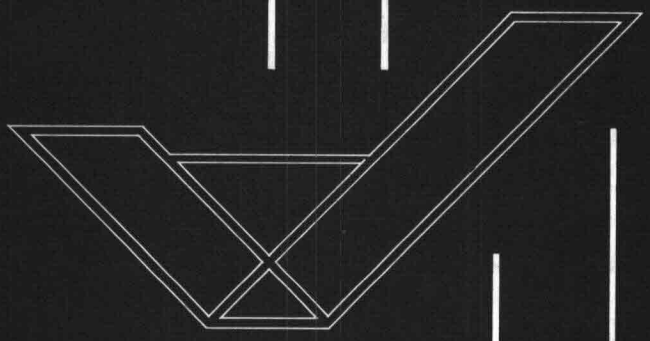


[redacted]

11-84

Delft University of Technology
Department of Civil Engineering
Group of Fluid Mechanics

Report No. 11-84

Implementation (II) of the numerical shallow water wave hindcast model HISWA

T.H.C. Herbers
N. Booij
L.H. Holthuijsen

PROJECT REPORT	**Delft University of Technology** **Department of Civil Engineering** **Group of Fluid Mechanics**
Project title	GEOMOR wave model (HISWA)
Project description	Development of a two-dimensional model to hindcast spectral wave parameters in an estuary with tidal flats on the basis of bottomtopography, current and wind data.
Customer	Rijkswaterstaat Deltadienst, Afdeling Kustonderzoek van Alkemadelaan 400 2597 AT THE HAGUE, the Netherlands
represented by	J. v. Marle
Project leader	dr.ir. L.H. Holthuijsen
work carried out by	ir. T.H.C. Herbers dr.ir. N. Booij dr.ir. L.H. Holthuijsen
Conclusion	Diffraction-like propagation has been implemented and tested. Tests indicate that this type of propagation cannot be used for most applications. Wave growth and decay has been implemented and tested. The results are satisfactory. The input program has been finalized. The output program has been implemented and is being tested.
Status of report	Confidential, progress report
City/date:	Delft, Sept. 12, 1984

TABLE OF CONTENTS

1. INTRODUCTION

HISWA is a numerical model, developed to hindcast waves in shallow water. The progress in the implementation, documentation and testing of this model since the last progress report (Booij et al., 1984) is described in this report. The mathematical formulation of HISWA is only included in this report in so far as significant modifications have been implemented. For a complete description reference is made to Holthuijsen and Booij (1983, chapter 5) and Booij et al., (1984, chapter 4).

The status of the computer programs is described in chapter 2. In chapter 3 a test is presented to verify the applicability of the simple approximation of diffraction effects in HISWA. Tests for the various source terms are presented in chapter 4. The conclusions of this report are given in chapter 5.

2. STATUS OF THE PROGRAMS

The model HISWA consists of 3 computer programs PREP, COMPU and OUTP of which the status will be specified below. The latest version of the user's manual of HISWA is included in appendix I. A complete overview of subroutines is given in appendix II.

PREP is a program that reads the commands given by the user and prepares instructions for the programs COMPU and OUTP. It is a modified version of the program PREP of the refraction/diffraction model CREDIZ of Rijkswaterstaat. PREP has been implemented and tested. Presently no system documentation of this modified program is available. We plan to make a document with the modifications in this program which together with the system documentation of CREDIZ will form a provisional documentation of PREP.

COMPU forms the computational body of the model HISWA. In this program wave conditions are determined on a rectangular grid through an explicit scheme applied to two balance equations. This program has been implemented and tested. The latest version of the system documentation of COMPU is included in appendix III.

The original plan to develop two output programs (OUT1 and OUT2, see Booij et al., 1984) has not been carried out because the development of OUT2 has progressed more rapidly than was expected. OUTP processes the results of COMPU in print, plot and tape output. It is a new program in which various parts of the program UITV of CREDIZ are incorporated. OUTP has been implemented and partly tested. A preliminary version of the system documentation of this program is included in appendix IV.

