C       S8 - X-coordinate of the starting point of the parabolic part of
C            Vellinga's post-storm profile
C       Mov - Value by which S8 is changed in each iteration performed
C             to find the final position of Vellinga's parabolic profile
C       Q1, Sin, Aux1 - Auxiliary variables
C
C       MODELING VARIABLES:
C       ka - Maximum number of iterations allowed by the program to find
C            the final position of Vellinga's profile & the final
C            location of the surcharge face
C       DS - Auxiliary variables
C
C       OUTPUT VARIABLE:
C       Fim - Auxiliary variable
C
C       ################################################################
C
        Integer*4 je,ka,Aux1,Q1,Fim
        Parameter (ka=999)
        Real*8 Err(0:ka),Sin(0:ka),T2,T3,DS,S8(0:Ka),Mov

        If (Err(je).LT.0.) then
          Sin(je)=1.
        else
```

```
            Sin(je)=-1.
         Endif
         If (ABS(Err(je)).GE.1.) then
           DS=Err(je)/(T3-T2)
           je=je+1
           If (je.GT.999) then
             Write(*,9987)
             Write(50,9987)
 9987        Format(// 6X,
     1             'ERROR: The maximum number of iterations allowed by' /
     1             6X,'        the program to find the final position of'/
     1             6X,'        Vellinga`s profile has been exceeded !')
             STOP
           Endif
           S8(je)=S8(je-1)-DS
           If ((ABS(S8(je)-S8(je-2)).LT.0.001).OR.(Aux1.EQ.1)) then
             If (Sin(je-1).NE.Sin(je-2)) then
               Mov=Mov/10.
             Endif
             S8(je)=S8(je-1)+Sin(je-1)*Mov
             Aux1=1
           Endif
           Q1=1
           Fim=1
         else
           If (Q1.EQ.0) then
             Fim=3
           else
             Fim=2
           Endif
         Endif
         return
         End




C       ################################################################
C
        Subroutine Error(Q1,BH,BD,NumDep,NumHump,SDep,SHump,Fim,je,Err,
     1                   T2,T3,S8,Mov,Sin,Aux1)
C
C       ################################################################
C
C       Returns Err. Err is the error in the balance between erosion &
C       accretion (assuming movements of sand only seaward during the
C       storm surge). Then, subroutine Balance is called to move
C       Vellinga's profile depending on the value of Err.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       T2 - Y-coordinate of the intersection point between the
C            nourished profile & the gradient 1:mt of Vellinga's profile
C       T3 - Y-coordinate of the intersection point between the
C            nourished profile & the gradient 1:md of Vellinga's profile
C
C       INPUT/OUTPUT VARIABLES:
C       je - Number of the iteration to find the final position of
C            Vellinga's profile
C       S8 - X-coordinate of the starting point of the parabolic part of
C            Vellinga's post-storm profile
C       Mov - Value by which S8 is changed in each iteration performed
C             to find the final position of Vellinga's parabolic profile
C       NumHump - Number of humps
C       NumDep - Number of depressions
C       BH - Area of a hump
C       BD - Area of a depression
C       SDep - Cumulative area of the depressions starting from the
C              seaward end of the profiles
C       SHump - Cumulative area of the humps starting from the seaward
C               end of the profiles
C       Q1, Sin, Aux1 - Auxiliary variables
C
C       MODELING VARIABLES:
C       ka - Maximum number of iterations allowed by the program to find
C            the final position of Vellinga's profile & the final
C            location of the surcharge face
C       Er - Maximum number of points (XP,YP) allowed by the program
C       SHumpOld - Cumulative area of the humps starting from the
C                  seaward end of the profiles
C       SDepOld - Cumulative area of the depressions starting from the
C                 seaward end of the profiles
C       i, j, k - Auxiliary variables
```

```
C
C        OUTPUT VARIABLES:
C        Err - Error in the balance between erosion & accretion (assuming
C              movements of sand only seaward during the storm surge)
C        Fim - Auxiliary variable
C
C        #############################################################
C
         Integer*4 i,j,je,k,ka,Er,Q1,NumDep,NumHump,Fim,Aux1
         Parameter (Er=100)
         Parameter (ka=999)
         Real*8 SHump(0:Er),SDep(0:Er),SHumpOld(Er),SDepOld(Er),BH(Er),
     1          BD(Er),Err(0:ka),T2,T3,S8(0:ka),Mov,Sin(0:ka)

         Fim=0
         i=1
 15      If (ABS(BH(i)).GE.ABS(BD(i))) then
           If (i.NE.NumDep) then
             NumHump=NumHump-1
             NumDep=NumDep-1
             Do 2399 j=1,NumDep
               SDep(j)=SDep(j+1)-BD(1)
 2399        continue
             Do 2299 j=1,NumDep
               BD(j)=BD(j+1)
 2299        continue
             If (NumHump.GE.1) then
               Do 9522 j=1,NumHump
                 SHump(j)=SHump(j+1)-BH(1)
 9522          continue
               Do 9922 j=1,NumHump
                 BH(j)=BH(j+1)
 9922          continue
               goto 15
             else
               Err(je)=BD(NumDep)
               Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
               If (Fim.NE.0) return
             Endif
           else
             Fim=0
             If (Q1.EQ.0) then
               Fim=3
             Endif
             If (Q1.GE.1) then
               Err(je)=BD(i)-BH(i)
               Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
             Endif
             If (Fim.NE.0) return
           Endif
         Endif

         If (ABS(BH(i)).LT.ABS(BD(i))) then
           If (i.EQ.NumDep) then
             Fim=0
             If (Q1.EQ.0) then
               Fim=3
             Endif
             If (Q1.GE.1) then
               Err(je)=BD(i)-BH(i)
               Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
             Endif
             If (Fim.NE.0) return
           else
             Do 108 j=(i+1),NumDep
               If (j.NE.NumDep) then
                 If (ABS(SHump(j)).GE.ABS(SDep(j))) then
                   SDepOld(j)=SDep(j)
                   SHumpOld(j)=SHump(j)
                   NumHump=NumHump-j
                   NumDep=NumDep-j
                   Do 2033 k=1,NumDep
                     BD(k)=BD(j+k)
                     SDep(k)=SDep(j+k)-SDepOld(j)
 2033              continue
                   If (NumHump.GE.1) then
                     Do 3320 k=1,NumHump
                       BH(k)=BH(j+k)
                       SHump(k)=SHump(j+k)-SHumpOld(j)
 3320                continue
                     goto 15
                   else
                     Err(je)=BD(NumDep)
```

```
                    Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
                    If (Fim.NE.0) return
                  Endif
                else
                  If (NumHump.GT.j) then
                    goto 108
                  else
                    Err(je)=SDep(NumDep)-SHump(NumHump)
                    Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
                    If (Fim.NE.0) return
                  Endif
                Endif
              else
                If (NumHump.EQ.NumDep) then
                  Err(je)=SDep(j)-SHump(j)
                  Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
                  If (Fim.NE.0) return
                else
                  Err(je)=SDep(j)-SHump(j-1)
                  Call Balance(je,Err,T2,T3,S8,Q1,Fim,Mov,Sin,Aux1)
                  If (Fim.NE.0) return
                Endif
              Endif
 108        continue
          Endif
        Endif
      return
      End




C       ################################################################
C
        Subroutine SurchC(ctcurv,BE,W,S8,T1,S3,S4,T4,S10,SurD,SurchEros,
     1                    SurchLongT,TSurch)
C
C       ################################################################
C
C       Returns the total surcharge on erosion area C, TSurch, & the
C       corresponding surcharge distance, SurD, & the intersection
C       point, (S4,T4), between the nourished profile & the surcharge
C       gradient, 1:md (through subroutine Surcharge). It also returns
C       the surcharge, SurchEros, on erosion area C above surge level to
C       take into account the effects of the accuracy of the
C       computation, Ac, of the storm surge duration, SD, & of the gust
C       bumps, GB, respectively. It also returns the surcharge,
C       SurchLongT, on erosion area C above surge level to take into
C       account the effect of a gradient in the longshore transport.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       MuxN - Mean of the equivalent Normal distribution of X
C       Ac - Accuracy of the computation
C       GB - Gust bumps
C       SD - Storm surge duration
C       W - Fall velocity of dune sand in seawater (m/s)
C       ctcurv - Coastal curvature in degrees per 1000m
C               (0<=ctcurv<=24Deg/1000m)
C       T1 - Y-coordinate of the intersection point between the
C            nourished profile & the surge level
C       S8 - X-coordinate of the starting point of the parabolic part of
C            Vellinga's post-storm profile
C       (S3,T3) - Intersection point between the nourished profile & the
C                 gradient 1:md of Vellinga's profile
C       C - Area of erosion between the surge level, the nourished
C           profile above surge & the gradient 1:md
C       BE - Auxiliary variable
C
C       MODELING VARIABLES:
C       ka - Maximum number of iterations allowed by the program to find
C            the final position of Vellinga's profile & the final
C            location of the surcharge face
C       j - Number of the iteration to find the final location of the
C           surcharge face
C       G0 - Coefficient used in the calculation of SurchLongT
C       CTotal - Sum of the erosion area C plus the surcharge SurchEros
C       D - Area between the two 1:md gradients, the surge level & the
C           nourished profile
C       Err1 - Error in the balance between the required surcharge,
C              TSurch, & D
C       Jump - Value by which SurD is changed in each iteration
C              performed to find the final location of the surcharge
```

```
C                 face
C         Q2, Aux2, Sinal - Auxiliary variables
C
C         OUTPUT VARIABLES:
C         (S4,T4) - Intersection point between the nourished profile & the
C                    surcharge gradient, 1:md
C         S10 - X-coordinate of the point of intersection between the
C               surge level & the gradient 1:md of the surcharge
C         SurchEros - Surcharge on erosion area C above surge level to
C                     take into account the effects of the accuracy of
C                     the computation, of the storm surge duration, & of
C                     the gust bumps
C         SurchLongT - Surcharge on erosion area C above surge level to
C                      take into account the effect of a gradient in the
C                      longshore transport
C         TSurch - Total surcharge on erosion area C which is the sum of
C                  the surcharges SurchEros plus SurchLongT
C         SurD - Surcharge distance
C
C         #############################################################
C
          Integer*4 ka,j,Q2,Aux2
          Parameter (ka=999)
          Real*8 W,BE,T1,S8(0:Ka),C,S3,T3,SurchEros,MuxN,SurD(0:ka),SD,Ac,
     1          GB,S4,T4,D,S10(0:ka),Err1(0:ka),Jump,Sinal(0:ka),ctcurv,
     1          CTotal,SurchLongT,TSurch,G0
          Common/BLOCK3/MuxN(15),C,T3
          Common/BLOCK5/Ac,GB,SD

C         =================================================================
C         Calculation of the surcharge, SurchEros, on erosion area C above
C         surge level to take into account the effects of the accuracy of
C         the computation, Ac, of the storm surge duration, SD, & of the
C         gust bumps, GB, respectively.
C         =================================================================
          SurchEros=(20.+0.1*C)*Ac+0.1*C*SD+0.05*C*GB/MuxN(5)

C         =================================================================
C         Calculation of the surcharge, SurchLongT, on erosion area C
C         above surge level to take into account the effect of a gradient
C         in the longshore transport for not too strongly curved coastal
C         sections (the coastal curvature, 0<=ctcurv<=24Deg/1000m).
C         =================================================================
          If ((ctcurv.LE.24.).AND.(ctcurv.GT.6.)) then
            If ((ctcurv.LE.24.).AND.(ctcurv.GT.18.)) G0=100.
            If ((ctcurv.LE.18.).AND.(ctcurv.GT.12.)) G0=75.
            If ((ctcurv.LE.12.).AND.(ctcurv.GT.6.)) G0=50.
            CTotal=C+SurchEros
            SurchLongT=CTotal*(BE**0.72)*((W/0.0268)**0.56)*G0/300.
          else
            SurchLongT=0.
          Endif

C         =================================================================
C         Calculation of the total surcharge on erosion area C, TSurch,
C         which is the sum of the surcharges SurchEros plus SurchLongT.
C         =================================================================
          TSurch=SurchEros+SurchLongT

C         =================================================================
C         Calculation of the surcharge distance, SurD.
C         =================================================================
          Jump=10.
          Sinal(0)=0.
          Aux2=0
          Q2=1
          j=1
          S10(0)=S8(1)
          SurD(0)=0.
          Err1(0)=0.
          If (TSurch.LT.0.) Q2=-1

C         ----------
C         Calculation of a first approximation for SurD. Calculation of
C         the point, X=S10, of intersection between the surge level &
C         the gradient 1:md of the surcharge.
C         ----------
          SurD(j)=TSurch/(T3-T1)
 2319     S10(j)=S8(1)-SurD(j)
          If ((S10(j).GT.S8(1)).AND.(TSurch.GT.0.)) then
            S10(j)=(S10(j-1)+S10(j-2))/2.
            SurD(j)=S8(1)-S10(j)
            Aux2=1
```

```
              Endif
              If ((S10(j).LT.S8(1)).AND.(TSurch.LT.0.)) then
                S10(j)=(S10(j-1)+S10(j-2))/2.
                SurD(j)=S8(1)-S10(j)
                Aux2=1
              Endif

C        ----------
C        Calculation of the surcharge erosion, D, corresponding to SurD.
C        ----------
              Call Surcharge(j,S8,S10,T1,D,S3,T3,S4,T4,TSurch)

C        ----------
C        Calculation of Err1. Err1 is the error in the balance between
C        the required surcharge, TSurch, & D. If |Err1|>=1 then the
C        balance between TSurch & D is not satisfactory. If Err1>1
C        then the surcharge distance is too big (decrease it). If
C        Err1<-1 then the surcharge distance is too small (increase
C        it). If the face of the surcharge is moved, the new surcharge
C        distance, SurD, is returned.
C        ----------
              Err1(j)=D-ABS(TSurch)
              If (Err1(j).LT.0.) then
                Sinal(j)=-1.
              else
                Sinal(j)=1.
              Endif
              If (ABS(Err1(j)).GE.1.) then
                j=j+1
                If (j.GT.999) then
                  Write(*,9987)
                  Write(50,9987)
 9987             Format(// 6X,
     1            'ERROR: The maximum number of iterations allowed by'
     1            / 6X,'      the program to find the final location of the'
     1            / 6X,'      surcharge face has been exceeded !')
                  STOP
                Endif
                SurD(j)=SurD(j-1)-Err1(j-1)*Q2/(T4-T1)
                If ((ABS(SurD(j)-SurD(j-2)).LT.0.0001).OR.(Aux2.EQ.1)) then
                  If (j.GT.1) then
                    If (Sinal(j-1).NE.Sinal(j-2)) then
                      Jump=Jump/10.
                    Endif
                    SurD(j)=SurD(j-1)-Sinal(j-1)*Jump
                  Endif
                  Aux2=1
                Endif
                goto 2319
              Endif
              return
              End




C        ################################################################
C
         Subroutine LSeaProf(S9,T9,S1,M1,T1,B,Q,S2,T2)
C
C        ################################################################
C
C        Returns the intersection point, (S2,T2), between the nourished
C        profile & the gradient 1:mt of Vellinga's profile. It also
C        returns the area, B, between the surge level & the gradient
C        1:mt; & the area, Q, between the surge level & the nourished
C        profile below surge.
C
C        ################################################################
C
C        INPUT VARIABLES:
C        NPD - Number of points defining the nourished profile
C        (XP,YP) - Coordinates of the points defining the nourished
C                  profile
C        1:mt - Gradient of the toe of the post-storm profile
C        (S1,T1) - Intersection point between the nourished profile & the
C                  surge level
C        M1 - Number of the point, (XP(M1),YP(M1)), from seaward for
C             which YP(M1)<=T1
C        (S9,T9) - Point where the parabolic part of Vellinga's profile
C                  finishes
C
C        MODELING VARIABLES:
C        YPT9 - Y-coordinate of the point in the nourished profile which
```

```
C              has X=S9
C        i, M2, Zes, G - Auxiliary variables
C        md, nourtlev, mnour, ctcurv - Variables mentioned in the Common
C                                    statements but not used here
C
C        OUTPUT VARIABLES:
C        (S2,T2) - Intersection point between the nourished profile & the
C                  gradient 1:mt of Vellinga's profile
C        B - Area between the surge level & the gradient 1:mt of
C            Vellinga's profile
C        Q - Area between the surge level & the nourished profile below
C            surge
C
C        ################################################################
C
         Integer*4 NPD,i,M1,M2
         Real*8 XP,YP,S1,T1,S2,T2,S9,T9,B,Q,Zes,G,md,mt,mnour,
     1         nourtlev,YPT9,ctcurv
         Common/BLOCK1/NPD,XP(100),YP(100)
         Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv

C        ================================================================
C        Calculation of the intersection point, (S2,T2), between the
C        nourished profile & the gradient 1:mt of Vellinga's profile.
C        ================================================================
         Do 4499 i=1,NPD
           If (XP(i).GT.S9) then
             M2=i
             YPT9=YP(M2-1)+(YP(M2)-YP(M2-1))*(S9-XP(M2-1))/
     1           (XP(M2)-XP(M2-1))
             If (YPT9.GE.T9) then
               S2=S9
               T2=YPT9
               B=0.
               M2=M2-1
               goto 3902
             else
               goto 4398
             Endif
           Endif
 4499    continue
         Write(*,8275)
         Write(50,8275)
 8275    Format(// 11X,'Extend the Initial Profile Seaward !')
         STOP
 4398    Zes=XP(M2)-S9
         G=T9-Zes/mt
         If (G.GT.YP(M2)) then
           M2=M2+1
           If (M2.GT.NPD) then
             Write(*,8225)
             Write(50,8225)
 8225        Format(// 11X,'Extend the Initial Profile Seaward !')
             STOP
           Endif
           goto 4398
         Endif
         S2=XP(M2)-Zes*(YP(M2)-G)/(T9-G-(Zes*(YP(M2-1)-YP(M2))/
     1     (XP(M2)-XP(M2-1))))
         T2=YP(M2)+(XP(M2)-S2)*(YP(M2-1)-YP(M2))/(XP(M2)-XP(M2-1))
         M2=M2-1

C        ================================================================
C        Calculation of the area, B, between the surge level & the
C        gradient 1:mt.
C        ================================================================
         B=0.5*(S2-S9)*((T1-T9)+(T1-T2))

C        ================================================================
C        Calculation of area, Q, between the surge level & the nourished
C        profile below surge.
C        ================================================================
 3902    If ((XP(M2).GT.S1).AND.(XP(M1).LT.S2)) then
           Q=0.5*(XP(M1)-S1)*(T1-YP(M1))+0.5*(S2-XP(M2))*
     1       ((T1-T2)+(T1-YP(M2)))
         else
           Q=0.5*(S2-S1)*(T1-T2)
         Endif
         If ((M2-1).GE.M1) then
           Do 3381 i=M1,(M2-1)
             Q=Q+0.5*(XP(i+1)-XP(i))*((T1-YP(i))+(T1-YP(i+1)))
 3381      continue
         Endif
```

```
                return
                End



C         ################################################################
C
C          Subroutine XYFunc(i,N1,Aux3,XPV,YPV,XFunc,YFunc)
          Subroutine XYFunc(je,i,N1,Aux3,BE,AL,T1,S8,XFunc,YFunc)
C
C         ################################################################
C
C         Returns the intersection point (XFunc,YFunc) between the part of
C         Vellinga's profile defined by the points:
C                         (XPV(i-Aux3),YPV(i-Aux3))
C                         (XPV(i-Aux3+1),YPV(i-Aux3+1))
C         & the part of the nourished profile defined by the points:
C                         (XP(N1+i-Aux3-1),YP(N1+i-Aux3-1))
C                         (XP(N1+i-Aux3-2),YP(N1+i-Aux3-2))
C
C         Returns the intersection point (XFunc,YFunc) between the
C         parabolic part of Vellinga's profile & the part of the nourished
C         profile defined by the points:
C                         (XP(N1+i-Aux3-1),YP(N1+i-Aux3-1))
C                         (XP(N1+i-Aux3-2),YP(N1+i-Aux3-2))
C
C         ################################################################
C
C         INPUT VARIABLES:
C         (XP,YP) - Coordinates of the points defining the nourished
C                   profile
C         (XPV,YPV) - Coordinates of the points defining Vellinga's
C                     profile
C         T1 - Y-coordinate of the intersection point between the
C              nourished profile & the surge level
C         S8 - X-coordinate of the starting point of the parabolic part of
C              Vellinga's post-storm profile
C         i, je, N1, Aux2, Aux3, BE, AL - Auxiliary variables
C
C         MODELING VARIABLES:
C         L - Maximum number of points (XP,YP) & (XPV,YPV) allowed by the
C             program
C         ka - Maximum number of iterations allowed by the program to find
C              the final position of Vellinga's profile & the final
C              location of the surcharge face
C         A, B, C, Aux1,
C         XFunc1, XFunc2, YFunc1, YFunc2 - Auxiliary variables
C         NPD - Variable mentioned in the Common statement but not used
C                here
C
C         OUTPUT VARIABLES:
C         (XFUNC,YFUNC) - Coordinates of an intersection point between the
C                         nourished profile & Vellinga's profile
C
C         ################################################################
C
          Integer*4 i,N1,Aux3,NPD,L,ka,je
          Parameter (L=100)
          Parameter (ka=999)
          Real*8  XP,YP,XFunc,YFunc,XFunc1,YFunc1,XFunc2,YFunc2,A,B,C,AL,
     1          BE,S8(0:Ka),T1,Aux1,Aux2
          Common/BLOCK1/NPD,XP(100),YP(100)


            A=-((YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2))**2)/
     1        ((0.4714*BE*(XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2)))**2)
            If (A.EQ.0.) then
              YFunc=YP(N1+i-Aux3-1)
              XFunc=S8(je)+AL*(((T1-YFunc+2.*BE)/(0.4714*BE))**2)-18.*AL
              return
            Endif
            B=(AL**(-1))+
     1        ((2.*T1*(YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2))-
     1        2.*(YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2))*YP(N1+i-Aux3-2))/
     1        ((XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2))*((0.4714*BE)**2)))+
     1        ((4.*(YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2)))/
     1        (BE*(0.4714**2)*(XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2))))+
     1        ((2.*XP(N1+i-Aux3-2)*((YP(N1+i-Aux3-1)-
     1        YP(N1+i-Aux3-2))**2))/(((0.4714*BE)**2)*((XP(N1+i-Aux3-1)-
     1        XP(N1+i-Aux3-2))**2)))
            C=-(4./(0.4714**2))-
     1        (4.*T1-4.*YP(N1+i-Aux3-2))/(BE*(0.4714**2))-
     1        (T1*T1-2.*T1*YP(N1+i-Aux3-2)+((YP(N1+i-Aux3-2))**2))/
```

```
      1         ((0.4714*BE)**2)-
      1         S8(je)*(AL**(-1))+18.-
      1         (4.*XP(N1+i-Aux3-2)*(YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2)))/
      1         (BE*(0.4714**2)*(XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2)))-
      1         ((XP(N1+i-Aux3-2)**2)*((YP(N1+i-Aux3-1)
      1         -YP(N1+i-Aux3-2))**2))/((((0.4714*BE)**2)*((XP(N1+i-Aux3-1)-
      1         XP(N1+i-Aux3-2))**2))-
      1         (2.*T1*XP(N1+i-Aux3-2)*(YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2))-
      1         2.*XP(N1+i-Aux3-2)*YP(N1+i-Aux3-2)*(YP(N1+i-Aux3-1)-
      1         YP(N1+i-Aux3-2)))/((XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2))*
      1         ((0.4714*BE)**2))
            If ((B*B-4.*A*C).LT.0.) then
              Write(*,9997)
              Write(50,9997)
 9997         Format(// 6X,'ERROR: (B^2-4AC) negative in the quadratic'
      1                 / 6X,'        equation AX^2+BX+C=0 ! Roots of the'
      1                 / 6X,'        equation are complex conjugate! ')
              STOP
            Endif
            Aux2=1.
            Aux1=-0.5*(B+DSIGN(Aux2,B)*SQRT(B*B-4.*A*C))
            XFunc1=Aux1/A
            XFunc2=C/Aux1
            If (XFunc1.GT.XP(N1+i-Aux3-2).AND.XFunc1.LT.
      1         XP(N1+i-Aux3-1).AND.XFunc1.GT.S8(je)) then
              YFunc1=YP(N1+i-Aux3-2)+(XFunc1-XP(N1+i-Aux3-2))*
      1             (YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2))/
      1             (XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2))
              If ((YFunc1.GT.YP(N1+i-Aux3-1).AND.
      1           YFunc1.LT.YP(N1+i-Aux3-2).AND.YFunc1.LE.T1).OR.
      1           (YFunc1.LT.YP(N1+i-Aux3-1).AND.
      1           YFunc1.GT.YP(N1+i-Aux3-2).AND.YFunc1.LE.T1)) then
                XFunc=XFunc1
                YFunc=YFunc1
              Endif
            Endif
            If (XFunc2.GT.XP(N1+i-Aux3-2).AND.
      1         XFunc2.LT.XP(N1+i-Aux3-1).AND.XFunc2.GT.S8(je)) then
              YFunc2=YP(N1+i-Aux3-2)+(XFunc2-XP(N1+i-Aux3-2))*
      1             (YP(N1+i-Aux3-1)-YP(N1+i-Aux3-2))/
      1             (XP(N1+i-Aux3-1)-XP(N1+i-Aux3-2))
              If ((YFunc2.GT.YP(N1+i-Aux3-1).AND.
      1           YFunc2.LT.YP(N1+i-Aux3-2).AND.YFunc2.LE.T1).OR.
      1           (YFunc2.LT.YP(N1+i-Aux3-1).AND.
      1           YFunc2.GT.YP(N1+i-Aux3-2).AND.YFunc2.LE.T1)) then
                XFunc=XFunc2
                YFunc=YFunc2
              Endif
            Endif
          return
          End




C       ################################################################
C
        Subroutine Surcharge(j,S8,S10,T1,D,S3,T3,S4,T4,TSurch)
C
C       ################################################################
C
C       Returns the intersection point, (S4,T4), between the nourished
C       profile & the surcharge gradient, 1:md.  It also returns the
C       area D between the two 1:md gradients, the surge level & the
C       nourished profile.
C
C       ################################################################
C
C       INPUT VARIABLES:
C       j - Number of the iteration to find the final location of the
C           surcharge face
C       NPD - Number of points defining the nourished profile
C       (XP,YP) - Coordinates of the points defining the nourished
C                 profile
C       T1 - Y-coordinate of the intersection point between the
C            nourished profile & the surge level
C       1:md - Gradient of the eroded dune face
C       S8 - X-coordinate of the starting point of the parabolic part of
C            Vellinga's post-storm profile
C       (S3,T3) - Intersection point between the nourished profile & the
C                 gradient 1:md of Vellinga's profile
C       S10 - X-coordinate of the point of intersection between the
C             surge level & the gradient 1:md of the surcharge
C       TSurch - Total surcharge on erosion area C which is the sum of
```

```
C                    the surcharges SurchEros plus SurchLongT
C
C        MODELING VARIABLES:
C        ka - Maximum number of iterations allowed by the program to find
C             the final position of Vellinga's profile & the final
C             location of the surcharge face
C        i, M3, M31, U, V - Auxiliary variables
C        mt, nourtlev, mnour, ctcurv - Variables mentioned in the Common
C                                      statements but not used here
C
C        OUTPUT VARIABLES:
C        (S4,T4) - Intersection point between the nourished profile & the
C                  surcharge gradient, 1:md
C        D - Area between the two 1:md gradients, the surge level & the
C            nourished profile
C
C        ################################################################
C
         Integer*4 NPD,i,j,M3,M31,ka
         Parameter (ka=999)
         Real*8 D,XP,YP,T1,S3,T3,S8(0:ka),S4,T4,S10(0:ka),U,V,md,mt,
     1        mnour,nourtlev,TSurch,ctcurv
         Common/BLOCK1/NPD,XP(100),YP(100)
         Common/BLOCK4/md,mt,mnour,nourtlev,ctcurv

C        ================================================================
C        Calculation of the intersection point, (S4,T4), between the
C        nourished profile & the surcharge gradient, 1:md.
C        ================================================================
         If (S10(j).LT.XP(1)) then
           Write(*,8925)
           Write(50,8925)
 8925      Format(// 11X,'Extend the Initial Profile Landward !')
           STOP
         Endif
         Do 0903 i=1,NPD
           If (XP(i).GT.S10(j)) then
             M3=i
             goto 4977
           Endif
 0903    continue
         Write(*,8825)
         Write(50,8825)
 8825    Format(// 11X,'Extend the Initial Profile Seaward !')
         STOP
 4977    U=S10(j)-XP(M3-1)
         V=T1+U/md
         If (V.LT.YP(M3-1)) then
           M3=M3-1
           If (M3.LE.1) then
             Write(*,8625)
             Write(50,8625)
 8625        Format(// 11X,'Extend the Initial Profile Landward !')
             STOP
           Endif
           goto 4977
         Endif
         S4=XP(M3-1)+U*(V-YP(M3-1))/(V-T1-(U*(YP(M3-1)-YP(M3))/
     1     (XP(M3)-XP(M3-1))))
         If (S4.LT.S10(j)) then
           T4=YP(M3-1)-(S4-XP(M3-1))*(YP(M3-1)-YP(M3))/(XP(M3)-XP(M3-1))
         else
           D=0.
           goto 2591
         Endif

C        ================================================================
C        Calculation of area D between the two 1:md faces, the surge
C        level & the nourished profile.
C        ================================================================
C
C        ----------
C        TSurch>0
C        ----------
         If (TSurch.GT.0.) then
           Do 0309 i=1,NPD
             If (XP(i).GT.S3) then
               M31=i-1
               goto 7794
             Endif
 0309      continue
           Write(*,8695)
           Write(50,8695)
```

```
8695      Format(// 11X,'Extend the Initial Profile Seaward !')
          STOP

7794      If ((XP(M31).GT.S4).AND.(XP(M3).LT.S3)) then
             D=0.5*(S8(1)-S3)*(T3-T1)+0.5*(XP(M3)-S4)*
     1          ((T4-T1)+(YP(M3)-T1))+
     1          0.5*(S3-XP(M31))*((YP(M31)-T1)+(T3-T1))
          else
             D=0.5*(S8(1)-S3)*(T3-T1)+0.5*(S3-S4)*((T3-T1)+(T4-T1))
          Endif
          If (M31.GT.M3) then
            Do 4701 i=M3,(M31-1)
              D=D+0.5*(XP(i+1)-XP(i))*((YP(i+1)-T1)+(YP(i)-T1))
4701        continue
          Endif
          D=D-0.5*(S10(j)-S4)*(T4-T1)
        Endif

C        ----------
C        TSurch<0
C        ----------
        If (TSurch.LT.0.) then
          Do 5137 i=1,NPD
            If (XP(i).GT.S3) then
              M31=i
              goto 7549
            Endif
5137      continue
          Write(*,8629)
          Write(50,8629)
8629      Format(// 11X,'Extend the Initial Profile Seaward !')
          STOP

7549      If ((XP(M31).LT.S4).AND.(XP(M3-1).GT.S3)) then
             D=0.5*(S10(j)-S4)*(T4-T1)+0.5*(S4-XP(M3-1))*
     1          ((T4-T1)+(YP(M3-1)-T1))+0.5*(XP(M31)-S3)*
     1          ((YP(M31)-T1)+(T3-T1))
          else
             D=0.5*(S10(j)-S4)*(T4-T1)+0.5*(S4-S3)*((T4-T1)+(T3-T1))
          Endif
          If (M3.GT.M31) then
            Do 7410 i=M31,(M3-1)
              D=D+0.5*(XP(i+1)-XP(i))*((YP(i+1)-T1)+(YP(i)-T1))
7410        continue
          Endif
          D=D-0.5*(S8(1)-S3)*(T3-T1)
        Endif
2591    return
        End
```

APPENDIX D

Input Data And Results From PARASODE

APPENDIX D1 - Examples Of Input And Output Files For Wave Overtopping

Input File *general.dad*: H&R Model

```
1                    ! Failure mode of overtopping (H&R model)
2                    ! Tide + Surge
1                    ! (R_max)37%
Y                    ! Derivatives of the failure function supplied
1                    ! Mode 1 - Reliability analysis for a specified design
1                    ! Design life (in years)
N                    ! No combinations of actions considered
10                   ! Target design parameter - seawall crest level
```

Input File *form.dad:* H&R Model

```
2                    ! Starting point: user specified values
4.5
2
4
1
0.5
0.95
1.049
2                    ! [XMin,XMax]: user specified values
1       20
0       20
-5.33   5.57
-5      5
0.25    1
0.5     1
-1E25   1E25
200                  ! Maximum number of iterations
6                    ! Number of FORM calculations
1E-1
1E-2
1E-3
1E-4
1E-5
1E-6
1                    ! Accuracy on Beta (%): default value (1%)
2                    ! Smoothing of the iteration: user specified value
0.9
1                    ! Accuracy on Z0 (%): default value (1%)
```

Input File *meandev.dad*: H&R Model

```
10                              !Type of distribution (Weibull)
0                               !Not truncated
6.4     1.152   4.224           !Mean; stand. dev.; lower limit
10                              !Type of distribution (Weibull)
1                               !Truncated for X above Xo
0                               !Toe level
1.2     0.7     0.45            !Mean; stand. dev.; lower limit
0                               !Type of distribution (Deterministic)
0.00753                         !Value
0                               !Type of distribution (Deterministic)
4.17                            !Value
13                              !Type of distribution (User-Defined)
0.2749          2.3619          !Mean value; standard deviation
3                               !Type of distribution (Gumbel)
0                               !Not truncated
0.019483        0.192382475     !Mean value; standard deviation
1                               !Type of distribution (Normal)
0                               !Not truncated
0.5             0.05            !Mean value; standard deviation
6                               !Type of distribution (Beta)
0                               !Not truncated
0.95    0.01    0.9     1       !Mean; stand. dev.; lower & upper limits
2                               !Type of distribution (Log-Normal)
0                               !Not truncated
1.049           0.241           !Mean value; standard deviation
```

Input File *coefcor.dad*: H&R Model

```
1       ! Rho(1,1)
0.6     ! Rho(1,2)
0       ! Rho(1,3)
0       !   .
0       !   .
0       !   .
0
0
0
0.6
1
0
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
0
0
1
```

```
0
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
0
1
```

D1-3

Output File *summary.dat*: H&R Model

```
WHAT IS THE DATA SOURCE ?

