



LABORATÓRIO NACIONAL  
DE ENGENHARIA CIVIL

**CONFIDENCIAL**

DEPARTAMENTO DE HIDRÁULICA E AMBIENTE  
Núcleo de Estuários e Zonas Costeiras

Proc. 0604/11/16262

## **MODELAÇÃO NUMÉRICA DE QUEBRA-MARES DESTACADOS**

Estudo integrado no Projecto Morfodinâmica de Praias  
do Plano de Investigação Programada (PIP) do LNEC  
para 2005-2008

Lisboa • Janeiro de 2009

**I&D** HIDRÁULICA E AMBIENTE

**RELATÓRIO 22/2009 – NEC**



# **MODELAÇÃO NUMÉRICA DE QUEBRA-MARES DESTACADOS**

## **RESUMO**

A crescente utilização internacional de quebra-mares destacados na estabilização de orlas costeiras levou à sua inclusão no modelo de evolução de linha de costa, Litmod, desenvolvido pelo LNEC.

Foi introduzido no modelo este tipo de obras e simulados os efeitos da difracção e da variação da altura ao longo da linha de rebentação sobre a intensidade do transporte das areias.

Foram efectuados testes de desenvolvimento de salientes e formação de tómbolos tendo-se verificado uma boa concordância entre os resultados obtidos e os provenientes de dados da natureza e de modelos físicos, apresentados por diferentes autores.

# **NUMERICAL MODELLING OF DETACHED BREAKWATERS**

## **ABSTRACT**

The increasing international use of coastal detached breakwaters for littoral protection led to the inclusion of this feature in the coastline evolution model LITMOD, developed at LNEC.

This type of coastal protection structures and its effect of wave diffraction and wave height variation along the breaker line in the sediment transport capacity were simulated.

The model tests concerned the development of salients and tombolos, where good agreement was found between our results and those from several authors, obtained from field observations and small-scale physical experiments.



# MODELAÇÃO NUMÉRICA DE QUEBRA-MARES DESTACADOS

## ÍNDICE

1.	INTRODUÇÃO .....	1
2.	CARACTERÍSTICAS GERAIS DO MODELO LITMOD .....	2
3.	QUEBRA-MARES DESTACADOS .....	3
4.	BASES FÍSICAS .....	4
	4.1 Considerações Gerais .....	4
	4.2 Perfil da praia .....	5
	4.3 Agitação marítima .....	6
	4.3.1 Representação da refração .....	6
	4.3.2 Representação da difração .....	8
	4.4 Transporte sólido litoral .....	11
5.	PROGRAMA DE CÁLCULO .....	14
	5.1 Programação geral do Litmod .....	14
	5.2 Programação do quebra-mar destacado .....	16
6.	TESTES DE VERIFICAÇÃO .....	18
	6.1 Considerações gerais .....	18
	6.2 Análise de sensibilidade de parâmetros .....	18
	6.3 Simetria das simulações .....	21
	6.4 Formação de salientes e tômbolos .....	22
	6.5 Simulação em condições naturais .....	25
7.	CONSIDERAÇÕES FINAIS .....	26
	BIBLIOGRAFIA .....	29
	ANEXO – LISTAGEM DA ROTINA DE CÁLCULO .....	31

## ÍNDICE DE FIGURAS

- Fig. 1 – Quebra-mares destacados com formação de salientes e tómbolos.
- Fig. 2 – Quebra-mares naturais com formação de saliente e tómbolo.
- Fig. 3 - Esquema de difracção.
- Fig. 4 – Gráficos de difracção de Hanson.
- Fig. 5 - Esquema de base para a programação.
- Fig. 6 – Formação de salientes com diferentes valores de abertura do leque de ondas ( $\Delta$ ), período (T) e alturas (m).
- Fig. 7 – Simetria das simulações com incidência oblíqua.
- Fig. 8- Esquema de um quebra-mar destacado com formação de saliente.
- Fig. 9 - Comparação de resultados do Litmod com a curva de correlação de Hsu e Silvester.
- Fig. 10 – Exemplos de simulação de tómbolos
- Fig. 11 – Série cronológica de dados de agitação marítima de Novembro de 1988 a Outubro de 1989.
- Fig. 12 – Simulação dos efeitos de três quebra-mares destacados.

# MODELAÇÃO NUMÉRICA DE QUEBRA-MARES DESTACADOS

## 1. INTRODUÇÃO

O Laboratório Nacional de Engenharia Civil (LNEC) dispõe de um modelo de simulação da evolução de linhas de costa (Vicente, 1991; Vicente e Clímaco, 2003; Clímaco e Vicente, 2007). Este modelo, desenvolvido no LNEC e designado por Litmod, utiliza algumas formulações relativamente simples, mas muito testadas pela experiência, para descrever os fenómenos físicos costeiros: leis de transporte sólido litoral baseadas no fluxo de energia das ondas; distribuições de transporte em perfil e profundidades de fecho, deduzidas de considerações empíricas e de investigações em modelo físico, disponíveis na bibliografia da especialidade; refacção de ondas de pequena amplitude sobre batimetria paralela, nas proximidades da rebentação; e difracção da onda devida à presença de obstáculos à sua propagação, constituídos por obras ou formações naturais.

O Litmod tem sido um instrumento de grande utilidade na abordagem de diversos problemas reais: reabilitação de praias afectadas por surtos erosivos; alargamento e melhoramento de praias; criação de praias artificiais; previsão da evolução de trechos de costa em desequilíbrio; previsão dos efeitos de obras costeiras sobre o equilíbrio das praias; e análise de medidas mitigadoras de erosões.

A crescente utilização internacional de quebra-mares destacados na estabilização de orlas costeiras levou à inclusão deste tipo de obras no Litmod. A formulação adoptada permite também simular o efeito sobre a morfologia da orla arenosa de formações naturais destacadas da costa, as quais apresentam um funcionamento hidráulico e sedimentar idêntico ao dessas obras.

## **2. CARACTERÍSTICAS GERAIS DO MODELO LITMOD**

O Litmod permite tratar trechos de grande comprimento, da ordem das dezenas de quilómetros, e testar evoluções de longo prazo, até algumas dezenas de anos. Apresenta grande flexibilidade de utilização, admitindo a adaptação dos módulos de cálculo às particularidades de cada situação concreta.

Trata-se de um modelo numérico unidimensional em que a evolução da costa é representada pelas variações de posição da linha de água. O seu funcionamento baseia-se no cálculo do transporte litoral, efectuado a partir das características das ondas na rebentação, e na aplicação da equação da continuidade aos volumes de areia movimentados.

A variação da posição da linha de água é calculada, em cada uma das numerosas células em que se divide o trecho de praia, a partir das diferenças de caudal sólido que nela penetram e a abandonam num curto intervalo de tempo. Admite-se que o perfil transversal no seu avanço ou recuo se desloca paralelamente a si próprio, sem alteração de forma.

São elementos de base no cálculo dos caudais sólidos as alturas, os períodos e os ângulos das ondas na rebentação. A posição da rebentação, e conseqüentemente os valores desses parâmetros, são afectados pela evolução da posição da linha de água, pelo que as características da onda na rebentação são objecto de um cálculo iterativo que tem em conta as profundidades que determinam a rebentação.

O modelo é aplicável a simulações da evolução de uma linha de costa, onde coexistam praias e trechos rochosos, em condições naturais ou com obras e intervenções de alimentação artificial. No caso de evoluções naturais, poderá permitir compreender a dinâmica e a configuração de uma costa sem obras mas onde possam existir fontes aluvionares, saliências rochosas que produzam retenção de sedimentos ou efeitos de difracção, e prever alterações decorrentes das oscilações naturais da alimentação de areias ou das taxas de transporte. Pode também simular os ganhos e perdas aluvionares provenientes: de cursos de água que debitem para a costa; da erosão de falésias; e de actividades de extracção de areias. Pode ainda ter em conta os efeitos de diversos tipos de obras que afectem o equilíbrio de uma costa: molhes, esporões e defesas frontais. Pode tratar não só

obras isoladas como combinações de vários tipos, associadas ou não a alimentação artificial.

### 3. QUEBRA-MARES DESTACADOS

O quebra-mar destacado é uma estrutura longitudinal desligada da costa, que tem por função acumular areias num determinado trecho de praia e protegê-lo da acção da ondulação. Este tipo de obra pode ser: emerso ou submerso; paralelo ou oblíquo à costa; contínuo ou segmentado.



Fig. 1 – Quebra-mares destacados com formação de salientes e tómbolos

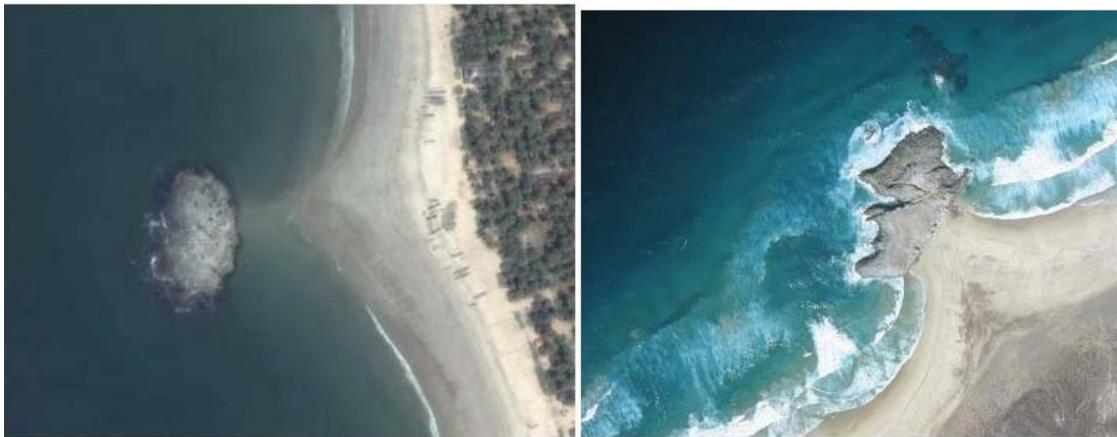


Fig. 2 – Quebra-mares naturais com formação de saliente e tómbolo

Estas obras ou formações naturais, ao interceptarem a propagação de um trecho da frente de onda originam uma atenuação da energia dissipada na zona que

protegem, alterando os padrões de movimentação e deposição dos sedimentos. Podem modificar significativamente a morfologia local, devido aos acentuados efeitos de difracção que introduzem na propagação das ondas. O equilíbrio sedimentar resultante traduz-se numa acumulação de areias na zona de sombra da obra, com avanço da linha de água. Este avanço pode ser limitado a uma protuberância designada por saliente, ou desenvolver-se até atingir a obra, situação em que toma o nome de tómbolo.

Trata-se de um tipo de obra que, no caso dos salientes, não bloqueia completamente o fluxo longitudinal das areias, permitindo alguma continuidade do transporte sólido litoral ao longo da linha de água.

A utilização de quebra-mares destacados pode ter os seguintes objectivos: controlar a erosão de uma orla arenosa; permitir a construção de praias artificiais, tirando partido da sua capacidade de concentrar massas de areia em trechos escolhidos da praia; ou proteger um trecho de costa de espraamentos e galgamentos indesejáveis, pela redução das alturas de onda que nele incidem.

O uso de quebra-mares destacados tem tido larga aplicação em especial na Itália, Japão, Espanha e Estados Unidos da América.

A tarefa de desenvolvimento de um módulo de simulação deste tipo de obras, e sua inclusão no modelo Litmod, abrangeu: a análise das bases físicas de funcionamento dos quebra-mares; a programação de uma nova rotina de cálculo e sua ligação ao programa geral; e a realização de testes para avaliar a capacidade de simulação das evoluções morfológicas.

## **4. BASES FÍSICAS**

### **4.1 Considerações Gerais**

As ondas são o factor determinante da movimentação e equilíbrio das areias das praias. As suas alturas e ângulos na rebentação condicionam o valor e sentido do transporte sólido litoral. Desta forma é indispensável, para interpretar a dinâmica e

morfologia de uma praia, caracterizar previamente a propagação das diferentes ondas do regime de agitação marítima, entre o largo e a rebentação.

Apresentam-se a seguir os parâmetros e fenómenos físicos que intervêm nessas situações, os quais se relacionam com a forma do perfil da praia, a profundidade limite da movimentação significativa das areias, a refração e difracção das ondas e o processamento do transporte sólido natural.

## 4.2 Perfil da praia

A forma do perfil é uma informação usada para definir a ordenada do ponto de rebentação e a profundidade nas extremidades do quebra-mar. Estes parâmetros são necessários para determinar os efeitos de difracção devidos à obra. A forma do perfil serve ainda para definir o declive da praia, grandeza que intervêm nalgumas expressões de cálculo do transporte litoral.

Existem diferentes propostas de expressões para descrição da **forma do perfil** da praia. Poderá em cada caso, e face aos dados do perfil natural, escolher-se a mais adequada. Dean (1991) propôs uma expressão simples para esse perfil, elaborada a partir de dados da natureza:

$$y = \left( \frac{h}{A} \right)^{1,5}$$

Em que é:

$y$  distância de um ponto genérico do perfil à linha de água

$h$  tirante de água no mesmo ponto, e

$A$  o coeficiente da parábola que descreve o perfil.

Este coeficiente calcula-se pela expressão:

$$A = 0,067 \cdot W^{0,44}$$

em que  $W$  é a velocidade de queda dos grãos de areia da praia.

O transporte sólido litoral processa-se paralelamente à costa, entre a linha de água e a **profundidade de fecho** do perfil. Este parâmetro corresponde à cota a partir da qual o transporte de areias é tão diminuto que não origina sensíveis variações

morfológicas. Define, juntamente com a altura média da berma da praia, a espessura da camada de areia que avança e recua quando se altera a posição da linha de água. Depende essencialmente da agitação marítima e pode ser calculado a partir da expressão de Birkemeier (1985):

$$P = 1,75 H - 57,9 (H^2 / g T^2)$$

em que é:

$P$  profundidade de fecho

$H$  e  $T$  altura e período significativos da onda com probabilidade de ser excedida, em média, 0,5 dias por ano, e

$g$  aceleração da gravidade.

A título de exemplo, com  $H = 5$  m e  $T = 13$  s, obtém-se:  $P = 7,9$  m.

### **4.3 Agitação marítima**

#### **4.3.1 Representação da refração**

A aplicação do Litmod exige a definição das distribuições das alturas e dos ângulos na rebentação, ao longo da linha de praia, para definição do transporte sólido litoral. Recebe, para esse efeito, o apoio de um modelo de propagação de ondas que simula a propagação entre o largo e a profundidade de aproximação,  $d_a$ , fornecendo os parâmetros das ondas nesta profundidade.

Por seu lado, o modelo Litmod dispõe de um módulo de cálculo de ondas que a partir dos valores à profundidade de aproximação calcula por iterações, supondo a batimetria paralela, a profundidade a que se dará a rebentação e as alturas e ângulos da onda nesse ponto.

Na refração sobre batimetria paralela utilizam-se as expressões da teoria da onda de pequena amplitude. A relação entre as alturas das ondas na rebentação e na batimétrica de aproximação é obtida pela expressão:

$$\frac{H_b}{H_a} = K_e \cdot K_r$$

em que  $K_e$  é o coeficiente de empolamento entre as duas posições e  $K_r$  o coeficiente de refração.