3. A TEST FOR DIFFRACTION

In the two balance equations implemented in HISWA (eq. 1 and 2) diffusion terms are included to obtain diffraction-like effects. The purpose of these terms is not to produce an accurate representation of diffraction but merely to spatially smooth wave conditions in areas with strong gradients in wave height. The equations are:

$$\frac{\partial}{\partial x}(c_{xo}A_o) + \frac{\partial}{\partial y}(c_{yo}A_o) + \frac{\partial}{\partial \theta}(c_{\theta o}A_o + \alpha\{c_{yo}\frac{\partial A_o}{\partial x} - c_{xo}\frac{\partial A_o}{\partial y}\}) = \frac{S_o}{\sigma_o} - \frac{A_o}{\omega_o}\frac{d\omega_o}{dt} \qquad (1)$$

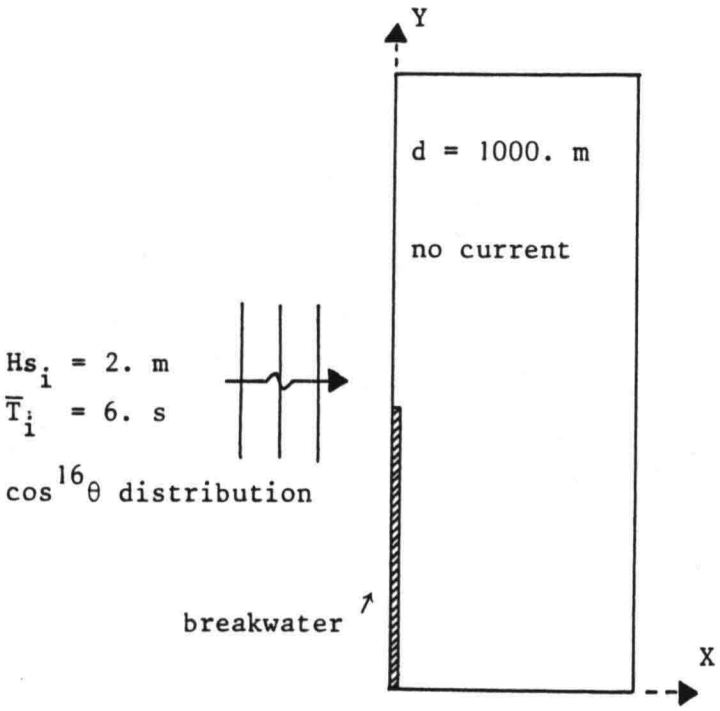
$$\frac{\partial}{\partial x}(c_{xo}\omega_o A_o) + \frac{\partial}{\partial y}(c_{yo}\omega_o A_o) + \frac{\partial}{\partial \theta}(c_{\theta o}\omega_o A_o + \alpha\{c_{yo}\frac{\partial(\omega_o A_o)}{\partial x} - c_{xo}\frac{\partial(\omega_o A_o)}{\partial y}\}) = \frac{\omega_o S_o}{\sigma_o} \qquad (2)$$

with

A_o frequency integrated wave action
ω_o mean action frequency
σ_o relative frequency
S_o source term
c_{xo}, c_{yo}, $c_{\theta o}$ wave action transport velocity components
α diffusion coefficient