        The Screen ..... [ 1 ]
        A Datafile ..... [ 2 ]

Select Option: 2



WHAT IS THE FAILURE MODE TO BE STUDIED:

        Overtopping (H&R) .................... [ 1 ]
        Overtopping (Owen) ................... [ 2 ]
        Dune Erosion (Vellinga) .............. [ 3 ]

Select Option:  1



HOW IS THE STILL-WATER-LEVEL DEFINED ?

        Total Level .... [ 1 ]
        Tide + Surge ... [ 2 ]

Select Option: 2



WHAT IS THE CONFIDENCE VALUE OF THE MAXIMUM RUN-UP
THAT YOU WOULD LIKE TO CONSIDER ?

        37 % ... [ 1 ]
        99 % ... [ 2 ]

Select Option: 1



ARE THE FIRST DERIVATIVES OF THE FAILURE FUNCTION SUPPLIED (Y/N) ? Y



DESCRIPTION OF THE VARIABLES

        X(  1) = Tp  = Peak Wave Period
        X(  2) = Hs  = Wave Height
        X(  3) = A   = H&R Parameter
        X(  4) = B   = H&R Parameter
        X(  5) = Tid = Tide Level
        X(  6) = Sur = Surge
        X(  7) = TAl = Seawall Slope
        X(  8) = r   = Roughness
        X(  9) = eB  = Model Parameter
```

Output File *summary.dat*: H&R Model

WHAT IS THE PURPOSE OF THE ANALYSIS ?

        Reliability Analysis for a Specified Design ... [1]
        Design for a Specified Reliability Level ...... [2]

Select Option: 1

DESIGN LIFE OF THE STRUCTURE =   1

WOULD YOU LIKE TO CONSIDER COMBINATION OF ACTIONS (Y/N) ? N

PRESCRIBED VALUE OF THE DESIGN PARAMETER

        Seawall Crest Level = 0.1000000000E+02

CHARACTERISTICS OF THE VARIABLES

        Probability Distribution of Tp = Minima Type III (Weibull)
        Mean Value of Tp = 0.6400000000E+01
        Standard Deviation of Tp = 0.1152000000E+01
        Lower Limit on Tp = 0.4224000000E+01

        Probability Distribution of Hs = Minima Type III (Weibull)
        The Distribution of Hs is truncated above Xo = 0.6(Tide+Surge-TL)
        Seawall Toe Level (TL) = 0.0000000000E+00
        Mean Value of Hs = 0.1200000000E+01
        Standard Deviation of Hs = 0.7000000000E+00
        Lower Limit on Hs = 0.4500000000E+00

        Probability Distribution of A   = Deterministic
        Mean Value of A   = 0.7530000000E-02
        Standard Deviation of A   = 0.0000000000E+00

        Probability Distribution of B   = Deterministic
        Mean Value of B   = 0.4170000000E+01
        Standard Deviation of B   = 0.0000000000E+00

        Probability Distribution of Tid = User-Defined Distribution
        Mean Value of Tid = 0.2749000000E+00
        Standard Deviation of Tid = 0.2361900000E+01

        Probability Distribution of Sur = Maxima Type I (Gumbel)
        Mean Value of Sur = 0.1948300000E-01
        Standard Deviation of Sur = 0.1923824750E+00

        Probability Distribution of TAl = Normal (Gaussian)
        Mean Value of TAl = 0.5000000000E+00
        Standard Deviation of TAl = 0.5000000000E-01

```
Probability Distribution of r   = Beta
Mean Value of r    =  0.9500000000E+00
Standard Deviation of r   =  0.1000000000E-01
Limits a and b of r    = [ 0.9000000000E+00, 0.1000000000E+01]



Probability Distribution of eB  = Log-Normal
Mean Value of eB  =  0.1049000000E+01
Standard Deviation of eB  =  0.2410000000E+00
```

```
CORRELATION COEFFICIENTS

      (Tp ,Tp ) =  0.1000000000E+01
      (Tp ,Hs ) =  0.6000000000E+00
      (Tp ,A  ) =  0.0000000000E+00
      (Tp ,B  ) =  0.0000000000E+00
      (Tp ,Tid) =  0.0000000000E+00
      (Tp ,Sur) =  0.0000000000E+00
      (Tp ,TAl) =  0.0000000000E+00
      (Tp ,r  ) =  0.0000000000E+00
      (Tp ,eB ) =  0.0000000000E+00
      (Hs ,Tp ) =  0.6000000000E+00
      (Hs ,Hs ) =  0.1000000000E+01
      (Hs ,A  ) =  0.0000000000E+00
      (Hs ,B  ) =  0.0000000000E+00
      (Hs ,Tid) =  0.0000000000E+00
      (Hs ,Sur) =  0.0000000000E+00
      (Hs ,TAl) =  0.0000000000E+00
      (Hs ,r  ) =  0.0000000000E+00
      (Hs ,eB ) =  0.0000000000E+00
      (A  ,Tp ) =  0.0000000000E+00
      (A  ,Hs ) =  0.0000000000E+00
      (A  ,A  ) =  0.1000000000E+01
      (A  ,B  ) =  0.0000000000E+00
      (A  ,Tid) =  0.0000000000E+00
      (A  ,Sur) =  0.0000000000E+00
      (A  ,TAl) =  0.0000000000E+00
      (A  ,r  ) =  0.0000000000E+00
      (A  ,eB ) =  0.0000000000E+00
      (B  ,Tp ) =  0.0000000000E+00
      (B  ,Hs ) =  0.0000000000E+00
      (B  ,A  ) =  0.0000000000E+00
      (B  ,B  ) =  0.1000000000E+01
      (B  ,Tid) =  0.0000000000E+00
      (B  ,Sur) =  0.0000000000E+00
      (B  ,TAl) =  0.0000000000E+00
      (B  ,r  ) =  0.0000000000E+00
      (B  ,eB ) =  0.0000000000E+00
      (Tid,Tp ) =  0.0000000000E+00
      (Tid,Hs ) =  0.0000000000E+00
      (Tid,A  ) =  0.0000000000E+00
      (Tid,B  ) =  0.0000000000E+00
      (Tid,Tid) =  0.1000000000E+01
      (Tid,Sur) =  0.0000000000E+00
      (Tid,TAl) =  0.0000000000E+00
```

```
            (Tid,r  ) =  0.0000000000E+00
            (Tid,eB ) =  0.0000000000E+00
            (Sur,Tp ) =  0.0000000000E+00
            (Sur,Hs ) =  0.0000000000E+00
            (Sur,A  ) =  0.0000000000E+00
            (Sur,B  ) =  0.0000000000E+00
            (Sur,Tid) =  0.0000000000E+00
            (Sur,Sur) =  0.1000000000E+01
            (Sur,TAl) =  0.0000000000E+00
            (Sur,r  ) =  0.0000000000E+00
            (Sur,eB ) =  0.0000000000E+00
            (TAl,Tp ) =  0.0000000000E+00
            (TAl,Hs ) =  0.0000000000E+00
            (TAl,A  ) =  0.0000000000E+00
            (TAl,B  ) =  0.0000000000E+00
            (TAl,Tid) =  0.0000000000E+00
            (TAl,Sur) =  0.0000000000E+00
            (TAl,TAl) =  0.1000000000E+01
            (TAl,r  ) =  0.0000000000E+00
            (TAl,eB ) =  0.0000000000E+00
            (r  ,Tp ) =  0.0000000000E+00
            (r  ,Hs ) =  0.0000000000E+00
            (r  ,A  ) =  0.0000000000E+00
            (r  ,B  ) =  0.0000000000E+00
            (r  ,Tid) =  0.0000000000E+00
            (r  ,Sur) =  0.0000000000E+00
            (r  ,TAl) =  0.0000000000E+00
            (r  ,r  ) =  0.1000000000E+01
            (r  ,eB ) =  0.0000000000E+00
            (eB ,Tp ) =  0.0000000000E+00
            (eB ,Hs ) =  0.0000000000E+00
            (eB ,A  ) =  0.0000000000E+00
            (eB ,B  ) =  0.0000000000E+00
            (eB ,Tid) =  0.0000000000E+00
            (eB ,Sur) =  0.0000000000E+00
            (eB ,TAl) =  0.0000000000E+00
            (eB ,r  ) =  0.0000000000E+00
            (eB ,eB ) =  0.1000000000E+01


STARTING POINT FOR THE FORM CALCULATIONS:

        Default Values (mean values) ... [ 1 ]
        User Specified Values .......... [ 2 ]

Select Option: 2

STARTING POINT

        Tp  =  0.4500000000E+01
        Hs  =  0.2000000000E+01
        Tid =  0.4000000000E+01
        Sur =  0.1000000000E+01
        TAl =  0.5000000000E+00
        r   =  0.9500000000E+00
        eB  =  0.1049000000E+01
```

```
LIMITING VALUES FOR THE VARIABLES:


        Default Values (+/- 1E25) ... [ 1 ]
        User Specified Values ....... [ 2 ]


Select Option: 2


LIMITING VALUES FOR THE VARIABLES


        XMin(Tp ) =  0.4224000000E+01   XMax(Tp ) =  0.2000000000E+02
        XMin(Hs ) =  0.4500000000E+00   XMax(Hs ) =  0.2000000000E+02
        XMin(A  ) =  0.7530000000E-02   XMax(A  ) =  0.7530000000E-02
        XMin(B  ) =  0.4170000000E+01   XMax(B  ) =  0.4170000000E+01
        XMin(Tid) = -0.5330000000E+01   XMax(Tid) =  0.5570000000E+01
        XMin(Sur) = -0.5000000000E+01   XMax(Sur) =  0.5000000000E+01
        XMin(TAl) =  0.2500000000E+00   XMax(TAl) =  0.1000000000E+01
        XMin(r  ) =  0.9000000000E+00   XMax(r  ) =  0.1000000000E+01
        XMin(eB ) =  0.1000000020E-24   XMax(eB ) =  0.1000000000E+26



MAXIMUM NUMBER OF ITERATIONS (Max=200) = 200

NUMBER OF FORM CALCULATIONS (Max=10) =  6

ALLOWABLE DISCHARGE - m3/s/m ( 1) =  0.100E+00
ALLOWABLE DISCHARGE - m3/s/m ( 2) =  0.100E-01
ALLOWABLE DISCHARGE - m3/s/m ( 3) =  0.100E-02
ALLOWABLE DISCHARGE - m3/s/m ( 4) =  0.100E-03
ALLOWABLE DISCHARGE - m3/s/m ( 5) =  0.100E-04
ALLOWABLE DISCHARGE - m3/s/m ( 6) =  0.100E-05



REQUIRED RELATIVE ACCURACY OF THE RELIABILITY INDEX:

        Default Value (1%) ......... [ 1 ]
        User Specified Value ....... [ 2 ]

Select Option: 1



REQUIRED SMOOTHING COEFFICIENT FOR THE ITERATION PROCESS:

        Default Value (0) .......... [ 1 ]
        User Specified Value ....... [ 2 ]

Select Option: 2



Required Smoothing Coefficient for the Iteration Process [0,1] =  0.9000000000E+00



REQUIRED ACCURACY OF THE FAILURE FUNCTION:

        Default Value (1%) ......... [ 1 ]
        User Specified Value ....... [ 2 ]

Select Option: 1
```

D1-8

```
ALLOWABLE DISCHARGE - m3/s/m ( 1) =  0.100E+00



FINAL RESULTS



        Total Number of Iterations = 65
        Failure Function Z (X) =  0.2620132425E-06
        Mean Value of Z =  0.3434493330E+00
        Standard Deviation of Z =  0.8345014442E-01
        Reliability Index =  0.4115622992E+01
        Relative Accuracy of the Reliability Index (%) =  0.2358248898E-05
        Probability of Failure (%) =   0.001940
        Difference in Pf Between the Last 2 Iterations =  0.7829946879E-11



        DESIGN POINT COORDINATES

                Tp  =  0.7563663838E+01
                Hs  =  0.2622799378E+01
                A   =  0.7530000000E-02
                B   =  0.4170000000E+01
                Tid =  0.4460740834E+01
                Sur =  0.1081726008E+00
                TAl =  0.4805044288E+00
                r   =  0.9517390888E+00
                eB  =  0.6076173730E+00


        Alpha(Y 1) = -0.2539299171E+00
        Influence of Y(Tp ) on the Reliability Index =  0.6448040E+01

        Alpha(Y 2) = -0.5427487582E+00
        Influence of Y(Hs ) on the Reliability Index =  0.2945762E+02

        Alpha(A  ) =  0.0000000000E+00
        Influence of A   on the Reliability Index =  0.0000000E+00

        Alpha(B  ) =  0.0000000000E+00
        Influence of B   on the Reliability Index =  0.0000000E+00

        Alpha(Tid) = -0.5447375721E+00
        Influence of Tid on the Reliability Index =  0.2967390E+02

        Alpha(Sur) = -0.1509738752E+00
        Influence of Sur on the Reliability Index =  0.2279311E+01

        Alpha(TAl) =  0.9473934424E-01
        Influence of TAl on the Reliability Index =  0.8975543E+00

        Alpha(r  ) = -0.4098585879E-01
        Influence of r   on the Reliability Index =  0.1679841E+00

        Alpha(eB ) =  0.5574548101E+00
        Influence of eB  on the Reliability Index =  0.3107559E+02
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 2) =  0.100E-01
```

FINAL RESULTS

```
        Total Number of Iterations = 66
        Failure Function Z (X) =  0.1202995422E-08
        Mean Value of Z =  0.7088017973E-01
        Standard Deviation of Z =  0.2367117310E-01
        Reliability Index =  0.2994367007E+01
        Relative Accuracy of the Reliability Index (%) =  0.7225613349E-06
        Probability of Failure (%) =   0.137535
        Difference in Pf Between the Last 2 Iterations =  0.9736262209E-10


        DESIGN POINT COORDINATES

                Tp  =  0.7509955325E+01
                Hs  =  0.2256461280E+01
                A   =  0.7530000000E-02
                B   =  0.4170000000E+01
                Tid =  0.4012849932E+01
                Sur =  0.5072785074E-01
                TAl =  0.4886679945E+00
                r   =  0.9511623469E+00
                eB  =  0.7826602205E+00


    Alpha(Y 1) = -0.3422216243E+00
    Influence of Y(Tp ) on the Reliability Index =  0.1171156E+02

    Alpha(Y 2) = -0.5978879019E+00
    Influence of Y(Hs ) on the Reliability Index =  0.3574699E+02

    Alpha(A  ) =  0.0000000000E+00
    Influence of A   on the Reliability Index =  0.0000000E+00

    Alpha(B  ) =  0.0000000000E+00
    Influence of B   on the Reliability Index =  0.0000000E+00

    Alpha(Tid) = -0.5919726309E+00
    Influence of Tid on the Reliability Index =  0.3504316E+02

    Alpha(Sur) = -0.1142617071E+00
    Influence of Sur on the Reliability Index =  0.1305574E+01

    Alpha(TAl) =  0.7568882121E-01
    Influence of TAl on the Reliability Index =  0.5728798E+00

    Alpha(r  ) = -0.3764135231E-01
    Influence of r   on the Reliability Index =  0.1416871E+00

    Alpha(eB ) =  0.3934226912E+00
    Influence of eB  on the Reliability Index =  0.1547814E+02
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 3) =  0.100E-02



FINAL RESULTS


        Total Number of Iterations = 67
        Failure Function Z (X) = -0.7517596647E-09
        Mean Value of Z =  0.1305319872E-01
        Standard Deviation of Z =  0.5122525833E-02
        Reliability Index =  0.2548195782E+01
        Relative Accuracy of the Reliability Index (%) =  0.2240196757E-06
        Probability of Failure (%) =   0.541433
        Difference in Pf Between the Last 2 Iterations =  0.8962282045E-10



        DESIGN POINT COORDINATES

                Tp  =  0.7426464897E+01
                Hs  =  0.2064834403E+01
                A   =  0.7530000000E-02
                B   =  0.4170000000E+01
                Tid =  0.3774956652E+01
                Sur =  0.3390393578E-01
                TAl =  0.4912886033E+00
                r   =  0.9509318641E+00
                eB  =  0.8709390940E+00


     Alpha(Y 1) = -0.3809074754E+00
     Influence of Y(Tp ) on the Reliability Index =  0.1450905E+02


     Alpha(Y 2) = -0.6205992250E+00
     Influence of Y(Hs ) on the Reliability Index =  0.3851434E+02


     Alpha(A  ) =  0.0000000000E+00
     Influence of A   on the Reliability Index =  0.0000000E+00


     Alpha(B  ) =  0.0000000000E+00
     Influence of B   on the Reliability Index =  0.0000000E+00


     Alpha(Tid) = -0.6139291896E+00
     Influence of Tid on the Reliability Index =  0.3769090E+02


     Alpha(Sur) = -0.9992632726E-01
     Influence of Sur on the Reliability Index =  0.9985271E+00


     Alpha(TAl) =  0.6837305630E-01
     Influence of TAl on the Reliability Index =  0.4674875E+00


     Alpha(r  ) = -0.3545923011E-01
     Influence of r   on the Reliability Index =  0.1257357E+00


     Alpha(eB ) =  0.2773797840E+00
     Influence of eB  on the Reliability Index =  0.7693954E+01
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 4) =  0.100E-03



FINAL RESULTS



        Total Number of Iterations = 50
        Failure Function Z (X) = -0.7579302935E-09
        Mean Value of Z =  0.2284606485E-02
        Standard Deviation of Z =  0.9794443295E-03
        Reliability Index =  0.2332553690E+01
        Relative Accuracy of the Reliability Index (%) =  0.2283641497E-06
        Probability of Failure (%) =   0.983635
        Difference in Pf Between the Last 2 Iterations =  0.1390272984E-09



        DESIGN POINT COORDINATES

                Tp  =  0.7369449843E+01
                Hs  =  0.1958496021E+01
                A   =  0.7530000000E-02
                B   =  0.4170000000E+01
                Tid =  0.3639478043E+01
                Sur =  0.2707465900E-01
                TAl =  0.4923807471E+00
                r   =  0.9508282991E+00
                eB  =  0.9214694518E+00



        Alpha(Y 1) = -0.3987835981E+00
        Influence of Y(Tp ) on the Reliability Index =  0.1590284E+02


        Alpha(Y 2) = -0.6316113956E+00
        Influence of Y(Hs ) on the Reliability Index =  0.3989330E+02


        Alpha(A  ) =  0.0000000000E+00
        Influence of A   on the Reliability Index =  0.0000000E+00


        Alpha(B  ) =  0.0000000000E+00
        Influence of B   on the Reliability Index =  0.0000000E+00


        Alpha(Tid) = -0.6238974930E+00
        Influence of Tid on the Reliability Index =  0.3892481E+02


        Alpha(Sur) = -0.9358862260E-01
        Influence of Sur on the Reliability Index =  0.8758830E+00


        Alpha(TAl) =  0.6532971068E-01
        Influence of TAl on the Reliability Index =  0.4267971E+00


        Alpha(r  ) = -0.3443007124E-01
        Influence of r   on the Reliability Index =  0.1185430E+00


        Alpha(eB ) =  0.1964137825E+00
        Influence of eB  on the Reliability Index =  0.3857837E+01
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 5) =  0.100E-04



FINAL RESULTS


        Total Number of Iterations = 59
        Failure Function Z (X) = -0.3298526059E-09
        Mean Value of Z =  0.3928047060E-03
        Standard Deviation of Z =  0.1771568459E-03
        Reliability Index =  0.2217270826E+01
        Relative Accuracy of the Reliability Index (%) =  0.4071944254E-06
        Probability of Failure (%) =   1.330288
        Difference in Pf Between the Last 2 Iterations =  0.3105839509E-09



        DESIGN POINT COORDINATES

                Tp  =  0.7333707910E+01
                Hs  =  0.1897379159E+01
                A   =  0.7530000000E-02
                B   =  0.4170000000E+01
                Tid =  0.3562349375E+01
                Sur =  0.2354824851E-01
                TAl =  0.4929539674E+00
                r   =  0.9507712973E+00
                eB  =  0.9541034378E+00


        Alpha(Y 1) = -0.4077597518E+00
        Influence of Y(Tp ) on the Reliability Index =  0.1662680E+02

        Alpha(Y 2) = -0.6362933160E+00
        Influence of Y(Hs ) on the Reliability Index =  0.4048692E+02

        Alpha(A  ) =  0.0000000000E+00
        Influence of A   on the Reliability Index =  0.0000000E+00

        Alpha(B  ) =  0.0000000000E+00
        Influence of B   on the Reliability Index =  0.0000000E+00

        Alpha(Tid) = -0.6298563707E+00
        Influence of Tid on the Reliability Index =  0.3967190E+02

        Alpha(Sur) = -0.8990601037E-01
        Influence of Sur on the Reliability Index =  0.8083091E+00

        Alpha(TAl) =  0.6355590418E-01
        Influence of TAl on the Reliability Index =  0.4039353E+00

        Alpha(r  ) = -0.3372981404E-01
        Influence of r   on the Reliability Index =  0.1137700E+00

        Alpha(eB ) =  0.1374176447E+00
        Influence of eB  on the Reliability Index =  0.1888361E+01
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 6) =  0.100E-05



FINAL RESULTS



        Total Number of Iterations = 73
        Failure Function Z (X) = -0.8329562325E-10
        Mean Value of Z =  0.6684666168E-04
        Standard Deviation of Z =  0.3105163469E-04
        Reliability Index =  0.2152758216E+01
        Relative Accuracy of the Reliability Index (%) =  0.9605813979E-06
        Probability of Failure (%) =   1.566915
        Difference in Pf Between the Last 2 Iterations =  0.8085487037E-09



        DESIGN POINT COORDINATES


                Tp  =  0.7311668376E+01
                Hs  =  0.1861443367E+01
                A   =  0.7530000000E-02
                B   =  0.4170000000E+01
                Tid =  0.3517012419E+01
                Sur =  0.2162300799E-01
                TAl =  0.4932691935E+00
                r   =  0.9507392836E+00
                eB  =  0.9760302809E+00



        Alpha(Y 1) = -0.4124204710E+00
        Influence of Y(Tp ) on the Reliability Index =  0.1700906E+02


        Alpha(Y 2) = -0.6384045466E+00
        Influence of Y(Hs ) on the Reliability Index =  0.4075604E+02


        Alpha(A  ) =  0.0000000000E+00
        Influence of A   on the Reliability Index =  0.0000000E+00


        Alpha(B  ) =  0.0000000000E+00
        Influence of B   on the Reliability Index =  0.0000000E+00


        Alpha(Tid) = -0.6329314191E+00
        Influence of Tid on the Reliability Index =  0.4006022E+02


        Alpha(Sur) = -0.8776866672E-01
        Influence of Sur on the Reliability Index =  0.7703339E+00


        Alpha(TAl) =  0.6253193176E-01
        Influence of TAl on the Reliability Index =  0.3910242E+00


        Alpha(r  ) = -0.3329704273E-01
        Influence of r   on the Reliability Index =  0.1108693E+00


        Alpha(eB ) =  0.9499754891E-01
        Influence of eB  on the Reliability Index =  0.9024534E+00



WOULD YOU LIKE TO RESTART (Y/N) ? N
```

Input File *general.dad*: Owen's Model

```
2                     ! Failure mode of overtopping (Owen)
2                     ! Tide + Surge
Y                     ! Derivatives of the failure function supplied
1                     ! Mode 1 - Reliability analysis for a specified design
1                     ! Design life (in years)
N                     ! No combinations of actions considered
10                    ! Target design parameter - seawall crest level
```

Input File *form.dad*: Owen's Model

```
2                     ! Starting point: user specified values
5
1.2
4
1
0.95
1.027
2                     ! [XMin,XMax]: user specified values
1       20
0       20
-5.33   5.57
-5      5
0.5     1
-1E25   1E25
200                   ! Maximum number of iterations
6                     ! Number of FORM calculations
1E-1
1E-2
1E-3
1E-4
1E-5
1E-6
1                     ! Accuracy on Beta (%): default value (1%)
2                     ! Smoothing of the iteration: user specified value
0.9
1                     ! Accuracy on Z0 (%): default value (1%)
```

Input File *meandev.dad*: Owen's Model

```
10                              !Type of distribution (Weibull)
0                               !Not truncated
5      0.9    3.3               !Mean; stand. dev.; lower limit
10                              !Type of distribution (Weibull)
1                               !Truncated for X above Xo
0                               !Toe level
1.2    0.7    0.45              !Mean; stand. dev.; lower limit
0                               !Type of distribution (Deterministic)
0.0117                          !Value
0                               !Type of distribution (Deterministic)
```

```
21.71                           !Value
13                              !Type of distribution (User-Defined)
0.2749          2.3619          !Mean value; standard deviation
3                               !Type of distribution (Gumbel)
0                               !Not truncated
0.019483        0.192382475     !Mean value; standard deviation
6                               !Type of distribution (Beta)
0                               !Not truncated
0.95    0.01    0.9     1       !Mean; stand. dev.; lower & upper limits
2                               !Type of distribution (Log-Normal)
0                               !Not truncated
1.027           0.15            !Mean value; standard deviation
```

Input File *coefcor.dad*: Owen's Model

```
1       ! Rho(1,1)
0.6     ! Rho(1,2)
0       ! Rho(1,3)
0       !   .
0       !   .
0       !   .
0
0
0.6
1
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
0
1
0
0
0
```

```
0
0
0
0
0
1
0
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
0
1
```

Output File *summary.dat*: Owen's Model

```
    WHAT IS THE DATA SOURCE ?

            The Screen ..... [ 1 ]
            A Datafile ..... [ 2 ]

    Select Option: 2


    WHAT IS THE FAILURE MODE TO BE STUDIED:

            Overtopping (H&R) .................... [ 1 ]
            Overtopping (Owen) ................... [ 2 ]
            Dune Erosion (Vellinga) .............. [ 3 ]

    Select Option:  2


    HOW IS THE STILL-WATER-LEVEL DEFINED ?

            Total Level .... [ 1 ]
            Tide + Surge ... [ 2 ]

    Select Option: 2


    ARE THE FIRST DERIVATIVES OF THE FAILURE FUNCTION SUPPLIED (Y/N) ? Y
```

```
DESCRIPTION OF THE VARIABLES

        X(  1) = Tm  = Mean Wave Period
        X(  2) = Hs  = Wave Height
        X(  3) = A   = Owen Parameter
        X(  4) = B   = Owen Parameter
        X(  5) = Tid = Tide Level
        X(  6) = Sur = Surge
        X(  7) = r   = Roughness
        X(  8) = eB  = Model Parameter



WHAT IS THE PURPOSE OF THE ANALYSIS ?