As expressões destes coeficientes são, por seu lado:

$$K_e = \sqrt{n_a \cdot \frac{L_a}{L_b}}$$

com

$$n_a = \frac{1}{2} \left[ 1 + \frac{\frac{4\pi d_a}{L_a}}{\sinh \frac{4\pi d_a}{L_a}} \right]$$

e

$$K_r = \sqrt{\frac{\cos \alpha_a}{\cos \alpha_b}}$$

em que  $d$  é a profundidade,  $L$  o comprimento de onda e  $\alpha$  o ângulo da onda com a costa, dizendo os índices  $a$  e  $b$  respeito à batimétrica de aproximação e à rebentação, respectivamente.

Utiliza-se um critério de rebentação que poderá ser o que é traduzido pela relação:

$$H_b = 0.78d_b$$

A determinação do ângulo na rebentação é feita pela aplicação da lei de Snell:

$$\frac{C_a}{C_b} = \frac{\sin \alpha_a}{\sin \alpha_b}$$

em que  $H$  é a altura da onda e  $C$  é a sua celeridade.

O cálculo das características da onda na rebentação é feito por aproximações sucessivas. Começa-se por calcular o produto  $K_e \cdot K_r$ , admitindo, em primeira aproximação, que  $H_b = H_a$ . Com esse produto determina-se novo valor, mais correcto, de  $H_b$ , repetindo-se o cálculo até que a variação seja inferior a determinada percentagem que se considere corresponder a uma aproximação satisfatória.

Com o último valor de  $H_b$ , fica a dispor-se também do correspondente  $\alpha_b$ , que foi necessário para o seu cálculo.

Refere-se ainda que os valores dos comprimentos de onda em função das profundidades e do período,  $T$ , necessários neste processo de cálculo, se obtêm também por iterações, a partir da expressão:

$$L = \frac{gT^2}{2\pi} \cdot \tanh \frac{2\pi d}{L}$$

### 4.3.2 Representação da difracção

Para considerar o efeito da difracção sobre os ângulos e as alturas das ondas na rebentação pode utilizar-se o método referido por Horikawa (1988), em que as zonas iluminadas e de sombra apresentam processos de cálculo diferentes (Fig. 3).

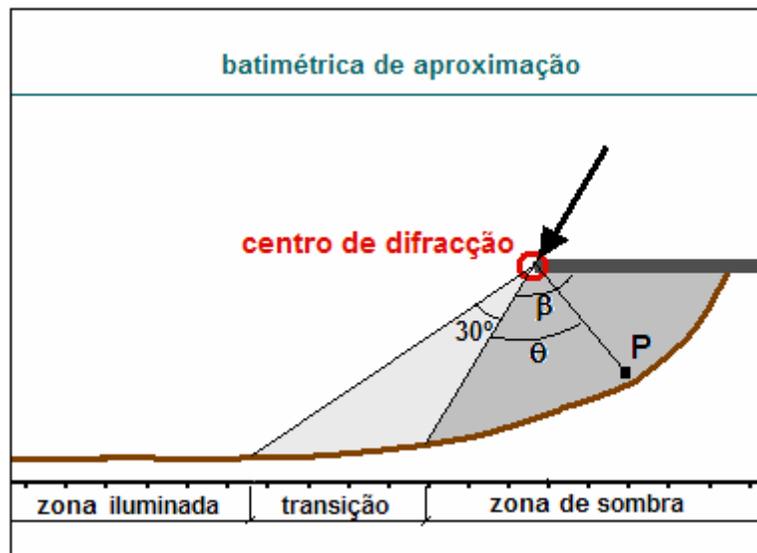


Fig. 3 - Esquema de difracção

#### Zonas iluminada e de transição

Os cálculos de  $H_b$  e  $\alpha_b$  quando o ponto  $P$  se encontra localizado numa destas zonas são efectuados, basicamente, segundo um processo de propagação da onda sobre batimetria paralela, a partir de uma batimétrica de aproximação, onde são conhecidos os diferentes parâmetros da onda.

O processo só é, no entanto, igual ao anteriormente referido, na parte da zona iluminada que não é afectada pela difracção. Já na zona de transição, onde se começa a sentir o efeito de difracção, o procedimento é ligeiramente diferente. No cálculo da altura da rebentação,  $H_b$ , é tido em conta o coeficiente local de difracção, vindo assim ligeiramente alterado  $H_b$  e consequentemente a profundidade de rebentação,  $d_b$ , e o ângulo na rebentação  $\alpha_b$ .

### **Zona de sombra**

Nos cálculos localizados nesta zona sobrepõem-se os efeitos de difracção e de refracção. Procede-se da forma seguinte:

- Considera-se, numa primeira aproximação, que a posição do ponto  $P$  onde se verifica a rebentação, coincide com a linha de água. Calcula-se o coeficiente de difracção nesse ponto por meio de um processo simplificado de cálculo da difracção. Utiliza-se opcionalmente o método de Wiegel (1964) ou o proposto por Goda *et al.* (1978). No primeiro caso utilizam-se as isolinhas de difracção para um obstáculo semi-indefinido propostas pelo autor. No segundo usam-se as curvas de Hanson (1987) que a seguir se apresentam (Fig. 4), em que o coeficiente de difracção é função de  $\Theta$  e do parâmetro de concentração  $S_{max}$ . Este dependerá da profundidade relativa na extremidade da obra, parâmetro designado por  $D_{TP}/L_0$ , e do grau de irregularidade da agitação.
- Calculado o coeficiente de difracção,  $K_d$ , determina-se em primeira aproximação o valor da altura na rebentação, a partir da altura da onda na extremidade do quebra-mar,  $H_Q$ :

$$H_b = H_Q \cdot K_d$$

e a profundidade de rebentação correspondente.

- Conhecida esta, é calculada a ordenada,  $y$ , do ponto de rebentação, tendo em conta a função que define o perfil da praia.
- É então feito um primeiro cálculo de refracção considerando a batimetria paralela à orientação da costa, na célula em causa. É também calculado o

coeficiente de empolamento,  $K_e$ , o qual é função do comprimento da onda e das profundidades no obstáculo e no ponto de rebentação.

- Calcula-se um novo valor de altura na rebentação:

$$H_b = H_Q \cdot K_d \cdot K_e \cdot K_r$$

- Repete-se o cálculo da refração até ser obtido um valor de  $H_b$ , cuja diferença em relação ao anterior seja aceitável.

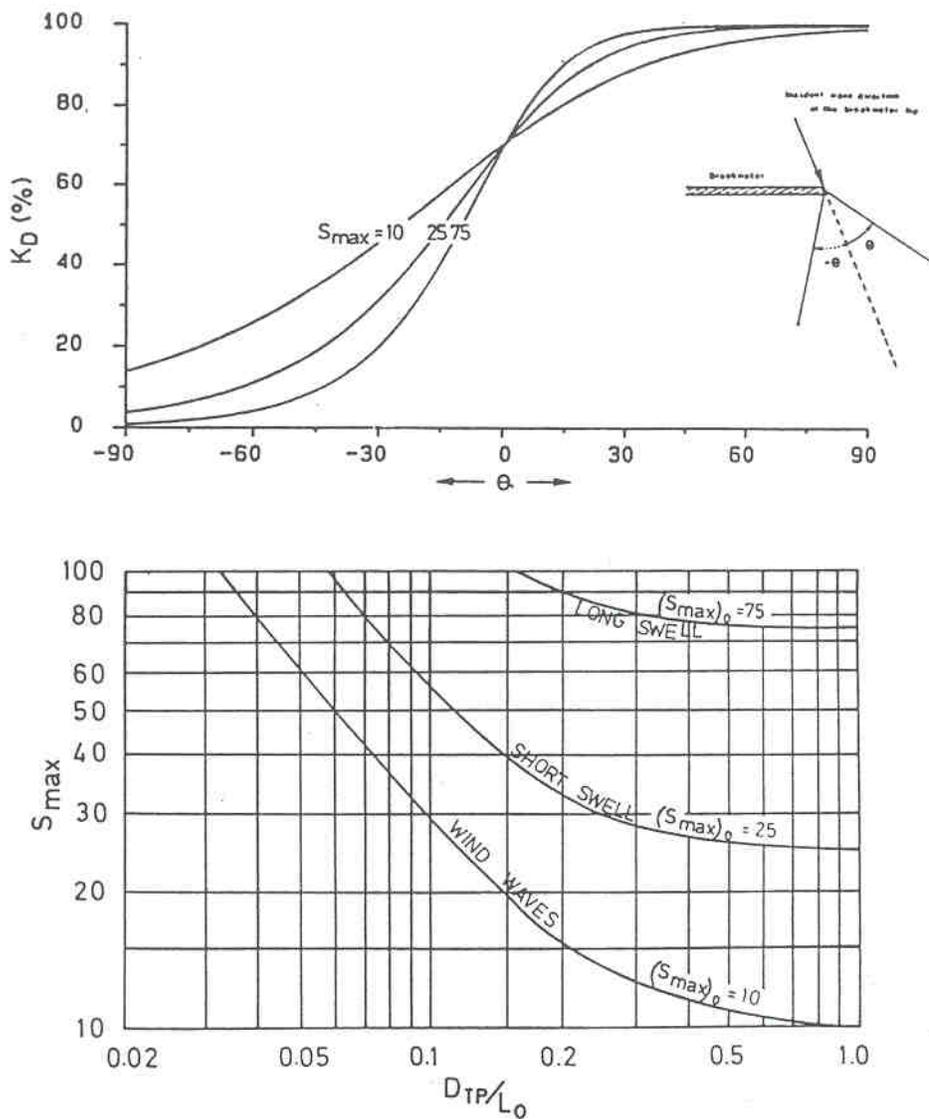


Fig. 4 - Gráficos de difracção de Hanson  
[Extraídos de Hanson (1987)]

Ficam a conhecer-se, terminado o cálculo, os valores de  $H_b$ ,  $\alpha_b$  e  $L_b$  na célula do modelo onde se localiza o ponto  $P$ , parâmetros indispensáveis à avaliação do transporte litoral pela fórmula do CERC. A variação de altura nas células vizinhas permitirá acrescentar uma parcela de transporte devida ao gradiente das alturas das ondas na rebentação.

#### 4.4 Transporte sólido litoral

As formulações mais utilizadas para o cálculo do transporte sólido litoral continuam ainda a basear-se na correlação entre o fluxo longitudinal da energia da onda na rebentação e o fluxo da massa de sedimentos. As fórmulas do tipo indicado, como a do Coastal Engineering Research Center (1975 e 1984) e a de Inman e Bagnold em 1963 (Komar, 1981 e 1988), são as mais utilizadas. A correlação que estabelecem é do tipo:

$$Q = K P_l$$

em que é:

$Q$  transporte sólido litoral

$P_l$  componente segundo a direcção da costa, e por unidade de comprimento desta, da potência dissipada pela onda na rebentação

$K$  constante de correlação obtida de medições na natureza.

Verifica-se que são idênticos os valores de  $K$  nas fórmulas do CERC e de Inman. De facto, segundo a apresentação feita pelo CERC,  $K$  é igual a 7500, com  $Q$  expresso em jardas cúbicas por ano e  $P_l$  em pés x libra por segundo e por pé de praia.

Transferindo as unidades para  $m^3/dia$  e  $kg/s$ , vem

$$Q (m^3/dia) = 34,6 P_l (Kg/s)$$

A fórmula de Inman é homogénea, sendo  $Q$  o caudal sólido em peso submerso e, portanto, em unidades de força por unidades de tempo, tal como a grandeza  $P_l$ . O valor atribuído por Inman a  $K$  foi 0.77, sendo um coeficiente adimensional.

Considerando na fórmula de Inman um peso específico para a areia submersa de  $1600 \text{ Kg/m}^3$ , um índice de vazios de 0.4 e efectuando a mesma transferência de unidades, obtém-se:

$$Q (\text{m}^3/\text{dia}) = 69.3 P_l (\text{Kg/s})$$

A aparente diferença entre os valores de  $K$  resulta do facto de a potência longitudinal na fórmula do CERC ser calculada com a altura significativa,  $H_S$ , enquanto que na de Inman se calcula com a raiz quadrada da média das alturas quadráticas,  $H_{rms}$ . Aplicando a relação:

$$H_S = 1.42 H_{rms}$$

e tendo em atenção que a altura da onda contribui com o seu valor elevado ao quadrado para o cálculo da potência longitudinal, verifica-se que as duas constantes são praticamente iguais.

Embora isso aconteça nas duas expressões mais usadas e a constante seja a mais ajustada aos conjuntos de dados experimentais dos numerosos investigadores que abordaram o tema, não se pode esquecer que existe uma apreciável dispersão nesses conjuntos. Este facto explica que alguns autores de programas de evolução da linha de costa considerem  $K$  como um parâmetro de calibração dos seus modelos.

#### Expressão de $P_l$

O fluxo de energia de uma onda, por unidade de comprimento da crista, é traduzido pelo produto:

$$E \cdot C_g$$

em que é:

$E$  densidade de energia

$C_g$  celeridade de grupo.

Estes parâmetros têm as expressões, na teoria das ondas de pequena amplitude:

$$E = \frac{\gamma H^2}{8}$$

e

$$C_g = n \cdot C = \frac{1}{2} \left[ 1 + \frac{\frac{4\pi d}{L}}{\sinh \frac{4\pi d}{L}} \right] \cdot \frac{L}{T}$$

em que é:

$H$  altura de onda

$C=L/T$  celeridade, sendo  $L$  o comprimento e  $T$  o período

$\gamma$  peso específico da água

$d$  profundidade.

Como na rebentação é  $n \simeq 1$ , a expressão do fluxo de energia por unidade tempo (potência) e de comprimento de costa é:

$$P = \frac{\gamma H_b^2 L_b}{8T} \cdot \cos \alpha_b$$

e a componente segundo a linha de costa e por unidade desta:

$$P_l = \frac{\gamma H_b^2 L_b}{8T} \cdot \text{sen} \alpha_b \cdot \cos \alpha_b$$

em que  $\alpha_b$  é o ângulo da onda com a costa na rebentação.

Resulta

$$Q (m^3/dia) = 34,6 \cdot \frac{\gamma H_b^2 L_b}{8T} \cdot \text{sen} \alpha_b \cdot \cos \alpha_b$$

Quando o gradiente da altura de onda ao longo da praia não é desprezável, como sucede nas zonas de difracção, é importante usar fórmulas de cálculo do transporte sólido litoral que tenham em conta esse efeito. Este foi o procedimento adoptado no modelo Genesis, tendo-se Hanson apoiado nas análises de Kraus *et al.* (1981) e Kraus e Harikai (1983).

Segundo esta formulação é adicionada à expressão tradicional, baseada no fluxo de energia, uma segunda parcela que introduz os efeitos da variação de altura.