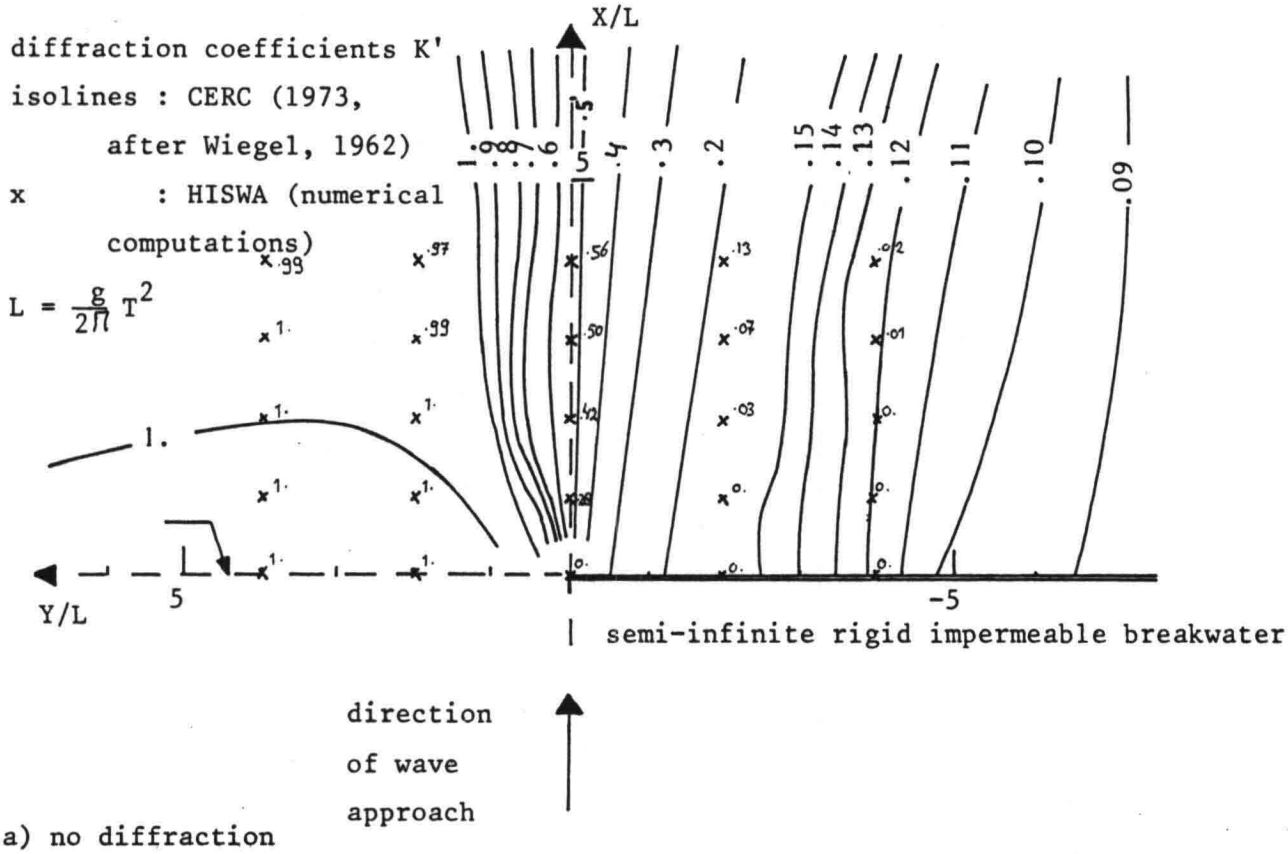
The terms between brackets and preceded by the coefficient α represent the diffraction-like behaviour in the model.

To investigate the applicability of this formulation and the value of the coefficient α, the following test is carried out. In an area with a constant depth (large compared with the wavelength) waves diffract around the tip of a semi-infinite breakwater. In this test the mean wave direction is perpendicular to the breakwater and the wave field is almost uni-directional (narrow directional energy distribution). The situation and information on the numerical grid and incident waves are shown in fig. 1. The results of this test, for three values of the coefficient α(α = 0, 0.01, 0.02) are presented in fig. 2. In fig. 2 the value of the diffraction coefficient K' (defined as the ratio of the local significant wave height over the incident significant wave height) is