        Reliability Analysis for a Specified Design ... [ 1 ]
        Design for a Specified Reliability Level ...... [ 2 ]

Select Option: 1



DESIGN LIFE OF THE STRUCTURE =   1



WOULD YOU LIKE TO CONSIDER COMBINATION OF ACTIONS (Y/N) ? N



PRESCRIBED VALUE OF THE DESIGN PARAMETER

        Seawall Crest Level =  0.1000000000E+02



CHARACTERISTICS OF THE VARIABLES

        Probability Distribution of Tm  = Minima Type III (Weibull)
        Mean Value of Tm  =  0.5000000000E+01
        Standard Deviation of Tm  =  0.9000000000E+00
        Lower Limit on Tm  =  0.3300000000E+01


        Probability Distribution of Hs  = Minima Type III (Weibull)
        The Distribution of Hs  is truncated above Xo = 0.6(Tide+Surge-TL)
        Seawall Toe Level (TL) =  0.0000000000E+00
        Mean Value of Hs  =  0.1200000000E+01
        Standard Deviation of Hs  =  0.7000000000E+00
        Lower Limit on Hs  =  0.4500000000E+00


        Probability Distribution of A   = Deterministic
        Mean Value of A   =  0.1170000000E-01
        Standard Deviation of A   =  0.0000000000E+00


        Probability Distribution of B   = Deterministic
        Mean Value of B   =  0.2171000000E+02
        Standard Deviation of B   =  0.0000000000E+00
```

```
        Probability Distribution of Tid = User-Defined Distribution
        Mean Value of Tid =  0.2749000000E+00
        Standard Deviation of Tid =  0.2361900000E+01



        Probability Distribution of Sur = Maxima Type I (Gumbel)
        Mean Value of Sur =  0.1948300000E-01
        Standard Deviation of Sur =  0.1923824750E+00



        Probability Distribution of r   = Beta
        Mean Value of r   =  0.9500000000E+00
        Standard Deviation of r   =  0.1000000000E-01
        Limits a and b of r   = [ 0.9000000000E+00, 0.1000000000E+01]



        Probability Distribution of eB  = Log-Normal
        Mean Value of eB  =  0.1027000000E+01
        Standard Deviation of eB  =  0.1500000000E+00




    CORRELATION COEFFICIENTS


        (Tm ,Tm ) =  0.1000000000E+01
        (Tm ,Hs ) =  0.6000000000E+00
        (Tm ,A  ) =  0.0000000000E+00
        (Tm ,B  ) =  0.0000000000E+00
        (Tm ,Tid) =  0.0000000000E+00
        (Tm ,Sur) =  0.0000000000E+00
        (Tm ,r  ) =  0.0000000000E+00
        (Tm ,eB ) =  0.0000000000E+00
        (Hs ,Tm ) =  0.6000000000E+00
        (Hs ,Hs ) =  0.1000000000E+01
        (Hs ,A  ) =  0.0000000000E+00
        (Hs ,B  ) =  0.0000000000E+00
        (Hs ,Tid) =  0.0000000000E+00
        (Hs ,Sur) =  0.0000000000E+00
        (Hs ,r  ) =  0.0000000000E+00
        (Hs ,eB ) =  0.0000000000E+00
        (A  ,Tm ) =  0.0000000000E+00
        (A  ,Hs ) =  0.0000000000E+00
        (A  ,A  ) =  0.1000000000E+01
        (A  ,B  ) =  0.0000000000E+00
        (A  ,Tid) =  0.0000000000E+00
        (A  ,Sur) =  0.0000000000E+00
        (A  ,r  ) =  0.0000000000E+00
        (A  ,eB ) =  0.0000000000E+00
        (B  ,Tm ) =  0.0000000000E+00
        (B  ,Hs ) =  0.0000000000E+00
        (B  ,A  ) =  0.0000000000E+00
        (B  ,B  ) =  0.1000000000E+01
        (B  ,Tid) =  0.0000000000E+00
        (B  ,Sur) =  0.0000000000E+00
        (B  ,r  ) =  0.0000000000E+00
        (B  ,eB ) =  0.0000000000E+00
        (Tid,Tm ) =  0.0000000000E+00
```

```
          (Tid,Hs ) =  0.0000000000E+00
          (Tid,A  ) =  0.0000000000E+00
          (Tid,B  ) =  0.0000000000E+00
          (Tid,Tid) =  0.1000000000E+01
          (Tid,Sur) =  0.0000000000E+00
          (Tid,r  ) =  0.0000000000E+00
          (Tid,eB ) =  0.0000000000E+00
          (Sur,Tm ) =  0.0000000000E+00
          (Sur,Hs ) =  0.0000000000E+00
          (Sur,A  ) =  0.0000000000E+00
          (Sur,B  ) =  0.0000000000E+00
          (Sur,Tid) =  0.0000000000E+00
          (Sur,Sur) =  0.1000000000E+01
          (Sur,r  ) =  0.0000000000E+00
          (Sur,eB ) =  0.0000000000E+00
          (r  ,Tm ) =  0.0000000000E+00
          (r  ,Hs ) =  0.0000000000E+00
          (r  ,A  ) =  0.0000000000E+00
          (r  ,B  ) =  0.0000000000E+00
          (r  ,Tid) =  0.0000000000E+00
          (r  ,Sur) =  0.0000000000E+00
          (r  ,r  ) =  0.1000000000E+01
          (r  ,eB ) =  0.0000000000E+00
          (eB ,Tm ) =  0.0000000000E+00
          (eB ,Hs ) =  0.0000000000E+00
          (eB ,A  ) =  0.0000000000E+00
          (eB ,B  ) =  0.0000000000E+00
          (eB ,Tid) =  0.0000000000E+00
          (eB ,Sur) =  0.0000000000E+00
          (eB ,r  ) =  0.0000000000E+00
          (eB ,eB ) =  0.1000000000E+01


STARTING POINT FOR THE FORM CALCULATIONS:

          Default Values (mean values) ... [ 1 ]
          User Specified Values .......... [ 2 ]

Select Option: 2

STARTING POINT

          Tm  =  0.5000000000E+01
          Hs  =  0.1200000000E+01
          Tid =  0.4000000000E+01
          Sur =  0.1000000000E+01
          r   =  0.9500000000E+00
          eB  =  0.1027000000E+01

LIMITING VALUES FOR THE VARIABLES:

          Default Values (+/- 1E25) ... [ 1 ]
          User Specified Values ....... [ 2 ]

Select Option: 2
```

```
LIMITING VALUES FOR THE VARIABLES


        XMin(Tm ) =  0.3300000000E+01   XMax(Tm ) =  0.2000000000E+02
        XMin(Hs ) =  0.4500000000E+00   XMax(Hs ) =  0.2000000000E+02
        XMin(A  ) =  0.1170000000E-01   XMax(A  ) =  0.1170000000E-01
        XMin(B  ) =  0.2171000000E+02   XMax(B  ) =  0.2171000000E+02
        XMin(Tid) = -0.5330000000E+01   XMax(Tid) =  0.5570000000E+01
        XMin(Sur) = -0.5000000000E+01   XMax(Sur) =  0.5000000000E+01
        XMin(r  ) =  0.9000000000E+00   XMax(r  ) =  0.1000000000E+01
        XMin(eB ) =  0.1000000020E-24   XMax(eB ) =  0.1000000000E+26



MAXIMUM NUMBER OF ITERATIONS (Max=200) = 200


NUMBER OF FORM CALCULATIONS (Max=10) =  6


ALLOWABLE DISCHARGE - m3/s/m ( 1) =  0.100E+00
ALLOWABLE DISCHARGE - m3/s/m ( 2) =  0.100E-01
ALLOWABLE DISCHARGE - m3/s/m ( 3) =  0.100E-02
ALLOWABLE DISCHARGE - m3/s/m ( 4) =  0.100E-03
ALLOWABLE DISCHARGE - m3/s/m ( 5) =  0.100E-04
ALLOWABLE DISCHARGE - m3/s/m ( 6) =  0.100E-05



REQUIRED RELATIVE ACCURACY OF THE RELIABILITY INDEX:

        Default Value (1%) ......... [ 1 ]
        User Specified Value ....... [ 2 ]

Select Option: 1



REQUIRED SMOOTHING COEFFICIENT FOR THE ITERATION PROCESS:

        Default Value (0) .......... [ 1 ]
        User Specified Value ....... [ 2 ]

Select Option: 2



Required Smoothing Coefficient for the Iteration Process [0,1] =  0.9000000000E+00



REQUIRED ACCURACY OF THE FAILURE FUNCTION:

        Default Value (1%) ......... [ 1 ]
        User Specified Value ....... [ 2 ]

Select Option: 1
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 1) =  0.100E+00



FINAL RESULTS


        Total Number of Iterations = 64
        Failure Function Z (X) = -0.3304572431E-07
        Mean Value of Z =  0.3929601531E+00
        Standard Deviation of Z =  0.1097844458E+00
        Reliability Index =  0.3579379120E+01
        Relative Accuracy of the Reliability Index (%) =  0.8743530582E-06
        Probability of Failure (%) =   0.017243
        Difference in Pf Between the Last 2 Iterations =  0.2190748757E-10



        DESIGN POINT COORDINATES

                Tm  =  0.7867682058E+01
                Hs  =  0.2288306898E+01
                A   =  0.1170000000E-01
                B   =  0.2171000000E+02
                Tid =  0.3959377043E+01
                Sur =  0.4621900285E-01
                r   =  0.9511036460E+00
                eB  =  0.8250366186E+00



        Alpha(Y 1) = -0.7472518744E+00
        Influence of Y(Tm ) on the Reliability Index =  0.5583854E+02


        Alpha(Y 2) = -0.2005296001E+00
        Influence of Y(Hs ) on the Reliability Index =  0.4021212E+01


        Alpha(A  ) =  0.0000000000E+00
        Influence of A   on the Reliability Index =  0.0000000E+00


        Alpha(B  ) =  0.0000000000E+00
        Influence of B   on the Reliability Index =  0.0000000E+00


        Alpha(Tid) = -0.4816107580E+00
        Influence of Tid on the Reliability Index =  0.2319489E+02


        Alpha(Sur) = -0.8911581216E-01
        Influence of Sur on the Reliability Index =  0.7941628E+00


        Alpha(r  ) = -0.2990006018E-01
        Influence of r   on the Reliability Index =  0.8940136E-01


        Alpha(eB ) =  0.4007716955E+00
        Influence of eB  on the Reliability Index =  0.1606180E+02
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 2) =  0.100E-01



FINAL RESULTS



         Total Number of Iterations = 54
         Failure Function Z (X) =  0.8715043405E-09
         Mean Value of Z =  0.4954358667E-01
         Standard Deviation of Z =  0.2093853466E-01
         Reliability Index =  0.2366143929E+01
         Relative Accuracy of the Reliability Index (%) =  0.3568124208E-06
         Probability of Failure (%) =   0.898770
         Difference in Pf Between the Last 2 Iterations =  0.2051574986E-09



         DESIGN POINT COORDINATES

                 Tm  =  0.6714341777E+01
                 Hs  =  0.1768924154E+01
                 A   =  0.1170000000E-01
                 B   =  0.2171000000E+02
                 Tid =  0.3324831734E+01
                 Sur =  0.1484800976E-01
                 r   =  0.9506214224E+00
                 eB  =  0.9037968322E+00



      Alpha(Y 1) = -0.7365302465E+00
      Influence of Y(Tm ) on the Reliability Index =  0.5424768E+02

      Alpha(Y 2) = -0.2590956446E+00
      Influence of Y(Hs ) on the Reliability Index =  0.6713055E+01

      Alpha(A  ) =  0.0000000000E+00
      Influence of A   on the Reliability Index =  0.0000000E+00

      Alpha(B  ) =  0.0000000000E+00
      Influence of B   on the Reliability Index =  0.0000000E+00

      Alpha(Tid) = -0.5189503568E+00
      Influence of Tid on the Reliability Index =  0.2693095E+02

      Alpha(Sur) = -0.6424241822E-01
      Influence of Sur on the Reliability Index =  0.4127088E+00

      Alpha(r  ) = -0.2546293174E-01
      Influence of r   on the Reliability Index =  0.6483609E-01

      Alpha(eB ) =  0.3410391781E+00
      Influence of eB  on the Reliability Index =  0.1163077E+02
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 3) =  0.100E-02


FINAL RESULTS


        Total Number of Iterations = 64
        Failure Function Z (X) =  0.8701465193E-11
        Mean Value of Z =  0.5323353420E-02
        Standard Deviation of Z =  0.3180215221E-02
        Reliability Index =  0.1673897221E+01
        Relative Accuracy of the Reliability Index (%) =  0.1547060913E-06
        Probability of Failure (%) =   4.707768
        Difference in Pf Between the Last 2 Iterations =  0.2540418154E-09


        DESIGN POINT COORDINATES

                Tm  =  0.6104414348E+01
                Hs  =  0.1383504337E+01
                A   =  0.1170000000E-01
                B   =  0.2171000000E+02
                Tid =  0.2805428663E+01
                Sur =  0.3618371385E-02
                r   =  0.9504050441E+00
                eB  =  0.9414699343E+00


        Alpha(Y 1) = -0.7203268632E+00
        Influence of Y(Tm ) on the Reliability Index =  0.5188708E+02

        Alpha(Y 2) = -0.2818109186E+00
        Influence of Y(Hs ) on the Reliability Index =  0.7941739E+01

        Alpha(A  ) =  0.0000000000E+00
        Influence of A   on the Reliability Index =  0.0000000E+00

        Alpha(B  ) =  0.0000000000E+00
        Influence of B   on the Reliability Index =  0.0000000E+00

        Alpha(Tid) = -0.5473526962E+00
        Influence of Tid on the Reliability Index =  0.2995950E+02

        Alpha(Sur) = -0.5360142620E-01
        Influence of Sur on the Reliability Index =  0.2873113E+00

        Alpha(r  ) = -0.2346256681E-01
        Influence of r   on the Reliability Index =  0.5504920E-01

        Alpha(eB ) =  0.3141547983E+00
        Influence of eB  on the Reliability Index =  0.9869324E+01
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 4) =  0.100E-03



FINAL RESULTS


        Total Number of Iterations = 68
        Failure Function Z (X) = -0.1654101392E-10
        Mean Value of Z =  0.5174503166E-03
        Standard Deviation of Z =  0.4304664431E-03
        Reliability Index =  0.1202068883E+01
        Relative Accuracy of the Reliability Index (%) =  0.1902001380E-06
        Probability of Failure (%) =  11.467071
        Difference in Pf Between the Last 2 Iterations =  0.4412629845E-09



        DESIGN POINT COORDINATES

            Tm  =  0.5709854971E+01
            Hs  =  0.1112225125E+01
            A   =  0.1170000000E-01
            B   =  0.2171000000E+02
            Tid =  0.2387767135E+01
            Sur = -0.2096607008E-02
            r   =  0.9502779598E+00
            eB  =  0.9643145164E+00



    Alpha(Y 1) = -0.6927297329E+00
    Influence of Y(Tm ) on the Reliability Index =  0.4798745E+02

    Alpha(Y 2) = -0.2879491765E+00
    Influence of Y(Hs ) on the Reliability Index =  0.8291473E+01

    Alpha(A  ) =  0.0000000000E+00
    Influence of A   on the Reliability Index =  0.0000000E+00

    Alpha(B  ) =  0.0000000000E+00
    Influence of B   on the Reliability Index =  0.0000000E+00

    Alpha(Tid) = -0.5867796063E+00
    Influence of Tid on the Reliability Index =  0.3443103E+02

    Alpha(Sur) = -0.4781748192E-01
    Influence of Sur on the Reliability Index =  0.2286512E+00

    Alpha(r  ) = -0.2242283715E-01
    Influence of r   on the Reliability Index =  0.5027836E-01

    Alpha(eB ) =  0.3001852550E+00
    Influence of eB  on the Reliability Index =  0.9011119E+01
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 5) =  0.100E-04



FINAL RESULTS


        Total Number of Iterations = 60
        Failure Function Z (X) = -0.4879941218E-10
        Mean Value of Z =  0.4701425035E-04
        Standard Deviation of Z =  0.5477444869E-04
        Reliability Index =  0.8583244829E+00
        Relative Accuracy of the Reliability Index (%) =  0.1489272992E-05
        Probability of Failure (%) =  19.535878
        Difference in Pf Between the Last 2 Iterations =  0.3538277560E-08



        DESIGN POINT COORDINATES

                Tm  =  0.5432632364E+01
                Hs  =  0.9176506972E+00
                A   =  0.1170000000E-01
                B   =  0.2171000000E+02
                Tid =  0.2031448014E+01
                Sur = -0.5574932557E-02
                r   =  0.9501921396E+00
                eB  =  0.9800548223E+00


    Alpha(Y 1) = -0.6511433929E+00
    Influence of Y(Tm ) on the Reliability Index =  0.4239877E+02

    Alpha(Y 2) = -0.2749863860E+00
    Influence of Y(Hs ) on the Reliability Index =  0.7561751E+01

    Alpha(A  ) =  0.0000000000E+00
    Influence of A   on the Reliability Index =  0.0000000E+00

    Alpha(B  ) =  0.0000000000E+00
    Influence of B   on the Reliability Index =  0.0000000E+00

    Alpha(Tid) = -0.6430924749E+00
    Influence of Tid on the Reliability Index =  0.4135679E+02

    Alpha(Sur) = -0.4388713472E-01
    Influence of Sur on the Reliability Index =  0.1926081E+00

    Alpha(r  ) = -0.2170666113E-01
    Influence of r   on the Reliability Index =  0.4711791E-01

    Alpha(eB ) =  0.2905676829E+00
    Influence of eB  on the Reliability Index =  0.8442958E+01
```

```
ALLOWABLE DISCHARGE - m3/s/m ( 6) =  0.100E-05
```

```
FINAL RESULTS
```

```
        Total Number of Iterations = 62
        Failure Function Z (X) = -0.4468194738E-10
        Mean Value of Z =  0.4055884630E-05
        Standard Deviation of Z =  0.6658564046E-05
        Reliability Index =  0.6091230184E+00
        Relative Accuracy of the Reliability Index (%) =  0.9391025663E-05
        Probability of Failure (%) =  27.122233
        Difference in Pf Between the Last 2 Iterations =  0.1900280305E-07
```

```
        DESIGN POINT COORDINATES
```

```
             Tm  =  0.5242180513E+01
             Hs  =  0.7713292909E+00
             A   =  0.1170000000E-01
             B   =  0.2171000000E+02
             Tid =  0.1698078052E+01
             Sur = -0.7783444552E-02
             r   =  0.9501336265E+00
             eB  =  0.9909332316E+00
```

```
    Alpha(Y 1) = -0.5999624020E+00
    Influence of Y(Tm ) on the Reliability Index =  0.3599549E+02
```

```
    Alpha(Y 2) = -0.2415765075E+00
    Influence of Y(Hs ) on the Reliability Index =  0.5835921E+01
```

```
    Alpha(A  ) =  0.0000000000E+00
    Influence of A   on the Reliability Index =  0.0000000E+00
```

```
    Alpha(B  ) =  0.0000000000E+00
    Influence of B   on the Reliability Index =  0.0000000E+00
```

```
    Alpha(Tid) = -0.7060358221E+00
    Influence of Tid on the Reliability Index =  0.4984866E+02
```

```
    Alpha(Sur) = -0.4108099036E-01
    Influence of Sur on the Reliability Index =  0.1687648E+00
```

```
    Alpha(r  ) = -0.2127042864E-01
    Influence of r   on the Reliability Index =  0.4524311E-01
```

```
    Alpha(eB ) =  0.2847090553E+00
    Influence of eB  on the Reliability Index =  0.8105925E+01
```