No modelo Litmod segue-se essa mesma orientação, tendo-se adoptado uma expressão com a seguinte estrutura:

$$Q_t = Q + Q_H$$

Em que são:

$Q_t$  o caudal total

$Q$  o caudal devido à componente longitudinal do fluxo de energia

$Q_H$  o caudal devido à variação da altura de onda ao longo da praia

Segundo os autores citados a componente  $Q_H$  cresce com o gradiente da altura da onda e com a potência dissipada na rebentação, sofrendo ainda algum efeito do declive do perfil da praia,  $m$ . É proposta a seguinte expressão:

$$Q_H = K_H \cdot P \cdot 1/m \cdot dH/dx = K_H \cdot \frac{\gamma H_b^2 L_b}{8T} \cdot \cos \alpha_b \cdot 1/m \cdot dH/dx$$

em que  $K_H$  é um coeficiente de correlação.

## 5. PROGRAMA DE CÁLCULO

### 5.1 Programação geral do Litmod

O programa começa por efectuar a leitura de um primeiro conjunto de dados: características gerais do modelo constituídas pelo número de células, abcissa da primeira célula, comprimento das células, passo temporal de cálculo e ângulo da linha de base com o norte; definição da linha de costa, nível da berma e forma de perfis da praia; condições nos limites, constituídas pelos valores ou expressões de cálculo dos caudais sólidos na primeira e na última célula em que o trecho de costa foi subdividido; características da agitação marítima incidente na batimétrica de aproximação; tipo de fórmula de cálculo de transporte sólido litoral dentre as disponíveis na programação do modelo.

Lê seguidamente outros dados complementares de caracterização das condições naturais: fontes aluvionares, localização e definição de trechos rochosos resistentes à erosão; definição da implantação de trechos de falésia erodível, com indicação de alturas de falésia, recuo médio anual, e taxa de transformação em areia dos volumes erodidos. Os últimos dados são relativos à definição de intervenções do tipo alimentação artificial, esporões e defesas frontais.

O programa actua sobre esses dados considerando sucessivos passos de tempo elementares, e efectuando em cada um deles o varrimento do conjunto de células em que se dividiu o trecho de costa.

Os principais cálculos que efectua são os seguintes:

- Interpolação, a partir dos pontos da batimétrica de aproximação onde são definidos os regimes, dos ângulos com a linha de base e das alturas das ondas em todas as células, chamando a subrotina de interpolação IPT.
- Determinação, na batimétrica de aproximação, dos ângulos da onda com a linha de costa e cálculo da propagação da onda até à rebentação, chamando a subrotina H\_REB.
- Avaliação dos caudais de transporte litoral em todas as células, através da subrotina FOR\_CAUDAL, de acordo com a fórmula de cálculo escolhida, e as condições nos limites.
- Aplicação da lei da continuidade e determinação da nova posição da linha de praia.
- Determinação da distribuição de caudais em perfis, se essa opção figurar nos dados, através das subrotinas COMP e Q\_PERFIL.
- Armazenamento dos valores das coordenadas das linhas de água a apresentar.
- Comparação do tempo de duração do cálculo com a duração de cada uma das ondas do regime e introdução dos parâmetros da onda seguinte do regime.

Para além deste quadro de cálculos e tarefas que são sempre efectuados, existem diversas opções complementares que só são accionadas quando o caso de aplicação o justifique. Essas opções que estão traduzidas em subrotinas próprias dizem respeito à existência de falésias erodíveis, defesas frontais, esporões, intervenções de alimentação artificial e extracções de areia.

O objectivo do presente trabalho foi o desenvolvimento de uma nova componente opcional, destinada a simular o funcionamento de um quebra-mar destacado ou de um conjunto de obras deste tipo.

## 5.2 Programação do quebra-mar destacado

Apresenta-se no Anexo A a programação do quebra-mar destacado, cujos aspectos principais se descrevem a seguir.

O modelo efectua um primeiro varrimento de todas as células para cálculo dos parâmetros da onda na rebentação, ainda sem considerar a presença da obra. Numa segunda passagem aplica a programação específica do quebra-mar, a qual introduz os efeitos de difracção (Fig. 5).

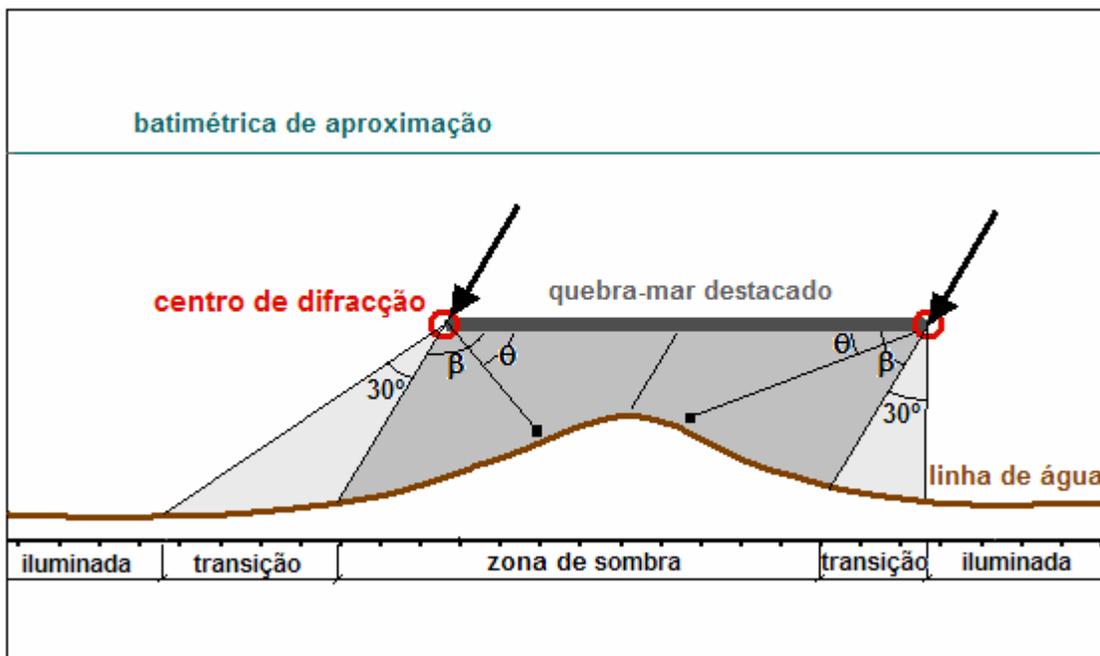


Fig. 5 - Esquema de base para a programação

A programação implica os seguintes acréscimos e alterações do programa geral:

- **Novos dados de entrada**

- Número de quebra-mares.
- Coordenadas das extremidades de cada uma das obras (centros de difracção).
- Opção pelo método de cálculo da difracção de Goda ou de Wiegel.
- Perfis da praia nos trechos de implantação das obras.
- Coeficiente de calibração da parcela da fórmula de transporte sólido litoral que introduz o efeito da variação longitudinal da altura de onda. Nesta análise utilizou-se para esse coeficiente o valor de 0,6, de acordo com simulações para análise de sensibilidade desse parâmetro efectuadas em anterior estudo (Clímaco *et al.*, 2009).

- **Cálculos adicionais de propagação das ondas**

- Determinam-se os parâmetros da onda nos centros de difracção. Considera-se propagação sobre batimetria paralela, entre a batimétrica de aproximação e a obra.
- Conhecida a direcção da onda incidente no centro de difracção delimita-se a zona de sombra. Define-se também a zona de transição, ainda influenciada pela difracção, a qual se estende por um sector de 30°.
- Determinam-se os parâmetros da onda na rebentação, nas zonas de sombra e de transição, de acordo com o procedimento anteriormente referido. É utilizado opcionalmente o método de Goda ou de Wiegel, dispondo o programa geral do modelo das correspondentes tabelas para interpolação dos valores dos parâmetros.
- Os cálculos nas células da metade esquerda da zona de sombra são efectuados a partir do centro de difracção esquerdo e nas células da metade direita a partir do centro de difracção direito.
- Substituem-se os parâmetros da onda, nas zonas de sombra e de transição, pelos novos valores que incluem os efeitos da difracção.

- **Alteração do cálculo do transporte**

- Após os cálculos de propagação que se acabam de referir o modelo fica a dispor dos parâmetros da agitação marítima em todas as células e pode prosseguir a quantificação dos transportes e balanços aluvionares. São tidos em conta na fórmula de transporte sólido litoral os efeitos da acentuada variação da altura da onda ao longo da praia, característicos da difracção, segundo a expressão de cálculo acima indicada.

## **6. TESTES DE VERIFICAÇÃO**

### **6.1 Considerações gerais**

A avaliação da eficácia da programação foi efectuada segundo duas vertentes:

- Reprodução de aspectos qualitativos da morfologia de salientes e tómbolos no que respeita: à sensibilidade às variações das alturas das ondas, períodos e abertura do leque de direcções; à simetria em actuações frontais da onda e nos casos de igual obliquidade da onda num e noutro sentido da praia.
- Simulação de aspectos quantitativos de dimensão do saliente e de passagem de saliente a tómbolo.

### **6.2 Análise de sensibilidade de parâmetros**

Testou-se uma incidência frontal durante o tempo necessário para alcançar o completo desenvolvimento do saliente. Foi considerado em todas as simulações um mesmo quebra-mar destacado com 160 m de comprimento, distanciado 100 m da linha de água rectilínea inicial. Usou-se um leque simétrico de direcções fazendo-se variar: alturas,  $H$ ; períodos,  $T$ ; e abertura do leque,  $\Delta$ . Tratava-se de um leque de 5 ondas que actuavam repetidamente de forma simétrica.

Nestas circunstâncias ocorria sempre a formação de um saliente com um eixo de simetria que passava pelo meio da obra. Apresentam-se no Quadro 1 as condições de simulação e o desenvolvimento atingido pelo saliente.

Estes resultados mostram, na gama de variação utilizada, que é reduzida a influência do período e da abertura do leque e que há um crescimento acentuado do saliente com a altura da onda.

Das numerosas simulações efectuadas nesta análise de desenvolvimento de salientes e formação de tómbolos apresentam-se como exemplo os 3 casos da Fig. 6, assinalados no Quadro 1.

Quadro 1 – Desenvolvimento atingido pelo saliente (m)

<b><math>\Delta = 10^\circ</math></b>	<b>H(m)</b>	<b>0,5</b>	<b>1,0</b>	<b>2,0</b>
	<b>T(s)</b>			
	<b>7</b>	41	52	67
	<b>9</b>	45	55	71
	<b>11</b>	46	56	74
<b><math>\Delta = 20^\circ</math></b>	<b>H(m)</b>	<b>0,5</b>	<b>1,0</b>	<b>2,0</b>
	<b>T(s)</b>			
	<b>7</b>	41	51	67
	<b>9</b>	43	53	71
	<b>11</b>	44	54	73
<b><math>\Delta = 30^\circ</math></b>	<b>H(m)</b>	<b>0,5</b>	<b>1,0</b>	<b>2,0</b>
	<b>T(s)</b>			
	<b>7</b>	44	55	73
	<b>9</b>	46	57	76
	<b>11</b>	48	59	79

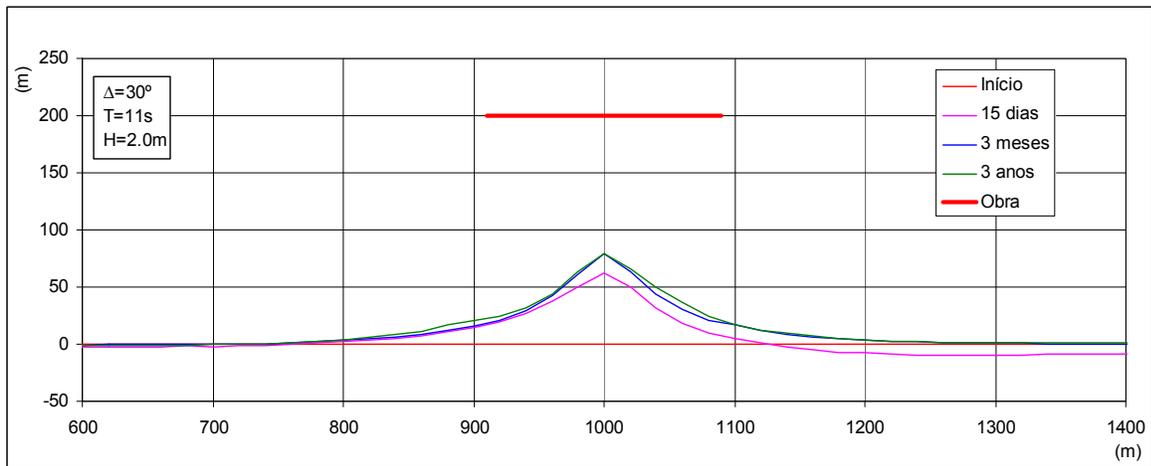
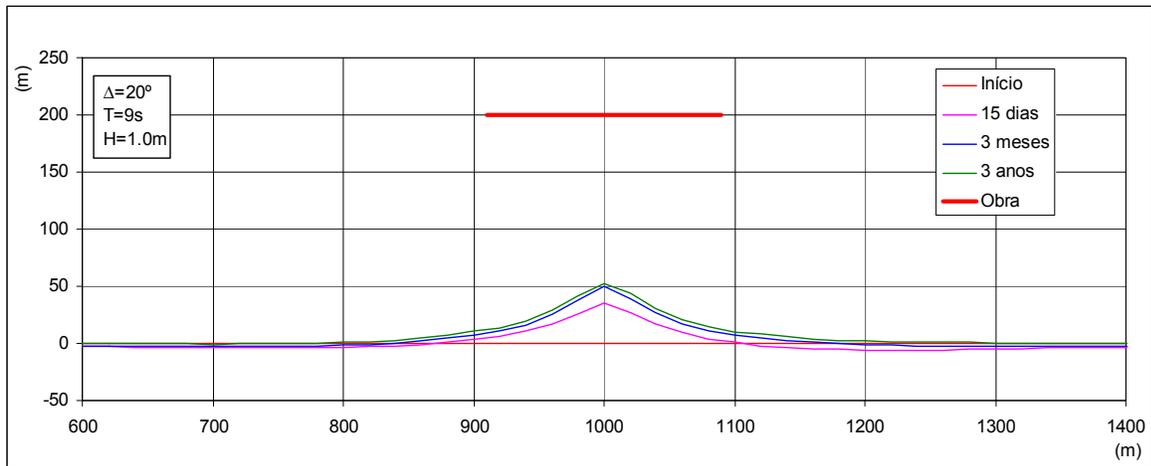
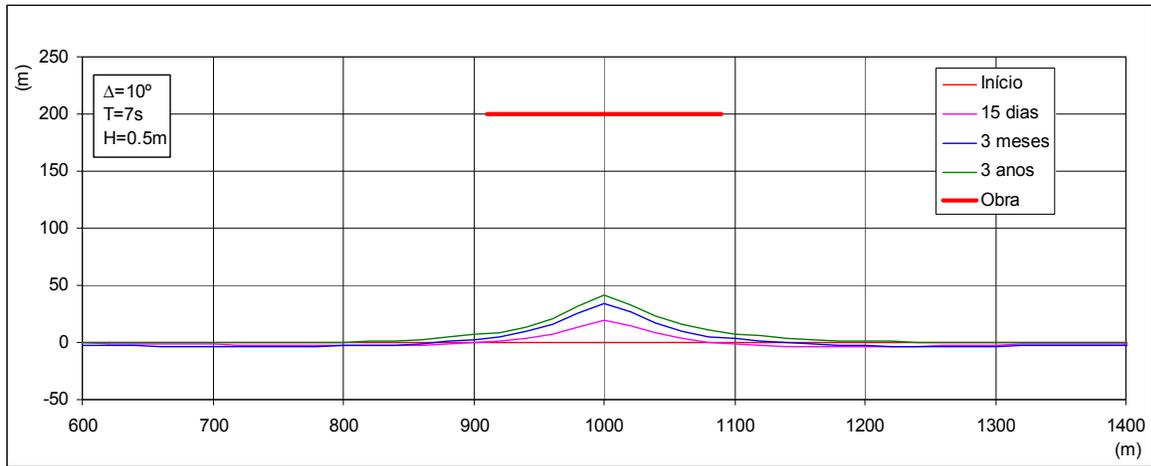


Fig. 6 – Formação de salientes com diferentes valores de abertura do leque de ondas ( $\Delta$ ), período (T) e alturas (m)

### 6.3 Simetria das simulações

Quando a incidência é frontal a linha de água resultante da simulação deve ser simétrica, com o eixo de simetria a passar pelo meio da obra. Os resultados das simulações referidas no ponto anterior mostraram que a programação respeita essa simetria.