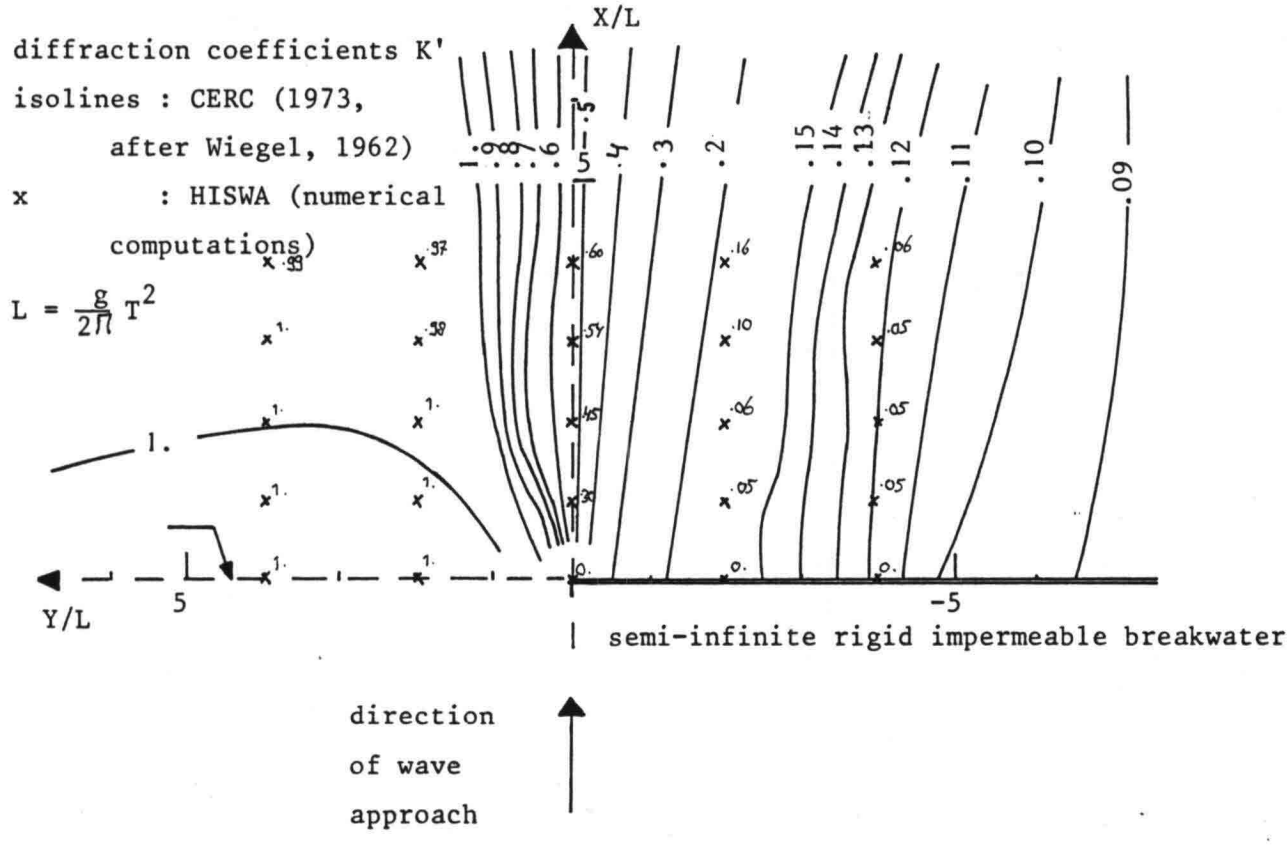
area 270 x 674 m
grid 25 x 25 x 21 points
minimum direction of propagation -60°
maximum direction of propagation +60°
mesh size 11.3 m x 28.1 m x 6°

Fig. 1 Situation of diffraction test.