```
WOULD YOU LIKE TO RESTART (Y/N) ? N
```

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.3E-05	0.5E-08	0.5E-09	0.3E-10	0.2E-11	0.2E-12
Standard Deviation of Z	0.103	0.306E-01	0.662E-02	0.127E-02	0.230E-03	0.404E-04
Relative Accuracy of Beta (%)	0.2E-05	0.2E-06	0.1E-06	0.4E-07	0.3E-07	0.4E-08
Probability of Failure P_f (%/Year)	0.083640	1.236960	2.998720	4.440951	5.407432	6.010773
Difference in P_f Between Last 2 Iterations	0.2E-09	0.2E-09	0.1E-09	0.6E-10	0.6E-10	0.7E-11

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)
$X_1=T_p$	7.544	-----	7.305	-----	7.154	-----	7.070	-----	7.023	-----	6.997	-----
$X_2=H_s$	2.333	-----	1.877	-----	1.661	-----	1.550	-----	1.490	-----	1.456	-----
$^*Y_1=f(T_p,H_s)$	-----	11.29	-----	15.88	-----	17.03	-----	17.24	-----	17.23	-----	17.18
$^*Y_2=f(T_p,H_s)$	-----	33.21	-----	34.66	-----	34.77	-----	34.65	-----	34.61	-----	34.56
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=$Tide	4.139	36.37	3.600	40.27	3.336	43.15	3.194	46.15	3.113	46.29	3.065	46.95
$X_6=Y_6=$Surge	0.064	1.68	0.025	0.86	0.015	0.67	0.011	0.60	0.92E-02	0.57	0.82E-02	0.55
$X_7=Y_7=\tan\alpha$	0.4888	0.51	0.4940	0.28	0.4954	0.23	0.4960	0.22	0.4963	0.21	0.4964	0.21
$X_8=Y_8=r$	0.9509	0.07	0.9505	0.06	0.9504	0.05	0.9504	0.05	0.9504	0.05	0.9504	0.05
$X_9=Y_9=e_B$	0.7629	16.87	0.8853	7.99	0.9378	4.10	0.9669	2.09	0.9850	1.04	0.9972	0.50

Table D2.1: Summary of PARASODE results: normal conditions, H&R model, $(R_{max})_{37\%}$, slope 1:2, CL=8m OD. (*Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.3E-05	0.1E-05	-0.8E-09	-0.8E-09	-0.3E-09	-0.8E-10
Standard Deviation of Z	0.833E-01	0.237E-01	0.513E-02	0.979E-03	0.177E-03	0.311E-04
Relative Accuracy of Beta (%)	0.2E-05	0.7E-05	0.2E-05	0.2E-05	0.4E-05	0.1E-05
Probability of Failure Pf (%/Year)	0.001940	0.130535	0.541433	0.583535	1.330288	1.598815
Difference In Pf Between Last 2 Iterations	0.8E-11	0.1E-09	0.9E-10	0.1E-09	0.3E-09	0.8E-09

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1=T_p$	7.564	—	7.510	—	7.426	—	7.369	—	7.334	—	7.312	—
$X_2=H_s$	2.623	—	2.256	—	2.065	—	1.958	—	1.897	—	1.851	—
$Y_1=f(T_p,H_s)$	—	6.45	—	11.71	—	14.51	—	15.90	—	16.63	—	17.01
$Y_2=f(T_p,H_s)$	—	29.46	—	36.75	—	38.51	—	39.89	—	40.49	—	40.76
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=$Tide	4.461	29.67	4.013	35.04	3.775	37.69	3.639	38.92	3.562	39.67	3.517	40.06
$X_6=Y_6=$Surge	0.108	2.28	0.051	1.31	0.034	1.00	0.027	0.88	0.024	0.81	0.022	0.77
$X_7=Y_7=$Tm	0.4806	0.90	0.4883	0.57	0.4913	0.47	0.4934	0.43	0.4910	0.40	0.4933	0.39
$X_8=Y_8=r$	0.9517	0.17	0.9512	0.14	0.9508	0.13	0.9506	0.12	0.9508	0.11	0.9507	0.11
$X_9=Y_9=\sigma_s$	0.6076	31.07	0.7823	15.48	0.8709	7.69	0.9215	3.86	0.9541	1.89	0.9760	0.90

Table D2.2: Summary of PARASODE results: Normal conditions, H&R model, $(P_{rms})_{3\%}$, slope 1:2, CL=10m OD. (Note that the correlation coefficient between T_p and H_s is $\rho_{1,2}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.2E-7	-0.4E-8	-0.3E-8	-0.2E-8	0.4E-11	0.4E-12
Standard Deviation of Z	0.755E-01	0.199E-01	0.406E-02	0.760E-03	0.136E-03	0.234E-04
Relative Accuracy of Beta (%)	0.2E-06	0.8E-06	0.3E-05	0.1E-05	0.4E-06	0.2E-06
Probability of Failure P_f (%/Year)	0.060012	0.005717	0.053427	0.133560	0.213479	0.272780
Difference in P_f Between Last 2 Iterations	0.9E-13	0.6E-11	0.2E-10	0.1E-10	0.8E-10	0.4E-10

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)
$X_1=T_p$	7.441	—	7.541	—	7.535	—	7.520	—	7.507	—	7.498	—
$X_2=H_s$	2.829	—	2.965	—	2.417	—	2.331	—	2.279	—	2.246	—
$Y_1=f(T_p,H_s)$	—	3.33	—	7.24	—	9.87	—	11.55	—	12.59	—	13.16
$Y_2=f(T_p,H_s)$	—	25.25	—	33.55	—	38.70	—	41.56	—	43.00	—	43.76
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=$Tide	4.674	23.56	4.376	31.64	4.188	35.37	4.078	37.38	4.013	38.48	3.971	38.58
$X_6=Y_6=$Surge	0.178	3.22	0.094	2.12	0.069	1.77	0.057	1.57	0.051	1.43	0.047	1.36
$X_7=Y_7=$settle	0.4702	1.33	0.4806	1.03	0.482	0.94	0.4839	0.88	0.4870	0.83	0.4875	0.80
$X_8=Y_8=r$	0.9530	0.32	0.9522	0.31	0.9519	0.30	0.9517	0.29	0.9516	0.28	0.9515	0.27
$X_9=Y_9=q$	0.4742	42.99	0.6684	24.11	0.7818	13.05	0.8563	6.77	0.9073	3.39	0.9425	1.67

Table D2.3: Summary of PARASODE results: normal conditions, H&R model, $(R_{max})_{2\%}$, slope 1:2, CL=12m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{1,2}=0.6$).

D (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.3E-10	0.1E-10	0.2E-09	-0.1E-09	-0.4E-10	-0.4E-10
Standard Deviation of Z	0.744E-01	0.183E-01	0.396E-02	0.644E-03	0.110E-03	0.184E-04
Relative Accuracy of Beta b (%)	0.6E-06	0.2E-06	0.1E-06	0.1E-06	0.1E-06	0.5E-06
Probability of Failure P_f (%/Year)	0.000000	0.000115	0.002599	0.008888	0.017747	0.025591
Difference in P_f Between Last 2 Iterations	0.9E-15	0.5E-12	0.5E-11	0.2E-10	0.3E-10	0.1E-10

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)
$X_1 = T_p$	7.298	—	7.423	—	7.478	—	7.499	—	7.507	—	7.510	—
$X_2 = H_s$	3.026	—	2.833	—	2.732	—	2.670	—	2.630	—	2.604	—
$Y_1 = f(T_p, H_s)$	—	1.71	—	3.86	—	5.68	—	6.97	—	7.71	—	8.28
$Y_2 = f(T_p, H_s)$	—	22.28	—	30.75	—	36.73	—	40.53	—	42.81	—	44.14
$X_3 = Y_3 = A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4 = Y_4 = B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5 = Y_5 = $ Tide	4.844	19.62	4.677	28.22	4.570	33.93	4.498	37.14	4.448	38.84	4.415	39.72
$X_6 = Y_6 = $ Surge	0.305	5.14	0.179	3.88	0.136	3.34	0.116	3.06	0.106	2.91	0.100	2.81
$X_7 = Y_7 = \gamma_f$	0.4591	1.75	0.4663	1.69	0.4713	1.66	0.4760	1.64	0.4772	1.63	0.4779	1.62
$X_8 = Y_8 = r$	0.9547	0.55	0.9538	0.63	0.9534	0.65	0.9532	0.65	0.9530	0.65	0.9529	0.65
$X_9 = Y_9 = \gamma_b$	0.2828	48.95	0.5631	30.97	0.6906	20.01	0.7813	10.01	0.8471	5.39	0.8862	2.78

Table D2.4: Summary of PARASODE results: normal conditions, H&R model, $(P_{mod})_{5\%}$, slope 1:2, CL=1.4m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.6$).

D (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.2E-06	0.1E-06	0.2E-06	0.4E-10	-0.4E-11	-0.7E-11
Standard Deviation of Z	0.783E-01	0.183E-01	0.343E-02	0.585E-03	0.957E-04	0.158E-04
Relative Accuracy of Beta (%)	0.2E-06	0.4E-06	0.6E-06	0.4E-06	0.3E-06	0.2E-06
Probability of Failure Pf (%/Year)	0.000000	0.000001	0.000030	0.000191	0.000568	0.000875
Difference in Pf Between Last 2 Iterations	0.8E-18	0.1E-13	0.5E-12	0.2E-11	0.3E-11	0.3E-11

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_p$	7.129	—	7.229	—	7.276	—	7.304	—	7.319	—	7.328	—
$X_2 = H_s$	3.262	—	3.102	—	3.038	—	2.984	—	2.956	—	2.937	—
$Y_1 = f(T_p, H_s)$	—	0.99	—	1.89	—	2.67	—	3.28	—	3.69	—	3.96
$Y_2 = f(T_p, H_s)$	—	20.45	—	28.20	—	34.16	—	38.26	—	40.90	—	42.53
$X_3 = Y_3 = A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4 = Y_4 = B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5 = Y_5 = $ Tide	4.951	16.86	4.887	24.57	4.845	30.27	4.815	34.12	4.792	36.52	4.775	37.93
$X_6 = Y_6 = $ Surge	0.576	9.70	0.384	8.58	0.308	8.01	0.269	7.62	0.248	7.37	0.236	7.21
$X_7 = Y_7 = \tan\alpha$	0.4491	2.06	0.4561	2.40	0.4594	2.65	0.4613	2.80	0.4625	2.89	0.4632	2.94
$X_8 = Y_8 = r$	0.9565	0.81	0.9561	1.10	0.9558	1.27	0.9556	1.35	0.9554	1.40	0.9553	1.43
$X_9 = Y_9 = \theta_a$	0.3313	49.12	0.4570	33.26	0.6088	20.97	0.7080	12.57	0.7810	7.23	0.8412	4.80

Table D2.5: Summary of PARASODE results: normal conditions, H&R model, $(R_{max})_{2\%}$, slope 1:2, CL=16m OD. (Note that the correlation coefficient between the two e's in T_p and H_s is $\rho_{12} = 0.6$).

Q(m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.4E-06	0.1E-06	-0.7E-10	-0.1E-09	-0.6E-10	-0.1E-10
Standard Deviation of Z	0.990E-01	0.288E-01	0.491E-02	0.803E-03	0.123E-03	0.190E-04
Relative Accuracy of Beta (%)	0.3E-06	0.3E-06	0.1E-06	0.1E-06	0.4E-06	0.6E-06
Probability of Failure P_f (%/Year)	0.069673	1.247739	3.595343	6.001907	8.057221	9.659011
Difference in P_f Between Last 2 Iterations	0.2E-10	0.2E-09	0.2E-09	0.2E-09	0.8E-09	0.1E-08

Q(m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_p$	7.559	—	7.303	—	7.113	—	6.995	—	6.917	—	6.864	—
$X_2 = H_s$	2.372	—	1.868	—	1.595	—	1.437	—	1.388	—	1.272	—
$Y_1 = f(T_p, H_s)$	—	11.13	—	15.79	—	16.86	—	16.82	—	16.96	—	16.24
$Y_2 = f(T_p, H_s)$	—	33.37	—	35.09	—	35.04	—	35.14	—	35.12	—	34.91
$X_3 = Y_3 = A$	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0
$X_4 = Y_4 = B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5 = Y_5 = $ Tide	4.191	37.19	3.575	39.33	3.225	41.80	2.999	43.53	2.849	45.03	2.748	45.42
$X_6 = Y_6 = $ Surge	0.069	1.86	0.024	0.81	0.012	0.57	0.70E-02	0.49	0.44E-02	0.45	0.27E-02	0.42
$X_7 = Y_7 = $ CITM	0.4884	0.53	0.4939	0.30	0.4956	0.24	0.4963	0.23	0.4965	0.22	0.4969	0.22
$X_8 = Y_8 = r$	0.9509	0.07	0.9505	0.05	0.9504	0.05	0.9503	0.05	0.9503	0.04	0.9503	0.04
$X_9 = Y_9 = e_m$	0.8053	15.85	0.9048	8.63	0.9448	5.44	0.9684	3.74	0.9624	2.68	0.9604	1.75

Table D2.6: Summary of PARASODE results: normal conditions, H&R model, $(R_{max})_{100\%}$, slope 1:2, CL=8m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.5$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.4E-09	-0.1E-07	-0.6E-08	0.1E-10	0.9E-12	0.5E-13
Standard Deviation of Z	0.763E-01	0.193E-01	0.376E-02	0.621E-03	0.976E-04	0.143E-04
Relative Accuracy of Beta (%)	0.2E-05	0.9E-05	0.6E-05	0.2E-05	0.1E-05	0.9E-07
Probability of Failure P_f (%/Year)	0.001251	0.148078	0.781139	1.737792	2.722584	3.574279
Difference in P_f Between Last 2 Iterations	0.5E-11	0.1E-09	0.3E-09	0.2E-09	0.2E-09	0.1E-09

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1=T_p$	7.559	—	7.493	—	7.370	—	7.271	—	7.198	—	7.147	—
$X_2=H_s$	2.655	—	2.216	—	1.952	—	1.78+	—	1.674	—	1.600	—
$\gamma_1=f(T_p,H_s)$	—	6.08	—	11.55	—	14.66	—	16.26	—	17.10	—	17.52
$\gamma_2=f(T_p,H_s)$	—	29.20	—	35.51	—	38.35	—	39.71	—	40.24	—	40.74
$X_3=\gamma_3=A$	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0
$X_4=\gamma_4=B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5=\gamma_5=$ Tide	4.469	29.38	3.950	33.31	3.608	35.01	3.387	36.17	3.244	37.36	3.141	37.96
$X_6=\gamma_6=$ Surge	0.116	2.42	0.045	1.12	0.026	0.75	0.017	0.60	0.012	0.52	0.994E-02	0.48
$X_7=\gamma_7=$ Tides	0.478	0.96	0.4888	0.57	0.4900	0.44	0.494	0.39	0.4943	0.35	0.494	0.34
$X_8=\gamma_8=r$	0.9618	0.17	0.9511	0.13	0.9508	0.10	0.9507	0.09	0.9506	0.08	0.9505	0.08
$X_9=\gamma_9=s$	0.6529	31.80	0.8861	17.81	0.8825	10.69	0.923	6.78	0.9500	4.35	0.9515	2.88

Table D2.7: Summary of PARASODE results: normal conditions, H&R model, $(P_{max})_{dom}$, slope 1:2, CL=10m OD. (Note that the correlation coefficients between T_p and H_s and H_s & $\rho_{12}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.7E-09	0.8E-08	0.4E-09	-0.2E-10	-0.3E-10	-0.2E-10
Standard Deviation of Z	0.660E-01	0.193E-01	0.291E-02	0.486E-03	0.757E-04	0.113E-04
Relative Accuracy of Beta (%)	0.4E-06	0.1E-06	0.9E-06	0.5E-06	0.3E-06	0.4E-06
Probability of Failure Pf (%/Year)	0.000005	0.008181	0.108005	0.360635	0.696062	1.047365
Difference In Pf Between Last 2 Iterations	0.6E-13	0.2E-10	0.1E-09	0.1E-09	0.1E-09	0.3E-09

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
X_1=T_p	7.416	—	7.520	—	7.489	—	7.438	—	7.392	—	7.356	—
X_2=H_s	2.821	—	2.472	—	2.246	—	2.063	—	1.987	—	1.913	—
Y_1=f(T_p,H_s)	—	3.60	—	7.19	—	10.55	—	12.85	—	14.36	—	15.37
Y_2=f(T_p,H_s)	—	24.17	—	32.34	—	37.61	—	40.75	—	42.68	—	43.84
X_3=Y_3=A	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0
X_4=Y_4=B	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
X_5=Y_5=Tide	4.663	21.87	4.246	28.27	3.952	31.33	3.748	32.94	3.606	33.82	3.507	34.46
X_6=Y_6=Surge	0.172	2.87	0.075	1.51	0.046	1.06	0.032	0.84	0.025	0.73	0.021	0.65
X_7=Y_7=β_m	0.4691	1.34	0.4816	0.95	0.4866	0.77	0.4880	0.67	0.4906	0.62	0.4912	0.58
X_8=Y_8=r	0.9530	0.30	0.9519	0.25	0.9516	0.21	0.9512	0.19	0.9511	0.17	0.9510	0.16
X_9=Y_9=e_n	0.5146	45.45	0.6651	29.49	0.7563	18.47	0.8306	11.76	0.9015	7.62	0.9361	4.94

Table D2.8: Summary of PARASODE results: normal conditions, H&R model, $(R_{max})_{allow}$, slope 1:2, CL=12m OD (Note that the correlation coefficient ρ models the link between T_p and H_s & ρ_2=0.6).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.8E-08	0.4E-08	0.1E-08	0.1E-08	0.1E-10	0.2E-12
Standard Deviation of Z	0.623E-01	0.138E-01	0.243E-02	0.395E-03	0.608E-04	0.886E-05
Relative Accuracy of Beta (%)	0.4E-05	0.2E-05	0.1E-05	0.8E-05	0.8E-05	0.5E-05
Probability of Failure Pf (%/Year)	0.000000	0.000190	0.007651	0.045509	0.12440+	0.223585
Difference In Pf Between Last 2 Iterations	0.1E-15	0.7E-12	0.1E-10	0.4E-10	0.1E-09	1.0E-10

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
X₁=Tp	7.237	—	7.441	—	7.481	—	7.481	—	7.474	—	7.451	—
X₂=Hs	2.947	—	2.670	—	2.496	—	2.388	—	2.287	—	2.206	—
Y₁=f(Tp,Hs)	—	1.50	—	4.15	—	6.65	—	8.73	—	10.44	—	11.66
Y₂=f(Tp,Hs)	—	20.40	—	28.43	—	34.78	—	39.41	—	42.51	—	44.57
X₃=Y₃=A	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0
X₄=Y₄=B	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
X₅=Y₅=Tide	4.777	17.04	4.484	24.08	4.246	28.06	4.073	30.48	3.954	32.29	3.864	33.35
X₆=Y₆=Surge	0.238	3.27	0.113	1.93	0.075	1.90	0.057	1.27	0.045	1.09	0.039	0.99
X₇=Y₇=Batter	0.4590	1.63	0.47731	1.35	0.4750	1.23	0.4822	1.15	0.4844	1.05	0.4865	1.01
X₈=Y₈=r	0.9543	0.43	0.9531	0.42	0.9534	0.39	0.9521	0.37	0.9518	0.34	0.9517	0.32
X₉=Y₉=q	0.4129	55.73	0.5900	39.64	0.7040	27.38	0.7816	18.59	0.8365	12.27	0.8795	8.10

Table D2.3: Summary of PARASODE result: normal conditions, H&R model, (P_max)₅₀%, slope 1:2, CL=14m OD (Note that the correlation coefficient between Tp and Hs & Ha & ρ₁₂=0.6).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.3E-09	0.1E-09	0.4E-09	0.6E-10	-0.3E-09	0.2E-10
Standard Deviation of Z	0.613E-01	0.129E-01	0.222E-02	0.345E-03	0.515E-04	0.739E-05
Relative Accuracy of Beta (%)	0.7E-05	0.4E-05	0.2E-05	0.2E-05	0.5E-05	0.1E-05
Probability of Failure P_f (%o/Year)	0.000000	0.000002	0.000051	0.003073	0.013576	0.002803
Difference in P_f Between Last 2 Iterations	0.2E-13	0.2E-13	0.1E-11	0.1E-10	0.9E-11	0.4E-10

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_p$	7.053	—	7.281	—	7.381	—	7.416	—	7.436	—	7.443	—
$X_2 = H_s$	3.063	—	2.832	—	2.694	—	2.591	—	2.517	—	2.461	—
$Y_1 = f(T_p, H_s)$	—	0.75	—	2.18	—	3.86	—	5.30	—	6.59	—	7.57
$Y_2 = f(T_p, H_s)$	—	17.87	—	25.09	—	31.95	—	36.7+	—	40.72	—	43.73
$X_3 = Y_3 = A$	0.0054	0	0.0054	0	0.0054	0	0.0054	0	0.0054	0	0.0054	0
$X_4 = Y_4 = B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5 = Y_5 = $ Tide	4.858	13.82	4.660	20.20	4.504	25.29	4.370	28.52	4.273	30.93	4.197	32.39
$X_6 = Y_6 = $ Surge	0.332	4.04	0.171	2.62	0.117	2.11	0.093	1.89	0.079	1.73	0.070	1.63
$X_7 = Y_7 = $ Batter	0.4488	1.88	0.4538	1.80	0.4658	1.76	0.4734	1.76	0.4759	1.73	0.4775	1.72
$X_8 = Y_8 = r$	0.9558	0.59	0.9546	0.65	0.9538	0.65	0.9538	0.66	0.9530	0.65	0.9538	0.64
$X_9 = Y_9 = \varepsilon_m$	0.3358	61.05	0.4970	47.46	0.6135	34.77	0.7002	25.13	0.7638	17.65	0.815	12.32

Table D2.10: Summary of PARASODE results: normal conditions. H&R model, $(R_{mean})_{ave}$, slope 1:2, CL=16m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.2E-07	-0.1E-08	-0.1E-08	-0.5E-09	-0.5E-09	-0.7E-10
Standard Deviation of Z	0.137	0.254E-01	0.375E-02	0.598E-03	0.648E-04	0.767E-05
Relative Accuracy of Beta (%)	0.5E-05	0.3E-05	0.8E-07	0.1E-05	0.9E-05	0.7E-05
Probability of Failure Pf (%/Year)	0.320838	4.150943	12.506962	22.343071	30.811748	36.859838
Difference in Pf Between Last 2 Iterations	0.2E-09	0.4E-09	0.2E-09	0.3E-08	0.2E-07	0.9E-08

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1 = T_m$	6.955	—	6.012	—	5.609	—	5.315	—	5.134	—	5.034	—
$X_2 = H_s$	2.041	—	1.475	—	1.119	—	0.888	—	0.725	—	0.599	—
$Y_1 = f(T_m, H_s)$	—	51.59	—	47.27	—	41.65	—	33.99	—	25.88	—	20.10
$Y_2 = f(T_m, H_s)$	—	6.39	—	7.58	—	7.81	—	6.53	—	4.18	—	1.89
$X_3 = Y_3 = A$	0.01π	0	0.01π	0	0.01π	0	0.01π	0	0.01π	0	0.01π	0
$X_4 = Y_4 = B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5 = Y_5 = $ Tide	3.758	32.34	3.064	37.67	2.903	43.67	2.036	53.10	1.610	63.99	1.227	71.96
$X_6 = Y_6 = $ Surge	0.033	0.84	0.83E-02	0.44	-0.78E-03	0.32	-0.55E-02	0.25	-0.82E-02	0.20	-0.97E-02	0.18
$X_7 = Y_7 = r$	0.9505	0.06	0.9504	0.04	0.9502	0.04	0.9501	0.03	0.9501	0.03	0.9501	0.03
$X_8 = Y_8 = $ b	0.9505	8.79	0.9507	7.00	0.9718	6.51	0.9688	6.10	0.9683	5.72	1.004	5.84

Table D2.11: Summary of PARASODE results: normal conditions, Owen's model, slope 1:2, CL=8m, OD. (Note that the correlation coefficient between T_m and H_s is $\rho_{12} = 0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	-0.3E-0	0.9E-09	0.9E-11	-0.2E-10	-0.5E-10	-0.4E-10
Standard Deviation of Z	0.110	0.206E-01	0.318E-02	0.430E-03	0.548E-04	0.666E-05
Relative Accuracy of Beta (%)	0.9E-05	0.4E-05	0.2E-05	0.2E-05	0.1E-05	0.9E-05
Probability of Failure P_f (%/Year)	0.017243	0.898770	4.707768	11.457071	19.536878	27.122238
Difference in P_f Between Last 2 Iterations	0.2E-10	0.2E-09	0.3E-09	0.4E-09	0.4E-08	0.2E-07

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1=T_m$	7.868	--	6.714	--	6.104	--	5.710	--	5.433	--	5.242	--
$X_2=H_s$	2.288	--	1.769	--	1.384	--	1.112	--	0.918	--	0.771	--
$Y_1=f(T_m,H_s)$	--	55.84	--	54.25	--	51.89	--	47.99	--	42.40	--	36.00
$Y_2=f(T_m,H_s)$	--	4.02	--	6.71	--	7.94	--	8.29	--	7.55	--	5.84
$X_3=Y_3=A$	0.017	0	0.017	0	0.017	0	0.017	0	0.017	0	0.017	0
$X_4=Y_4=B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5=Y_5=$ Tide	3.969	23.20	3.325	26.94	2.805	29.96	2.388	34.43	2.031	41.36	1.698	49.85
$X_6=Y_6=$ Surge	0.046	0.79	0.015	0.41	0.36E-02	0.29	-0.21E-02	0.23	-0.55E-02	0.19	-0.78E-02	0.17
$X_7=Y_7=r$	0.9511	0.09	0.9506	0.06	0.9504	0.06	0.9503	0.06	0.9502	0.06	0.9501	0.05
$X_8=Y_8=\phi_n$	0.8230	16.06	0.9038	11.63	0.9415	9.86	0.9543	9.01	0.9601	8.44	0.9609	8.09

Table D2.12: Summary of PARASODE results: normal conditions. Owen's model, slope 1:2, CL=10m OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	-0.5E-08	-0.8E-08	0.6E-10	0.5E-12	-0.2E-11	-0.4E-11
Standard Deviation of Z	0.955E-01	0.175E-01	0.279E-02	0.383E-03	0.484E-04	0.588E-05
Relative Accuracy of Beta (%)	0.1E-05	0.7E-06	0.4E-06	0.5E-06	0.3E-06	0.1E-06
Probability of Failure P_f (%/Year)	0.000640	0.014606	1.487913	5.064674	10.696238	17.406728
Difference in P_f Between Last 2 Iterations	0.1E-11	0.1E-08	0.4E-09	0.8E-10	0.8E-09	0.3E-08

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1=T_m$	8.686	—	7.340	—	6.594	—	6.122	—	5.783	—	5.527	—
$X_2=H_s$	2.408	—	1.964	—	1.597	—	1.312	—	1.096	—	0.932	—
$Y_1=f(T_m,H_s)$	—	56.70	—	56.99	—	56.26	—	54.78	—	52.06	—	48.00
$Y_2=f(T_m,H_s)$	—	2.02	—	4.96	—	6.87	—	7.96	—	8.37	—	7.99
$X_3=Y_3=A$	0.01f	0	0.01f	0	0.01f	0	0.01f	0	0.01f	0	0.01f	0
$X_4=Y_4=B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5=Y_5=Tide$	4.059	17.07	3.495	20.45	3.021	22.83	2.620	24.93	2.290	28.26	2.065	33.38
$X_6=Y_6=Surge$	0.065	0.69	0.021	0.38	0.74E-02	0.26	0.83E-03	0.20	-0.31E-02	0.17	-0.58E-02	0.15
$X_7=Y_7=r$	0.9516	0.13	0.9909	0.10	0.9906	0.08	0.9904	0.07	0.9908	0.06	0.9902	0.06
$X_8=Y_8=\beta_0$	0.7439	23.39	0.8496	17.12	0.904	13.70	0.9385	12.06	0.959	11.08	0.9725	10.42

Table D2.13: Summary of PARASODE results: normal conditions, Owen's model, slope 1:2, CL=12m OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.2E-07	-0.4E-08	-0.1E-08	0.3E-11	0.8E-13	-0.1E-12
Standard Deviation of Z	0.881E-01	0.160E-01	0.249E-02	0.344E-03	0.446E-04	0.548E-05
Relative Accuracy of Beta (%)	0.1E-05	0.5E-05	0.1E-05	0.2E-05	0.8E-07	0.4E-06
Probability of Failure Pf (%/Year)	0.000012	0.017938	0.397621	1.97985+	5.28969?	10.067409
Difference in Pf Between last 2 iterations	0.5E-14	0.1E-10	0.3E-09	0.2E-09	0.1E-09	0.9E-09

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
X₁=Tm	9.406	—	7.919	—	7.069	—	6.528	—	6.134	—	5.838	—
X₂=Hs	2.465	—	2.086	—	1.756	—	1.475	—	1.255	—	1.079	—
Y₁=f(Tm,Hs)	—	56.05	—	57.60	—	58.09	—	57.93	—	56.74	—	54.79
Y₂=f(Tm,Hs)	—	0.93	—	3.25	—	5.41	—	6.88	—	7.84	—	8.25
X₃=Y₃=A	0.0017	0	0.0017	0	0.0017	0	0.0017	0	0.0017	0	0.0017	0
X₄=Y₄=B	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
X₅=Y₅=Tide	4.106	12.99	3.607	16.05	3.177	18.21	2.792	19.50	2.480	21.43	2.210	23.98
X₆=Y₆=Surge	0.660	0.57	0.036	0.35	0.011	0.24	0.34E-02	0.18	-0.11E-02	0.15	-0.40E-02	0.13
X₇=Y₇=r	0.9521	0.16	0.9513	0.13	0.9309	0.10	0.9906	0.09	0.9906	0.08	0.9904	0.07
X₈=Y₈=ea	0.5769	29.30	0.7941	22.62	0.8630	17.95	0.9036	15.42	0.9314	13.76	0.9510	12.77

Table D2.14: Summary of PARASODE results: normal conditions, Owen's model, slope 1:2, CL=14m OD (Note that the correlation coefficient between Tm and Hs is ρ₁₂ = 0.5).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.2E-08	-0.3E-08	-0.1E-08	-0.1E-09	0.7E-13	0.2E-13
Standard Deviation of Z	0.833E-01	0.148E-01	0.227E-02	0.315E-03	0.410E-04	0.568E-05
Relative Accuracy of Beta (%)	0.5E-06	0.5E-06	0.3E-06	0.1E-06	0.5E-07	0.2E-06
Probability of Failure Pf (%/Year)	0.000000	0.001769	0.04008+	0.690028	2.378177	5.377900
Difference in Pf Between Last 2 Iterations	0.5E-15	0.2E-11	0.3E-10	0.5E-09	0.6E-10	0.4E-09

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1=T_m$	10.013	—	8.447	—	7.519	—	6.920	—	6.484	—	6.150	—
$X_2=H_s$	2.492	—	2.160	—	1.869	—	1.604	—	1.396	—	1.207	—
$Y_1=f(T_m,H_s)$	—	54.47	—	57.02	—	58.52	—	59.2+	—	59.1+	—	58.33
$Y_2=f(T_m,H_s)$	—	0.42	—	2.01	—	3.97	—	5.57	—	6.80	—	7.64
$X_3=Y_3=A$	0.017	0	0.017	0	0.017	0	0.017	0	0.017	0	0.017	0
$X_4=Y_4=B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5=Y_5=$ Tide	4.125	10.26	3.675	12.87	3.288	14.88	2.923	15.91	2.620	17.66	2.361	18.55
$X_6=Y_6=$ Surge	0.062	0.46	0.029	0.30	0.014	0.22	0.57E-02	0.17	0.83E-03	0.14	-0.24E-02	0.12
$X_7=Y_7=r$	0.9525	0.19	0.9517	0.15	0.9511	0.12	0.9508	0.11	0.9506	0.09	0.9506	0.09
$X_8=Y_8=\beta_h$	0.6170	34.20	0.7409	27.65	0.8204	22.29	0.8665	19.00	0.9032	16.77	0.9275	15.27

Table D2.15: Summary of PARASODE results: normal conditions. Owen's model, slope 1:2, CL=16m OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.3E-06	0.6E-06	-0.1E-06	-0.2E-07	-0.4E-08	-0.6E-09
Standard Deviation of Z	0.193	0.403E-01	0.717E-02	0.121E-02	0.203E-03	0.34IE-04
Relative Accuracy of Beta (%)	0.2E-04	0.2E-04	0.3E-03	0.3E-05	0.4E-05	0.4E-05
Probability of Failure Pf (%/Year)	10.46228	33.36498	50.03187	59.88353	65.50187	68.70631
Difference In Pf Between Last 2 Iterations	0.5E-07	0.3E-07	0.1E-08	0.3E-08	0.6E-08	0.8E-08

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
X₁=Tp	7.134	—	6.549	—	6.282	—	6.105	—	6.015	—	5.963	—
X₂=Hs	1.883	—	1.263	—	0.986	—	0.858	—	0.792	—	0.756	—
Y₁=f(Tp,Hs)	—	57.12	—	46.26	—	39.83	—	36.62	—	34.89	—	33.91
Y₂=f(Tp,Hs)	—	38.45	—	49.47	—	56.29	—	60.06	—	62.29	—	63.62
X₃=Y₃=A	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
X₄=Y₄=B	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
X₅=Y₅=SWL	5.633	0.77	5.618	0.91	5.612	1.12	5.607	1.32	5.604	1.49	5.602	1.61
X₆=Y₆=σm	0.4579	0.11	0.4968	0.12	0.5000	0.15	0.5005	0.18	0.5009	0.21	0.5012	0.23
X₃=Y₇=r	0.9501	0.01	0.9501	0.02	0.9500	0.02	0.9500	0.03	0.9499	0.03	0.9499	0.04
X₆=Y₈=σb	0.9591	3.54	1.0051	3.22	1.0222	2.59	1.0308	1.79	1.0334	1.09	1.0311	0.59

Table D2.16: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{37\%}$, slope 12, CL=8m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.5$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	-0.1E-05	-0.1E-05	-0.2E-05	-0.3E-05	-0.5E-05	-0.8E-05
Standard Deviation of Z	0.168	0.490E-01	0.877E-02	0.195E-02	0.398E-03	0.498E-04
Relative Accuracy of Beta (%)	0.2E-04	0.5E-05	0.2E-05	0.2E-05	0.1E-05	0.2E-05
Probability of Failure P_f (%/Year)	2.049887	11.152396	19.782596	25.710291	29.412394	31.613696
Difference in P_f Between Last 2 Iterations	0.2E-07	0.1E-07	0.5E-06	0.4E-06	0.3E-06	0.3E-06

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)
$X_1=T_p$	7.651	—	7.108	—	6.846	—	6.707	—	6.630	—	6.596	—
$X_2=H_s$	2.726	—	1.946	—	1.604	—	1.438	—	1.351	—	1.303	—
$Y_1=f(T_p,H_s)$	—	48.62	—	56.84	—	53.54	—	51.10	—	49.63	—	48.76
$Y_2=f(T_p,H_s)$	—	42.40	—	38.79	—	43.59	—	47.01	—	49.08	—	50.29
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=SWL$	5.647	0.69	5.636	0.43	5.620	0.43	5.618	0.44	5.617	0.45	5.616	0.46
$X_6=Y_6=\tan\alpha$	0.495	0.19	0.4960	0.11	0.4966	0.11	0.4988	0.12	0.4991	0.12	0.4992	0.12
$X_7=Y_7=r$	0.9504	0.03	0.9501	0.03	0.9501	0.03	0.9501	0.03	0.9501	0.03	0.9501	0.03
$X_8=Y_8=\theta_n$	0.8998	8.07	0.9967	3.81	0.9929	2.30	1.0051	1.39	1.0123	0.69	1.0164	0.34

Table D2.17: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{3/4\%}$, slope 1:2, CL=10m, OD. (Note that the correlation coefficient model between T_p and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.1E-05	-0.9E-05	-0.1E-05	-0.2E-07	-0.4E-08	-0.5E-09
Standard Deviation of Z	0.108	0.406E-01	0.873E-02	0.164E-02	0.293E-03	0.511E-04
Relative Accuracy of Beta (%)	0.2E-05	0.1E-04	0.4E-05	0.1E-05	0.1E-05	0.1E-07
Probability of Failure P_f (%/Year)	0.166868	3.325311	7.363337	10.457024	12.514215	13.796318
Difference in P_f Between Last 2 Iterations	0.3E-08	0.2E-07	0.7E-08	0.2E-08	0.3E-09	0.3E-10

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1=T_p$	7.899	—	7.538	—	7.283	—	7.147	—	7.072	—	7.029	—
$X_2=H_s$	3.164	—	2.959	—	2.190	—	1.966	—	1.897	—	1.859	—
$Y_1=f(T_p,H_s)$	—	23.60	—	53.77	—	58.55	—	58.72	—	58.32	—	57.95
$Y_2=f(T_p,H_s)$	—	49.82	—	39.95	—	38.26	—	38.44	—	40.58	—	41.29
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=SWL$	5.697	1.77	5.634	0.39	5.626	0.31	5.623	0.29	5.621	0.28	5.620	0.28
$X_6=Y_6=\tan\alpha$	0.4852	0.55	0.4964	0.16	0.4974	0.12	0.4978	0.12	0.4980	0.12	0.4982	0.12
$X_7=Y_7=r$	0.9510	0.12	0.9504	0.04	0.9503	0.04	0.9502	0.04	0.9502	0.04	0.9502	0.04
$X_8=Y_8=\zeta_m$	0.7404	24.14	0.9258	5.69	0.9884	2.72	0.9886	1.39	1.0001	0.69	1.0068	0.32

Table D2.18: Summary of PARASODE results: extreme conditions, H&R model. $(R_{max})_{37\%}$, slope 1:2, CL=12m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12}=0.6$).

q(m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.2E-6	-0.1E-5	-0.1E-6	-0.1E-7	-0.2E-8	-0.3E-9
Standard Deviation of Z	0.865E-01	0.288E-01	0.731E-02	0.147E-02	0.289E-03	0.476E-04
Relative Accuracy of Beta (%)	0.5E-04	0.1E-04	0.1E-04	0.5E-05	0.2E-05	0.1E-05
Probability of Failure P_f (%/Year)	0.00460+	0.618692	2.247271	3.733632	4.787147	5.468444
Difference in P_f Between Last 2 Iterations	0.4+E-08	0.5E-08	0.1E-07	0.7E-08	0.4E-08	0.2E-08

q(m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1=T_p$	7.849	—	7.853	—	7.660	—	7.527	—	7.451	—	7.407	—
$X_2=H_s$	3.369	—	3.045	—	2.737	—	2.542	—	2.431	—	2.367	—
$Y_1=f(T_p,H_s)$	—	10.76	—	33.21	—	50.16	—	56.21	—	58.41	—	59.30
$Y_2=f(T_p,H_s)$	—	+0.84	—	+52.25	—	+44.86	—	+1.39	—	40.25	—	39.86
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=$SWL	5.843	+.91	5.660	0.81	5.636	0.37	5.629	0.29	5.627	0.26	5.636	0.25
$X_6=Y_6=$tanα	0.479B	1.06	0.4921	0.40	0.4955	0.20	0.4954	0.16	0.4958	0.15	0.4970	0.14
$X_7=Y_7=r$	0.9522	0.29	0.9508	0.13	0.9505	0.07	0.9504	0.06	0.9504	0.05	0.9504	0.05
$X_8=Y_8=e_a$	0.5746	+2.1+	0.8319	13.20	0.9320	4.34	0.9570	1.89	0.9588	0.88	0.9860	0.40

Table D2.19: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{2\%}$, slope 1:2, CL=14m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.3E-05	0.5E-07	-0.2E-07	-0.1E-07	-0.2E-08	-0.2E-09
Standard Deviation of Z	0.838E-01	0.197E-01	0.454E-02	0.993E-03	0.199E-03	0.357E-04
Relative Accuracy of Beta (%)	0.1E-04	0.9E-05	0.3E-04	0.1E-04	0.9E-05	0.7E-05
Probability of Failure Pf (%/Year)	0.000066	0.032120	0.334650	0.864799	1.329916	1.653077
Difference in Pf Between Last 2 Iterations	0.2E-11	0.4E-09	0.8E-08	0.6E-08	0.7E-08	0.6E-08

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1=T_p$	7.757	—	7.850	—	7.891	—	7.852	—	7.797	—	7.757	—
$X_2=H_s$	3.574	—	3.319	—	3.172	—	3.041	—	2.945	—	2.883	—
$Y_1=\xi(T_p,H_s)$	—	6.10	—	14.44	—	26.34	—	36.61	—	42.84	—	46.26
$Y_2=\xi(T_p,H_s)$	—	33.48	—	80.62	—	59.23	—	57.68	—	54.44	—	52.17
$X_3=Y_3=A$	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0	0.00753	0
$X_4=Y_4=B$	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0	4.17	0
$X_5=Y_5=SWL$	6.125	10.05	5.778	3.82	5.682	1.37	5.654	0.69	5.644	0.48	5.639	0.41
$X_6=Y_6=\tan\alpha$	0.4716	1.38	0.4815	1.18	0.4886	0.72	0.4921	0.44	0.4936	0.33	0.4943	0.28
$X_7=Y_7=r$	0.9534	0.45	0.9524	0.46	0.9515	0.29	0.9510	0.18	0.9508	0.14	0.9508	0.12
$X_8=Y_8=\gamma_\beta$	0.4760	48.54	0.6715	29.48	0.8238	12.05	0.9128	4.40	0.9551	1.77	0.9802	0.76

Table D2.20: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{2\%}$, slope 1:2, CL=16m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	-0.6E-05	-0.5E-05	-0.1E-05	-0.1E-07	-0.2E-08	-0.3E-09
Standard Deviation of Z	0.197	0.373E-01	0.577E-02	0.810E-03	0.108E-03	0.138E-04
Relative Accuracy of Beta (%)	0.2E-06	0.2E-06	0.3E-06	0.4E-06	0.4E-06	0.4E-06
Probability of Failure Pf (%/Year)	9.984048	32.900951	52.142062	65.206304	73.832340	79.958976
Difference in Pf Between Last 2 Iterations	0.5E-08	0.3E-08	0.7E-09	0.5E-08	0.9E-08	0.1E-07

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1 = T_p$	7.156	—	6.557	—	6.228	—	6.022	—	5.882	—	5.784	—
$X_2 = H_s$	2.012	—	1.271	—	0.9567	—	0.7966	—	0.7045	—	0.6475	—
$Y_1 = f(T_p, H_s)$	—	57.96	—	46.58	—	38.95	—	34.47	—	31.90	—	29.30
$Y_2 = f(T_p, H_s)$	—	38.41	—	49.44	—	59.86	—	61.15	—	63.93	—	65.95
$X_3 = Y_3 = A$	0.6542	0	0.6542	0	0.6542	0	0.6542	0	0.6542	0	0.6542	0
$X_4 = Y_4 = B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5 = Y_5 = SWL$	5.635	0.84	5.618	0.87	5.611	1.05	5.605	1.27	5.600	1.55	5.595	1.87
$X_6 = Y_6 = \tan\alpha$	0.4968	0.11	0.4982	0.12	0.5001	0.15	0.5009	0.21	0.5017	0.27	0.5025	0.36
$X_7 = Y_7 = r$	0.9502	0.01	0.9501	0.02	0.9500	0.02	0.9499	0.03	0.9499	0.04	0.9499	0.05
$X_8 = Y_8 = \theta_u$	0.9835	3.07	1.0112	2.97	1.0271	2.96	1.0384	2.87	1.0461	2.71	1.0513	2.47

Table D2.21: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{50\%}$, slope 12, CL=8m OD (Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.5$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	-0.3E-07	-0.6E-07	-0.9E-07	-0.2E-07	-0.2E-08	-0.4E-09
Standard Deviation of Z	0.153	0.392E-01	0.670E-02	0.103E-02	0.151E-03	0.214E-04
Relative Accuracy of Beta (%)	0.2E-04	0.3E-05	0.3E-05	0.2E-05	0.2E-05	0.2E-05
Probability of Failure P_f (%/Year)	1.837258	11.375034	22.371933	31.510145	38.374448	43.371755
Difference in P_f Between Last 2 Iterations	0.2E-07	0.7E-08	0.4E-08	0.3E-08	0.3E-08	0.1E-08

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1=T_p$	7.686	—	7.099	—	6.780	—	6.586	—	6.459	—	6.372	—
$X_2=H_s$	2.778	—	1.934	—	1.524	—	1.302	—	1.170	—	1.086	—
$Y_1=f(T_p,H_s)$	—	47.18	—	46.77	—	51.90	—	47.87	—	45.12	—	43.27
$Y_2=f(T_p,H_s)$	—	44.05	—	38.90	—	44.71	—	49.40	—	52.68	—	54.96
$X_3=Y_3=A$	0.00642	0	0.00642	0	0.00642	0	0.00642	0	0.00642	0	0.00642	0
$X_4=Y_4=B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5=Y_5=SWL$	5.651	0.77	5.625	0.40	5.619	0.39	5.616	0.39	5.614	0.40	5.613	0.41
$X_6=Y_6=\tan\alpha$	0.4953	0.21	0.4950	0.11	0.4957	0.11	0.4962	0.12	0.4965	0.13	0.4967	0.14
$X_7=Y_7=r$	0.9504	0.04	0.9502	0.02	0.9501	0.02	0.9501	0.02	0.9500	0.03	0.9500	0.03
$X_8=Y_8=e_n$	0.9181	7.75	0.9865	3.80	1.0012	2.87	1.0123	2.20	1.0184	1.64	1.0230	1.19

Table D2.22: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{crit}$, slope 1:2, CL=10m OD (Note that the correlation coefficient between T_p and H_s & ρ_{12} =06).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.2E-06	0.7E-06	0.2E-07	0.1E-06	0.3E-07	0.4E-08
Standard Deviation of Z	0.954E-01	0.390E-01	0.657E-02	0.163E-02	0.157E-03	0.229E-04
Relative Accuracy of Beta (%)	0.4E-08	0.5E-04	0.1E-04	0.2E-04	0.3E-04	0.3E-04
Probability of Failure P_f (%/Year)	0.13850	3.63579	9.36304	14.88088	19.45578	23.06942
Difference in P_f Between Last 2 Iterations	0.5E-07	0.8E-07	0.2E-07	0.5E-07	0.7E-07	0.7E-07

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_p$	7.803	—	7.509	—	7.183	—	6.985	—	6.857	—	6.770	—
$X_2 = H_s$	3.196	—	2.516	—	2.049	—	1.781	—	1.618	—	1.512	—
$Y_1 = f(T_p, H_s)$	—	21.53	—	54.37	—	57.78	—	56.12	—	54.22	—	52.69
$Y_2 = f(T_p, H_s)$	—	49.82	—	39.23	—	38.35	—	41.13	—	43.75	—	45.80
$X_3 = Y_3 = A$	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0
$X_4 = Y_4 = B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5 = Y_5 = SWL$	5.709	2.14	5.631	0.33	5.622	0.25	5.619	0.23	5.618	0.22	5.617	0.22
$X_6 = Y_6 = \tan\alpha$	0.4881	0.63	0.4965	0.15	0.4977	0.12	0.4983	0.11	0.4985	0.11	0.4987	0.11
$X_7 = Y_7 = r$	0.9511	0.1	0.9504	0.04	0.9502	0.03	0.9502	0.03	0.9502	0.03	0.9501	0.03
$X_8 = Y_8 = m$	0.7682	25.74	0.9441	5.88	0.9786	3.47	0.9946	2.38	1.0040	1.67	1.0100	1.15

Table D2.23: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{conv}$, slope 1:2, CL=12m OD. (Note that the correlation coefficient between T_p and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.4E-07	-0.6E-05	-0.1E-05	-0.2E-07	-0.4E-08	-0.5E-09
Standard Deviation of Z	0.752E-01	0.251E-01	0.516E-02	0.583E-03	0.151E-03	0.223E-04
Relative Accuracy of Beta (%)	0.6E-04	0.1E-04	0.8E-05	0.3E-05	0.2E-05	0.3E-05
Probability of Failure Pf (%/Year)	0.003327	0.830008	3.963881	6.691976	9.507379	11.81352
Difference in Pf Between Last 2 Iterations	0.3E-08	0.8E-08	0.1E-07	0.6E-08	0.5E-08	0.6E-08

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_p$	7.835	—	7.815	—	7.522	—	7.318	—	7.185	—	7.065	—
$X_2 = H_s$	3.37+	—	2.975	—	2.535	—	2.239	—	2.052	—	1.908	—
$Y_1 = f(T_p, H_s)$	—	9.92	—	36.69	—	54.57	—	58.40	—	58.75	—	58.32
$Y_2 = f(T_p, H_s)$	—	39.35	—	49.51	—	40.00	—	38.30	—	39.01	—	40.09
$X_3 = Y_3 = A$	0.06542	0	0.06542	0	0.06542	0	0.06542	0	0.06542	0	0.06542	0
$X_4 = Y_4 = B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_5 = Y_5 = SWL$	5.843	4.61	5.648	0.53	5.627	0.23	5.622	0.18	5.620	0.17	5.619	0.16
$X_6 = Y_6 = \tan\alpha$	0.4790	1.09	0.4932	0.32	0.4954	0.16	0.4973	0.13	0.4977	0.12	0.4980	0.12
$X_7 = Y_7 = r$	0.9522	0.27	0.9508	0.09	0.9504	0.05	0.9508	0.04	0.9508	0.04	0.9502	0.04
$X_8 = Y_8 = \alpha$	0.6147	44.76	0.8716	12.86	0.9463	4.99	0.9764	2.95	0.9905	1.91	0.9906	1.27

Table D2.24: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{...}$, slope 1:2, CL=14m, OD. (Note that the correlation coefficient between T_p and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	0.2E-07	-0.8E-08	-0.1E-06	-0.1E-07	-0.2E-08	-0.3E-09
Standard Deviation of Z	0.696E+01	0.166E+01	0.436E-02	0.827E-03	0.139E-03	0.205E-04
Relative Accuracy of Beta (%)	0.6E-04	0.1E-04	0.2E-04	0.9E-05	0.4E-05	0.2E-05
Probability of Failure Pf (%/Year)	0.000030	0.078323	1.022197	2.647434	4.287714	5.707717
Difference in Pf Between Last 2 iterations	0.5E-11	0.1E-08	0.1E-07	0.1E-07	0.7E-08	0.4E-08

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)
$X_i = T_p$	7.724	—	7.891	—	7.758	—	7.615	—	7.478	—	7.385	—
$X_i = H_s$	3.517	—	3.223	—	2.948	—	2.670	—	2.471	—	2.335	—
$Y_i = f(T_p, H_s)$	—	5.46	—	18.31	—	39.16	—	51.97	—	56.94	—	58.83
$Y_i = f(T_p, H_s)$	—	31.34	—	49.54	—	50.25	—	43.15	—	40.15	—	39.26
$X_i = Y_i = A$	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0	0.00542	0
$X_i = Y_i = B$	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0	7.16	0
$X_i = Y_i = SWL$	6.025	7.00	5.693	1.36	5.640	0.37	5.628	0.21	5.623	0.17	5.621	0.14
$X_i = Y_i = \tan\alpha$	0.4711	1.34	0.4890	0.79	0.4935	0.31	0.4936	0.18	0.4967	0.15	0.4971	0.14
$X_i = Y_i = r$	0.9532	0.39	0.9517	0.26	0.9508	0.11	0.9506	0.06	0.9504	0.05	0.9504	0.05
$X_i = Y_i = \gamma_b$	0.5094	54.47	0.7380	29.74	0.8933	9.80	0.9491	4.43	0.9734	2.54	0.9874	1.58

Table D2.25: Summary of PARASODE results: extreme conditions, H&R model, $(R_{max})_{50\%}$, slope 1:2, CL=16m, OD. (Note that the correlation coefficient between T_p and H_s is $\rho_{12} = 0.5$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.2E-06	-0.4E-06	-0.1E-06	-0.8E-08	-0.9E-09	-0.1E-09
Standard Deviation of Z	0.173	0.244E-01	0.300E-02	0.233E-03	0.233E-04	0.231E-05
Relative Accuracy of Beta (%)	0.2E-04	0.5E-05	0.4E-05	0.6E-07	0.2E-05	0.1E-07
Probability of Failure P_f (%/Year)	17.252849	56.131605	85.360694	97.651107	99.888835	99.997254
Difference in P_f Between Last 2 Iterations	0.5E-07	0.3E-08	0.1E-07	0.7E-10	0.2E-10	0.6E-13

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1=T_m$	5.640	—	4.780	—	4.195	—	3.780	—	3.573	—	3.485	—
$X_2=H_s$	1.641	—	0.909	—	0.618	—	0.511	—	0.479	—	0.468	—
$Y_1=f(T_m, H_s)$	—	88.74	—	76.12	—	68.81	—	60.83	—	46.10	—	33.54
$Y_2=f(T_m, H_s)$	—	5.90	—	16.92	—	18.78	—	12.59	—	7.17	—	4.42
$X_3=Y_3=A$	0.017	0	0.017	0	0.017	0	0.017	0	0.017	0	0.017	0
$X_4=Y_4=B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5=Y_5=SWL$	5.633	1.39	5.609	1.50	5.588	2.18	5.558	3.30	5.528	3.71	5.510	3.41
$X_6=Y_6=r$	0.9501	0.02	0.9500	0.03	0.9499	0.06	0.9459	0.13	0.9466	0.24	0.9476	0.33
$X_7=Y_7=\theta_n$	0.9689	3.95	1.023	5.43	1.0672	10.17	1.1681	23.15	1.3634	42.72	1.5901	58.30

Table D2.26: Summary of PARASODE results: extreme conditions, Owen's model, slope 1:2, CL=8m, OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12}=0.6$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.6E-05	0.5E-05	-0.4E-07	-0.1E-07	-0.2E-08	-0.1E-09
Standard Deviation of Z	0.163	0.274E-01	0.361E-02	0.417E-03	0.433E-04	0.418E-05
Relative Accuracy of Beta (%)	0.2E-04	0.4E-04	0.6E-05	0.2E-05	0.4E-05	0.7E-05
Probability of Failure P_f (%/Year)	2.924600	20.353686	45.865913	68.600770	84.688447	94.104100
Difference in P_f Between Last 2 Iterations	0.2E-07	0.9E-07	0.3E-08	0.4E-08	0.1E-07	0.1E-08

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_m$	6.521	—	5.549	—	4.970	—	4.549	—	4.211	—	3.937	—
$X_2 = H_s$	2.484	—	1.536	—	1.042	—	0.775	—	0.636	—	0.546	—
$Y_1 = f(T_m, H_s)$	—	89.58	—	87.49	—	79.57	—	73.25	—	69.29	—	65.22
$Y_2 = f(T_m, H_s)$	—	2.40	—	6.01	—	13.23	—	17.67	—	17.78	—	1+8+
$X_3 = Y_3 = A$	0.0017	0	0.0017	0	0.0017	0	0.0017	0	0.0017	0	0.0017	0
$X_4 = Y_4 = B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5 = Y_5 = SWL$	5.647	0.78	5.621	0.56	5.613	0.57	5.606	0.66	5.597	0.86	5.586	1.17
$X_6 = Y_6 = r$	0.9504	0.04	0.9500	0.03	0.9500	0.04	0.9489	0.05	0.9487	0.07	0.9485	0.10
$X_7 = Y_7 = \phi_a$	0.9489	6.80	0.9889	5.91	1.0122	6.59	1.0371	8.37	1.0703	12.00	1.121+	18.67

Table D2.27: Summary of PARASODE results: extreme conditions, Owen's model, slope 1:2, CL=10m OD (Note that the correlation coefficients between T_m and H_s is ρ_{12}=0.6).

Q (m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	-0.9E-05	0.3E-06	0.1E-06	-0.2E-08	-0.1E-08	-0.2E-09
Standard Deviation of Z	0.133	0.366E-01	0.377E-02	0.472E-03	0.545E-04	0.591E-05
Relative Accuracy of Beta (%)	0.4E-04	0.2E-04	0.2E-06	0.4E-06	0.2E-06	0.3E-06
Probability of Failure Pf (%/Year)	0.374653	6.788809	21.989960	40.989400	58.770594	73.4+16331
Difference in Pf Between Last 2 Iterations	0.1E-07	0.3E-07	0.4E-08	0.4E-08	0.1E-08	0.6E-08

Q (m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)	DP	α²(%)
$X_1 = T_m$	7.346	—	6.136	—	5.503	—	5.066	—	4.731	—	4.455	—
$X_2 = H_s$	2.967	—	2.106	—	1.487	—	1.114	—	0.880	—	0.729	—
$Y_1 = f(T_m, H_s)$	—	81.70	—	89.86	—	86.64	—	81.01	—	75.87	—	72.12
$Y_2 = f(T_m, H_s)$	—	5.12	—	2.30	—	6.07	—	11.37	—	15.53	—	17.50
$X_3 = Y_3 = A$	0.010	0	0.010	0	0.010	0	0.010	0	0.010	0	0.010	0
$X_4 = Y_4 = B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5 = Y_5 = SWL$	5.662	0.75	5.628	0.30	5.618	0.30	5.613	0.30	5.610	0.32	5.606	0.37
$X_6 = Y_6 = r$	0.9507	0.07	0.9508	0.04	0.9500	0.04	0.9500	0.04	0.9500	0.05	0.9498	0.06
$X_7 = Y_7 = \varepsilon_u$	0.8865	12.36	0.9578	7.45	0.9665	6.95	1.0073	7.28	1.0261	8.23	1.0463	9.95

Table D2.28: Summary of PARASODE results: extreme conditions. Owen's model, slope 1:2, CL=12m. OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12} = 0.6$).

Q(m³/s/m)	10⁻¹	10⁻²	10⁻³	10⁻⁴	10⁻⁵	10⁻⁶
Failure Function Z	-0.1E-05	-0.2E-05	0.4E-07	0.2E-07	0.5E-09	-0.1E-09
Standard Deviation of Z	0.111	0.243E-01	0.366E-02	0.481E-03	0.581E-04	0.664E-05
Relative Accuracy of Beta (%)	0.5E-04	0.1E-04	0.2E-04	0.3E-07	0.5E-05	0.2E-05
Probability of Failure P_f (%/Year)	0.031054	2.063650	9.979423	22.982653	38.001697	52.561676
Difference in P_f Between Last 2 Iterations	0.2E-08	0.1E-07	0.5E-07	0.6E-10	0.6E-08	0.5E-09

Q(m³/s/m)	10⁻¹		10⁻²		10⁻³		10⁻⁴		10⁻⁵		10⁻⁶	
	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)	DP	α^2(%)
$X_1 = T_m$	8.127	—	6.668	—	5.946	—	5.476	—	5.125	—	4.845	—
$X_2 = H_s$	3.204	—	2.570	—	1.910	—	1.459	—	1.159	—	0.953	—
$Y_1 = f(T_m, H_s)$	—	73.82	—	87.59	—	89.03	—	86.08	—	81.75	—	77.55
$Y_2 = f(T_m, H_s)$	—	5.58	—	2.68	—	2.76	—	6.11	—	10.27	—	13.89
$X_3 = Y_3 = A$	0.0017	0	0.0017	0	0.0017	0	0.0017	0	0.0017	0	0.0017	0
$X_4 = Y_4 = B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5 = Y_5 = SWL$	5.678	0.78	5.632	0.38	5.620	0.21	5.616	0.19	5.614	0.19	5.611	0.19
$X_6 = Y_6 = r$	0.9512	0.11	0.9506	0.05	0.9503	0.0+	0.9502	0.0+	0.9501	0.0+	0.9500	0.05
$X_7 = Y_7 = h_b$	0.8150	19.71	0.9253	10.00	0.9542	7.56	0.9666	7.58	1.0042	7.75	1.0192	8.32

Table D2.29: Summary of PARASODE results: extreme conditions, Owen's model, slope 1:2, CL=14m OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12} = 0.5$).

Q (m³/s/m)	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Failure Function Z	0.4E-05	-0.1E-05	-0.2E-05	-0.1E-07	0.2E-08	0.2E-09
Standard Deviation of Z	0.991E-01	0.543E-01	0.390E-02	0.473E-03	0.588E-04	0.688E-05
Relative Accuracy of Beta (%)	0.4E-05	0.2E-04	0.4E-05	0.2E-05	0.3E-05	0.6E-05
Probability of Failure Pf (%/year)	0.001842	0.54509+	4.31133+	12.43342	23.64836	36.06036
Difference in Pf Between Last 2 Iterations	0.1E-10	0.8E-08	0.7E-09	0.4E-08	0.7E-09	0.8E-08

Q (m³/s/m)	10^{-1}		10^{-2}		10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)	DP	α^2 (%)
$X_1 = T_m$	8.904	—	7.186	—	6.347	—	5.831	—	5.498	—	5.165	—
$X_2 = H_s$	3.299	—	2.860	—	2.288	—	1.793	—	1.441	—	1.190	—
$Y_1 = f(T_m, H_s)$	—	68.96	—	82.02	—	88.57	—	88.30	—	85.70	—	82.20
$Y_2 = f(T_m, H_s)$	—	3.74	—	3.70	—	1.73	—	3.20	—	6.13	—	9.55
$X_3 = Y_3 = A$	0.015	0	0.015	0	0.015	0	0.015	0	0.015	0	0.015	0
$X_4 = Y_4 = B$	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0	21.71	0
$X_5 = Y_5 = SWL$	5.691	0.75	5.638	0.27	5.623	0.16	5.618	0.14	5.615	0.13	5.613	0.12
$X_6 = Y_6 = r$	0.9516	0.15	0.9507	0.08	0.9504	0.05	0.9503	0.05	0.9502	0.04	0.9501	0.05
$X_7 = Y_7 = \theta_B$	0.7457	26.40	0.8851	13.93	0.9411	9.49	0.9458	8.31	0.9667	8.00	1.0013	8.08

Table D2.30: Summary of PARASODE results: extreme conditions, Owen's model, slope 1:2, CL=16m, OD (Note that the correlation coefficient between T_m and H_s is $\rho_{12} = 0.6$).

APPENDIX D3 - Examples Of Input And Output Files For Dune Erosion

Input File *general.dad*: Example 1