No caso de incidências oblíquas simétricas as linhas de água resultantes deverão ser simétricas entre si. Este comportamento foi também testado no modelo para diferentes condições, apresentando-se um exemplo na Fig. 7.

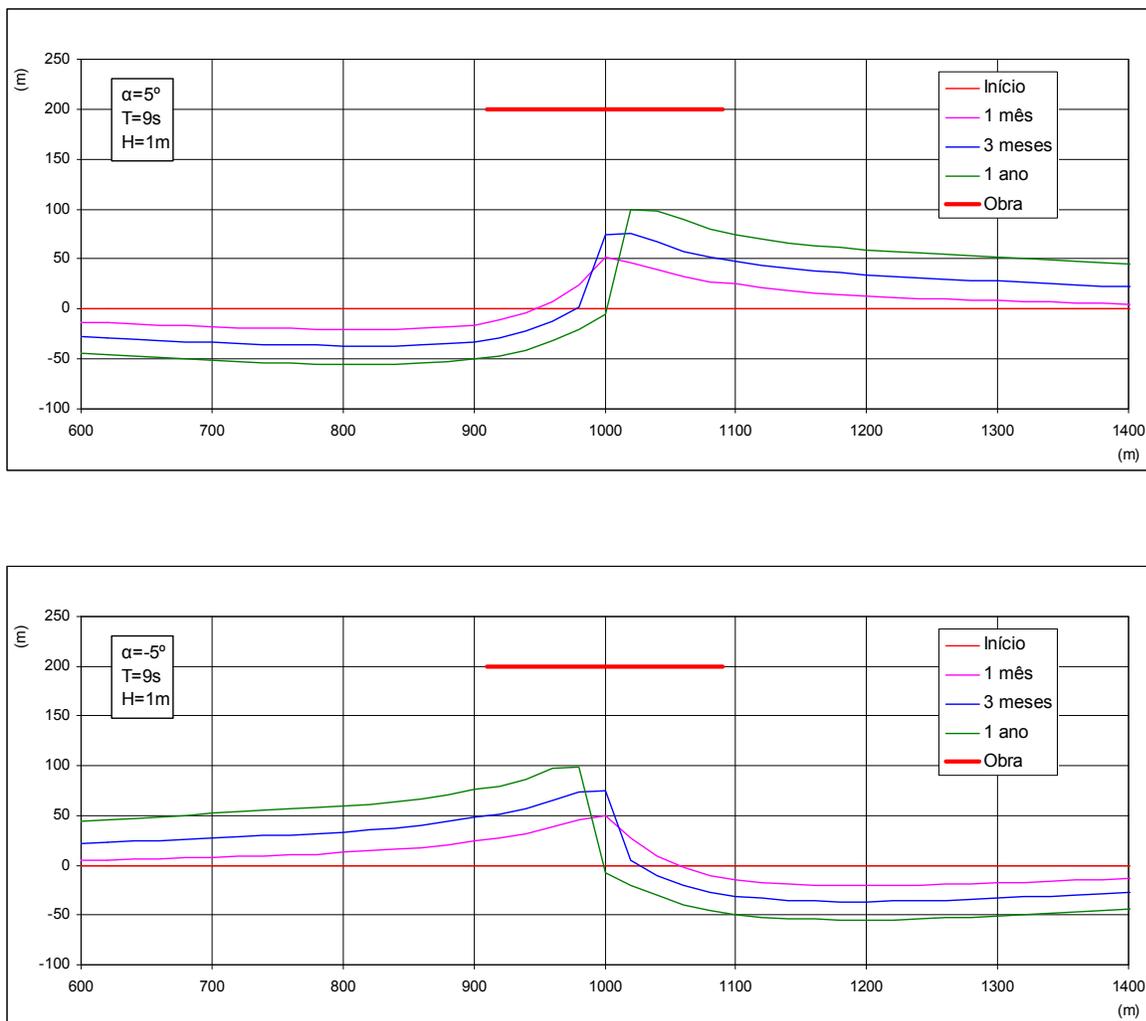


Fig. 7 – Simetria das simulações com incidência oblíqua.

## 6.4 Formação de salientes e tómbolos

A formação e desenvolvimento de salientes têm sido objecto de estudos por diversos autores com vista à determinação de parâmetros e critérios de previsão.

Para aferir os resultados do modelo recorreu-se ao estudo de Hsu e Silvester (1990). Trata-se de um estudo adequado a esse fim pois tem por base dados reais provenientes de modelação física e da natureza. Considera as variáveis  $B$ ,  $S$  e  $X$  como determinantes da morfologia do saliente e utiliza os parâmetros adimensionais  $B/S$  e  $X/B$  para as correlacionar (Fig.8).

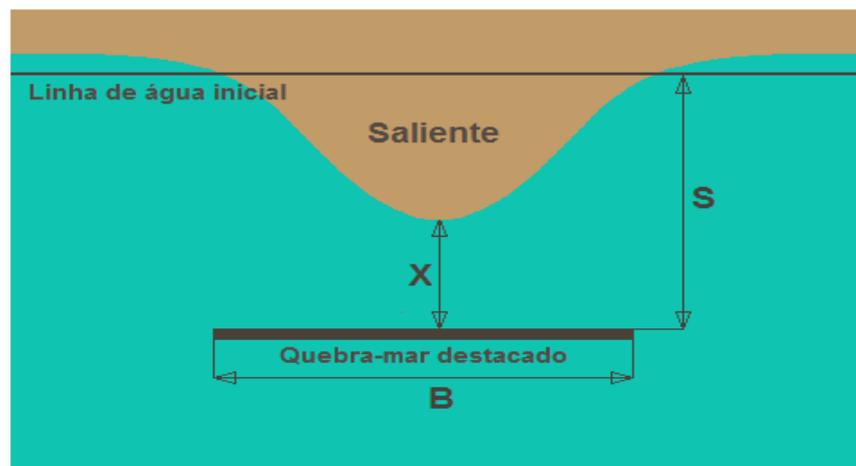


Fig. 8- Esquema de um quebra-mar destacado com formação de saliente

O critério de aferição do modelo consistiu em realizar diversas simulações, com incidência frontal, fazendo variar o parâmetro  $B/S$ , e comparar os resultados obtidos para o parâmetro  $X/B$  com a curva de correlação referida (Quadro 2 e Fig. 9). Verifica-se que o desenvolvimento dos salientes obtidos no modelo está em boa concordância com a curva de correlação do conjunto de dados de modelo físico e da natureza.

A definição das condições determinantes da formação de tómbolos, foi também objecto de análise por diversos investigadores da área da engenharia costeira (CERC, 1984; Gourlay, 1981; e Dally e Pope, 1986). Embora os critérios que propõem apresentem algumas diferenças, é possível extrair uma condição geral de passagem de saliente a tómbolo para valores de  $B/S$  entre 1,5 e 2,5, com maior

incidência no valor 2. Os resultados das simulações efectuadas com o modelo, embora se afastem um pouco da curva de Hsu e Silvester, são consentâneas com essa gama de valores.

Quadro 2 – Resultados da simulação com o Litmod

<b>B=220 m</b>		<b>B=180 m</b>		<b>B=140 m</b>	
<b>B/S</b>	<b>X/B</b>	<b>B/S</b>	<b>X/B</b>	<b>B/S</b>	<b>X/B</b>
0,25	3,89	0,25	3,88	0,25	3,86
0,5	1,84	0,5	1,82	0,5	1,78
1,0	0,66	1,0	0,60	1,0	0,59
1,5	0	1,5	0	1,5	0
2,0	0	2,0	0	2,0	0
2,5	0	2,5	0	2,5	0

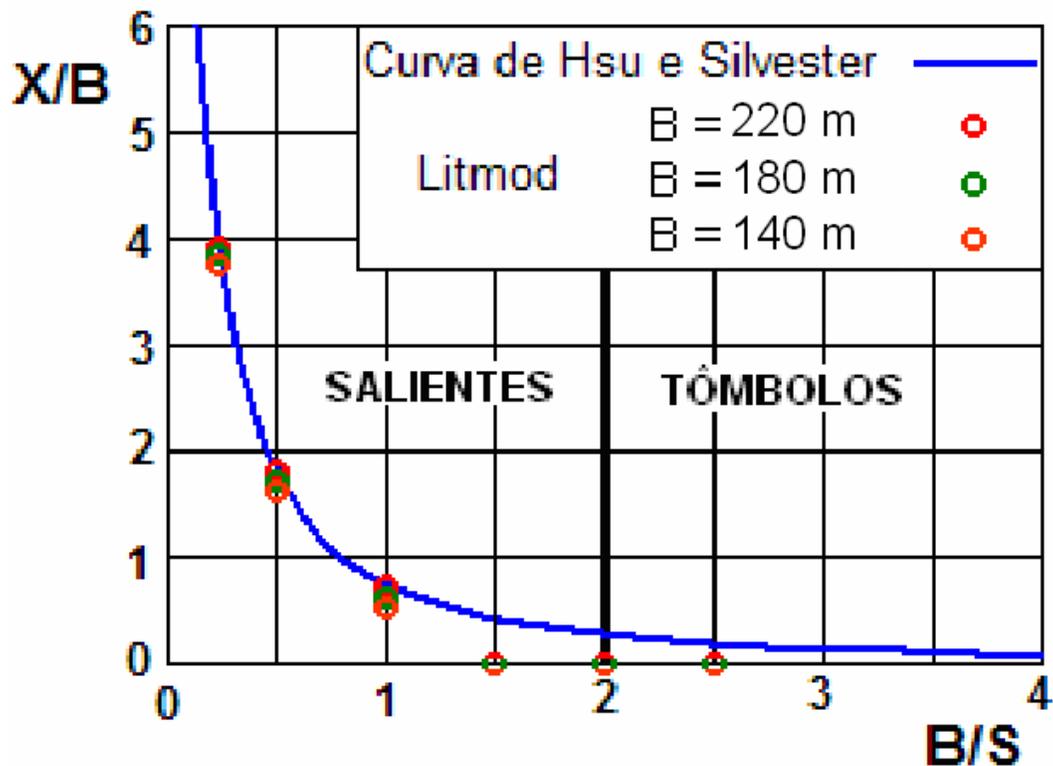


Fig.9 – Comparação de resultados do Litmod com a curva de correlação de Hsu e Silvester

Apresentam-se na Fig. 10 alguns exemplos de simulação de tómbolos assinalados no Quadro 2.

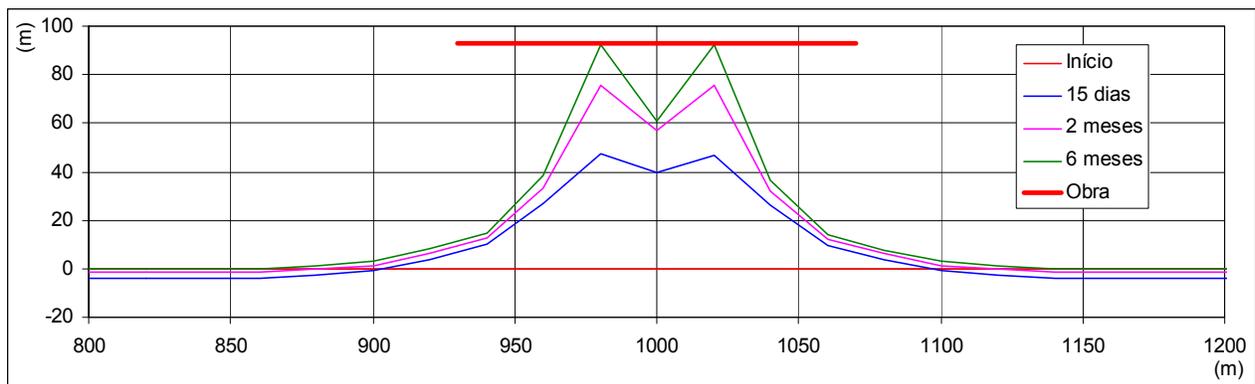
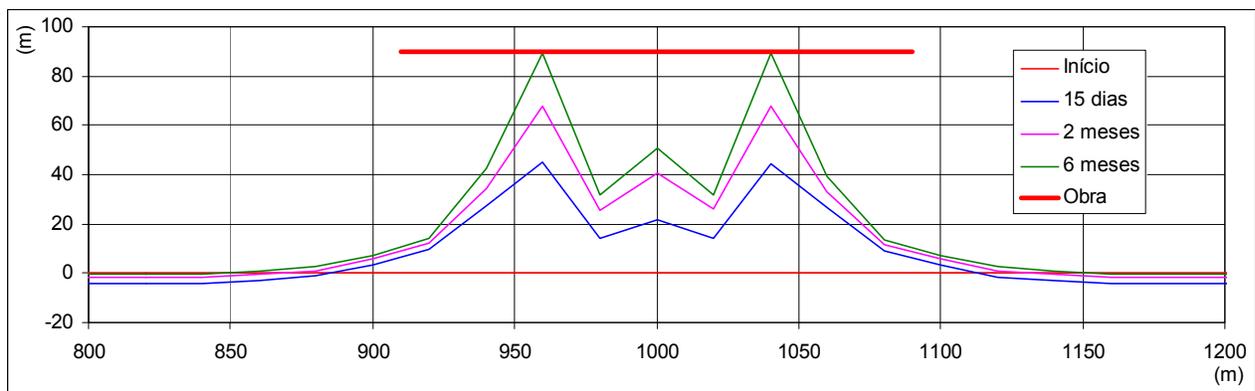
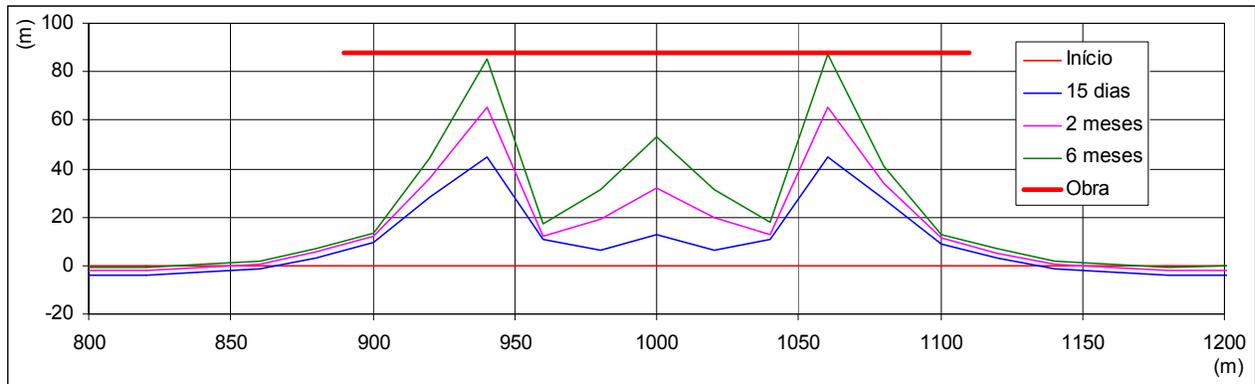


Fig. 10 – Exemplos de simulação de tómbolos

## 6.5 Simulação em condições naturais

Teria sido vantajoso aplicar a modelação a um quebra-mar destacado existente, para avaliar a acuidade da simulação. Não foi possível fazê-lo pois não existem obras deste tipo na costa portuguesa e embora frequentes noutros países não se dispõe aí dos dados necessários, especialmente os relativos à agitação marítima.