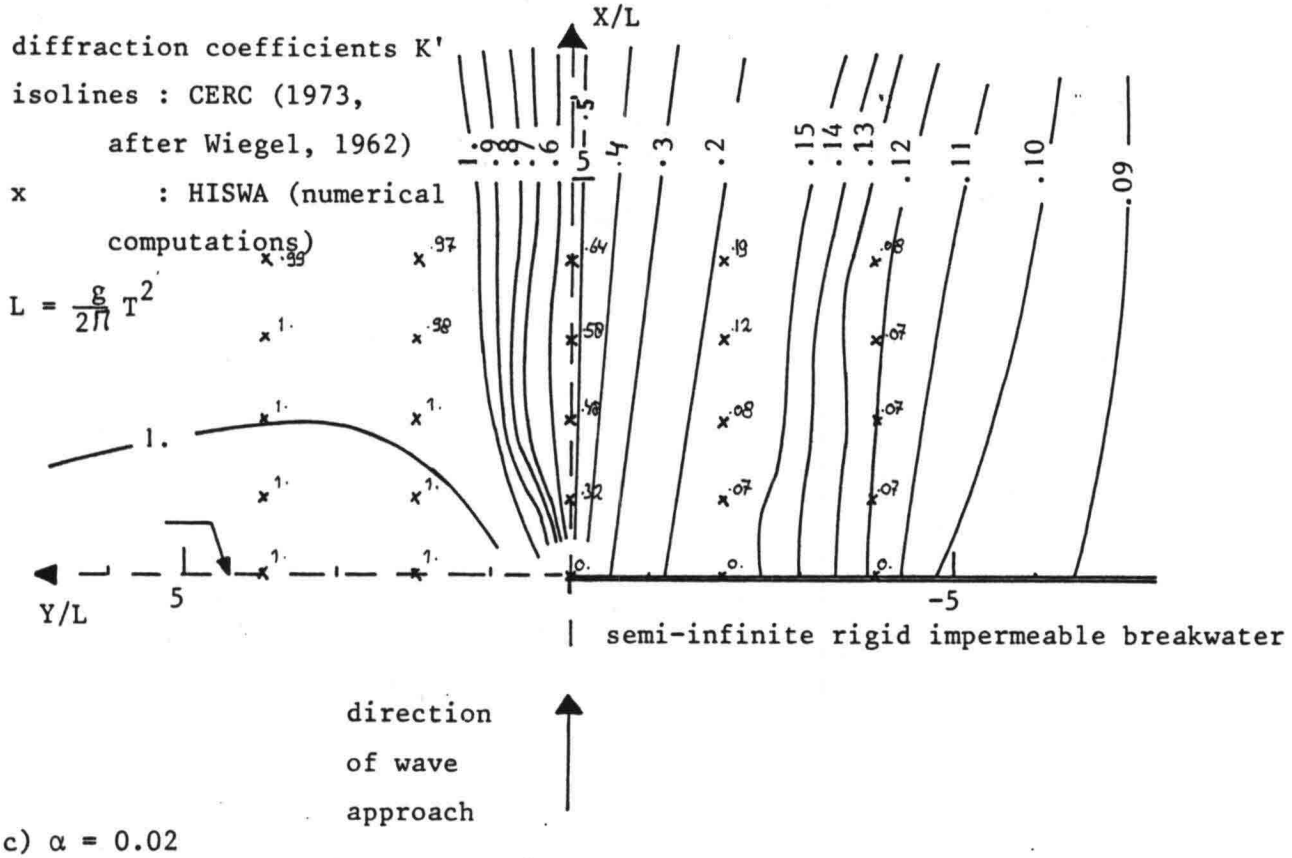


a) no diffraction

Fig. 2 Results of diffraction tests.



b) $\alpha = 0.01$.



c) $\alpha = 0.02$

Fig. 2 (continued).

plotted at various locations behind the breakwater.

For the case of an infinitely thin, semi-infinite, rigid breakwater Wiegel (1962, quoted from Wiegel, 1964) presented diffraction coefficients obtained through numerical computations based on the Sommerfeld solution of the diffraction of light of one single wave component. In the Shore Protection Manual (CERC, 1973) these results are presented in the form of diagrams which show isolines of equal wave height reduction. These isolines (for incident wave direction perpendicular to the breakwater) are included in fig. 2.