```
3               ! Failure mode of dune erosion
1               ! Total level
2               ! Movements of sand only seaward
1               ! Mode 1 - Reliability analysis for a specified design
1               ! Design life (in years)
N               ! No combinations of actions considered
0               ! Target design parameter - nourishment width
```

Input File *form.dad*: Example 1

```
2               ! Starting point: user specified values
0
2.25E-4
-0.3
0
0.4
0
4.2
2               ! [XMin,XMax]: user specified values
0       1E25
0       1E25
-1E25   1E25
-1E25   1E25
-1E25   1E25
-1E25   1E25
3       1E25
200             ! Maximum number of iterations
1               ! Number of FORM calculations
90
1               ! Accuracy on Beta (%): default value (1%)
1               ! Smoothing of the iteration: default value (0)
1               ! Accuracy on Z0 (%): default value (1%)
```

Input File *meandev.dad*: Example 1

```
1                                       ! Type of distribution (Normal)
0                                       ! Not truncated
0           0.60                        ! Mean value; standard deviation
1                                       ! Type of distribution (Normal)
0                                       ! Not truncated
```

```
2.25E-4       2.25E-5                            ! Mean value; standard deviation
1                                                ! Type of distribution (Normal)
0                                                ! Not truncated
0.            0.6                                ! Mean value; standard deviation
1                                                ! Type of distribution (Normal)
0                                                ! Not truncated
0.            1                                  ! Mean value; standard deviation
1                                                ! Type of distribution (Normal)
0                                                ! Not truncated
0.4           0.1                                ! Mean value; standard deviation
1                                                ! Type of distribution (Normal)
0                                                ! Not truncated
0.            1                                  ! Mean value; standard deviation
10                                               ! Type of distribution (Weibull)
0                                                ! Not truncated
2.52          0.33          2.19                 ! Mean; stand. dev.; lower limit
```

Input File *coefcor.dad*: Example 1

```
1       ! Rho(1,1)
0       ! Rho(1,2)
0       ! Rho(1,3)
0       !   .
0       !   .
0       !   .
0
0
1
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
1
0
0
0
0
0
0
0
1
0
0
0
```

```
0
0
0
0
1
0
0
0
0
0
0
0
1
```

Input File *perfil.dad*: Example 1

```
0                 !Coastal curvature in degrees per 1000m
21                !Number of points in the initial profile
-200    15        !Coordinates of the initial profile (XP,YP)
-150    13.5
-100    11
-75     13.5
-50     12
-25     15
-20     12.68571429
-10     8.057142857
0       3.428571429
10      -1.2
15      -1.7
50      -2.082474227
100     -2.628865979
150     -3.175257732
200     -3.721649485
250     -4.268041237
300     -4.81443299
350     -5.360824742
400     -5.907216495
450     -6.453608247
500     -7
2                 !Number of points to be changed in the initial profile
10                !First point to be changed, point No.
1.                !Gradient of the eroded dune face, 1:md
12.5              !Gradient of the toe of the post-storm profile, 1:mt
15                !Nourishment top level
1.5               !Gradient of the nourished face, 1:mnour
```

Output File *summary.dat*: Example 1

```
  WHAT IS THE DATA SOURCE ?

        The Screen ..... [ 1 ]
        A Datafile ..... [ 2 ]

  Select Option: 2



  WHAT IS THE FAILURE MODE TO BE STUDIED:

        Overtopping (H&R) .................... [ 1 ]
        Overtopping (Owen) ................... [ 2 ]
        Dune Erosion (Vellinga) .............. [ 3 ]
```

```
Select Option:  3



HOW IS THE STILL-WATER-LEVEL DEFINED ?

        Total Level .... [ 1 ]
        Tide + Surge ... [ 2 ]

Select Option: 1



DURING A STORM SURGE, WOULD YOU LIKE TO TAKE INTO ACCOUNT:

        Movements of Sand in Both Directions ? ... [ 1 ]
        Movements of Sand only Seaward ? ......... [ 2 ]

Select Option: 2

DUNE EROSION

        Coastal Curvature (Deg/1000m) =  0.0000000000E+00

        Number of Points Defining the Initial Profile (Max=100) =  21

                  Initial Profile

              X                     Y
        -0.2000000000E+03   0.1500000000E+02
        -0.1500000000E+03   0.1350000000E+02
        -0.1000000000E+03   0.1100000000E+02
        -0.7500000000E+02   0.1350000000E+02
        -0.5000000000E+02   0.1200000000E+02
        -0.2500000000E+02   0.1500000000E+02
        -0.2000000000E+02   0.1268571429E+02
        -0.1000000000E+02   0.8057142857E+01
         0.0000000000E+00   0.3428571429E+01
         0.1000000000E+02  -0.1200000000E+01
         0.1500000000E+02  -0.1700000000E+01
         0.5000000000E+02  -0.2082474227E+01
         0.1000000000E+03  -0.2628865979E+01
         0.1500000000E+03  -0.3175257732E+01
         0.2000000000E+03  -0.3721649485E+01
         0.2500000000E+03  -0.4268041237E+01
         0.3000000000E+03  -0.4814432990E+01
         0.3500000000E+03  -0.5360824742E+01
         0.4000000000E+03  -0.5907216495E+01
         0.4500000000E+03  -0.6453608247E+01
         0.5000000000E+03  -0.7000000000E+01


        Number of Points to be Changed in the Initial Profile =   2

        First Point to be Changed = Point No. 10

        Gradient of the Eroded Dune Face = 1: 1.0

        Gradient of the Toe of the Post-Storm Profile = 1:12.5
```

```
        Nourishment Top Level =  0.1500000000E+02


        Gradient of the Nourished Face = 1: 1.5



DESCRIPTION OF THE VARIABLES


        X(  1) = Hs  = Wave Height
        X(  2) = D50 = Particle Size
        X(  3) = DP  = Initial Profile
        X(  4) = SD  = Surge Duration
        X(  5) = GB  = Gust Bumps
        X(  6) = Ac  = Accuracy Comput.
        X(  7) = h   = Surge Level



WHAT IS THE PURPOSE OF THE ANALYSIS ?


        Reliability Analysis for a Specified Design ... [ 1 ]
        Design for a Specified Reliability Level ...... [ 2 ]

Select Option: 1


DESIGN LIFE OF THE STRUCTURE =    1

WOULD YOU LIKE TO CONSIDER COMBINATION OF ACTIONS (Y/N) ? N


PRESCRIBED VALUE OF THE DESIGN PARAMETER

        Nourishment Width  = 0.0000000000E+00


CHARACTERISTICS OF THE VARIABLES

        Probability Distribution of Hs  = Normal (Gaussian)
        Mean Value of Hs  = 0.0000000000E+00
        Standard Deviation of Hs  = 0.6000000000E+00


        Probability Distribution of D50 = Normal (Gaussian)
        Mean Value of D50 = 0.2250000000E-03
        Standard Deviation of D50 = 0.2250000000E-04


        Probability Distribution of DP  = Normal (Gaussian)
        Mean Value of DP  = 0.0000000000E+00
        Standard Deviation of DP  = 0.6000000000E+00


        Probability Distribution of SD  = Normal (Gaussian)
        Mean Value of SD  = 0.0000000000E+00
        Standard Deviation of SD  = 0.1000000000E+01


        Probability Distribution of GB  = Normal (Gaussian)
        Mean Value of GB  = 0.4000000000E+00
```

```
        Standard Deviation of GB  =  0.1000000000E+00


        Probability Distribution of Ac  = Normal (Gaussian)
        Mean Value of Ac  =  0.0000000000E+00
        Standard Deviation of Ac  =  0.1000000000E+01


        Probability Distribution of h   = Minima Type III (Weibull)
        Mean Value of h   =  0.2520000000E+01
        Standard Deviation of h   =  0.3300000000E+00
        Lower Limit on h   =  0.2190000000E+01


CORRELATION COEFFICIENTS

        (Hs ,Hs ) =  0.1000000000E+01
        (Hs ,D50) =  0.0000000000E+00
        (Hs ,DP ) =  0.0000000000E+00
        (Hs ,SD ) =  0.0000000000E+00
        (Hs ,GB ) =  0.0000000000E+00
        (Hs ,Ac ) =  0.0000000000E+00
        (Hs ,h  ) =  0.0000000000E+00
        (D50,Hs ) =  0.0000000000E+00
        (D50,D50) =  0.1000000000E+01
        (D50,DP ) =  0.0000000000E+00
        (D50,SD ) =  0.0000000000E+00
        (D50,GB ) =  0.0000000000E+00
        (D50,Ac ) =  0.0000000000E+00
        (D50,h  ) =  0.0000000000E+00
        (DP ,Hs ) =  0.0000000000E+00
        (DP ,D50) =  0.0000000000E+00
        (DP ,DP ) =  0.1000000000E+01
        (DP ,SD ) =  0.0000000000E+00
        (DP ,GB ) =  0.0000000000E+00
        (DP ,Ac ) =  0.0000000000E+00
        (DP ,h  ) =  0.0000000000E+00
        (SD ,Hs ) =  0.0000000000E+00
        (SD ,D50) =  0.0000000000E+00
        (SD ,DP ) =  0.0000000000E+00
        (SD ,SD ) =  0.1000000000E+01
        (SD ,GB ) =  0.0000000000E+00
        (SD ,Ac ) =  0.0000000000E+00
        (SD ,h  ) =  0.0000000000E+00
        (GB ,Hs ) =  0.0000000000E+00
        (GB ,D50) =  0.0000000000E+00
        (GB ,DP ) =  0.0000000000E+00
        (GB ,SD ) =  0.0000000000E+00
        (GB ,GB ) =  0.1000000000E+01
        (GB ,Ac ) =  0.0000000000E+00
        (GB ,h  ) =  0.0000000000E+00
        (Ac ,Hs ) =  0.0000000000E+00
        (Ac ,D50) =  0.0000000000E+00
        (Ac ,DP ) =  0.0000000000E+00
        (Ac ,SD ) =  0.0000000000E+00
        (Ac ,GB ) =  0.0000000000E+00
        (Ac ,Ac ) =  0.1000000000E+01
        (Ac ,h  ) =  0.0000000000E+00
```

```
          (h  ,Hs ) =  0.0000000000E+00
          (h  ,D50) =  0.0000000000E+00
          (h  ,DP ) =  0.0000000000E+00
          (h  ,SD ) =  0.0000000000E+00
          (h  ,GB ) =  0.0000000000E+00
          (h  ,Ac ) =  0.0000000000E+00
          (h  ,h  ) =  0.1000000000E+01


STARTING POINT FOR THE FORM CALCULATIONS:

          Default Values (mean values) ... [ 1 ]
          User Specified Values .......... [ 2 ]

Select Option: 2

STARTING POINT

          Hs  =  0.0000000000E+00
          D50 =  0.2250000000E-03
          DP  = -0.3000000000E+00
          SD  =  0.0000000000E+00
          GB  =  0.4000000000E+00
          Ac  =  0.0000000000E+00
          h   =  0.4200000000E+01

LIMITING VALUES FOR THE VARIABLES:

          Default Values (+/- 1E25) ... [ 1 ]
          User Specified Values ....... [ 2 ]

Select Option: 2

LIMITING VALUES FOR THE VARIABLES

          XMin(Hs ) =  0.0000000000E+00   XMax(Hs ) =  0.1000000000E+26
          XMin(D50) =  0.0000000000E+00   XMax(D50) =  0.1000000000E+26
          XMin(DP ) = -0.1000000000E+26   XMax(DP ) =  0.1000000000E+26
          XMin(SD ) = -0.1000000000E+26   XMax(SD ) =  0.1000000000E+26
          XMin(GB ) = -0.1000000000E+26   XMax(GB ) =  0.1000000000E+26
          XMin(Ac ) = -0.1000000000E+26   XMax(Ac ) =  0.1000000000E+26
          XMin(h  ) =  0.3000000000E+01   XMax(h  ) =  0.1000000000E+26


MAXIMUM NUMBER OF ITERATIONS (Max=200) = 200

NUMBER OF FORM CALCULATIONS (Max=10) =  1

ALLOWABLE EROSION DISTANCE - m ( 1) =  90.00


REQUIRED RELATIVE ACCURACY OF THE RELIABILITY INDEX:

          Default Value (1%) ......... [ 1 ]
          User Specified Value ....... [ 2 ]

Select Option: 1
```

```
REQUIRED SMOOTHING COEFFICIENT FOR THE ITERATION PROCESS:


        Default Value (0) .......... [ 1 ]
        User Specified Value ....... [ 2 ]


Select Option: 1




REQUIRED ACCURACY OF THE FAILURE FUNCTION:


        Default Value (1%) ......... [ 1 ]
        User Specified Value ....... [ 2 ]


Select Option: 1




ALLOWABLE EROSION DISTANCE - m ( 1) =  90.00



FINAL RESULTS


        Total Number of Iterations =   6
        Failure Function Z (X) =  0.5098040390E-07
        Mean Value of Z =  0.1053991340E+03
        Standard Deviation of Z =  0.3238986303E+02
        Reliability Index =  0.3254077794E+01
        Relative Accuracy of the Reliability Index (%) =  0.1540871846E-03
        Probability of Failure (%) =   0.056884
        Difference in Pf Between the Last 2 Iterations =  0.1002823371E-07



        DESIGN POINT COORDINATES


                Hs  =  0.2366829604E+00
                D50 =  0.2061481291E-03
                DP  = -0.7271166310E-01
                SD  =  0.6350647216E+00
                GB  =  0.4079383090E+00
                Ac  =  0.8719123188E+00
                h   =  0.4302783557E+01



        Alpha(Hs ) = -0.1212237770E+00
        Influence of Hs  on the Reliability Index =  0.1469520E+01


        Alpha(D50) =  0.2574803009E+00
        Influence of D50 on the Reliability Index =  0.6629611E+01


        Alpha(DP ) =  0.3724130548E-01
        Influence of DP  on the Reliability Index =  0.1386915E+00


        Alpha(SD ) = -0.1951596618E+00
        Influence of SD  on the Reliability Index =  0.3808729E+01


        Alpha(GB ) = -0.2439495770E-01
        Influence of GB  on the Reliability Index =  0.5951140E-01
```

```
Alpha(Ac ) = -0.2679445219E+00
Influence of Ac  on the Reliability Index =  0.7179427E+01


Alpha(h  ) = -0.8984125452E+00
Influence of h   on the Reliability Index =  0.8071451E+02
```