Optou-se por uma simulação de previsão do efeito de três quebra-mares destacados num trecho de costa natural sujeito a um regime de agitação real, tendo-se seleccionado a zona compreendida entre a Quarteira e a praia do Ancão.

Partindo da linha de água actual, aplicou-se o modelo com dados de agitação marítima de 5 anos, obtidos pela repetição da série cronológica do ano de Novembro de 1988 a Outubro de 1989, único período de observação de que se dispunha informação suficientemente completa (Fig. 11).

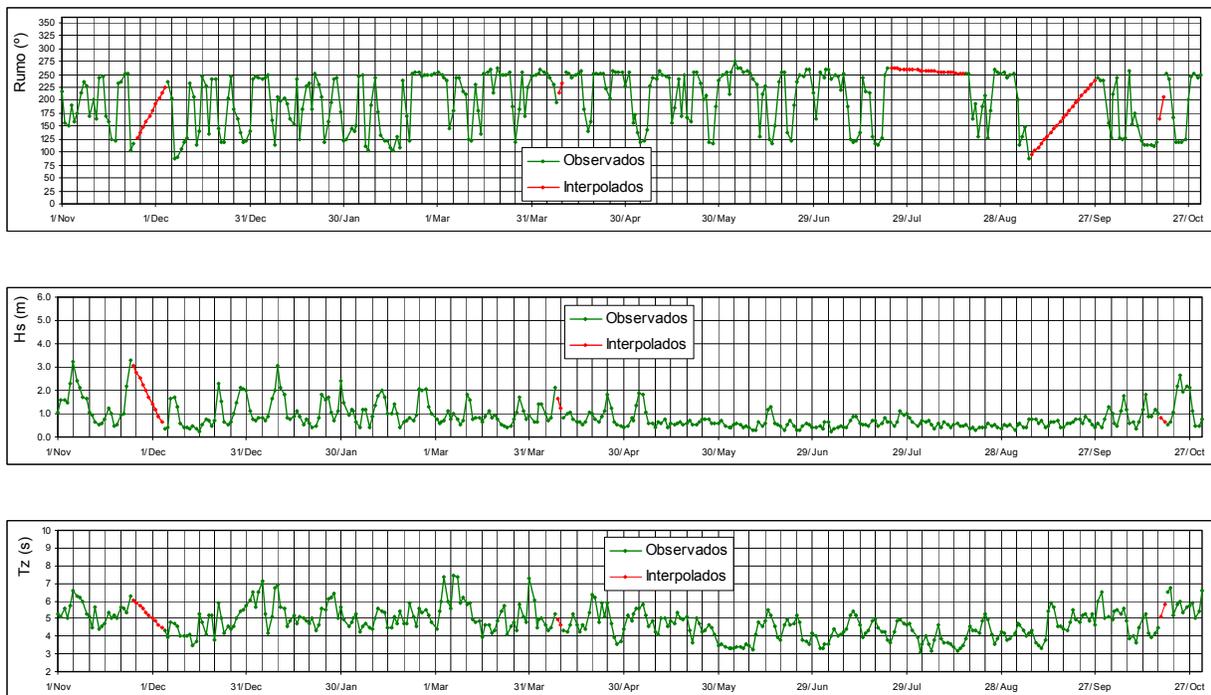


Fig. 11 – Série cronológica de dados de agitação marítima de Novembro de 1988 a Outubro de 1989.

Começou-se por efectuar a simulação sem quebra-mares, com vista a calibrar o modelo reproduzindo os valores do transporte sólido litoral e as tendências erosivas desse trecho de praia (Fig. 12).

Foi possível simular o moderado processo erosivo que resulta da saída anual do trecho de cerca de 70 000 m<sup>3</sup> de areias, para nascente, sem compensação por afluxos da fronteira poente.

Repetiu-se a simulação introduzindo 3 quebra-mares destacados com 300 m de comprimento, implantados a distâncias de 150, 200 e 300 m da costa. Verificou-se que os resultados da simulação estão em concordância com os critérios anteriormente referidos para a passagem de saliente a tómbolo.

Faz-se notar que o esquema de protecção simulado tem uma função meramente ilustrativa e não configura uma solução particularmente adequada para este trecho de costa.

## **7. CONSIDERAÇÕES FINAIS**

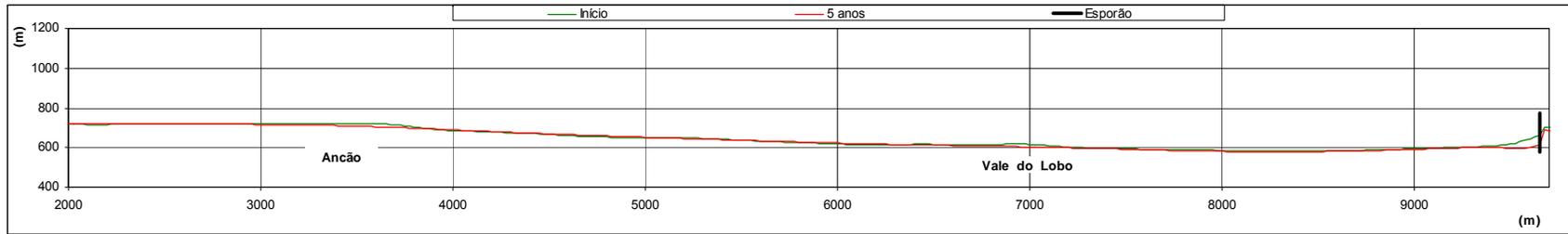
O Laboratório Nacional de Engenharia Civil tem vindo a desenvolver e a aplicar, desde o início da década de 90, um modelo de simulação da evolução de linhas de costa, designado por Litmod.

A crescente utilização internacional de quebra-mares destacados na estabilização de orlas costeiras levou à inclusão no modelo deste tipo de obras, com simulação da difracção e dos efeitos, sobre a intensidade do transporte de areias, da variação da altura das ondas ao longo da linha de rebentação.

Foram efectuadas numerosas simulações de desenvolvimento de salientes e formação de tómbolos, em que se fizeram variar os parâmetros característicos: da obra, comprimento e distância à linha de água; e das ondas, período, altura e obliquidade.

Verificou-se uma boa concordância entre os resultados obtidos e os provenientes de dados da natureza e de modelos físicos apresentados por diferentes autores.

### Simulação da situação existente



### Simulação com três quebra-mares destacados

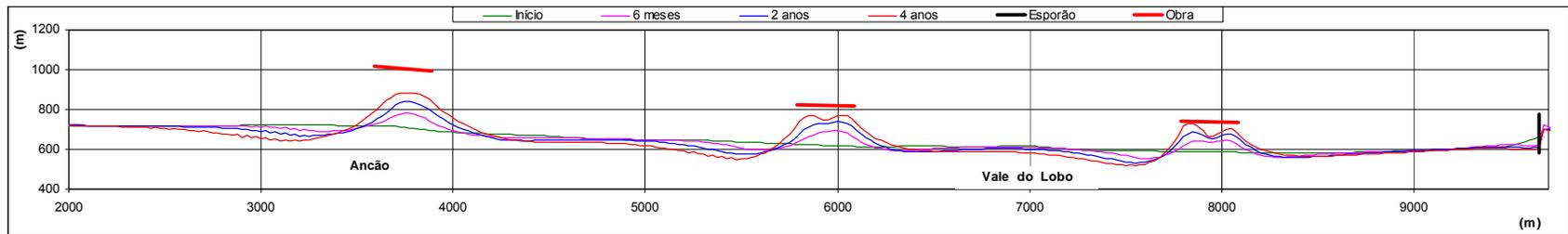


Fig. 12 – Simulação dos efeitos de três quebra-mares destacados

Lisboa, Laboratório Nacional de Engenharia Civil, Janeiro de 2009.

**VISTOS**

O Chefe do Núcleo de  
Estuários e Zonas Costeiras



André B. Fortunato

A Directora do Departamento de  
Hidráulica e Ambiente



Rafaela de Saldanha Matos

**AUTORIAS**



Claudino Martins Vicente  
Engº Civil, Inv. Coordenador



Manuel Clímaco  
Engº Civil, Inv. Principal



Vítor Pisco  
Técnico Especialista Principal

## BIBLIOGRAFIA

- Birkemeier, W. (1985) - Field Data on Seaward Limit of Profile Changes. *Journal of Waterways Ports, Coastal and Ocean Engineering*, Vol. II, nº 3, p. 598-602. ASCE.
- Clímaco, M.; Vicente, C.; Pisco, V. (2009) – Modelação Numérica de Praias Encaixadas. Rel. 15/09-NEC, Janeiro, LNEC, Lisboa.
- Clímaco, M.; Vicente, C. (2007) – Metodologia de Aplicação de um Modelo de Evolução de Linhas de Costa. Rel. 283/07-NEC, Setembro, LNEC, Lisboa.
- Coastal Engineering Research Center - CERC (1975) - Shore Protection Manual, 3<sup>th</sup> Edition Vol. 1 & 2, U.S. Army Engineer Waterways Experiment Station, U.S. Government Printing Office, Washington, DC.
- Coastal Engineering Research Center - CERC (1984) – Shore Protection Manual, 4<sup>th</sup> Edition Vol. 1 & 2, U.S. Army Engineer Waterways Experiment Station, U.S. Government Printing Office, Washington, DC.
- Dally, W.R., Pope, J. (1986). Detached Breakwaters for Shore Protection. Technical Report CERC-86-1, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.
- Dean, R. G. (1991)- Equilibrium Beach Profiles. Characteristics and Application. *Journal of Coastal Resarch*, Winter 1991, pp.53-84.
- Goda, Y.; Takayama, T.; Suzuki, Y. (1978) - Diffraction Diagrams for Directional Random Waves. 16<sup>th</sup> Conference on Coastal Engineering, Hamburgo.
- Gourlay, M.R. (1981). Beach Processes in the Vicinity of Offshore Breakwaters. *Proceedings, 5<sup>th</sup> Australian Conference on Coastal and Ocean Engineering*, pp 129-134.
- Hanson, H. (1987) – Genesis - A Generalized Shoreline Change Numerical Model for Engineerig Use. Lund University, Sweden.
- Horikawa, K. (1988) - Nearshore Dynamics and Coastal Processes - Theory, Measurements and Predictive Models. University of Tokyo Press.

- Hsu, J. C.; Silvester, R. (1990) – Accretion Behind Single Offshore Breakwater. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, Vol.116, N.º 3, May/June.
- Hsu, J. and Evans, C. (1989) – Parabolic Bay Shapes and Applications. *Proc. Institution of Civil Engineers, London, England*, vol. 87 (Part 2), pp. 556-570.
- Komar, P. D. (1988) – Environment Controls on Littoral Sand Transport. *21<sup>st</sup> Conference on Coastal Engineering*. Málaga, Spain.
- Komar, P. D. (1981) – *Handbook of Coastal Process and Erosion*.
- Komar, P. D. (1976) – *Beach Processes and Sedimentation*. Prentice Hall. Englewood Cliffs, New Jersey.
- Kraus, N.; Harikai, S.; and Kubota, S. (1981) – A Numerical Simulation of Shoreline Evolution at Oarai Beach. *28<sup>th</sup> Japanese Conference on Coastal Engineering*. Japanese Society of Civil Engineers. pp. 632-645.
- Kraus, N. and Harikai, S. (1983) – Numerical Model of the Shoreline Change at Oarai Beach. *Coastal Engineering Vol. 7*, February. Elsevier Publishing Company.
- Moreno, L. and Kraus, N. (1999) – Equilibrium Shape of Headland-Bay Beaches for Engineerin Design. *Proc. Coastal Sediments`99*, ASCE, pp. 860-875.
- Silvester, R. (1970) – Development of Crenulated Shaped Bays to Equilibrium. *J. Waterways and Harbour Division, ASCE 96 (WW2)*, pp. 275-287.
- Silvester, R. (1974) – *Coastal Engineering*. Elsevier Publishing Company.
- Silvester, R. and Hsu, J. C. (1993) – *Coatal Stabilization. Innovative Concepts*. Prentice Hall. Englewood Cliffs, pp. 199-271.
- Vicente, C. (1991) - *Aperfeiçoamento de Métodos de Modelação Matemática e Física Aplicáveis a Problemas de Dinâmica Costeira*. LNEC, Lisboa.
- Vicente, C.; Clímaco, M. (2003) - *Evolução de Linhas de Costa. Desenvolvimento e Aplicação de Um Modelo Numérico*. ICT-ITH 42. LNEC, Lisboa.
- Wiegel, R. (1964) – *Oceanographical Engineering*.