Fig. 2 shows that increasing the value of α yields more wave energy in the lee of the breakwater but the diffraction coefficients for $\alpha = 0.02$ (fig. 2c) are still considerably lower than the theoretical values of Wiegel. Unfortunately for higher values of α numerical instabilities appeared. In this test the mesh size in the computational grid was chosen a fraction (1/5 and 1/2) of the wave length. Since HISWA is designed for computations with considerably larger mesh sizes, an even lower value of will have to be chosen for most applications of HISWA to avoid numerical instabilities. This implies that the smoothing of the wave field by the diffraction-like behaviour of HISWA is much less than by diffraction proper. In fact, considering the required low value of α to avoid the numerical instabilities one may well decide to remove the diffusion terms from the HISWA model (unless a very small mesh size is used). The effect of such a removal need not be dramatic for the results as the wave field in HISWA will be smooth anyway (compared with a monochromatic model) due to the inherent distribution of wave energy over the directions in HISWA.

4. TESTS FOR SOURCE TERMS

4.1 Wind induced wave growth

The wind induced growth of the directional energy density has been implemented as formulated in Holthuijsen and Booij (1983):

$$\left(\frac{dE_o}{dt}\right)_{wind} = \frac{U_{10}^3}{g} B\, abcd \left(\frac{\tilde{E}_o/B}{a}\right)^{\frac{d-1}{d}} \left\{1 - \left(\frac{\tilde{E}_o/B}{a}\right)^{\frac{2}{d}}\right\} \left\{\frac{1}{b}\, \text{atanh}\left[\left(\frac{\tilde{E}_o/B}{a}\right)^{\frac{1}{d}}\right]\right\}^{\frac{c-1}{c}} \qquad (3)$$

with

$E_o = E_o g^2/U_{10}^4$, U_{10} is the wind speed relative to the mean current

B = the directional distribution: $E_o(\theta) = B(\theta)E$.

Values for the coefficients a, b, c and d are determined from the SWAMP study (1981, fig. 7.6):

$a = 3.6 \cdot 10^{-3}$

$b = 0.21 \cdot 10^{-21}$

$c = 4.667$

$d = 0.3$

The test results indicate a satisfactory performance of this formulation.

The formulation of the wind induced evolution of the wave frequency in Holthuijsen and Booij (1983, eq. 62) did not give satisfactory results. A different formulation is therefore presented here.

This formulation is based on an assumed universal relationship between the dimensionless energy $\tilde{E}$ and the dimensionless peak frequency $\tilde{f}$. From the SWAMP study (1981, figs. 7.6 and 7.7) we found (roughly):

$$\tilde{f} = a_2 \tilde{E}^{b_2} \qquad (4)$$

with

$a_2 = 2.91 \times 10^{-2}$

$b_2 = -0.283$

This relationship agrees well with the growth curves of the Shore Protection Manual (see fig. 3). From equation (4) the rate of change of the dimensionless frequency is readily obtained:

$$\left(\frac{d\tilde{f}}{d\tilde{t}}\right)_{wind} = a_2^{1/b_2} b_2 \tilde{f}^{(b_2-1)/b_2} \left(\frac{d\tilde{E}}{d\tilde{t}}\right)_{wind} \qquad (5)$$

Fig. 3 A universal relation between wave energy and -frequency.