```
WOULD YOU LIKE TO RESTART (Y/N) ? N
```

APPENDIX D4 - Results From PARASODE For Dune Erosion

Variable	Design Point	α^2 (%)
H_S	0.237	1.47
D_{50}	0.206E-03	6.63
DP	-0.073	0.14
SD	0.635	3.81
GB	0.408	0.06
Ac	0.872	7.18
h	4.303	80.71

Total Number of Iterations = 6
Failure Function Z(X) = 0.5E-07
Mean Value of Z = 105.399
Standard Deviation of Z = 32.390
Reliability Index = 3.254
Relative Accuracy of the Reliability Index (%) = 0.2E-03
Probability of Failure (%/Year) = 0.056884
Difference in P_f Between the Last 2 Iterations = 0.1E-07

Table D4.1: Summary of PARASODE results for example 1.

Variable	Design Point	α^2 (%)
H_S	-0.029	0.01
D_{50}	0.215E-03	0.90
DP	-0.135	0.23
SD	0.779	2.82
GB	0.410	0.04
Ac	1.178	6.45
h	6.207	89.55

Total Number of Iterations = 7
Failure Function Z(X) = 0.5E-03
Mean Value of Z = 168.796
Standard Deviation of Z = 36.378
Reliability Index = 4.640
Relative Accuracy of the Reliability Index (%) = 0.2E-03
Probability of Failure (%/Year) = 0.000175
Difference in P_f Between the Last 2 Iterations = 0.7E-10

Table D4.2: Summary of PARASODE results for example 2.

Variable	Design Point	α^2 (%)
H_S	0.331	1.62
D_{50}	0.199E-03	7.07
DP	-0.052	0.04
SD	0.542	1.57
GB	0.407	0.02
Ac	0.780	3.26
h	5.669	86.42

Total Number of Iterations = 20
Failure Function Z(X) = 0.8E-04
Mean Value of Z = 224.852
Standard Deviation of Z = 51.968
Reliability Index = 4.327
Relative Accuracy of the Reliability Index (%) = 0.3E-04
Probability of Failure (%/Year) = 0.000761
Difference in P_f Between the Last 2 Iterations = 0.4E-10

Table D4.3: Summary of PARASODE results for example 3.

Variable	Design Point	α^2 (%)
H_S	0.298	1.63
D_{50}	0.202E-03	7.06
DP	-0.058	0.06
SD	0.584	2.26
GB	0.407	0.04
Ac	0.822	4.48
h	5.058	84.47

Total Number of Iterations = 7
Failure Function Z(X) = -0.3E-06
Mean Value of Z = 165.857
Standard Deviation of Z = 42.682
Reliability Index = 3.886
Relative Accuracy of the Reliability Index (%) = 0.2E-03
Probability of Failure (%/Year) = 0.005083
Difference in P_f Between the Last 2 Iterations = 0.2E-08

Table D4.4: Summary of PARASODE results for example 4.

Variable	Design Point	α^2 (%)
H_S	0.337	1.73
D_{50}	0.198E-03	7.97
DP	0.000	0.00
SD	0.607	2.03
GB	0.408	0.03
Ac	0.855	4.02
h	5.512	84.22

Total Number of Iterations = 8
Failure Function Z(X) = -0.7E-07
Mean Value of Z = 204.380
Standard Deviation of Z = 47.939
Reliability Index = 4.263
Relative Accuracy of the Reliability Index (%) = 0.2E-03
Probability of Failure (%/Year) = 0.001011
Difference in P_f Between the Last 2 Iterations = 0.4E-09

Table D4.5: Summary of PARASODE results for example 5.

Variable	Design Point	α^2 (%)
H_S	0.285	1.17
D_{50}	0.201E-03	5.82
DP	-0.061	0.05
SD	0.906	4.26
GB	0.411	0.07
Ac	1.177	7.19
h	5.581	81.44

Total Number of Iterations = 14
Failure Function Z(X) = -0.03
Mean Value of Z = 203.719
Standard Deviation of Z = 46.421
Reliability Index = 4.388
Relative Accuracy of the Reliability Index (%) = 0.6E-03
Probability of Failure (%/Year) = 0.000573
Difference in P_f Between the Last 2 Iterations = 0.7E-09

Table D4.6: Summary of PARASODE results for example 6.

Variable	Design Point	α^2 (%)
H_S	0.273	1.19
D_{50}	0.202E-03	5.80
DP	-0.063	0.06
SD	0.873	4.39
GB	0.411	0.07
Ac	1.143	7.52
h	5.291	80.97

Total Number of Iterations = 16
Failure Function Z(X) = 0.051
Mean Value of Z = 189.313
Standard Deviation of Z = 45.425
Reliability Index = 4.168
Relative Accuracy of the Reliability Index (%) = 0.8E-07
Probability of Failure (%/Year) = 0.001547
Difference in P_f Between the Last 2 Iterations = 0.2E-12

Table D4.7: Summary of PARASODE results for example 7.

Variable	Design Point	α^2 (%)
H_S	0.264	1.24
D_{50}	0.203E-03	5.90
DP	-0.065	0.08
SD	0.805	4.15
GB	0.410	0.06
Ac	1.063	7.24
h	5.043	81.33

Total Number of Iterations = 11
Failure Function Z(X) = 0.157
Mean Value of Z = 183.466
Standard Deviation of Z = 46.447
Reliability Index = 3.950
Relative Accuracy of the Reliability Index (%) = 0.3E-05
Probability of Failure (%/Year) = 0.003900
Difference in P_f Between the Last 2 Iterations = 0.2E-10

Table D4.8: Summary of PARASODE results for example 8.

Variable	Design Point	α^2 (%)
H_S	0.256	1.30
D_{50}	0.204E-03	6.09
DP	-0.067	0.09
SD	0.748	4.00
GB	0.409	0.06
Ac	0.997	7.11
h	4.811	81.35

Total Number of Iterations = 9
Failure Function Z(X) = 0.4E-06
Mean Value of Z = 176.884
Standard Deviation of Z = 47.326
Reliability Index = 3.738
Relative Accuracy of the Reliability Index (%) = 0.2E-03
Probability of Failure (%/Year) = 0.009298
Difference in P_f Between the Last 2 Iterations = 0.3E-08

Table D4.9: Summary of PARASODE results for example 9.

Variable	Design Point	α^2 (%)
H_S	0.248	1.37
D_{50}	0.205E-03	6.30
DP	-0.072	0.11
SD	0.693	3.85
GB	0.409	0.06
Ac	0.934	7.01
h	4.589	81.30

Total Number of Iterations = 19
Failure Function Z(X) = -0.7E-02
Mean Value of Z = 168.325
Standard Deviation of Z = 47.707
Reliability Index = 3.528
Relative Accuracy of the Reliability Index (%) = 0.9E-04
Probability of Failure (%/Year) = 0.020938
Difference in P_f Between the Last 2 Iterations = 0.3E-08

Table D4.10: Summary of PARASODE results for example 10.

Variable	Design Point	α^2 (%)
H_S	0.219	1.59
D_{50}	0.208E-03	6.96
DP	-0.076	0.19
SD	0.588	4.12
GB	0.407	0.06
Ac	0.830	8.21
h	3.947	78.87

Total Number of Iterations = 6
Failure Function Z(X) = -0.2E-07
Mean Value of Z = 69.676
Standard Deviation of Z = 24.053
Reliability Index = 2.897
Relative Accuracy of the Reliability Index (%) = 0.8E-04
Probability of Failure (%/Year) = 0.188552
Difference in P_f Between the Last 2 Iterations = 0.1E-07

Table D4.11: Summary of PARASODE results for example 11.

Variable	Design Point	α^2 (%)
H_S	0.195	1.77
D_{50}	0.210E-03	7.45
DP	-0.079	0.28
SD	0.533	4.72
GB	0.407	0.07
Ac	0.780	10.13
h	3.552	75.58

Total Number of Iterations = 6
Failure Function Z(X) = -0.2E-07
Mean Value of Z = 53.567
Standard Deviation of Z = 21.858
Reliability Index = 2.451
Relative Accuracy of the Reliability Index (%) = 1.0E-04
Probability of Failure (%/Year) = 0.712880
Difference in P_f Between the Last 2 Iterations = 0.5E-07

Table D4.12: Summary of PARASODE results for example 12.

Variable	Design Point	α^2 (%)
H_S	0.169	2.09
D_{50}	0.212E-03	8.39
DP	-0.079	0.46
SD	0.463	5.63
GB	0.406	0.09
Ac	0.707	13.15
h	3.178	70.19

Total Number of Iterations =   6
Failure Function Z(X) = -0.4E-08
Mean Value of Z = 35.121
Standard Deviation of Z = 18.005
Reliability Index = 1.951
Relative Accuracy of the Reliability Index (%) = 0.9E-04
Probability of Failure (%/Year) =   2.555196
Difference in P_f Between the Last 2 Iterations = 1.0E-07

Table D4.13: Summary of PARASODE results for example 13.

Variable	Design Point	α^2 (%)
H_S	0.336	1.73
D_{50}	0.198E-03	7.98
DP	0.000	0.00
SD	0.608	2.05
GB	0.408	0.03
Ac	0.857	4.06
h	5.498	84.15

Total Number of Iterations = 18
Failure Function Z(X) = -0.3E-02
Mean Value of Z = 203.088
Standard Deviation of Z = 47.734
Target Probability of Failure (%/Year) =   0.001011
Reliability Index = 4.255
Design Parameter = 74.87

Table D4.14: Summary of PARASODE results for example 14.

APPENDIX E

Results From @Risk

CONVERGENCE OF Q STATISTICS USING LATIN HYPERCUBE SAMPLING

NORMAL CONDITIONS: H&R MODEL, $(R_{max})_{37\%}$ SLOPE 1:2, CL=10m OD

Number Of Samples	Mean Of Samples	Standard Deviation Of Samples	Coefficient Of Variation Of Samples
100	2.59E-06	2.57E-05	9.95E+00
500	8.51E-05	1.90E-03	2.23E+01
1000	4.48E-05	7.93E-04	1.77E+01
2000	2.30E-05	5.12E-04	2.23E+01
3000	4.80E-05	1.65E-03	3.44E+01
4000	4.65E-05	1.29E-03	2.78E+01
5000	4.78E-05	1.03E-03	2.16E+01
6000	4.28E-05	1.17E-03	2.74E+01
7000	2.23E-05	7.16E-04	3.21E+01
8000	3.45E-05	8.82E-04	2.56E+01
9000	3.15E-05	8.28E-04	2.63E+01
10000	3.10E-05	6.56E-04	2.12E+01
20000	2.83E-05	6.83E-04	2.41E+01
30000	2.95E-05	8.00E-04	2.71E+01
40000	3.88E-05	1.03E-03	2.65E+01
50000	4.93E-05	1.52E-03	3.08E+01
60000	3.78E-05	1.10E-03	2.91E+01
70000	4.01E-05	1.16E-03	2.90E+01
80000	4.30E-05	1.40E-03	3.25E+01
90000	4.11E-05	1.14E-03	2.76E+01
100000	4.28E-05	1.30E-03	3.03E+01

Table E1: Example of the convergence of the mean, standard deviation and coefficient of variation of Q using Latin Hypercube Sampling for normal conditions, H&R model, $(R_{max})_{37\%}$, slope 1:2, CL=10m OD.

CONVERGENCE OF THE PROBABILITY OF FAILURE (%/YEAR) USING LATIN HYPERCUBE SAMPLING

NORMAL CONDITIONS: H&R MODEL, $(R_{max})_{37\%}$
SLOPE 1:2, CL=10m OD

Number Of Samples	Allowable Discharge, Q (m^3/s/m)					
	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
10000	0.00E+00	1.00E-01	3.70E-01	7.70E-01	1.03E+00	1.23E+00
20000	0.00E+00	9.00E-02	3.70E-01	6.45E-01	8.40E-01	1.02E+00
30000	0.00E+00	7.67E-02	3.27E-01	6.10E-01	7.97E-01	9.77E-01
40000	0.00E+00	1.05E-01	3.78E-01	6.53E-01	8.80E-01	1.03E+00
50000	6.00E-03	1.12E-01	3.88E-01	6.84E-01	8.94E-01	1.08E+00
60000	0.00E+00	8.67E-02	3.67E-01	6.62E-01	9.02E-01	1.08E+00
70000	1.43E-03	1.00E-01	3.77E-01	6.57E-01	8.57E-01	1.01E+00
80000	2.50E-03	1.00E-01	3.74E-01	6.44E-01	9.04E-01	1.06E+00
90000	1.11E-03	1.12E-01	3.86E-01	6.69E-01	9.00E-01	1.08E+00
100000	3.00E-03	1.15E-01	3.65E-01	6.82E-01	9.45E-01	1.11E+00

Table E2: Example of the convergence of the probability of failure for different allowable discharges using Latin Hypercube Sampling for normal conditions, H&R model, $(R_{max})_{37\%}$, slope 1:2, CL=10m OD.

CONVERGENCE OF Q STATISTICS USING LATIN HYPERCUBE SAMPLING

NORMAL CONDITIONS: OWEN'S MODEL
SLOPE 1:2, CL=10m OD

Number Of Samples	Mean Of Samples	Standard Deviation Of Samples	Coefficient Of Variation Of Samples
100	1.57E-04	8.52E-04	5.44E+00
500	2.62E-04	1.64E-03	6.25E+00
1000	4.95E-04	7.32E-03	1.48E+01
2000	2.57E-04	2.12E-03	8.23E+00
3000	2.76E-04	2.30E-03	8.34E+00
4000	3.26E-04	3.11E-03	9.54E+00
5000	2.91E-04	3.09E-03	1.06E+01
6000	3.77E-04	4.18E-03	1.11E+01
7000	3.56E-04	4.14E-03	1.16E+01
8000	3.21E-04	2.84E-03	8.84E+00
9000	3.64E-04	3.91E-03	1.07E+01
10000	2.68E-04	2.77E-03	1.03E+01
20000	3.20E-04	3.50E-03	1.09E+01
30000	2.91E-04	3.15E-03	1.08E+01
40000	3.14E-04	3.35E-03	1.07E+01
50000	3.00E-04	3.10E-03	1.03E+01
75000	3.01E-04	3.22E-03	1.07E+01
100000	2.99E-04	3.04E-03	1.02E+01

Table E3: Example of the convergence of the mean, standard deviation and coefficient of variation of Q using Latin Hypercube Sampling for normal conditions, Owen's model, slope 1:2, CL=10m OD.

CONVERGENCE OF THE PROBABILITY OF FAILURE (%/YEAR) USING LATIN HYPERCUBE SAMPLING

NORMAL CONDITIONS: OWEN'S MODEL
SLOPE 1:2, CL=10m OD

Number Of Samples	Allowable Discharge, Q (m³/s/m)					
	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
10000	1.00E-02	5.60E-01	3.08E+00	8.48E+00	1.50E+01	2.19E+01
20000	1.50E-02	7.10E-01	3.67E+00	8.69E+00	1.57E+01	2.26E+01
30000	1.33E-02	6.77E-01	3.58E+00	8.92E+00	1.57E+01	2.27E+01
40000	2.25E-02	6.95E-01	3.49E+00	8.78E+00	1.53E+01	2.23E+01
50000	1.60E-02	6.62E-01	3.62E+00	8.99E+00	1.55E+01	2.24E+01
75000	2.13E-02	6.99E-01	3.50E+00	8.82E+00	1.55E+01	2.25E+01
100000	2.10E-02	6.67E-01	3.59E+00	8.95E+00	1.55E+01	2.22E+01

Table E4: Example of the convergence of the probability of failure for different allowable discharges using Latin Hypercube Sampling for normal conditions, Owen's model, slope 1:2, CL=10m OD.

PROBABILITY OF FAILURE (%/YEAR) FOR NORMAL CONDITIONS
H&R MODEL, $(R_{max})_{37\%}$, SLOPE 1:2

CL (m OD)	Allowable Discharge, Q (m³/s/m)					
	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
8	0.05666	0.84167	2.15000	3.18166	3.85833	4.33667
10	0.00000	0.08667	0.36667	0.66167	0.90166	1.07833
12	0.00000	0.00833	0.04166	0.08667	0.15334	0.19666
14	0.00000	0.00000	0.00166	0.00833	0.01667	0.02000
16	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

Table E5: Probabilities of failure produced by @Risk, Latin Hypercube Sampling (60000 samples) for normal conditions, H&R model, $(R_{max})_{37\%}$, slope 1:2.

PROBABILITY OF FAILURE (%/YEAR) FOR NORMAL CONDITIONS
H&R MODEL, $(R_{max})_{99\%}$, SLOPE 1:2

CL (m OD)	Allowable Discharge, Q (m^3/s/m)					
	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
8	0.23333	3.05334	10.00333	18.73333	26.98666	33.62334
10	0.01334	0.67667	3.57667	8.92333	15.68000	22.67000
12	0.00333	0.13000	1.05000	3.94000	8.34333	13.84666
14	0.00000	0.01667	0.32000	1.54667	4.23000	7.89667
16	0.00000	0.00333	0.09666	0.61000	1.91333	4.41666

Table E6: Probabilities of failure produced by @Risk, Latin Hypercube Sampling (60000 samples) for normal conditions, H&R model, $(R_{max})_{99\%}$, slope 1:2.

PROBABILITY OF FAILURE (%/YEAR) FOR NORMAL CONDITIONS
OWEN'S MODEL, SLOPE 1:2

CL (m OD)	Allowable Discharge, Q (m^3/s/m)					
	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
8	0.23333	3.05334	10.00333	18.73333	26.98666	33.62334
10	0.01334	0.67667	3.57667	8.92333	15.68000	22.67000
12	0.00333	0.13000	1.05000	3.94000	8.34333	13.84666
14	0.00000	0.01667	0.32000	1.54667	4.23000	7.89667
16	0.00000	0.00333	0.09666	0.61000	1.91333	4.41666

Table E7: Probabilities of failure produced by @Risk, Latin Hypercube Sampling (30000 samples) for normal conditions, Owen's model, slope 1:2.

APPENDIX F

Published Papers

A new regression model for wave overtopping data

T.S. Hedges[a] & M.T. Reis[b]

[a] *Department of Civil Engineering, University of Liverpool, P.O.Box 147, Liverpool, L69 3BX, UK*

[b] *National Laboratory of Civil Engineering, Av do Brasil 101, 1799 Lisbon, Portugal*

Abstract

Random waves with a significant height H_s produce a maximum run-up of CH_s on the face of a coastal structure, where coefficient C is determined by the wave and wall characteristics. Unless this run-up is greater than R_C, the freeboard of the structure, then there is no overtopping (apart from wind-blown spray). That is $1 - R_c / CH_s > 0$ for overtopping to occur. The parameter $1 - R_c / CH_s$ is used in an analysis of overtopping data and a new regression model is evaluated which, unlike existing expressions, satisfies the relevant physical boundary conditions. The new model is inherently suitable for representing the small overtopping discharges associated with normal design conditions.

1 Introduction

An important criterion for the design of a seawall is the allowable degree of wave overtopping which depends upon the activities normally performed in the lee of the structure, the need to prevent erosion of the rear face of the seawall, and the economic consequences of flooding. During 1978 and 1979, Owen (at HRS[1], Wallingford) carried out an extensive series of model tests to determine the overtopping discharges for a range of seawall designs subjected to different random wave climates. The modelled seawalls were all of the same general type: a flat-topped embankment fronted in some cases by a flat berm. The tests were aimed at establishing the impact on overtopping discharge of the wave climate, the seawall slope, the crest and berm elevations, and the berm width.

This paper presents a re-analysis of some of Owen's data: the results for simple seawalls possessing uniform seaward slopes of 1:1, 1:2 and 1:4, subjected to random waves approaching normal to the slope. The purpose has been to construct a new regression model to represent the data which is more

reliable than Owen's expression. Care has been taken to consider the physical boundary conditions. Regression coefficients determined using the least-absolute-deviations (LAD) method are recommended in preference to those obtained using the least-squares (LS) technique.

2 Dimensional analysis

In general, the mean overtopping discharge per unit length of seawall, Q, depends upon the wave motion, the seawall profile, the foreshore characteristics and the water properties. Written in symbolic form:

$$Q = \text{function}\,(H_s, T_m, \beta, R_c, \alpha, d_s, g, ...) \tag{1}$$

H_s is the significant height of the incident waves; T_m is the mean zero-crossing wave period; β is the angle of wave approach measured normal to the seawall; R_c is the seawall's freeboard (the height of the crest of the structure above the still-water-level); α is the angle of the seawall front slope measured from the horizontal; d_s is the still-water-depth at the toe of the structure; and g is the acceleration due to gravity. Alternatively,

$$\frac{Q}{\sqrt{gH_s^3}} = \text{function}\,(\frac{R_c}{H_s}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, ...) \tag{2}$$

H_s / gT_m^2 is a measure of the incident wave steepness. Owen combined this dimensionless group both with $Q / \sqrt{gH_s^3}$ and with R_c / H_s, to write:

$$\frac{Q}{T_m gH_s} = \text{function}\,(\frac{R_c}{T_m\sqrt{gH_s}}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, ...) \tag{3}$$

However, other arrangements are possible (Hedges & Reis[2]), including use of the wave period of peak spectral density, T_p, rather than the zero-crossing period, T_m. The dimensionless groups are generally related using one of the two following functions:

$$Q_* = A\exp(-BR_*) \tag{4}$$
$$Q_* = A\,(R_*)^{-B} \tag{5}$$

where Q_* is the dimensionless overtopping discharge, R_* is the dimensionless freeboard, and A and B are best-fit coefficients determined from the experimental data.

Dimensional analysis provides no means for determining which sets of dimensionless groups may be especially informative or helpful in dealing with a particular data set. A possible problem in using certain pairings of groups is the potential for spurious correlation. A spurious correlation may arise when dimensionless groups plotted against one another contain a common variable.

Care must be taken in interpreting such plots. Scatter in the data may be suppressed simply by the presence of this variable.

3 Regression analysis

3.1 Introduction

Once experimental data have been collected, they may be used to confirm the validity of some theory or, where no satisfactory theory exists, they may be used to construct regression models. However, it is always useful to have some theoretical basis for choosing amongst the possible models. Furthermore, there are many techniques available for fitting regression models. In describing a regression model, care should be taken to emphasise the range of conditions over which there are data to support its use.

As a start, let us consider the physical boundary conditions to be met in addressing wave overtopping:

i) when the embankment has a large freeboard (i.e. when its crest elevation is well above the level of wave uprush), the predicted overtopping discharge should be zero (assuming that the effects of wind-blown spray are ignored);

ii) when the embankment has zero freeboard (i.e. when still-water-level is at the crest level of the embankment) then the predicted overtopping discharge may be large but should still remain finite.

Equations (4) and (5) represent two of the more common functions used to predict wave overtopping. However, when R_* is large, both expressions suggest that the discharge will be finite rather than zero (though it is small provided that A is not very large and provided also that B>1). When R_* is zero, the first of these expressions gives $Q_* = A$, a finite quantity, whilst the second expression gives $Q_* = \infty$. Thus neither expression satisfies both boundary conditions, with the second of them satisfying neither. Since most seawalls are designed to permit only relatively small overtopping discharges, the first of the two boundary conditions is likely to be the more important to satisfy.

In addition to considering the boundary conditions, we also need to establish the line of "best fit" to the observed data. There are many criteria for defining the best fit. One possibility is to minimise the sum of the squared deviations of the observations from the values predicted from our expression, but real data usually do not completely satisfy the classical assumptions for LS fitting. Reliable inferences may be drawn from regression models fitted by the LS method only if the assumptions are valid. Furthermore, an LS fitting has the disadvantage that the result is not "robust": it is sensitive to outlying data points. Whilst we could remove "outliers", such a procedure should only be considered if there is reason to doubt their validity. Such data must not be removed merely because they do not support our regression model: it may be the model which is wrong.

Minimising the sum of the absolute deviations rather than the sum of the squared deviations does not rely upon the Gaussian error assumption and allows us to retain outliers but prevents these points from exerting a disproportionate influence on the values of the regression coefficients. If the errors are assumed to follow a double exponential distribution, which has thicker tails than the Gaussian distribution, then the parameter values are maximum likelihood estimates. In this paper, we have chosen to fit our regression lines using the LAD method. However, we have also compared these results with an LS fitting.

3.2   A new regression model

3.2.1   A simple overtopping theory for regular waves

Let us step back from the complications of random waves to the simpler problem of regular waves of height H approaching normal to a seawall. We will assume that the instantaneous discharge of water over unit length of the seawall, q, is given by the weir formula:

$$q = C_d \frac{2}{3}\sqrt{2g}\,(\eta - R_c)^{3/2} \quad \text{for } \eta > R_c \tag{6}$$

in which η is the water surface elevation above still-water-level at the seawall (a periodic function of time); C_d is a discharge coefficient. Obviously, overtopping occurs only when the water surface is above the structure's crest. We will also assume that $\eta = kHF(t)$ in which F(t) denotes a function of time, t. For simple, sinusoidal, progressive waves, k=0.5 and F(t)=cos(2πt/T), where T is the wave period. However, following
Kikkawa et al[3], we will adopt the simpler form for F(t) shown in Figure 1;
k remains a coefficient determined by the particular wave and wall details. Then, the mean discharge, Q, is determined as follows:

$$Q = C_d \frac{2}{3}\sqrt{2g}\,\frac{1}{T}\int_{t_1}^{t_2} \left\{ kHF(t) - R_c \right\}^{3/2} dt \tag{7}$$

in which $t_1 < t < t_2$ corresponds to the interval during each wave period for which $kHF(t) > R_c$. Using the form for F(t) given in Figure 1 then yields:

$$\frac{Q}{\sqrt{g(kH)^3}} = C_d \frac{2\sqrt{2}}{15}\left\{ 1 - \frac{R_c}{kH} \right\}^{5/2} \quad \text{for } 0 < R_c < kH$$

$$= 0 \qquad\qquad\qquad \text{for } R_c \geq kH \tag{8}$$

Note that overtopping occurs only when $R_c < kH$. In other words, kH represents the run-up on the face of the seawall.
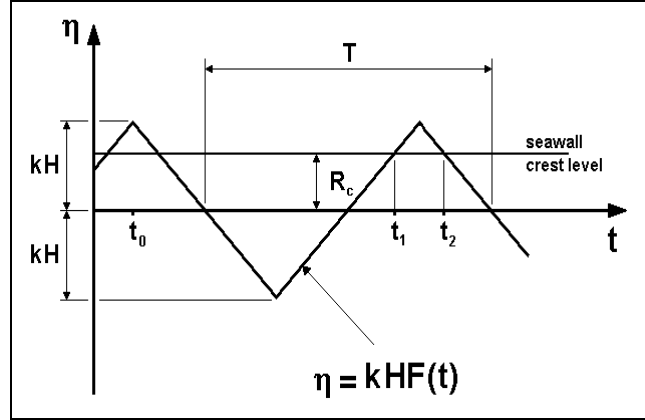
Figure 1: Form of function F(t).

3.2.2 The Hedges & Reis (H&R) overtopping model

The above theory suggests a regression equation for the random overtopping data of the following form:

$$
\begin{aligned}
Q_* &= A\left(1 - R_*\right)^B &&\text{for } 0 \le R_* < 1 \\
&= 0 &&\text{for } R_* \ge 1
\end{aligned}
\tag{9}
$$

in which

$$
Q_* = \frac{Q}{\sqrt{gR_{max}^3}} = \frac{Q}{\sqrt{g(CH_s)^3}}
\qquad\qquad
R_* = \frac{R_c}{R_{max}} = \frac{R_c}{CH_s}
\tag{10}
$$

Coefficient k in the expression for regular waves has been replaced by C in this regression model for random waves characterised by H_s. Note that CH_s represents R_{max}, the maximum run-up induced by the random waves, not the run-up induced by a wave of height H_s. Consequently, C will depend upon the duration of the incident wave conditions unless the wave heights in front of the wall are limited by the available water depth. Until the maximum run-up exceeds the freeboard, R_c, there will be no overtopping. It is also clear that coefficient B is related, in the case of regular waves, to the shape of the function F(t) which describes the water surface variation on the seaward face of the wall. There will be a similar dependence on the detailed behaviour of the water surface at the face of the wall in the case of random waves. Finally, coefficient A represents the dimensionless discharge over the seawall when the freeboard is zero. All three coefficients will be influenced by the seaward profile of the structure.

The above model for overtopping has the advantage that $Q_* = 0$ when $R_* \ge 1$ and that $Q_* = A$ when $R_* = 0$, in accordance with our required boundary conditions. Figure 2 shows the influences of coefficients A, B and C in the new overtopping model.
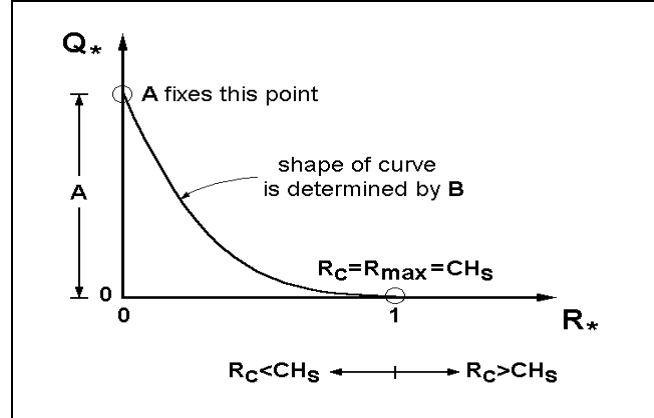
Figure 2: Influences of coefficients A, B and C in the new overtopping model.