# **MODELAÇÃO NUMÉRICA DE QUEBRA-MARES DESTACADOS**

## **ANEXO**

### **LISTAGEM DA ROTINA DE CÁLCULO**



# ANEXO

## LISTAGEM DA ROTINA DE CÁLCULO

```
C*****C
C
C Programa LITMOD - Subrotina Quebra C
C Versão de Outubro 2008 C
C C
C Cálculo do efeito de difracção provocado por um quebra-mar no caudalc
C sólido litoral C
C-----C
C
C Parâmetros
C AA.....Ang. da onda na bat. de aproximação em cada célula
C HA.....Alt. da onda na bat. de aproximação em cada célula
C LA.....Comprimento da onda na bat. de aprox. em cada célula
C AR.....Ang. da onda na reb. em cada célula
C HB.....Alt. da onda na reb. em cada célula
C DB.....Prof. na rebentação em cada célula
C LB.....Comprimento de onda na reb. em cada célula
C AC.....Ang. da costa com a linha de base em cada célula
C ARC.....Ang. da reb. com a costa em cada célula
C T.....Período da onda (variavel auxiliar)
C Q.....Caudal em cada célula
C N.....N. de células
C X1.....Abcissa do meio da primeira célula
C DX.....Largura das células
C DT.....Passo de calculo
C KP.....(KP=(PM+NB)*DX/DT) (variavel auxiliar)
C
C Parâmetros (Variaveis em COMMON)
C NQM.....N. de quebra-mares
C AQM.....Vector dos coef. de cada um dos perfis nos quebra-mares
C EQM ...Vector dos expoentes de cada um dos perfis
C XQMI.....Abcissa do início de cada quebra-mar
C XQMC.....Abcissa do centro quebra-mar
C NCQMI....Vector dos n. de célula do início de cada um dos quebra-mares
C NCQMC....Vector dos n. de célula do centro de cada um dos quebra-mares
C NCQMF....Vector dos n. de célula do fim de cada um dos quebra-mares
C
C INC.....Inclinação media da praia
C D50.....Diametro D50 da areia da praia
C FCQ.....Factores de correcção de caudais (valores interpolados)
C
C Variaveis locais
C A.....Variável auxiliar
C ALFAA....Ang. na batimétrica de aproximação
C ALFAOB...Ang. na cabeça da obra
C ANGCEL...Ang. na cabeça da obra
C ANGOBR...Ang. da obra com a batimétrica de aproximação
C BETA....Ang. relativo às formulas de cálculo da difracção
C BETAF....Ang. relativo às formulas de cálculo da difracção
```

```

c   DA.....Profundidade na batimétrica de aproximação
c   DH.....Difrença de altura de onda entre I & I+1
c   DHDX.....Relação entre DH & DX
c   DOB.....Profundidade na cabeça da obra
c   E.....Variável auxiliar
c   HOB.....Alt. da onda na cabeça da obra
c   I.....Variável auxiliar
c   OPQK.....Tipo de formula de difracção (OPQK=1 > Hanson ; OPQK=2 >
Wiegel)
c   QD.....Caudal devida ao efeito de difracção
c   QQ.....Variável auxiliar
c   R.....Distância da cabeça da obra à célula
c   PI.....3,14159
c   RADII.....Variavel de transformação de graus em radianos
c   GRAU.....Variavel de transformação de radianos em graus
c   RL.....Relação entre o comp. De onda e a distância da cabeça da obra à
c   célula
c   TETA.....Ang. relativo às formulas de cálculo da difracção
c   XX.....Variável auxiliar
c   YOBR.....Variável auxiliar
c   YQF.....Ordenada do início do quebra-mar
c   YQI.....Ordenada do fim do quebra-mar
c   YY.....Variável auxiliar
c   KDF.....Coef. de transmissão lateral de energia
c   LOB.....Comprimento da onda na cabeça da obra
c   LOB2.....Comprimento da onda na cabeça da obra (variável auxiliar)
c   M.....Variável auxiliar
c   NN.....Variável auxiliar
c   NQF.....N. da célula do início do quebra-mar
c   NQI.....N. da célula do fim do quebra-mar
c   OP0.....Tipo de formula de caudal (OP0=1 > CERC ; OP0=2 > Kamphuis)
c   OP1.....OP1=1 > ponto fixo ; OP1=2 > evolução paralela
c   OP2.....OP2=1 > ponto fixo ; OP1=2 > evolução paralela
c   ICD.....Célula que indica o meio do quebra-mar (à direita)
c   ICE.....Célula que indica o meio do quebra-mar (à esquerda)
c   IDQ0.....Variável auxiliar
c   IEQ0.....Variável auxiliar
c   IID.....Variável auxiliar
c   ILD.....Limite da zona de influência da difracção (à direita)
c   ILE.....Limite da zona de influência da difracção (à esquerda)
c   LD.....Limite da zona de sombra (à direita)
c   LE.....Limite da zona desombra (à esquerda)
c   INCQ.....Inclinação da praia na zona da obra
c   INE.....Variável auxiliar
c   ITQ0.....Variável auxiliar
c   K.....Variável auxiliar
c   KD.....Coef. de difracção

```

```

c
c
c-----c

```

subroutine quebra

1 (op1,op2,x1,n,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,kp,nql)

```
c
```

implicit none

```
c
```

common/queb/opq,opqk,aqm,eqm,nqm,ncqmi,ncqmc,ncqmf,



```

        alfaa=aa(nqf+1)-ac(nqf+1)
c
        call h_pto(t,da,ha(nqf+1),alfaa,dob,hob,alfaob,lob)
c
c
        do i=nqi,ild
            xx=(nqf-i)*dx
            yy=yqf-(y(i)+y(i+1))/2
            if(abs(yqf-(y(i)+y(i+1))/2).gt.0.1) then
                angcel=atan(xx/yy)
            else
                angcel=pi/2
            end if
c
            teta=-angcel+alfaob
c
            if(teta.lt.30*radi) then
                call hanson(teta,kd)
c
                if(teta.lt.0.0) then
                    ar(i)=angcel-ac(i)
                    call h_reb
1                    (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
                else
                    call h_reb
1                    (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
                end if
            end if
        end do
cc
c OPQK=2
        else if(opqk.eq.2) then !Direita Weigel
c
            call lim_qdw
1            (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ild,ld)
c
c
            dob=a*(yqf-y(nqf+1))**(1/e)
c
            alfaa=aa(nqf+1)-ac(nqf+1)
c
            call h_pto(t,da,ha(nqf+1),alfaa,dob,hob,alfaob,lob)
c
            xx=(nqf-nqi)*dx
            yy=(yqf-yqi)
c
            angobr=atan(yy/xx)
c
            teta=(-pi/2+angobr)-alfaob
c
            teta=abs(teta)
            beta=0.0
            betaf=teta+30*radi
c
c
            do i=nqi,ild
                xx=(nqf-i)*dx

```

```

yy=yqf-(y(i)+y(i+1))/2
if(abs(yqf-(y(i)+y(i+1))/2).gt.0.1) then
  angcel=atan(xx/yy)
else
  angcel=pi/2
end if

c
beta=(pi/2+angobr)-angcel
c
if(beta.le.0.0) then
  q(i)=0
else if(beta.lt.betaf) then
  r=sqrt(xx**2+yy**2)
  rl=r/lob
  call kdw3(beta,rl,teta,kd)
c
  if(beta.lt.teta) then
    ar(i)=angcel-ac(i)
    call h_reb
1      (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
  else
    call h_reb
1      (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
  end if
end if
end do

c
end if !Hanson & Weigel

c
c
c Direita
c
do i=nqi,ild
  if(op0.eq.1) then !Formula do CERC
1    q(i)=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*
      sin(arc(i))*cos(arc(i))
  else if(op0.eq.2) then !Formula de Kamphuis
1    q(i)=fcq(i)*(64000.0/365.0)*(hb(i)**2)*((t*1.84)**1.5)
2    *(inc**0.75)*(d50**(-.25))
2    *abs(sin(2*(arc(i))))**0.6
    if(sin(2*arc(i)).lt.0) q(i)=-q(i)
  end if
end do

c
c
do i=nqi,ild
  dhdx=(hb(i+1)-hb(i))/dx
  qd=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*cos(arc(i))
  qd=kdf*dhdx*(1/incq)*abs(qd)
  qq=q(i)
  q(i)=q(i)+qd
end do

c
idq0=nqi
c
do i=nqi,nqf
  yobr=yqi+(yqf-yqi)/(nqf-nqi)*(i-nqi)

```

```

        if(y(i).ge.yobr*0.99) then
            idq0=i
        end if
    end do
c
c
        do i=nqi,idq0
            q(i)=0.0
        end do
c
        q(nqi)=0
c
c
        nql(2*k-1)=nqi-1
        nql(2*k)=idq0+1
c
c
        ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c OPQ=2
        else if(opq(k).eq.2) then !Quebra-mar ao Centro -
c
        yqi=yqmi(k)
        yqf=yqmf(k)
        nqi=ncqmi(k)
        nqf=ncqmf(k)
c
        if(opqk.eq.1) then !Hanson
c
            call lim_geh
1            (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ile,le)
c
            call lim_qdh
1            (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ild,ld)
c
c
            nn=(ld-le)/2
            ice=le+nn
            icd=ice+1
c
            ine=nqf
            iid=nqi
c
            do i=nqi,nqf
                if(y(i).ge.yqi*0.99) then
                    if(ine.gt.i) ine=i
                    if(iid.lt.i) iid=i
                    ice=ine
                    icd=iid+1
                end if
            end do
c
c Esquerda Hanson
c
        dob=a*(yqi-y(nqi-1))**(1/e)
c
        alfaa=aa(nqi-1)-ac(nqi-1)
c
        call h_pto(t,da,ha(nqi-1),alfaa,dob,hob,alfaob,lob)

```

```

c
do i=ile,ice
  xx=(nqi-i)*dx
  yy=yqi-(y(i)+y(i+1))/2
  if(abs(yqi-(y(i)+y(i+1))/2).gt.0.1) then
    angcel=atan(xx/yy)
  else
    angcel=-pi/2
  end if
c
  teta=angcel-alfaob
c
  if(teta.lt.30*radi) then
    call hanson(teta,kd)
c
    if(teta.lt.0.0) then
      ar(i)=angcel-ac(i)
      call h_reb
1      (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
    else
      call h_reb
1      (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
    end if
  end if
c  write(2,125)i,angcel*grau,lb(i),hb(i),arc(i)
c
  end do
c
c
c Direita Hanson
c
  dob=a*(yqf-y(nqf+1))**(1/e)
c
  alfaa=aa(nqf+1)-ac(nqf+1)
c
  call h_pto(t,da,ha(nqf+1),alfaa,dob,hob,alfaob,lob)
c
c
do i=icd,ild
  xx=(nqf-i)*dx
  yy=yqf-(y(i)+y(i+1))/2
  if(abs(yqf-(y(i)+y(i+1))/2).gt.0.1) then
    angcel=atan(xx/yy)
  else
    angcel=pi/2
  end if
c
  teta=-angcel+alfaob
c
  if(teta.lt.30*radi) then
    call hanson(teta,kd)
c
    if(teta.lt.0.0) then
      ar(i)=angcel-ac(i)
      call h_reb
1      (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
    else

```

```

                call h_reb
1                (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
                end if
                end if
c                write(2,125)i,angcel*grau,lb(i),hb(i),arc(i)
c
                end do
c
c
                else if(opqk.eq.2) then !Weigel Centro
c
                call lim_gew
1                (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ile,le)
c
                call lim_qdw
1                (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ild,ld)
c
                nn=(ld-le)/2
                ice=le+nn
                icd=ice+1
c
                ine=nqf
                iid=nqi
c
                do i=nqi,nqf
                    if(y(i).ge.yqi*0.99) then
                        if(ine.gt.i) ine=i
                        if(iid.lt.i) iid=i
                            ice=ine
                            icd=iid+1
                    end if
                end do
c
c Esquerda Weigel
c
                dob=a*(yqi-y(nqi-1))**(1/e)
c
                alfaa=aa(nqi-1)-ac(nqi-1)
c
                call h_pto(t,da,ha(nqi-1),alfaa,dob,hob,alfaob,lob)
c
                xx=(nqi-nqf)*dx
                yy=(yqi-yqf)
c
                angobr=atan(yy/xx)
c
                teta=(pi/2+angobr)-alfaob
c
c
                teta=abs(teta)
                beta=0.0
                betaf=teta+30*radi
c
                do i=ile,ice
                    xx=(nqi-i)*dx
                    yy=yqi-(y(i)+y(i+1))/2
                    if(abs(yqi-(y(i)+y(i+1))/2).gt.0.1) then

```

```

                angcel=atan(xx/yy)
            else
                angcel=pi/2
            end if
c
                beta=(pi/2+angobr)+angcel
c
c
                if(beta.le.0.0) then
                    q(i)=0
                else if(beta.lt.betaf) then
                    r=sqrt(xx**2+yy**2)
                    rl=r/lob
                    call kdw3(beta,rl,teta,kd)
c
                    if(beta.lt.teta) then
                        ar(i)=angcel-ac(i)
                        call h_reb
1                        (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
                    else
                        call h_reb
1                        (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
                    end if
                end if
c
            end do
c
c Direita Weigel
c
                dob=a*(yqf-y(nqf+1))**(1/e)
c
                alfaa=aa(nqf+1)-ac(nqf+1)
c
                call h_pto(t,da,ha(nqf+1),alfaa,dob,hob,alfaob,lob)
c
                xx=(nqf-nqi)*dx
                yy=(yqf-yqi)
c
                angobr=atan(yy/xx)
c
                teta=(-pi/2+angobr)-alfaob
c
                teta=abs(teta)
                beta=0.0
                betaf=teta+30*radi
c
                do i=icd,ild
                    xx=(nqf-i)*dx
                    yy=yqf-(y(i)+y(i+1))/2
                    if(abs(yqf-(y(i)+y(i+1))/2).gt.0.1) then
                        angcel=atan(xx/yy)
                    else
                        angcel=pi/2
                    end if
c
                beta=(pi/2+angobr)-angcel
c

```

```

        if(beta.le.0.0) then
            q(i)=0
        else if(beta.lt.betaf) then
            r=sqrt(xx**2+yy**2)
            rl=r/lob
            call kdw3(beta,rl,teta,kd)
c
            if(beta.lt.teta) then
                ar(i)=angcel-ac(i)
                call h_reb
1                (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
            else
                call h_reb
1                (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
            end if
        end if
c
    end do
c
    end if !Hanson & Weigel
c
c
do i=ile,ice
    if(op0.eq.1) then !Formula do CERC
        q(i)=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*
1        sin(arc(i))*cos(arc(i))
    else if(op0.eq.2) then !Formula de Kamphuis
        q(i)=fcq(i)*(64000.0/365.0)*(hb(i)**2)*((t*1.84)**1.5)
1        *(inc**0.75)*(d50**(-.25))
2        *abs(sin(2*(arc(i))))**0.6
        if(sin(2*arc(i)).lt.0) q(i)=-q(i)
    end if
end do
c
c
do i=icd,ild
    if(op0.eq.1) then !Formula do CERC
        q(i)=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*
1        sin(arc(i))*cos(arc(i))
    else if(op0.eq.2) then !Formula de Kamphuis
        q(i)=fcq(i)*(64000.0/365.0)*(hb(i)**2)*((t*1.84)**1.5)
1        *(inc**0.75)*(d50**(-.25))
2        *abs(sin(2*(arc(i))))**0.6
        if(sin(2*arc(i)).lt.0) q(i)=-q(i)
    end if
end do
c
c
do i=ile,ice
    dhdx=(hb(i)-hb(i-1))/dx
    qd=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*cos(arc(i))
    qd=kdf*dhdx*(1/incq)*abs(qd)
    qq=q(i)
c    write(2,125) i,q(i),qd
        q(i)=q(i)+qd
c    write(2,125) i,q(i)
    end do