This expression is the basic expression for the wind induced evolution of the peak frequency in HISWA. However, HISWA accepts wave boundary conditions which are given by the user. These wave conditions need not be in agreement with the assumed universal relationship (4). The use of the evolution equation (5) is then not appropriate. To remedy this situation, which may also arise due to the effect of the other source terms in HISWA, we have chosen to let the waves develop towards the universal relationship (4) whenever the wave situation deviates from this relationship. The rationale for this development is based on the effects of nonlinear wave-wave interaction (see Günther, 1981). The manner in which this development takes place is modelled with a wave dependent coefficient for the rate of change of the dimensionless frequency f:

$$\left(\frac{d\tilde{f}}{dt}\right)_{\text{wind}}^{\text{modified}} = \left(\frac{d\tilde{f}}{dt}\right)_{\text{wind}}^{\text{original}} \times C \qquad (6)$$

in which

$$C = \left(\frac{\tilde{f}}{a_2 \tilde{E}^{b_2}}\right)^m \qquad (7)$$

The effect of this term is tested in a one-dimensional model in which eq. 6 and the same wave growth formula used in HISWA (Holthuijsen and Booij, 1983, eq. 52) were implemented.

The computed behaviour of the development of E and f is illustrated in fig. 4 for m = 5 and m = 10. This can be compared with the results of Günther (1981) who obtained his results with a five parameter deep water parametric model (his fig. 14 transformed to $\tilde{E}$-$\tilde{f}$ domain). The agreement in general behaviour and in time scale seems fair considering the absence of any empirical information. The value of m is arbitrarily chosen to be 5 in HISWA.

The formula for wind induced evolution of directional wave frequency ω_o implemented in HISWA is consequently:

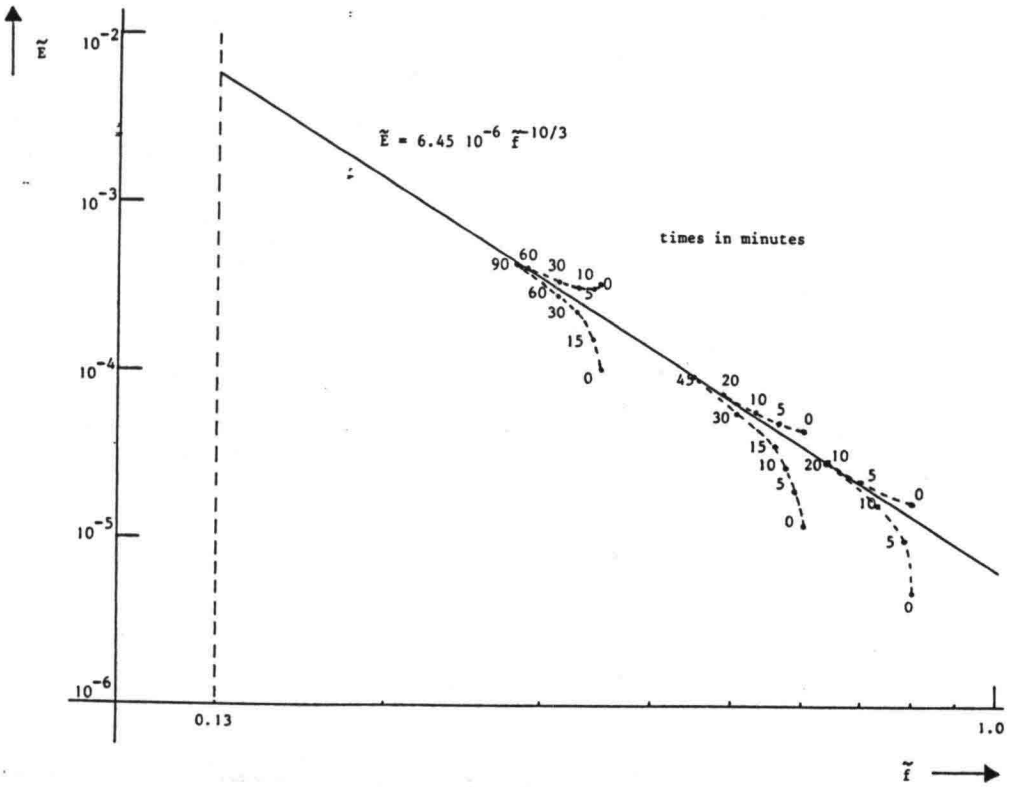
$$\left(\frac{d\omega_o}{dt}\right)_{\text{wind}} = \frac{g^3}{U_{10}^5}\, 2\pi\, a_2^{1/b_2}\, b_2 \left(\frac{\tilde{\omega}_o}{2\pi}\right)^{\frac{b_2-1}{b_2}} \frac{1}{B} \left(\frac{dE_o}{dt}\right)_{\text{wind}} \left[\frac{\tilde{\omega}_o}{2\pi a_2 (\tilde{E}_o/B)^{b_2}}\right]^5 \qquad (8)$$