Although C (=R_{max}/H_s) is not normally evaluated during tests involving wave overtopping, its value may be estimated from run-up measurements for random waves acting on slopes for which there is no overtopping. We have adopted this option rather than including C alongside A and B as a regression coefficient.

Owen recorded his overtopping discharges during tests involving sets of five different runs, each of 100 waves, characterised by the same value of H_s. Assuming that run-up values may be described by a Rayleigh distribution, then the expected maximum run-up, R_{max}, during each run is related to the significant run-up, R_s, by $R_{max} = \sqrt{(\ln 100)/2}\ R_s = 1.52\,R_s$. The CIRIA/CUR[4] manual gives two equations for evaluating R_s for smooth slopes without overtopping. Rewritten in our notation and allowing for a printing error, the following expressions for C may be derived:

$$C = 1.52\,R_s\,/\,H_s\ = 1.52\,(1.35\,\xi_p) \qquad \text{for}\ \xi_p < 2$$
$$C = 1.52\,R_s\,/\,H_s\ = 1.52\,(3.00 - 0.15\,\xi_p) \quad \text{for}\ \xi_p > 2$$

(11)

Here, ξ_p is the surf similarity parameter calculated using T_p ($\xi_p = \tan\alpha\,/\,\sqrt{H_s\,/\,L_{op}}$; $L_{op} = gT_p^2\,/\,2\pi$). T_p was estimated for Owen's data using the relationships between H_s, T_m and T_p provided by Isherwood[5].

4 Results of analysis

Figure 3 shows an example of the overtopping data collected by Owen: the results for a simple seawall with a uniform front slope of 1:2. The data are plotted in the formats required for fitting regression equations using both the H&R and the Owen models. Figure 3(a) shows the best-fit lines established using LS and LAD procedures for the H&R model. Comparison of the regression coefficients shows the relatively stronger influence which outlying data points have on the LS values. For example, the magnitude of B obtained

from LAD is only about 92% of the LS result. Similar comments may be made about the regression lines obtained for Owen's model. Note, that the values of A and B reported in Figure 3(b) for Owen's model are not those which Owen himself recommended.

(a)

H&R Model - Slope 1:2

$Ln(Q_*)$

Least Squares (A=0.00790 B=4.55)

Least Absolute Deviations (A=0.00753 B=4.17)

$$Q_* = A(1-R_*)^B$$
$$Q_* = \frac{Q}{\sqrt{g(CH_S)^3}}$$
$$R_* = \frac{R_c}{CH_S}$$

$Ln(1-R_*)$

(b)

Owen's Model - Slope 1:2

$Ln(Q_*)$

$$Q_* = A\exp(-BR_*)$$
$$Q_* = \frac{Q}{T_m g H_s}$$
$$R_* = \frac{R_c}{T_m\sqrt{gH_s}}$$

Least Squares (A=0.0125 B=22.80)

Least Absolute Deviations (A=0.0117 B=21.71)

R_*
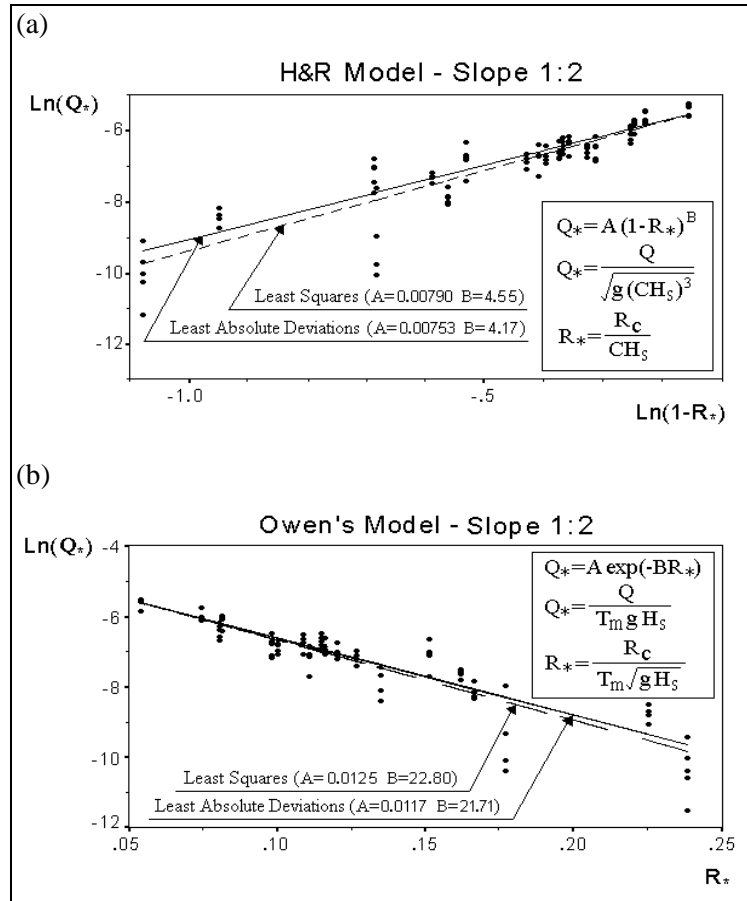
Figure 3: Wave overtopping data for slope 1:2 plotted in the formats required for fitting regression equations.

Table 1: Regression coefficients obtained for the H&R model and for Owen's model. Also included are Owen's recommended values.

		H&R MODEL		OWEN'S MODEL		
		LAD	LS	LAD	LS	Recommended
Slope 1:1	**A**	0.00703	0.00581	0.00777	0.00758	0.00794
	B	3.42	3.22	20.44	21.27	20.12
Slope 1:2	**A**	0.00753	0.00790	0.0117	0.0125	0.0125
	B	4.17	4.55	21.71	22.80	22.06
Slope 1:4	**A**	0.0104	0.00792	0.0134	0.0164	0.0192
	B	6.27	5.94	42.92	46.12	46.96

Table 1 gives the regression coefficients for all three slopes which we have obtained for the H&R model and for Owen's model, using both LS and

LAD fitting. Also included for reference are Owen's recommended values. Owen restricted his analysis to a particular set of conditions whilst we have included all available data. These data fell within the following ranges:

$$0 < \frac{Q}{\sqrt{g(CH_s)^3}} < 0.0056 \qquad 0.14 < \frac{R_c}{CH_s} < 0.90$$

$$0 < \frac{Q}{T_m g H_s} < 0.0039 \qquad 0.053 < \frac{R_c}{T_m \sqrt{gH_s}} < 0.239 \qquad (12)$$

$$0.0053 < \frac{H_s}{gT_m^2} < 0.0095 \qquad 1.65 < \frac{d_s}{H_s} < 5.20$$
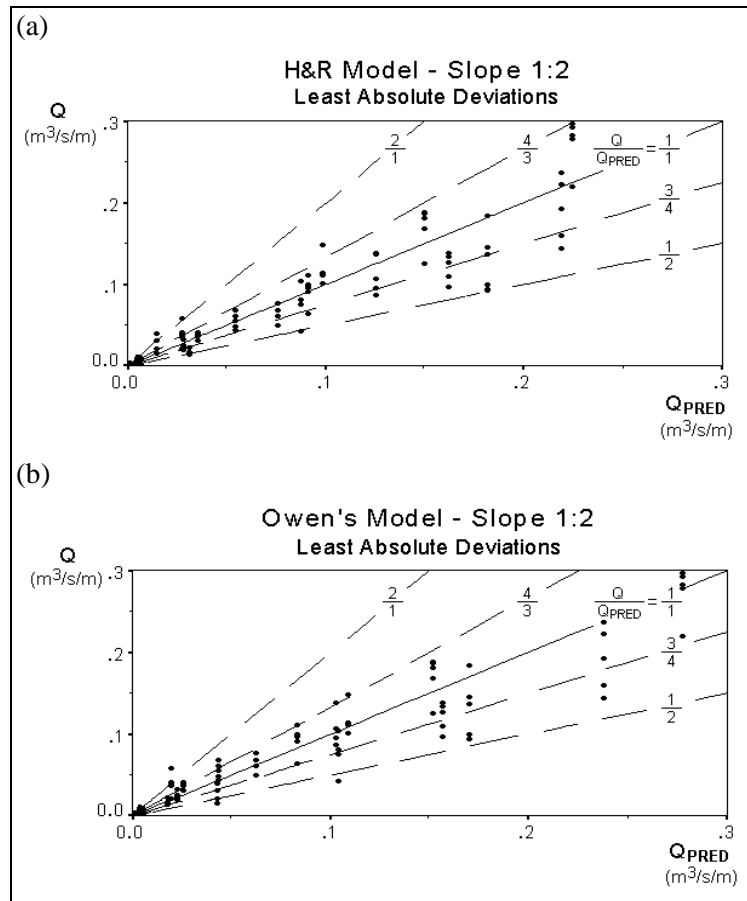
Figure 4: Wave overtopping data for slope 1:2, showing the level of agreement between Q and Q_{PRED}.

Earlier, we have mentioned briefly the problem of spurious correlation. Along with most other overtopping models, the H&R model employs a dimensionless discharge and a dimensionless freeboard which contain a common variable (R_{max} or CH_s). The presence of this common variable may reduce the apparent scatter in the data. Consequently, in Figure 4 we show

directly the level of agreement between Owen's measured values of Q (converted by Owen to full-scale discharges for a seawall in 4m water depth) and the predicted values, Q_{PRED}. Under random wave conditions, overtopping will be dominated by the few waves with large run-ups: most waves will contribute no overtopping if the seawall has a substantial freeboard. Thus, particularly for short runs of random waves, as in Owen's tests, we can expect some variability in the measured values of Q. Indeed, one of the purposes of Owen's tests was to show this inherent variability.

In Figure 4, most data points lie within a range for Q/Q_{PRED} of 3/4 to 4/3, whichever model is adopted. It is not obvious from the figure which model best fits the data, nor is it obvious from the plots for simple seawalls with 1:1 and 1:4 front slopes. However, it should be noted that full-scale discharges greater than about $0.001 \times 10^{-3} m^3/s/m$ will be unsafe for vehicles at high speed. Conditions become dangerous for pedestrians when the discharge exceeds $0.03 \times 10^{-3} m^3/s/m$. Discharges greater than about
$2 \times 10^{-3} m^3/s/m$ may damage embankment seawalls (CIRIA/CUR[4]). Consequently, we have looked in more detail at the data points for discharges in the ranges of practical interest. The H&R model appears generally better than Owen's model for discharges of less than
$5 \times 10^{-3} m^3/s/m$, owing to its ability to predict zero overtopping at finite values of freeboard. Furthermore, it tends to give lower required crest levels than Owen's model for small permissible discharges.

5 Concluding remarks

A new regression model has been developed for describing wave overtopping data. The important features of the model are as follows:
i)   it satisfies the relevant physical boundary conditions, a feature which is especially important when the model is used near these boundaries;
ii) it explicitly recognises (through its foundations in a simple theoretical model for regular waves) that regression coefficient A depends upon the shape of the structure since the shape, particularly at its crest, affects the discharge coefficient; coefficient A represents the dimensionless discharge when the dimensionless freeboard is zero;
iii) coefficient B depends upon the detailed behaviour of the water surface on the face of the structure; it increases as front slopes become flatter;
iv) coefficient C relates the maximum run-up to the significant height of the incident waves and may be chosen to allow for the influences of the seawall slope, the surface roughness and porosity, and the incident wave steepness; coefficient C can also account for storm duration in influencing R_{max}.
Ideally, any test programme would fix coefficient A by measuring the discharge over a seawall when the freeboard was zero. Likewise, C could be determined from the minimum freeboard giving zero overtopping discharge.

In the absence of this information, C has been estimated from available data on the significant wave run-up on smooth slopes.

For the present test results, the H&R model is little different from Owen's model in its ability to represent the data, except for small discharges for which the H&R model is better suited. However, there are a number of ways in which the H&R model could be improved. For example, the period of peak spectral density and the maximum run-up could be measured directly, whereas we have had to estimate these values. We would then expect more reliable estimates for coefficient C and, consequently, a closer agreement between our model and the data.

Finally, one of the purposes of Owen's tests was to show the variability in Q. This property must be considered in design procedures: it is necessary not only to model the expected value of Q, but also the distribution of Q about this value. As a consequence, approaches to coastal engineering design are shifting towards the assessment of the safety of coastal structures using risk analysis rather than using a deterministic procedure. One of the major stages of risk analysis is the formulation of equations to describe the failure mechanisms of a structure. In this connection, the main objective of the present paper has been to improve the mathematical description of wave overtopping of simple seawalls.

Acknowledgements

References

1. HRS. *Design of Seawalls Allowing for Wave Overtopping*, Report No. EX 924, Hydraulics Research Station, Wallingford, UK, 1980.
2. Hedges, T.S. & Reis, M.T. *A New Regression Model for Wave Overtopping Data*, Report No. CE/2/96, Department of Civil Engineering, University of Liverpool, Liverpool, UK, 1996.
3. Kikkawa, H., Shi-igai, H. & Kono, T. Fundamental study of wave overtopping on levees, *Coastal Engineering in Japan*, 1968, **11**, 107-115.
4. CIRIA/CUR. *Manual on the Use of Rock in Coastal and Shoreline Engineering*, Special Publication 83, Construction Industry Research and Information Association, London, 1991.
5. Isherwood, R.M. Technical note: a revised parameterisation of the Jonswap spectrum, *Applied Ocean Research*, 1993, **9**(1), 47-50.

Random Wave Overtopping of Simple Seawalls: a New Regression Model

T. S. Hedges[1], BEng, MEng, CEng, MICE, FRMetS
&
M. T. Reis[2], Engenheira Civil, MSc(Eng)

[1] Senior Lecturer, Department of Civil Engineering, University of Liverpool, P.O.Box 147, Liverpool, L69 3BX, UK

[2] Research Assistant, National Laboratory of Civil Engineering (LNEC), Avenida do Brasil 101, 1799 Lisbon, Portugal

Abstract

Seawalls are expensive, and fixing a seawall freeboard at too large a value has both a financial penalty and is unnecessarily damaging to the natural environment owing to the increased impact of the structure on its surroundings. On the other hand, if the crest of a seawall is set too low, then there are potential problems with structural safety and flooding from wave overtopping. Hence, it is important to strike the correct balance between satisfying the structural and functional requirements of the project, avoiding unnecessary expense, and having undesirable impacts on the surrounding environment.

The prediction of wave overtopping rates is usually based on empirical equations fitted to laboratory data. These equations do not have any theoretical basis. However, a new model has now been developed which, unlike existing expressions, accounts for the fact that no overtopping (apart from wind-blown spray) occurs if the seawall freeboard exceeds the maximum wave run-up on the face of the structure. This fact is of practical importance because allowable overtopping discharges to ensure the safety of people and property are quite small.

The paper starts with a brief review of existing overtopping equations, then presents the new model, and concludes by giving an example of its practical implications. It is shown that, for some conditions, the new model predicts seawall freeboards which are several metres less than those predicted by the well-known expression given by Owen (1982).

Notation

A, B	regression coefficients;
C	ratio of the maximum run-up to the significant height of the incident waves (=R_{max}/H_s);
C_d	discharge coefficient;
d_s	still-water-depth at the toe of the seawall;
F(t)	a function of time (in the description of water surface elevation);
g	acceleration due to gravity;
H	wave height;
H_s	significant wave height;
k	coefficient (in the description of water surface elevation);
L_m	Airy wavelength at the toe of the seawall calculated using the mean zero-crossing wave period;

L_{op}	Airy wavelength in deep water calculated using the period of peak spectral density ($= gT_p^2 / 2\pi$);
L_p	Airy wavelength at the toe of the seawall calculated using the period of peak spectral density;
L_s	Airy wavelength at the toe of the seawall calculated using the significant wave period;
N	number of run-up values;
q	instantaneous discharge of water over unit length of seawall;
Q	mean overtopping discharge over unit length of seawall;
Q_{PRED}	predicted mean overtopping discharge over unit length of seawall;
Q_*	dimensionless overtopping discharge;
R_c	seawall freeboard (the height of the crest of the structure above the still-water-level);
R_{max}	maximum run-up ($=CH_s$);
$(R_{max})_{p\%}$	p% confidence value of the estimated maximum run-up;
R_s	significant wave run-up;
$R_{2\%}$	run-up exceeded by only 2% of the incident waves;
R_*	dimensionless freeboard;
t	time;
T	wave period;
T_m	mean zero-crossing wave period;
T_p	wave period corresponding to peak spectral density;
T_s	significant wave period;
α	angle of seawall front slope measured from horizontal;
β	angle of wave approach measured from the normal to the seawall;
γ	reduction factor to account for influences of berms, roughness, shallow water and oblique wave attack on wave run-up and overtopping;
η	water surface elevation above still-water-level;
π	3.14159... ;
ξ_p	surf similarity parameter calculated using the period of peak spectral density ($= \tan\alpha / \sqrt{H_s / L_{op}}$).

1. Introduction

An important criterion for the design of a seawall is the allowable degree of wave overtopping which depends upon the activities normally performed in the lee of the structure, the need to prevent erosion of the rear face of the seawall, and the economic consequences of flooding. During 1978 and 1979, Owen (Hydraulics Research Station, 1980; Owen, 1982) carried out an extensive series of model tests to determine the overtopping discharges for a range of seawall designs subjected to different random wave climates. The modelled seawalls were all of the same general type: a flat-topped embankment fronted in some cases by a flat berm. The tests were aimed at establishing the impact on overtopping discharge of the wave climate (including the angle of wave attack and the wave steepness), the seawall slope, the crest and berm elevations, and the berm width.

This paper presents a re-analysis of Owen's data: the results for simple seawalls possessing uniform seaward slopes of 1:1, 1:2 and 1:4, subjected to random waves approaching normal to the slope. The purpose has been to construct a new regression model to represent the data which accounts for the fact that there is no overtopping (apart from wind-blown spray) if the seawall freeboard exceeds the maximum wave run-up on the face of the structure. The practical implications of using the new model are illustrated.

2. Dimensional analysis

In general, the mean overtopping discharge per unit length of seawall, Q, depends upon the wave motion, the seawall profile, the foreshore characteristics and the water properties:

$$Q = function(H_s, T_m, \beta, R_c, \alpha, d_s, g, ...) \qquad (1)$$

H_s is the significant height of the incident waves; T_m is the mean zero-crossing wave period; β is the angle of wave approach measured from the normal to the seawall; R_c is the seawall's freeboard (the height of the crest of the structure above the still-water-level); α is the angle of the seawall front slope measured from the horizontal; d_s is the still-water-depth at the toe of the structure; and g is the acceleration due to gravity (see Figure 1).

Equation 1 may be rewritten in the form of dimensionless groups:

$$\frac{Q}{\sqrt{gH_s^3}} = \text{function}\left(\frac{R_c}{H_s}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, \ldots\right) \tag{2}$$

or

$$\frac{Q}{g^2 T_m^3} = \text{function}\left(\frac{R_c}{H_s}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, \ldots\right) \tag{3}$$

H_s / gT_m^2 is a measure of the incident wave steepness. Owen combined this group both with $Q / \sqrt{gH_s^3}$ (or $Q / g^2 T_m^3$) and with R_c / H_s, to write:

$$\frac{Q}{T_m gH_s} = \text{function}\left(\frac{R_c}{T_m \sqrt{gH_s}}, \frac{H_s}{gT_m^2}, \frac{d_s}{H_s}, \alpha, \beta, \ldots\right) \tag{4}$$

However, other arrangements are possible (Aminti & Franco, 1988; Ahrens & Bender, 1992), including use of the wave period of peak spectral density, T_p, rather than the mean zero-crossing period, T_m. Table 1 summarises some of the options for dimensionless discharge, Q_*, and dimensionless freeboard, R_*. Here, H_s has been used to denote the significant wave height calculated either as the mean height of the highest one third of the waves in a record or estimated from the zeroth moment of the surface elevation spectrum (IAHR, 1989). L_s is the Airy wavelength calculated using the water depth at the toe of the structure and the significant wave period, T_s. L_m and L_p are the corresponding wavelengths calculated using T_m and T_p. $R_{2\%}$ is the run-up exceeded by only 2% of the incident waves. γ is a reduction factor to account for influences of berms, roughness, shallow water and oblique wave attack on wave run-up and overtopping. ξ_p is the surf similarity parameter calculated using the wave period of peak spectral density ($\xi_p = \tan\alpha / \sqrt{H_s / L_{op}}$ in which $L_{op} = gT_p^2 / 2\pi$). Finally, note that R_* for De Waal & Van der Meer (1992) is not strictly a dimensionless freeboard but the dimensionless excess of the crest level above the 2% run-up level.

   The dimensionless groups in Table 1 are generally related using one of the two following functions:

$$Q_* = A \exp(-BR_*) \tag{5}$$

$$Q_* = A (R_*)^{-B} \tag{6}$$

where A and B are best-fit coefficients determined from the experimental data. Clearly, coefficients A and B must account for all influences on Q_* other than R_*.

Dimensional analysis provides no means for determining which sets of dimensionless groups may be especially informative or helpful in dealing with a particular data set. A possible problem in using many of the pairings in Table 1 is the potential for spurious correlation. A spurious correlation may arise when dimensionless groups plotted against one another contain a common variable (Massey, 1971). There is nothing wrong with the presence of a common variable, but care must be taken in interpreting such plots. Scatter in the data may be suppressed simply by the presence of this variable.

3. Regression analysis

3.1. Introduction

Once experimental data have been collected, they may be used to confirm the validity of some theory or, where no satisfactory theory exists, they may be used to construct regression models. However, it is always useful to have some theoretical basis for choosing amongst the possible models. Furthermore, there are many techniques available for fitting regression models (Gunst & Mason, 1980). Which ones are appropriate for a particular study depend upon its objectives. For example, it may be possible to develop a model which is good at predicting values of the response variable but which, nevertheless, is incorrectly specified (i.e. the model does not include all relevant predictor variables or it has an incorrect functional form). In describing a regression model, care should be taken to emphasise the range of conditions over which there are data to support its use. Unfortunately, it is sometimes impossible to collect data on the dependent or response variable (in this instance, overtopping) over the entire range of interest of the independent or predictor variables (wave height, structure profile, etc).

As a start, let us consider the physical boundary conditions to be met in addressing wave overtopping:

i) when the embankment has a large freeboard (i.e. when its crest elevation is well above the level of wave uprush), the predicted overtopping discharge should be zero (assuming that the effects of wind-blown spray are ignored);

ii) when the embankment has zero freeboard (i.e. when still-water-level is at the crest level of the embankment) then the predicted overtopping discharge may

be large but should still remain finite.

Equations 5 and 6 represent two of the more common functions used to predict wave overtopping. However, when R_* is large, both expressions suggest that the discharge will be finite rather than zero (though it is small provided that A is not very large and provided also that B>1). When R_* is zero, the first of these expressions gives Q_*=A, a finite quantity, whilst the second expression gives Q_*=∞. Thus neither expression satisfies both boundary conditions, with the second of them satisfying neither. Since most seawalls are designed to permit only relatively small overtopping discharges, it is especially important to satisfy the first of the two boundary conditions.

In addition to considering the boundary conditions, we also need to establish the line of "best fit" to the observed data. There are many criteria for defining the best fit. One possibility is to minimise the sum of the squared deviations of the observations from the values predicted from our expression. But real data usually do not completely satisfy the classical assumptions for a least-squares (LS) fitting (Rousseeuw & Leroy, 1987). For example, the deviations may not follow a Normal distribution. Reliable inferences may be drawn from regression models fitted by the LS method only if the assumptions are valid (Draper & Smith, 1981; Rousseeuw & Leroy, 1987). Furthermore, an LS fitting has the disadvantage that the result is not "robust": it is sensitive to outlying data points. Whilst we could remove "outliers", such a procedure should only be considered if there is reason to doubt their validity. Such data must not be removed merely because they do not support our regression model: it may be the model which is wrong.

Performing a least-absolute-deviations (LAD) fitting, involves minimising the sum of the absolute deviations rather than the sum of the squared deviations. It does not rely upon the Normal assumption and allows us to retain outliers but prevents these points from exerting a disproportionate influence on the values of the regression coefficients. If the deviations are assumed to follow a Double Exponential distribution, which has thicker tails than the Normal distribution, then the parameter values are maximum likelihood estimates. Figure 2 shows an example of the different results obtained from using the two techniques. Figure 2(a) is a plot of five points which lie almost on a straight line. Therefore, the LS fit and the LAD fit are essentially the same. Figure 2(b) displays a situation where, for some reason, point 4 has been wrongly moved from its original position (indicated by the dashed circle). This point is called an outlier in the y-direction; it has a rather strong influence on the LS line, which is quite different from the line in Figure 2(a). Figure 2(c) shows the robustness of the LAD fit with

respect to such an outlier; the line remains (approximately) where it was when observation 4 was correct. In this paper, we have chosen to fit our regression lines using the LAD method. However, we have also compared these results with an LS fitting.

3.2. A new regression model

3.2.1. A simple overtopping theory for regular waves

Let us step back from the complications of random waves to the simpler case of regular waves of height H approaching normal to a seawall. We will assume that the instantaneous discharge of water over unit length of the seawall, q, is given by the weir formula (Streeter & Wylie, 1979):

$$q = C_d \frac{2}{3} \sqrt{2g} (\eta - R_c)^{3/2} \qquad \text{for } \eta > R_c \qquad (7)$$

in which η is the water surface elevation above still-water-level at the seawall (a periodic function of time); C_d is a discharge coefficient. Obviously, overtopping occurs only when the water surface is above the structure's crest.

We will also assume that:

$$\eta = k \, H F(t) \qquad (8)$$

F(t) denotes a function of time, t. For simple, sinusoidal, progressive waves, k=0.5 and F(t)=cos($2\pi t/T$), where T is the wave period. However, following Kikkawa et al (1968), we will adopt the simpler form for F(t) shown in Figure 3; k remains a coefficient determined by the particular wave and wall details.

The mean discharge, Q, is determined as follows:

$$Q = C_d \frac{2}{3} \sqrt{2g} \frac{1}{T} \int_{t_1}^{t_2} \{k H F(t) - R_c\}^{3/2} \, dt \qquad (9)$$

in which $t_1 < t < t_2$ corresponds to the interval during each wave period for which $k H F(t) > R_c$. Using the form for F(t) given in Figure 3 then yields:

$$\frac{Q}{\sqrt{g(kH)^3}} = C_d \frac{2\sqrt{2}}{15} \left\{ 1 - \frac{R_c}{kH} \right\}^{5/2} \qquad \text{for } 0 < R_c < kH$$

$$= 0 \qquad\qquad\qquad\quad \text{for } R_c \geq kH \tag{10}$$

Note that overtopping occurs only when $R_c < kH$. In other words, kH represents the run-up on the face of the seawall. Since wave run-up is a function of the incident wave height and steepness, and of the seawall slope, the overtopping discharge can be expected also to depend upon these parameters.

3.2.2. The Hedges & Reis (H&R) overtopping model

The above theory suggests a regression equation for the random overtopping data of the following form:

$$Q_* = A(1 - R_*)^B \qquad \text{for } 0 \leq R_* < 1$$

$$= 0 \qquad\qquad\quad \text{for } R_* \geq 1 \tag{11}$$

in which

$$Q_* = \frac{Q}{\sqrt{gR_{max}^3}} = \frac{Q}{\sqrt{g(CH_s)^3}} \tag{12}$$

and

$$R_* = \frac{R_c}{R_{max}} = \frac{R_c}{CH_s} \tag{13}$$

Coefficient k in the expression for regular waves has been replaced by C in this regression model for random waves characterised by H_s. Note that CH_s represents R_{max}, the maximum run-up induced by the random waves, not the run-up induced by a wave of height H_s. Consequently, C will depend upon the duration of the incident wave conditions unless the wave heights in front of the wall are limited by the available water depth. Unless the maximum run-up, R_{max}, exceeds the freeboard, R_c, there is no overtopping (apart from wind-blown spray). It is also clear that coefficient B is related, in the case of regular waves, to the shape of the function $F(t)$ which describes the water surface variation on the seaward face of the wall. There will be a similar dependence on the detailed behaviour of the water surface at the face of the wall in the case of random waves. Finally, coefficient A represents the dimensionless discharge over the seawall when the freeboard is zero. All three coefficients will be

influenced by the seaward profile of the structure.

The above model for overtopping has the advantage that $Q_* = 0$ when $R_* \geq 1$ and that $Q_* = A$ when $R_* = 0$, in accordance with our required boundary conditions. Figure 4 shows the influences of coefficients A, B and C in the new overtopping model.

The value of C ($= R_{max}/H_s$) to be adopted would best be determined from experimental data. Unfortunately, Owen's data set (and others) do not provide an adequate number of cases involving zero or very small discharges. Consequently, its value has been estimated from run-up measurements for random waves acting on slopes for which there is no overtopping. Although not ideal for our purposes, these additional data on run-up complement Owen's overtopping results, allowing the new model to be applied outside the range of his experimental data. We have adopted this option rather than including C alongside A and B as a regression coefficient.

A number of equations describing random wave run-up are available (CIRIA/CUR, 1991; Van der Meer & Janssen, 1995). For example, the CIRIA/CUR (1991) manual gives two equations for evaluating the significant wave run-up, R_s, on smooth slopes without overtopping. It notes that the equations, based upon Ahrens' data (Ahrens, 1981), are probably conservative and that data from Allsop et al (1985) give values 20 to 30% lower. Rewritten in our notation and allowing for a printing error, the expressions are:

$$
\begin{aligned}
\frac{R_s}{H_s} &= 1.35\,\xi_p && \text{for} \quad \xi_p < 2 \\[2mm]
\frac{R_s}{H_s} &= 3.00 - 0.15\,\xi_p && \text{for} \quad \xi_p > 2
\end{aligned}
\tag{14}
$$

Here, ξ_p is the surf similarity parameter calculated using T_p which was estimated for Owen's data using the relationships between H_s, T_m and T_p provided by Isherwood (1987).

Assuming that run-up may be described by a Rayleigh distribution, then the p% confidence value of maximum run-up (defining a level below which p% of the cases should lie) is related to the significant wave run-up by (Hogben, 1990):

$$
(R_{max})_{p\%} = \left[\frac{1}{2}\left(\ell n\, N - \ell n\left(-\ell n\, \frac{p}{100} \right) \right) \right]^{1/2} R_s
\tag{15}
$$

N is the number of run-up values, here taken conservatively to be equal to the number of incident waves.

Owen recorded his overtopping discharges during tests involving sets of five different runs, each of 100 waves, characterised by the same significant wave height. The most probable maximum run-up during each run (the value not exceeded in 37% of the cases for a Rayleigh distribution of run-ups) is then:

$$(R_{max})_{37\%} = \sqrt{(\ell n\ 100)\ /\ 2}\ R_s = 1.52 R_s \qquad (16)$$

In none of Owen's cases were there overtoppings for freeboards greater than $(R_{max})_{37\%}$ if R_s was evaluated using equations 14. In fact, all nine reported cases of zero overtopping were for freeboards of less than this value. Hence, setting $C = (R_{max})_{37\%} / H_s$ is conservative in this instance and the following expressions for C then arise from equations 14 and 16:

$$
\begin{aligned}
C &= 1.52\,(1.35\,\xi_p) & \text{for}\ \ \xi_p &< 2 \\
C &= 1.52\,(3.00 - 0.15\,\xi_p) & \text{for}\ \ \xi_p &> 2
\end{aligned}
\qquad (17)
$$

The fact that these expressions for C are conservative may be a result either of the conservative nature of equations 14 or of deficiencies in the assumptions relating to the distribution of run-ups. However, setting $C = (R_{max})_{37\%} / H_s$ may not always be appropriate. Note that the value of C to be adopted in the regression model depends upon the level of confidence associated with the prediction of R_{max}. If C is changed then there will be corresponding changes in the values of A and B. The implications for seawall freeboards of adopting different levels of confidence in R_{max} are considered later.

4. Results of regression analysis

Figure 5 shows an example of the overtopping data collected by Owen: the results for a simple seawall with a uniform front slope of 1:2. The data are plotted in the formats required for fitting regression equations using both the H&R and the Owen models. Figure 5(a) shows the best-fit lines established using LS and LAD procedures for the H&R model. Comparison of the regression coefficients shows the relatively stronger influence which outlying data points have on the LS values. For example, the

magnitude of B obtained from the LAD fitting is only about 92% of the LS result. Similar comments may be made about the regression lines obtained for Owen's model. Note, that the values of A and B reported in Figure 5(b) for Owen's model are not those which Owen himself recommended.

Table 2 gives the regression coefficients for all three slopes which we have obtained for the H&R model and for Owen's model, using both LS and LAD fitting. Also included for reference are Owen's recommended values.

Owen restricted his analysis to a particular set of conditions whilst we have included all available data apart from eleven of the 110 results for the 1:4 slope. Of these eleven, nine had Q=0. Figure 4 shows that there will be many values of R_* for which $Q_*=0$ and data points with $Q_*=0$ must be excluded from a regression analysis, otherwise a regression line (if it could be fitted) would pass through these data rather than defining their lower limit. The other two excluded values (for which Q was not zero) were from a set of five runs with the same dimensionless freeboard, three of which had zero overtopping discharge. Including only two of these five data points would have severely biased the positions of the regression lines. Furthermore, the validity of these two data points is doubtful since a full set of five runs at a smaller dimensionless freeboard all had Q recorded as zero. Although removed for the purposes of regression analysis, the eleven points were reinstated for inclusion in Figure 6 (see later). The full data set fell within the following ranges:

$$0 < \frac{Q}{\sqrt{g(CH_s)^3}} < 0.0056 \qquad 0.14 < \frac{R_c}{CH_s} < 0.90$$

$$0 < \frac{Q}{T_m g H_s} < 0.0039 \qquad 0.053 < \frac{R_c}{T_m \sqrt{gH_s}} < 0.239 \tag{18}$$

$$0.0053 < \frac{H_s}{gT_m^2} < 0.0095$$

$$1.65 < \frac{d_s}{H_s} < 5.20$$

Earlier, we have mentioned briefly the problem of spurious correlation. Along with most other overtopping models (see Table 1), the H&R model employs a dimensionless discharge and a dimensionless freeboard which contain a common variable (in our case, it is R_{max} or CH_s). The presence of this common variable may reduce the apparent scatter in the data. Consequently, in Figure 6 we show directly the level of agreement between Owen's measured values of Q (converted by Owen to

full-scale discharges for a seawall in 4m water depth) and the predicted values, Q_{PRED}. Of course, we could also have attempted to disguise the scatter in the relationship between Q and Q_{PRED} by plotting against logarithmic scales (Massey, 1971). But such an attempt is both misleading and unnecessary. Under random wave conditions, overtopping will be dominated by the few waves with large run-ups: most waves will contribute no overtopping if the seawall has a substantial freeboard (Jensen & Juhl, 1987). Thus, particularly for short runs of random waves, as in Owen's tests, we can expect some variability in the measured values of Q. Indeed, one of the purposes of Owen's tests was to show this inherent variability.

In Figure 6, most data points lie within a range for Q/Q_{PRED} of 3/4 to 4/3, whichever model is adopted. It is not obvious from the figure which model best fits the data, nor is it obvious from the plots for simple seawalls with 1:1 and 1:4 front slopes. Consequently, we have looked in more detail at the data points for discharges in the ranges of practical interest.

Figure 7 shows the critical mean overtopping discharges currently used in the design of seawalls. The main point to note from this figure is that the range of critical discharges runs from as little as $0.001 \times 10^{-3} \text{m}^3/\text{s/m}$ to about $200 \times 10^{-3} \text{m}^3/\text{s/m}$. An overtopping rate greater than about $0.001 \times 10^{-3} \text{m}^3/\text{s/m}$ will be unsafe for vehicles at high speed and may cause minor damage to the fittings of buildings. Conditions become dangerous for pedestrians when the discharge exceeds $0.03 \times 10^{-3} \text{m}^3/\text{s/m}$. Discharges greater than about $2 \times 10^{-3} \text{m}^3/\text{s/m}$ may damage embankment seawalls, whilst $50 \times 10^{-3} \text{m}^3/\text{s/m}$ is approximately the critical discharge for seawalls without back slopes.

Designers will often wish to know the necessary crest height to limit overtopping to the relatively small amounts indicated above. For these purposes, the H&R model appears generally better than Owen's model owing to its ability to predict zero overtopping at finite values of freeboard. Furthermore, the next section shows that it tends to give lower required crest levels than Owen's model for small permissible discharges, offering lower environmental impact and potential cost savings.

5.   Some implications for seawall freeboards

According to Owen's model, the freeboard, R_c, necessary to limit overtopping to a specified value, Q, is given by:

$$R_c = \frac{T_m \sqrt{gH_s}}{B} \, \ell n \left\{ \frac{A T_m g H_s}{Q} \right\} \qquad (19)$$

The H&R model gives:

$$R_c = CH_s \left\{ 1 - \left[\frac{Q}{A\sqrt{g(CH_s)^3}} \right]^{1/B} \right\} \qquad (20)$$

Note that equation 19 incorporates the mean zero-crossing wave period, T_m, whilst equation 20 involves coefficient C which has been described in terms of the period of peak spectral density, T_p. In order to compare the output from the two expressions, it has been assumed that the incident waves conform to the Pierson-Moskowitz spectrum. In this case (for H_s in metres, with T_m and T_p in seconds):

$$T_m = 3.55\sqrt{H_s}; \quad T_p = 5.00\sqrt{H_s} \qquad (21)$$

Figure 8 provides a comparison between the freeboards predicted using equations 19 and 20 for embankments with uniform front slopes of 1:2 subject to random waves with a significant height of 2m. Similar figures could be prepared for embankments with seaward slopes of 1:1 and 1:4, for additional incident significant wave heights and for different values of the confidence level associated with R_{max}.

Owen stated (Hydraulics Research Station, 1980) that his empirical coefficients A and B were determined only for particular ranges of the dimensionless groupings given in equation 4. The conditions included the following: $10^{-6} < Q / T_m g H_s < 10^{-2}$ and $0.05 < R_c / T_m \sqrt{gH_s} < 0.30$. Many of the discharges shown in Figure 8 have $Q / T_m g H_s < 10^{-6}$. For the conditions of Figure 8, this limit is approximately equivalent to $Q < 10^{-4} m^3/s/m$. Nevertheless, Owen also suggested that it was possible to use his equation to extrapolate results when the dimensionless freeboard was such that the dimensionless discharge fell below 10^{-6}. Thus, for a typical seawall in 4m water depth, it is possible to compare the minimum necessary freeboards predicted by the H&R model with those predicted by Owen's expression if overtopping is to be limited to specified values.

Two points are worth noting:

i)   There is reasonable agreement between the H&R model and Owen's model only for overtopping discharges in the range of $10^{-2}m^3/s/m$ to $2\times10^{-1}m^3/s/m$. This is irrespective of the confidence level (37% or 99%) assigned in the evaluation of R_{max}. However, it is in the range where there are significant differences that most seawalls are designed (see Figure 7).

ii) As the confidence level in R_{max} is increased, the freeboards predicted by the H&R model approach those values obtained from Owen's model. Nevertheless, even using $(R_{max})_{99\%}$ there remain significant differences. This observation has important implications for seawall design. For example, for an expected overtopping discharge of $10^{-4}m^3/s/m$, the difference amounts to about 1.9m. It is even greater both for the lower expected overtopping rates associated with functional safety requirements (Figure 7) and for higher H_s values. Owen's model suggests that the freeboard must continue to increase in order to reduce the overtopping rate even when the crest of the seawall is clearly above any possible run-up level induced by the random waves.

6.   Concluding remarks

A new regression model has been presented for describing wave overtopping data. Part of our motivation was to improve the methods available to the designers of seawalls by developing a model closely related to the physics of wave overtopping. The important features of the model are as follows:

i)   It satisfies the relevant physical boundary conditions, a feature which is especially important when the model is used near these boundaries.

ii) It explicitly recognises (through its foundations in a simple theoretical model for regular waves) that regression coefficient A depends upon the shape of the structure since the shape, particularly at its crest, affects the discharge coefficient; coefficient A represents the dimensionless discharge when the dimensionless freeboard is zero.

iii) Coefficient B depends upon the detailed behaviour of the water surface on the seaward face of the structure; it increases as front slopes become flatter.

iv) Coefficient C relates the maximum run-up to the significant height of the incident waves and may be chosen to allow for the influences of the seawall slope, the surface roughness and porosity, and the incident wave steepness. Coefficient C can also account for storm duration in influencing R_{max} (though the regression coefficients in the present study have been established only for short

sequences of 100 random waves). Finally, it may be chosen so that there is a specified confidence level associated with R_{max}.

It is suggested that the regression coefficients contained within the model should be established using a robust regression technique. Examples are given of the differences between the LS and the LAD fitting methods in analysing overtopping data collected by Owen (Hydraulics Research Station, 1980; Owen, 1982).

For the present test results, the H&R model is little different from Owen's model in its ability to represent the data, except for small discharges for which the H&R model is better suited. An example is given of the application of Owen's model and the H&R model to predict the freeboards necessary to limit overtopping to specified values. The example shows that, for the small allowable discharges associated with normal design conditions, the H&R model predicts seawall crest elevations which may be up to several metres lower than values from Owen's model. These differences may have very significant financial and environmental consequences and are worthy of further investigation.

Acknowledgements

References

Ahrens, J.P., 1981. Irregular wave run-up on smooth slopes. Coastal Engineering Technical Aid No. 81-17, US Army, Corps of Engineers, Coastal Engineering Research Center, Fort Belvoir.

Ahrens, J.P. & Bender, T., 1992. Evaluating the performance of seawalls. In Proc. Conf. on Coastal Structures and Breakwaters, Institution of Civil Engineers, London, pp.13-24.

Ahrens, J.P. & Heimbaugh, M.S., 1988. Seawall overtopping model. In Proc. Twenty-First Coastal Engng. Conf., Malaga (Ed. Edge, B.L.), ASCE, New York, pp.795-806.

Allsop, N.W.H., Hawkes, P.J., Jackson, F.A. & Franco, L., 1985. Wave run-up on steep slopes - model tests under random waves. Report No. SR2 Hydraulics Research Ltd, Wallingford.

Aminti, P. & Franco, L., 1988. Wave overtopping on rubble mound breakwaters. In Proc. Twenty-First Coastal Engng. Conf., Malaga (Ed. Edge, B.L.), ASCE, New York, pp.770-781.

Bradbury, A.P. & Allsop, N.W.H., 1988. Hydraulic effects of breakwater crown walls. In Proc. Conf. on Design of Breakwaters, Institution of Civil Engineers, London, pp.385-396.

CIRIA/CUR, 1991. Manual on the use of rock in coastal and shoreline engineering. Special publication 83, Construction Industry Research and Information Association, London.

De Waal, J.P. & Van der Meer, J.W., 1992. Wave runup and overtopping on coastal structures. In Proc. Twenty-Third Coastal Engng. Conf., Venice (Ed. Edge, B.L.), ASCE, New York, pp.1758-1771.

Draper, N.R. & Smith, H., 1981. Applied regression analysis. John Wiley & Sons, New York.

Franco, L., de Gerloni, M. & Van der Meer, J.W., 1994. Wave overtopping on vertical and composite breakwaters. In Proc. Twenty-Fourth Coastal Engng. Conf., Kobe (Ed. Edge, B.L.), ASCE, New York, pp.1030-1045.

Gunst, R.F. & Mason, R.L., 1980. Regression analysis and its application: a data-oriented approach. Marcel Dekker Inc, New York.

Hogben, N., 1990. Long term wave statistics. In The Sea (Ed. Le Méhauté, B. & Hanes, D.M.), John Wiley & Sons, Inc., New York.

Hydraulics Research Station, 1980. Design of seawalls allowing for wave overtopping. Report No. EX 924, Hydraulics Research Station, Wallingford.

IAHR Working Group on Wave Generation and Analysis, 1989. List of sea-state parameters. J. Waterway, Port, Coastal, and Ocean Engineering, Vol.115(6), pp.793-808.

Isherwood, R.M., 1987. Technical note: a revised parameterisation of the Jonswap spectrum. Applied Ocean Research, Vol.9(1), pp.47-50.

Jensen, O.J. & Juhl, J., 1987. Wave overtopping on breakwaters and sea dikes. In Proc. Coastal and Port Engineering in Developing Countries (Ed. Nanjing Hydraulic Research Institute), China Ocean Press, Beijing, pp.716-730.

Kikkawa, H., Shi-igai, H. & Kono, T., 1968. Fundamental study of wave overtopping

on levees. Coastal Engineering in Japan, Vol.11, pp.107-115.

Massey, B.S., 1971. Units, dimensional analysis and physical similarity. Van Nostrand Reinhold Company, London.

Owen, M.W., 1982. The hydraulic design of seawall profiles. Proc. Conf. On Shoreline Protection, Institution of Civil Engineers, London, pp.129-136.

Pedersen, J. & Burcharth, H.F., 1992. Wave forces on crown walls. In Proc. Twenty-Third Coastal Engng. Conf., Venice (Ed. Edge, B.L.), ASCE, New York, pp.1489-1502.

Rousseeuw, P.J. & Leroy, A.M., 1987. Robust regression and outlier detection. John Wiley & Sons, New York.

Sawaragi, T., Deguchi, I. & Park, S-K., 1988. Reduction of wave overtopping rate by the use of artificial reefs. In Proc. Twenty-First Coastal Engng. Conf., Malaga (Ed. Edge, B.L.), ASCE, New York, pp.335-349.

Smith, G.M.; Seijffert, J.W.W. and Van der Meer, J.W., 1994. Erosion and overtopping of a grass dike: large scale model tests. In Proc. Twenty-Fourth Coastal Engng. Conf., Kobe (Ed. Edge, B.L.), ASCE, New York, pp. 2639-2652.

Streeter, V.L. & Wylie, E.B., 1979. Fluid mechanics. McGraw-Hill Kogakusha Ltd, Tokyo.

Van der Meer, J.W., 1993. Conceptual design of rubble mound breakwaters. Publication No. 483, Delft Hydraulics, Delft.

Van der Meer, J.W. & Janssen, J.P.F.M., 1995. Wave run-up and wave overtopping at dikes. In Wave Forces on Inclined and Vertical Wall Structures (Ed. Kobayashi, N. & Demirbilek, Z.), ASCE, New York.
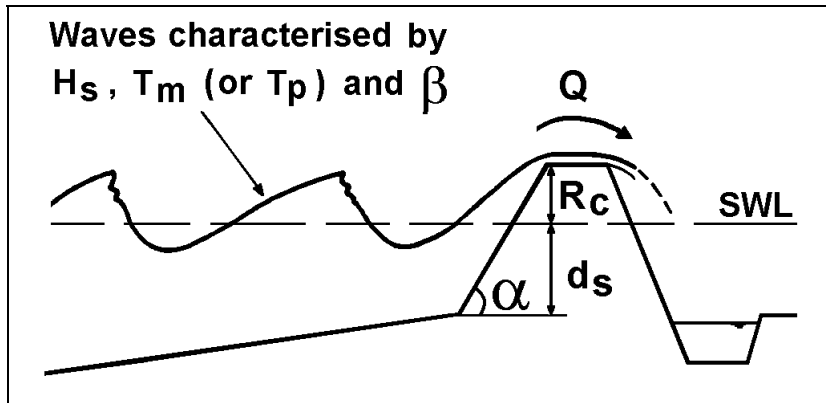
List of Figures

List of Tables

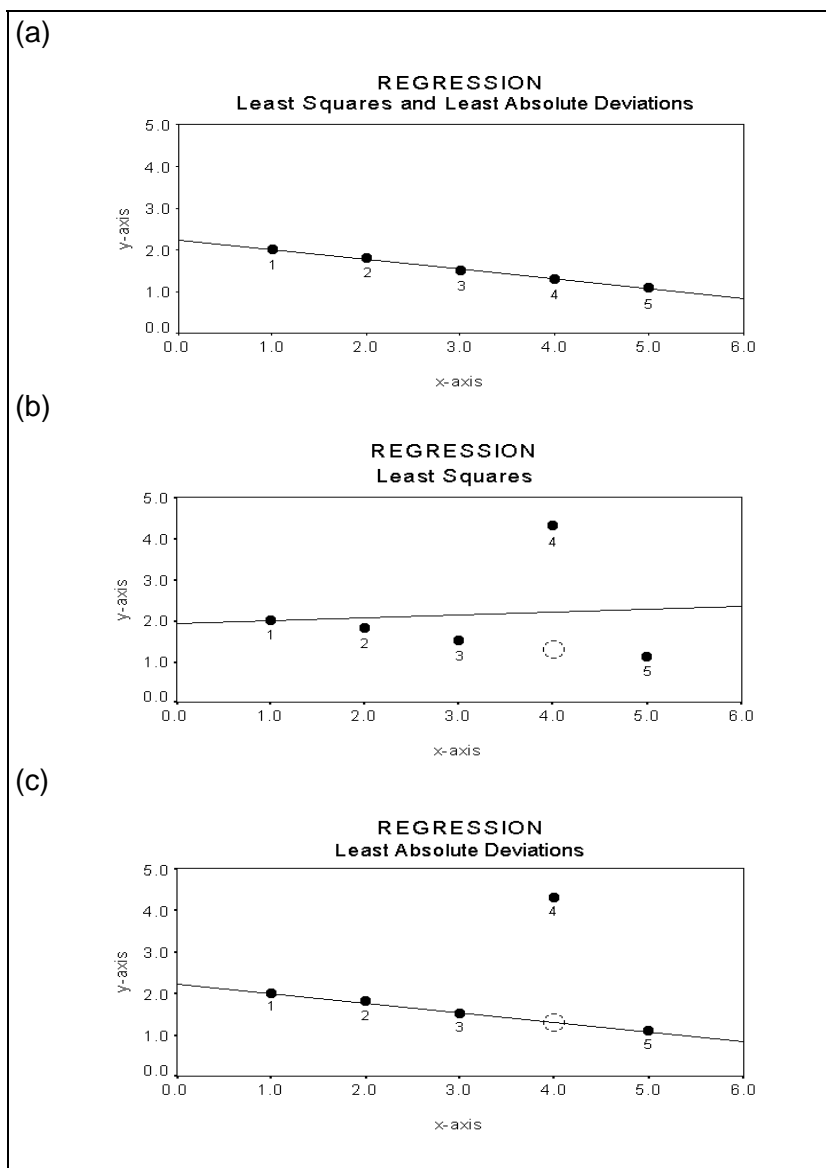**Figure 1:** Notation for seawall overtopping.

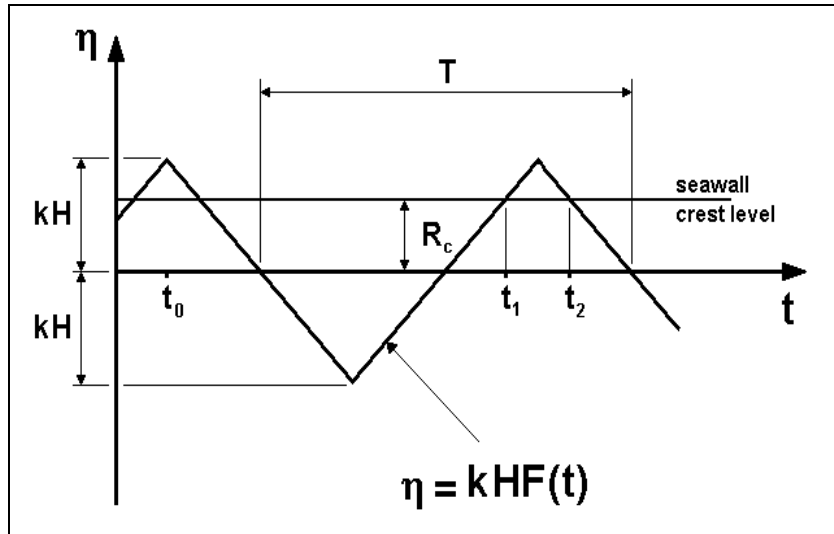**Figure 2:** LS and LAD regression lines (after Rousseeuw & Leroy, 1987).

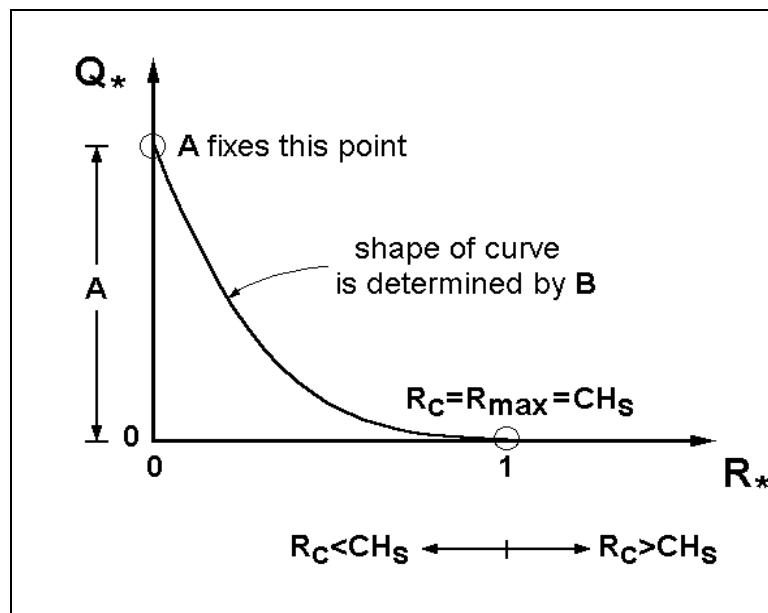**Figure 3:** Form of function F(t).

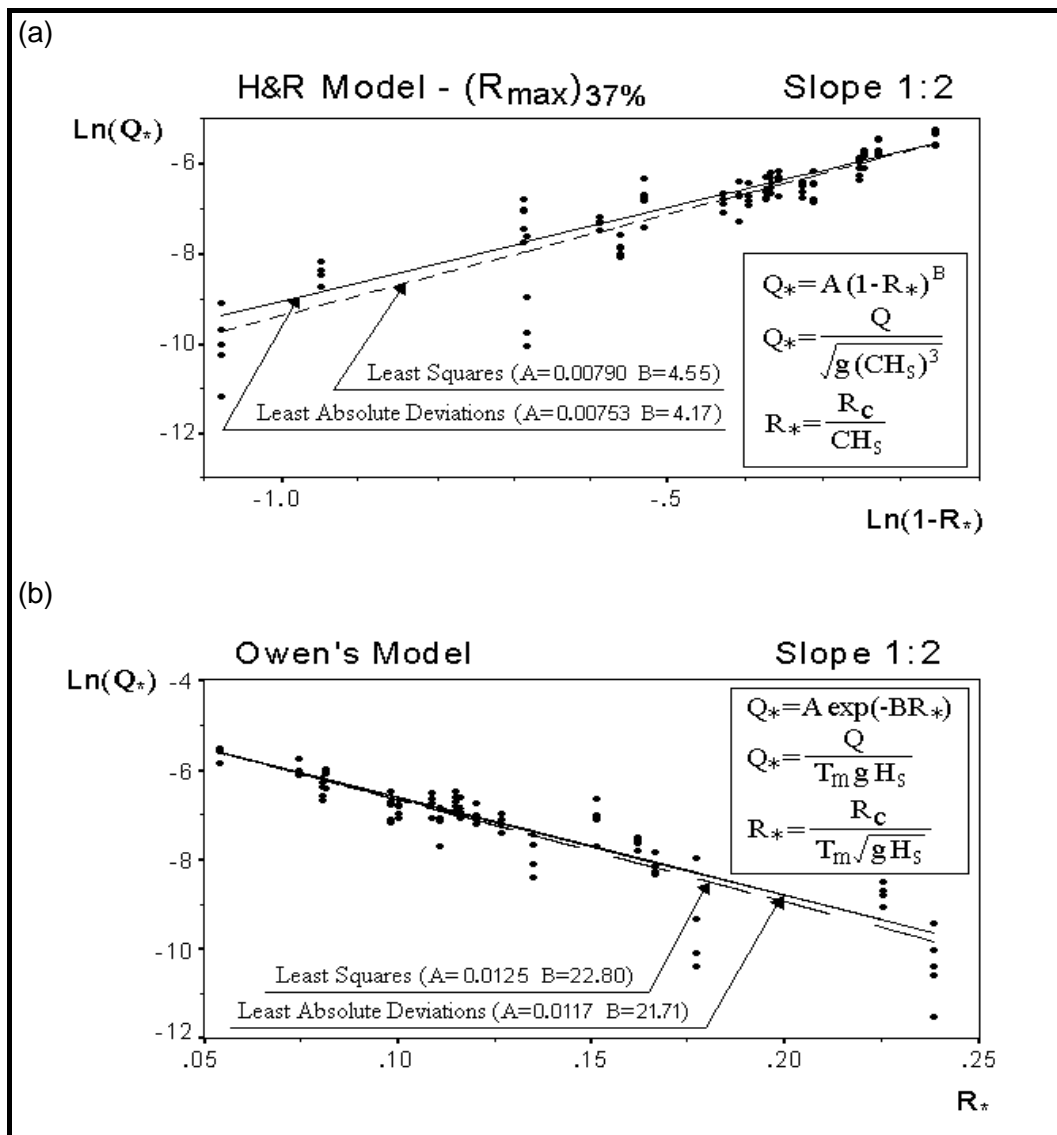**Figure 4:** Influences of coefficients A, B and C in the new overtopping model.

Figure 5: Wave overtopping data for slope 1:2 plotted in the formats required for fitting regression equations, (a) using the H&R model, $(R_{max})_{37\%}$, and (b) using Owen's model.
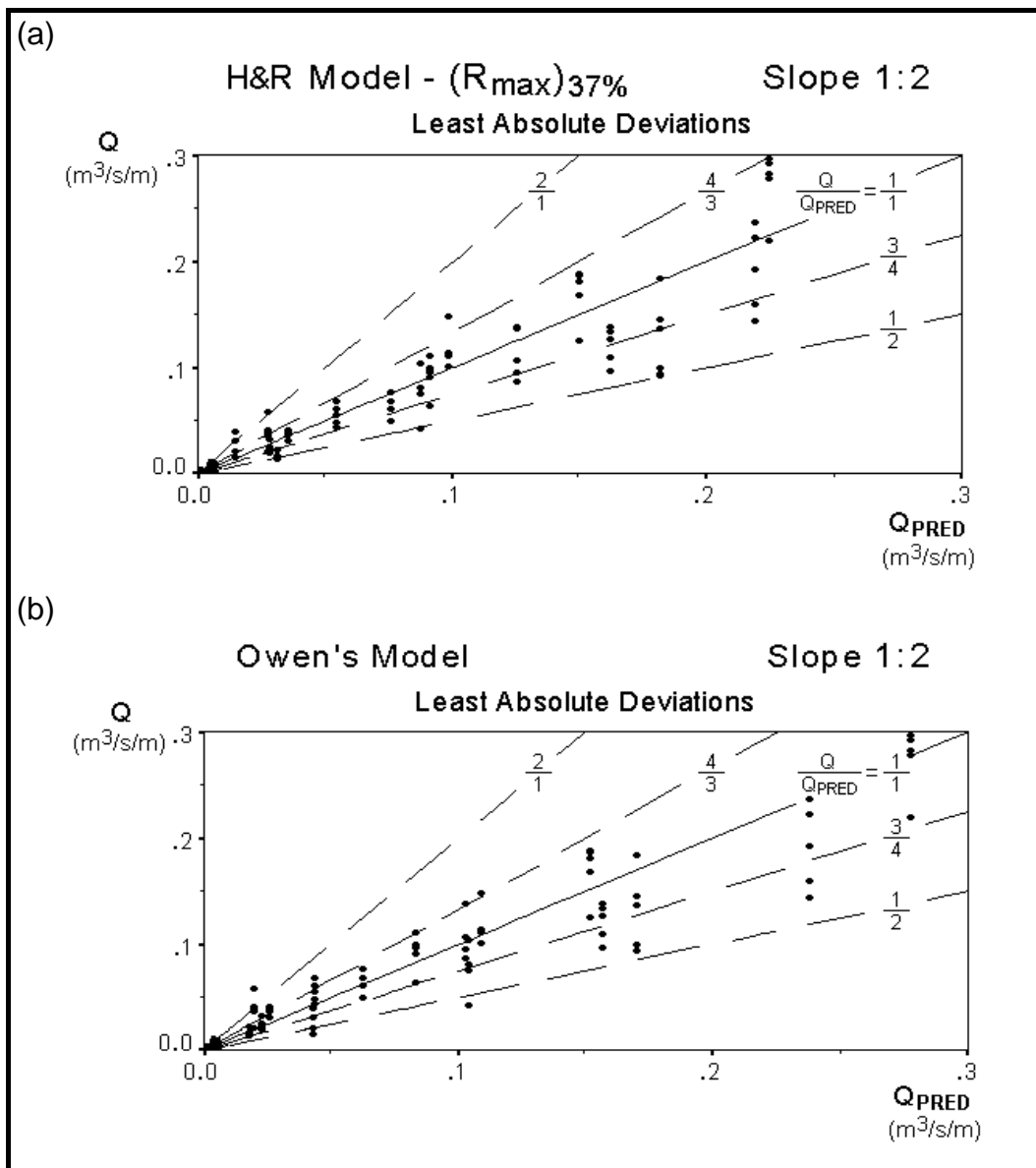
Figure 6: Wave overtopping data for slope 1:2, showing the level of agreement between Q and Q_{PRED}, (a) using the H&R model, $(R_{max})_{37\%}$, and (b) using Owen's model.
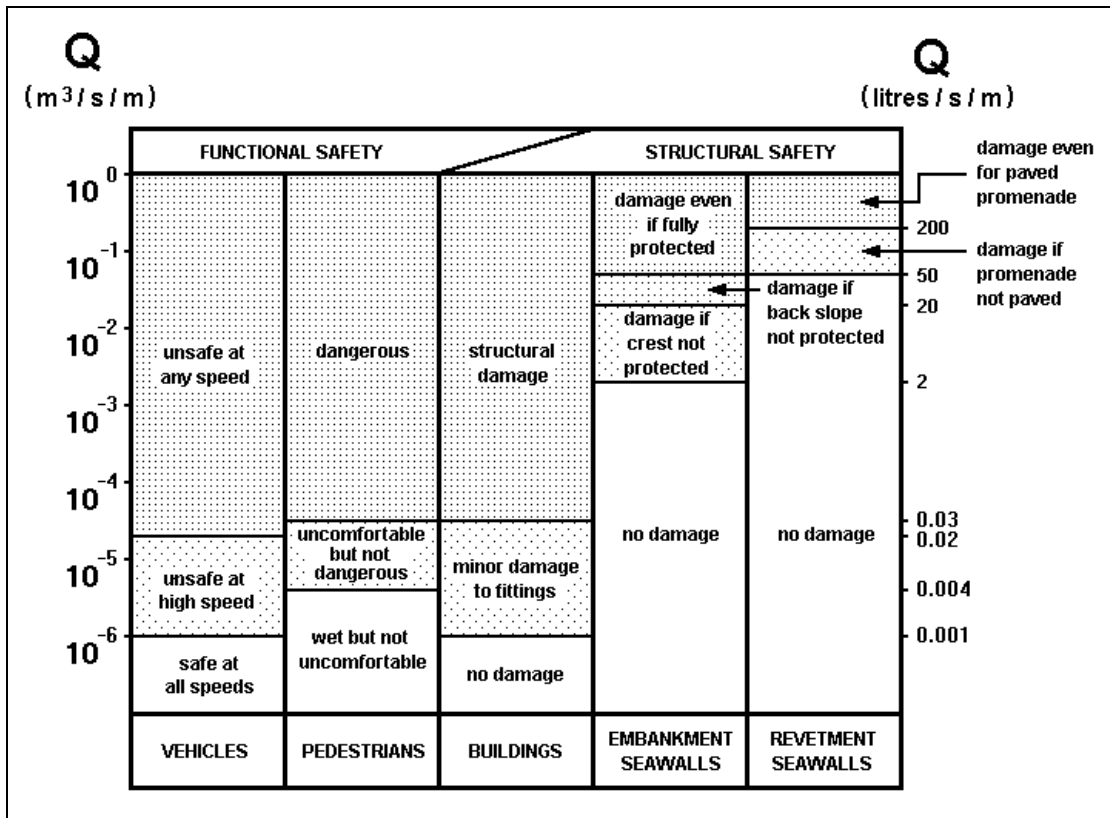
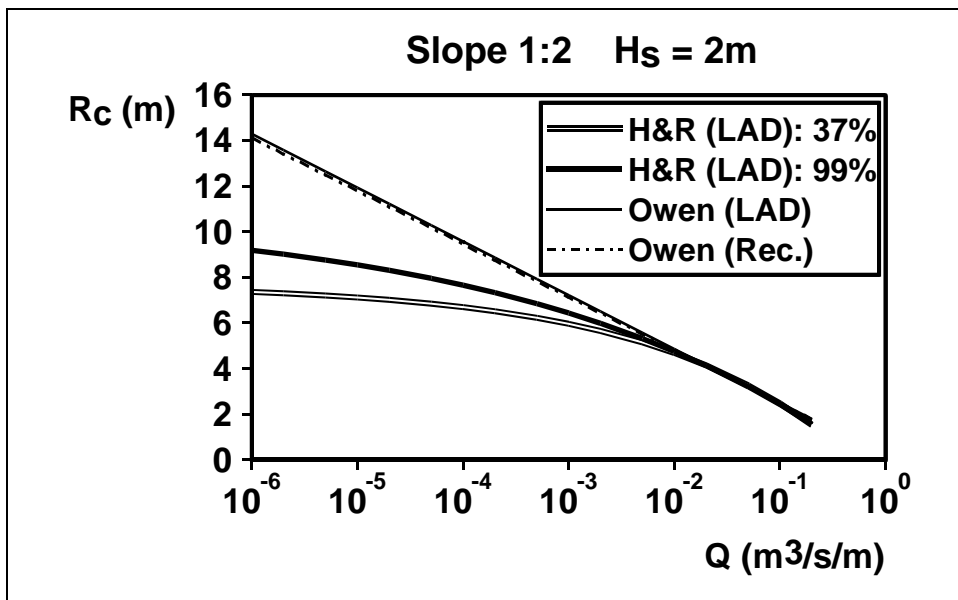**Figure 7:** Critical mean overtopping discharges for use in design (modified after CIRIA/CUR, 1991).

Figure 8: Comparison of predicted freeboards for slope 1:2 (LAD indicates that the values of the regression coefficients have been established using the method of least absolute deviations; Rec. refers to the coefficient values recommended by Owen for use in equation 5; 37% and 99% refer to the confidence level associated with R_{max}).

Authors	Dimensionless Discharge, Q_*	Dimensionless Freeboard, R_*	Overtopping Model
Hydraulics Research Station (1980); Owen (1982)	$\dfrac{Q}{T_m g H_s}$	$\dfrac{R_c}{T_m \sqrt{g H_s}}$	$Q_* = A \exp(-B R_*)$
Bradbury & Allsop (1988)	$\dfrac{Q}{T_m g H_s}$	$\dfrac{R_c^2}{T_m \sqrt{g H_s^3}}$	$Q_* = A (R_*)^{-B}$
Ahrens & Heimbaugh (1988)	$\dfrac{Q}{\sqrt{g H_s^3}}$	$\dfrac{R_c}{(H_s^2 L_p)^{1/3}}$	$Q_* = A \exp(-B R_*)$
Sawaragi et al (1988)	$\dfrac{Q}{\sqrt{g L_s H_s^2}}$	$\dfrac{R_c}{H_s}$	-----------------
Aminti & Franco (1988)	$\dfrac{Q}{T_m g H_s}$	$\dfrac{R_c}{H_s}$	$Q_* = A (R_*)^{-B}$
Pedersen & Burcharth (1992)	$\dfrac{Q T_m}{L_m^2}$	$\dfrac{R_c}{H_s}$	$Q_* = A (R_*)^{-B}$
De Waal & Van der Meer (1992)	$\dfrac{Q}{\sqrt{g H_s^3}}$	$\dfrac{R_c - R_{2\%}}{H_s}$	$Q_* = A \exp(-B R_*)$
Van der Meer (1993); Smith et al (1994); Van der Meer & Janssen (1995)	$\dfrac{Q}{\sqrt{g H_s^3}} \dfrac{\sqrt{\tan \alpha}}{\xi_p}$ for $\xi_p < 2$ $\dfrac{Q}{\sqrt{g H_s^3}}$ for $\xi_p > 2$	$\dfrac{R_c}{H_s} \dfrac{1}{\xi_p} \dfrac{1}{\gamma}$ for $\xi_p < 2$ $\dfrac{R_c}{H_s} \dfrac{1}{\gamma}$ for $\xi_p > 2$	$Q_* = A \exp(-B R_*)$
Franco et al (1994)	$\dfrac{Q}{\sqrt{g H_s^3}}$	$\dfrac{R_c}{H_s}$	$Q_* = A \exp(-B R_*)$

Table 1: Some options for dimensionless discharge, dimensionless freeboard and overtopping model.

		H&R MODEL (C given by $(R_{max})_{37\%}$)		H&R MODEL (C given by $(R_{max})_{99\%}$)		OWEN'S MODEL		
		LAD	LS	LAD	LS	LAD	LS	Rec.
Slope 1:1	A	0.00703	0.00581	0.00515	0.00474	0.00777	0.00758	0.00794
	B	3.42	3.22	6.06	6.04	20.44	21.27	20.12
Slope 1:2	A	0.00753	0.00790	0.00542	0.00614	0.0117	0.0125	0.0125
	B	4.17	4.55	7.16	7.98	21.71	22.80	22.06
Slope 1:4	A	0.0104	0.00792	0.00922	0.00870	0.0134	0.0164	0.0192
	B	6.27	5.94	10.96	11.12	42.92	46.12	46.96

Table 2: Regression coefficients for use in the H&R model and Owen's model. Also included for reference are Owen's recommended (Rec.) values.