```



```

c      alfaa=aa(nqi-1)-ac(nqi-1)
c
c      call h_pto(t,da,ha(nqi-1),alfaa,dob,hob,alfaob,lob)
c
c      do i=ile,nqf
c          xx=(nqi-i)*dx
c          yy=yqi-(y(i)+y(i+1))/2
c          if(abs(yqi-(y(i)+y(i+1))/2).gt.0.1) then
c              angcel=atan(xx/yy)
c          else
c              angcel=-pi/2
c          end if
c
c          teta=angcel-alfaob
c
c          if(teta.lt.30*radi) then
c              call hanson(teta,kd)
c
c              if(teta.lt.0.0) then
c                  ar(i)=angcel-ac(i)
c                  call h_reb
1              (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
c              else
c                  call h_reb
1              (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
c              end if
c          end if
c
c      end do
c
c
c      else if(opqk.eq.2) then !Weigel
c
c          call lim_qew
1          (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ile,le)
c
c          dob=a*(yqi-y(nqi-1))**(1/e)
c
c          alfaa=aa(nqi-1)-ac(nqi-1)
c
c          call h_pto(t,da,ha(nqi),alfaa,dob,hob,alfaob,lob)
c
c
c          xx=(nqi-nqf)*dx
c          yy=(yqi-yqf)
c
c          angobr=atan(yy/xx)
c
c          teta=(pi/2+angobr)+alfaob
c
c          teta=abs(teta)
c          beta=0.0
c          betaf=teta+30*radi
c
c          do i=ile,nqf
c              xx=(nqi-i)*dx

```

```

yy=yqi-(y(i)+y(i+1))/2
if(abs(yqi-(y(i)+y(i+1))/2).gt.0.1) then
  angcel=atan(xx/yy)
else
  angcel=-pi/2
end if
c
beta=(pi/2+angobr)+angcel
c
if(beta.le.0.0) then
  q(i)=0
else if(beta.lt.betaf) then
  r=sqrt(xx**2+yy**2)
  rl=r/lob
  call kdw3(beta,rl,teta,kd)
c
  if(beta.lt.teta) then
    ar(i)=angcel-ac(i)
    call h_reb
1 (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
  else
    call h_reb
1 (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
  end if
end if
c
end do
end if !Hanson & Weigel
c
c
c Esquerda
c
do i=ile,nqf
  if(op0.eq.1) then !Formula do CERC
1 q(i)=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*
  sin(arc(i))*cos(arc(i))
  else if(op0.eq.2) then !Formula de Kamphuis
1 q(i)=fcq(i)*(64000.0/365.0)*(hb(i)**2)*((t*1.84)**1.5)
2 *(inc**0.75)*(d50**(-.25))
  *abs(sin(2*(arc(i))))**0.6
  if(sin(2*arc(i)).lt.0) q(i)=-q(i)
  end if
end do
c
c
do i=ile,nqf
  dhdx=(hb(i)-hb(i-1))/dx
  qd=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*cos(arc(i))
  qd=kdf*dhdx*(1/incq)*abs(qd)
  qq=q(i)
  q(i)=q(i)+qd
end do
c
ieq0=nqf
itq0=0
do i=nqi,nqf
  yobr=yqi+(yqf-yqi)/(nqf-nqi)*(i-nqi)

```



```

        ar(i)=angcel-ac(i)
        call h_reb
1         (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
        else
        call h_reb
1         (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
        end if
    end if
c
    end do
c
c OPQK=2
else if(opqk.eq.2) then !Weigel
c
    call lim_qew
1     (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ile,le)
c
    dob=a*(yqi-y(nqi-1))**(1/e)
c
    alfaa=aa(nqi-1)-ac(nqi-1)
c
    call h_pto(t,da,ha(nqi),alfaa,dob,hob,alfaob,lob)
c
c
    xx=(nqi-nqf)*dx
    yy=(yqi-yqf)
c
    angobr=atan(yy/xx)
c
    teta=(pi/2+angobr)-alfaob
c
    teta=abs(teta)
    beta=0.0
    betaf=teta+30*radi
c
    do i=ile,nqf
        xx=(nqi-i)*dx
        yy=yqi-(y(i)+y(i+1))/2
        if(abs(yqi-(y(i)+y(i+1))/2).gt.0.1) then
            angcel=atan(xx/yy)
        else
            angcel=-pi/2
        end if
c
        beta=(pi/2+angobr)+angcel
c
        if(beta.le.0.0) then
            q(i)=0
        else if(beta.lt.betaf) then
            r=sqrt(xx**2+yy**2)
            rl=r/lob
            call kdw3(beta,rl,teta,kd)
c
            if(beta.lt.teta) then
                ar(i)=angcel-ac(i)
                call h_reb
1                (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))

```

```

                else
                    call h_reb
1                (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
                end if
            end if
        end do
c
        end if !Hanson & Weigel
c
c
c Esquerda
c
        do i=ile,nqf
            if(op0.eq.1) then !Formula do CERC
                q(i)=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*
1                sin(arc(i))*cos(arc(i))
            else if(op0.eq.2) then !Formula de Kamphuis
                q(i)=fcq(i)*(64000.0/365.0)*(hb(i)**2)*((t*1.84)**1.5)
1                *(inc**0.75)*(d50**(-.25))
2                *abs(sin(2*(arc(i))))**0.6
                if(sin(2*arc(i)).lt.0) q(i)=-q(i)
            end if
c
        end do
c
c
c
        do i=ile,nqf
            dhdx=(hb(i)-hb(i-1))/dx
            qd=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*cos(arc(i))
            qd=kdf*dhdx*(1/incq)*abs(qd)
            qq=q(i)
            q(i)=q(i)+qd
        end do
c
        ieq0=nqf
        itq0=0
        do i=nqi,nqf
            yobr=yqi+(yqf-yqi)/(nqf-nqi)*(i-nqi)
            if(y(i).ge.yobr*0.99) then
                if(ieq0.ge.i) ieq0=i
                itq0=1
            end if
        end do
c
c
        if(itq0.eq.1)then
            do i=ieq0,nqf
                q(i-1)=0.0
                q(i)=0.0
            end do
        end if
c
        q(nqf)=0
c
c
c Direita
        yqi=yqmc(k)

```

```

    yqf=yqmf(k)
    nqi=ncqmc(k)
    nqf=ncqmf(k)
c
    if(opqk.eq.1) then !Direita Hanson
c
        call lim_qdh
1         (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ild,ld)
c
        dob=a*(yqf-y(nqf+1))**(1/e)
c
        alfaa=aa(nqf+1)-ac(nqf+1)
c
        call h_pto(t,da,ha(nqf+1),alfaa,dob,hob,alfaob,lob)
c
c
        do i=nqi,ild
            xx=(nqf-i)*dx
            yy=yqf-(y(i)+y(i+1))/2
            if(abs(yqf-(y(i)+y(i+1))/2).gt.0.1) then
                angcel=atan(xx/yy)
            else
                angcel=pi/2
            end if
c
            teta=-angcel+alfaob
c
            if(teta.lt.30*radi) then
                call hanson(teta,kd)
c
                if(teta.lt.0.0) then
                    ar(i)=angcel-ac(i)
                    call h_reb
1                    (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
                else
                    call h_reb
1                    (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
                end if
            end if
c
        end do
cc
c OPQK=2
    else if(opqk.eq.2) then !Direita Weigel
c
        call lim_qdw
1         (k,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,ild,ld)
c
c
        dob=a*(yqf-y(nqf+1))**(1/e)
c
        alfaa=aa(nqf+1)-ac(nqf+1)
c
        call h_pto(t,da,ha(nqf+1),alfaa,dob,hob,alfaob,lob)
c
        xx=(nqf-nqi)*dx
        yy=(yqf-yqi)

```

```

c      angobr=atan(yy/xx)
c
c      teta=(-pi/2+angobr)-alfaob
c
c      teta=abs(teta)
c      beta=0.0
c      betaf=teta+30*radi
c
c
c      do i=nqi,ild
c          xx=(nqf-i)*dx
c          yy=yqf-(y(i)+y(i+1))/2
c          if(abs(yqf-(y(i)+y(i+1))/2).gt.0.1) then
c              angcel=atan(xx/yy)
c          else
c              angcel=pi/2
c          end if
c
c          beta=(pi/2+angobr)-angcel
c
c          if(beta.le.0.0) then
c              q(i)=0
c          else if(beta.lt.betaf) then
c              r=sqrt(xx**2+yy**2)
c              rl=r/lob
c              call kdw3(beta,rl,teta,kd)
c
c              if(beta.lt.teta) then
c                  ar(i)=angcel-ac(i)
c                  call h_reb
1              (t,hob*kd,dob,ar(i),lob2,lb(i),hb(i),db(i),arc(i))
c              else
c                  call h_reb
1              (t,ha(i)*kd,da,ar(i),la,lb(i),hb(i),db(i),arc(i))
c              end if
c          end if
c      end do
c
c      end if !Hanson & Weigel
c
c
c      do i=nqi,ild
c          if(op0.eq.1) then !Formula do CERC
1              q(i)=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*
c                  sin(arc(i))*cos(arc(i))
c          else if(op0.eq.2) then !Formula de Kamphuis
1              q(i)=fcq(i)*(64000.0/365.0)*(hb(i)**2)*((t*1.84)**1.5)
1              *(inc**0.75)*(d50**(-.25))
2              *abs(sin(2*(arc(i))))**0.6
c              if(sin(2*arc(i)).lt.0) q(i)=-q(i)
c          end if
c      end do
c
c
c      do i=nqi,ild
c          dhdx=(hb(i+1)-hb(i))/dx

```

```

        qd=fcq(i)*34.6*(1026*hb(i)**2*lb(i))/(8*t)*cos(arc(i))
        qd=kdf*dhdx*(1/incq)*abs(qd)
        qq=q(i)
        q(i)=q(i)+qd
    end do
c
        idq0=nqi
        do i=nqi,nqf
            yobr=yqi+(yqf-yqi)/(nqf-nqi)*(i-nqi)
            if(y(i).ge.yobr*0.99) then
                idq0=i
            end if
        end do
c
        do i=nqi,idq0
            q(i)=0.0
        end do
c
        q(nqi)=0
c
        nql(2*k-1)=ieq0-1
        nql(2*k)=idq0+1
c
c
        end if ! OPQ
c
        end do
c
        if(op1.eq.1) then
            q(0)=q(1)
        else if(op1.eq.2) then
            q(0)=2*q(1)-q(2)
        end if
c
        if(op2.eq.1) then
            q(n)=q(n-1)
        else if(op2.eq.2) then
            q(n)=2*q(n-1)-q(n-2)
        end if
c
        return
101  format(1x,'Esq',1x,i3,3x,f12.2,6(1x,f7.2))
102  format(1x,'Dir',1x,i3,3x,f12.2,6(1x,f7.2))
103  format(1x,'Cel',12x,'Hbi',9x,'Hbi+1',9x,'Qi',9x,'QDi')
104  format(1x,i3,2x,10(1x,f8.2))
105  format(1x,8(1x,10i5))
        end
c*****
c*****c
c
c  Programa LITMOD - Subrotina Lim_QEH ---ESQUERDA ---
c  Versao de Outubro de 2008
c
c  Quebra-mares
c  Difraccão à esquerda (Hanson)
c  Determinação das células limite das zonas afectadas pela difracção
c-----c

```

```

c
c   Parâmetros
c   AA.....Ang. da onda na bat. de aproximação em cada célula
c   HA.....Alt. da onda na bat. de aproximação em cada célula
c   LA.....Comprimento da onda na bat. de aprox. em cada célula
c   AR.....Ang. da onda na reb. em cada célula
c   HB.....Alt. da onda na reb. em cada célula
c   DB.....Prof. na rebentação em cada célula
c   LB.....Comprimento de onda na reb. em cada célula
c   AC.....Ang. da costa com a linha de base em cada célula
c   ARC.....Ang. da reb. com a costa em cada célula
c   T.....Período da onda (variável auxiliar)
c   Q.....Caudal em cada célula
c
c   N.....N. de células
c   X1.....Abcissa do meio da primeira célula
c   DX.....Largura das células
c   DT.....Passo de cálculo
c   KP.....(KP=(PM+NB)*DX/DT) (variável auxiliar)
c
c   Parâmetros (Variáveis em COMMON)
c   NQM.....N. de quebra-mares
c   AQM.....Vector dos coef. de cada um dos perfis nos quebra-mares
c   EQM   ...Vector dos expoentes de cada um dos perfis
c
c   XQMI.....Abcissa do início de cada quebra-mar
c   XQMC   Abcissa do centro quebra-mar
c   XQMF   Abcissa do fim de cada quebra-mar
c   YQMI   Ordenada do início de cada quebra-mar
c   YQMC   Ordenada do centro quebra-mar
c   YQMF   Ordenada do fim de cada quebra-mar
c
c   NCQMI....Vector dos n. de célula do início de cada um dos quebra-mares
c   NCQMC....Vector dos n. de célula do centro de cada um dos quebra-mares
c   NCQMF....Vector dos n. de célula do fim de cada um dos quebra-mares
c
c   INCQM   Inclinação da praia
c   KDIF   Coef. que traduz a transmissão lateral de energia
c   OPQK   Tipo de difração (Hanson ou Wiegel)
c   OPQ    Tipo de quebra-mar
c   Variáveis locais

```

```

c
c -----c
c

```

```

c   subroutine lim_geh
c   1      (nq,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,n3,n4)
c
c   implicit none
c
c   common/queb/opq,opqk,aqm,eqm,nqm,ncqmi,ncqmc,ncqmf,
c   1 yqmi,yqmc,yqmf,kdif,incqm
c
c   integer nqm,opqk,opq(5),ncqmi(5),ncqmc(5),ncqmf(5)
c
c   real aqm(5),eqm(5),xqmi(5),yqmi(5)
c   real xqmc(5),yqmc(5),xqmf(5),yqmf(5)

```

```

real kdif(5),incqm(5)
c
integer i,m,n,n1,n2,n3,n4,nq,op0
c
real ha(500),hb(500),lb(500),db(500)
real y(500),q(0:500),fcq(500),inc,d50
real aa(500),ac(500),ar(500),arc(500),t
real dx,xx,yy,angobr,angcel
real alfaa,alfae,de,he,la,le,le2
real teta,beta,beta2,betaf,r,r1,kd
real a,e,da,y1,y2
real kp(500),yaux,yobr,dy
real qd,qq,dhdx
real pi,grau,radi
c
pi=4*atan(1.)
radi=pi/180.0
grau=180.0/pi
c
a=aqm(nq)
e=eqm(nq)
if(opq(nq).eq.4) then
  y1=yqmi(nq)
  n1=ncqmi(nq)
  y2=yqmc(nq)
  n2=ncqmc(nq)
else
  y1=yqmi(nq)
  n1=ncqmi(nq)
  y2=yqmf(nq)
  n2=ncqmf(nq)
end if
c
write(2,*) 'e ',nq,n1,n2
c
if(y1-y(n1-1).lt.10) then
  return
end if
c
de=a*(y1-y(n1-1))**(1/e)
c
alfaa=aa(n1-1)-ac(n1-1)
c
call h_pto(t,da,ha(n1-1),alfaa,de,he,alfae,le)
c
xx=(n1-n2)*dx
yy=(y1-y2)
c
angobr=atan(yy/xx)
c
teta=-pi/2
c
c
m=0
2 n3=n2-m
  i=n3
  if(n3.gt.2)then

```

```

c      if(n3.gt.(n1-(n2-n1)))then
          if(teta.le.30*radi) then
              xx=(n1-n2+m)*dx
              yy=y1-(y(i)+y(i+1))/2
              if(yy.gt.0.0) then
                  angcel=atan(xx/yy)
              else
                  angcel=-pi/2
              end if
              teta=angcel-alfae
              m=m+1
              goto 2
          end if
      end if
      n3=n3+1

c
      teta=-pi/2
      m=0
3      n4=n2-m
      i=n4
      if(n4.gt.2)then
c      if(n4.gt.(n1-(n2-n1)))then
          if(teta.le.0.0) then
              xx=(n1-n2+m)*dx
              yy=y1-(y(i)+y(i+1))/2
              if(yy.gt.0.0) then
                  angcel=atan(xx/yy)
              else
                  angcel=-pi/2
              end if
              teta=angcel-alfae
c      write(2,*) i,angcel*grau
              m=m+1
              goto 3
          end if
      end if
      n4=n4+1
c      write(2,*)n1,n2,n3,n4
c
104      format(1x,i3,3x,8(1x,f11.4))
120      format(1x,i3,10f9.3)
130      format(1x,2i3,10f9.4)
c
      return
c
      end
c
c*****c
c*****c
c
c      Programa LITMOD - Subrotina Lim_QDH ---DIREITA---
c      Versao de outubro de 2008
c
c      Quebra-mares
c      Difracção à direita (Hanson)
c      Determinação das células limite das zonas afectadas pela difracção
c

```

```

c-----c
c
c  Parâmetros
c    AA.....Ang. da onda na bat. de aproximação em cada célula
c    HA.....Alt. da onda na bat. de aproximação em cada célula
c    LA.....Comprimento da onda na bat. de aprox. em cada célula
c    AR.....Ang. da onda na reb. em cada célula
c    HB.....Alt. da onda na reb. em cada célula
c    DB.....Prof. na rebentação em cada célula
c    LB.....Comprimento de onda na reb. em cada célula
c    AC.....Ang. da costa com a linha de base em cada célula
c    ARC.....Ang. da reb. com a costa em cada célula
c    T.....Periodo da onda (variavel auxiliar)
c    Q.....Caudal em cada célula
c
c    N.....N. de células
c    X1.....Abcissa do meio da primeira célula
c    DX.....Largura das células
c    DT.....Passo de calculo
c    KP.....(KP=(PM+NB)*DX/DT) (variavel auxiliar)
c
c  Parâmetros (Variaveis em COMMON)
c    NQM.....N. de quebra-mares
c    AQM.....Vector dos coef. de cada um dos perfis nos quebra-mares
c    EQM    ...Vector dos expoentes de cada um dos perfis
c
c    XQMI....Abcissa do início de cada quebra-mar
c    XQMC    Abcissa do centro quebra-mar
c    XQMF    Abcissa do fim de cada quebra-mar
c    YQMI    Ordenada do início de cada quebra-mar
c    YQMC    Ordenada do centro quebra-mar
c    YQMF    Ordenada do fim de cada quebra-mar
c
c    NCQMI...Vector dos n. de célula do início de cada um dos quebra-mares
c    NCQMC...Vector dos n. de célula do centro de cada um dos quebra-mares
c    NCQMF...Vector dos n. de célula do fim de cada um dos quebra-mares
c
c    INCQM   Inclinação da praia
c    KDIF    Coef. que traduz a transmissão lateral de energia
c    OPQK    Tipo de difração (Hanson ou Wiegel)
c    OPQ     Tipo de quebra-mar
c  Variaveis locais
c
c-----c
c
c    subroutine lim_qdh
c    1 (nq,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,n3,n4)
c
c    implicit none
c
c    common/queb/opq,opqk,aqm,eqm,nqm,ncqmi,ncqmc,ncqmf,
c    1 yqmi,yqmc,yqmf,kdif,incqm
c
c    integer nqm,opqk,opq(5),ncqmi(5),ncqmc(5),ncqmf(5)
c
c    real aqm(5),eqm(5),xqmi(5),yqmi(5)

```

```

real xqmc(5),yqmc(5),xqmf(5),yqmf(5)
real kdif(5),incqm(5)
c
integer i,m,n,n1,n2,n3,n4,nq,op0
c
real ha(500),hb(500),lb(500),db(500)
real y(500),q(0:500),fcq(500),inc,d50
real aa(500),ac(500),ar(500),arc(500),t
real dx,xx,yy,angobr,angcel
real alfaa,alfae,de,he,la,le,le2
real teta,beta,beta2,betaf,r,r1,kd
real a,e,da,y1,y2
real kp(500),yaux,yobr,dy
real qd,qq,dhdx
real pi,grau,radi
c
pi=4*atan(1.)
radi=pi/180.0
grau=180.0/pi
c
a=aqm(nq)
e=eqm(nq)
if(opq(nq).eq.4) then
  y1=yqmc(nq)
  n1=ncqmc(nq)
  y2=yqmf(nq)
  n2=ncqmf(nq)
else
  y1=yqmi(nq)
  n1=ncqmi(nq)
  y2=yqmf(nq)
  n2=ncqmf(nq)
end if
c
write(2,*) 'd ',nq,n1,n2
c
if(y2-y(n2+1).lt.10) then
  return
end if
c
de=a*(y2-y(n2+1))**(1/e)
c
alfaa=aa(n2+1)-ac(n2+1)
c
call h_pto(t,da,ha(n2+1),alfaa,de,he,alfae,le)
c
xx=(n2-n1)*dx
yy=(y2-y1)
c
angobr=atan(yy/xx)
c
teta=-pi/2
c
c
m=0
n3=n1+m
i=n3

```

```

        if(n3.lt.n-2) then
c      if(n3.lt.(n2+(n2-n1))) then
          if(teta.le.30*radi) then
            xx=(n2-n1-m)*dx
            yy=y2-(y(i)+y(i+1))/2
            if(yy.gt.0.0) then
              angcel=atan(xx/yy)
            else
              angcel=pi/2
            end if
            teta=-(angcel-alfae)
            m=m+1
            goto 2
          end if
        end if
        n3=n3-1
c
        teta=-pi/2
        m=0
3      n4=n1+m
        i=n4
c      if(n4.lt.n-2) then
        if(n4.lt.(n2+(n2-n1))) then
          if(teta.le.0.0) then
            xx=(n2-n1-m)*dx
            yy=y2-(y(i)+y(i+1))/2
            if(yy.gt.0.0) then
              angcel=atan(xx/yy)
            else
              angcel=pi/2
            end if
            teta=-(angcel-alfae)
            m=m+1
c          write(2,120) n4,angcel*grau,teta*grau,alfae*grau
            goto 3
          end if
        end if
        n4=n4-1
c
c
104      format(1x,i3,3x,8(1x,f11.4))
120      format(1x,i3,10f9.3)
130      format(1x,2i3,10f7.2)
c
        return
c
        end
c
c*****c
c*****c
c
c      Programa LITMOD - Subrotina Lim_QEW ---ESQUERDA ---      c
c      Versao de Outubro 2008      c
c
c      Quebra-mares      c
c      Difraccão à esquerda (Weigel)      c
c      Determinação das células limite das zonas afectadas pela difracção c

```

```

c
c-----c
c
c   Parâmetros
c
c   Variaveis locais
c
c-----c
c
c   subroutine lim_gew
1      (nq,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,n3,n4)
c
c   implicit none
c
c   common/queb/opq,opqk,aqm,eqm,nqm,ncqmi,ncqmc,ncqmf,
1 yqmi,yqmc,yqmf,kdif,incqm
c   common/caudal/op0,inc,d50,fcq
c
c   integer nqm,opqk,opq(5),ncqmi(5),ncqmc(5),ncqmf(5)
c
c   real aqm(5),eqm(5),xqmi(5),yqmi(5)
c   real xqmc(5),yqmc(5),xqmf(5),yqmf(5)
c   real kdif(5),incqm(5)
c
c   integer i,m,n,n1,n2,n3,n4,nq,op0
c
c   real ha(500),hb(500),lb(500),db(500)
c   real y(500),q(0:500),fcq(500),inc,d50
c   real aa(500),ac(500),ar(500),arc(500),t
c   real dx,xx,yy,angobr,angcel
c   real alfaa,alfae,de,he,la,le,le2
c   real teta,beta,betaf,r,r1,kd
c   real a,e,da,y1,y2
c   real kp(500),yaux,yobr,dy
c   real qd,qq,dhdx
c   real pi,grau,radi
c
c   pi=4*atan(1.)
c   radi=pi/180.0
c   grau=180.0/pi
c
c
c   a=aqm(nq)
c   e=eqm(nq)
c   if(opq(nq).eq.4) then
c       y1=yqmc(nq)
c       n1=ncqmc(nq)
c       y2=yqmf(nq)
c       n2=ncqmf(nq)
c   else
c       y1=yqmi(nq)
c       n1=ncqmi(nq)
c       y2=yqmf(nq)
c       n2=ncqmf(nq)
c   end if
c

```

```

        if(y1-y(n1-1).lt.25) return
c
        de=a*(y1-y(n1-1))**(1/e)
c
        alfaa=aa(n1-1)-ac(n1-1)
c
        call h_pto(t,da,ha(n1),alfaa,de,he,alfae,le)
c
        xx=(n1-n2)*dx
        yy=(y1-y2)
c
        angobr=atan(yy/xx)
c
        teta=(pi/2+angobr)+alfae
c
        write(2,*) 'Esq',n1,teta*grau,angobr*grau,alfae*grau
c
        teta=abs(teta)
        beta=0.0
        betaf=teta+30*radi
c
        m=0
2       n3=n2-m
        i=n3
        if(n3.gt.1)then
c       if(n3.gt.(n1-(n2-n1)))then
            if(beta.le.betaf) then
                xx=(n1-n2+m)*dx
                yy=y1-(y(i)+y(i+1))/2
                if(abs(y1-(y(i)+y(i+1))/2).gt.0.1) then
                    angcel=atan(xx/yy)
                else
                    angcel=pi/2
                end if
                beta=(pi/2+angobr)+angcel
                m=m+1
                goto 2
            end if
c       end if
        end if
        n3=n3+1
c
        beta=0.0
        m=0
3       n4=n2-m
        i=n4
        if(n4.gt.1)then
c       if(n4.gt.(n1-(n2-n1)))then
            if(beta.le.teta) then
                xx=(n1-n2+m)*dx
                yy=y1-(y(i)+y(i+1))/2
                if(abs(y1-(y(i)+y(i+1))/2).gt.0.1) then
                    angcel=atan(xx/yy)
                else
                    angcel=pi/2
                end if
                beta=(pi/2+angobr)+angcel
            end if
        end if
    
```

```

        m=m+1
        goto 3
    end if
c   end if
    end if
    n4=n4+1
c   write(2,*) n4
c
c
c
104   format(1x,i3,3x,8(1x,f11.4))
120   format(1x,i3,10f9.2)
130   format(1x,2i3,10f9.4)
c
    return
c
    end
c
c*****c
c*****c
c
c   Programa LITMOD - Subrotina Lim_QDW ---DIREITA---
c   Versao de      2008
c
c   Quebra-mares
c   Difracção à direita (Weigel)
c   Determinação das células limite das zonas afectadas pela difracção
c
c-----c
c
c   Parâmetros
c
c   Variaveis locais
c
c-----c
c
    subroutine lim_qdw
1   (nq,da,ha,aa,ac,ar,arc,hb,lb,t,y,q,dx,n,kp,n3,n4)
c
    implicit none
c
    common/queb/opq,opqk,aqm,eqm,nqm,ncqmi,ncqmc,ncqmf,
1   yqmi,yqmc,yqmf,kdif,incqm
    common/caudal/op0,inc,d50,fcq
c
    integer nqm,opqk,opq(5),ncqmi(5),ncqmc(5),ncqmf(5)
c
    real aqm(5),eqm(5),xqmi(5),yqmi(5)
    real xqmc(5),yqmc(5),xqmf(5),yqmf(5)
    real kdif(5),incqm(5)
c
    integer i,m,n,n1,n2,n3,n4,nq,op0
c
    real ha(500),hb(500),lb(500),db(500)
    real y(500),q(0:500),fcq(500),inc,d50
    real aa(500),ac(500),ar(500),arc(500),t

```

```

real dx,xx,yy,angobr,angcel
real alfaa,alfae,de,he,la,le,le2
real teta,beta,betaf,r,r1,kd
real a,e,da,y1,y2
real kp(500),yaux,yobr,dy
real qd,qq,dhdx
real pi,grau,radi
c
pi=4*atan(1.)
radi=pi/180.0
grau=180.0/pi
c
a=aqm(nq)
e=eqm(nq)
if(opq(nq).eq.4) then
  y1=yqmc(nq)
  n1=ncqmc(nq)
  y2=yqmf(nq)
  n2=ncqmf(nq)
else
  y1=yqmi(nq)
  n1=ncqmi(nq)
  y2=yqmf(nq)
  n2=ncqmf(nq)
end if
c
if(y2-y(n2).lt.25) return
c
de=a*(y2-y(n2+1))**(1/e)
c
alfaa=aa(n2+1)-ac(n2+1)
c
call h_pto(t,da,ha(n2+1),alfaa,de,he,alfae,le)
c
xx=(n2-n1)*dx
yy=(y2-y1)
c
angobr=atan(yy/xx)
c
teta=(-pi/2+angobr)-alfae
c
write(2,*) 'Dir',teta*grau,angobr*grau,alfae*grau
c
teta=abs(teta)
beta=0.0
betaf=teta+30*radi
c
m=0
2 n3=n1+m
i=n3
if(n3.lt.n-2) then
c if(n3.lt.(n2+(n2-n1))) then
  if(beta.le.betaf) then
    xx=(n2-n1-m)*dx
    yy=y2-(y(i)+y(i+1))/2
    if(abs(y2-(y(i)+y(i+1)))/2).gt.0.1) then
      angcel=atan(xx/yy)

```

```

        else
            angcel=pi/2
        end if
        beta=(pi/2+angobr)-angcel
        m=m+1
        goto 2
    end if
c    end if
end if
n3=n3-1
c
beta=0.0
m=0
3    n4=n1+m
    i=n4
    if(n4.lt.n-2) then
c    if(n4.lt.(n2+(n2-n1))) then
        if(beta.le.teta) then
            xx=(n2-n1-m)*dx
            yy=y2-(y(i)+y(i+1))/2
            if(abs(y2-(y(i)+y(i+1))/2).gt.0.1) then
                angcel=atan(xx/yy)
            else
                angcel=pi/2
            end if
            beta=(pi/2+angobr)-angcel
            m=m+1
            goto 3
        end if
c    end if
end if
n4=n4-1
c    write(2,*) n4
c
c
c
104    format(1x,i3,3x,8(1x,f11.4))
120    format(1x,i3,10f9.2)
130    format(1x,2i3,10f7.2)
c
    return
c
end
c
c*****c

```