with

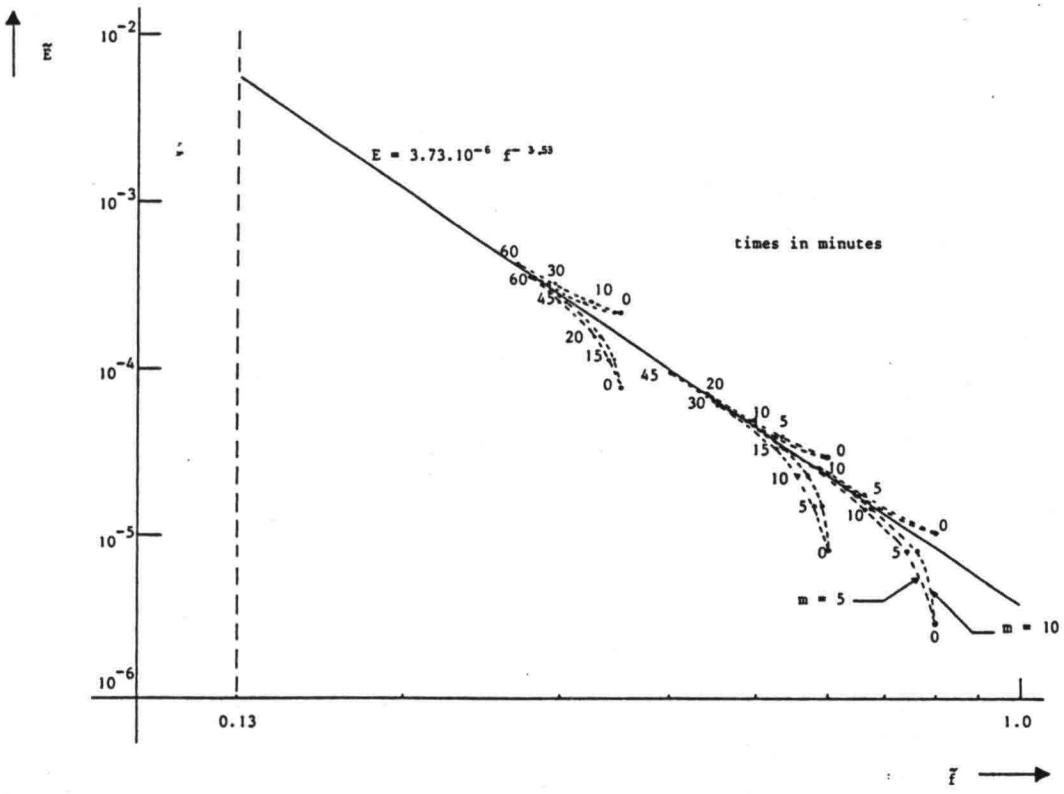
$$\tilde{\omega}_o = \omega_o U_{10}/g$$

For the coefficient a_2 a larger value was chosen (0.04) which proved to give a better fit with empirical data reviewed by Holthuijsen (see below).

In fig. 5 growth curves of wave height and -period in deep water, computed with HISWA, are plotted together in one graph with a few relations from literature



a) five-parameter model Günther (1981).



b) HISWA.

Fig. 4 Relaxation of wave energy and -frequency to equilibrium.

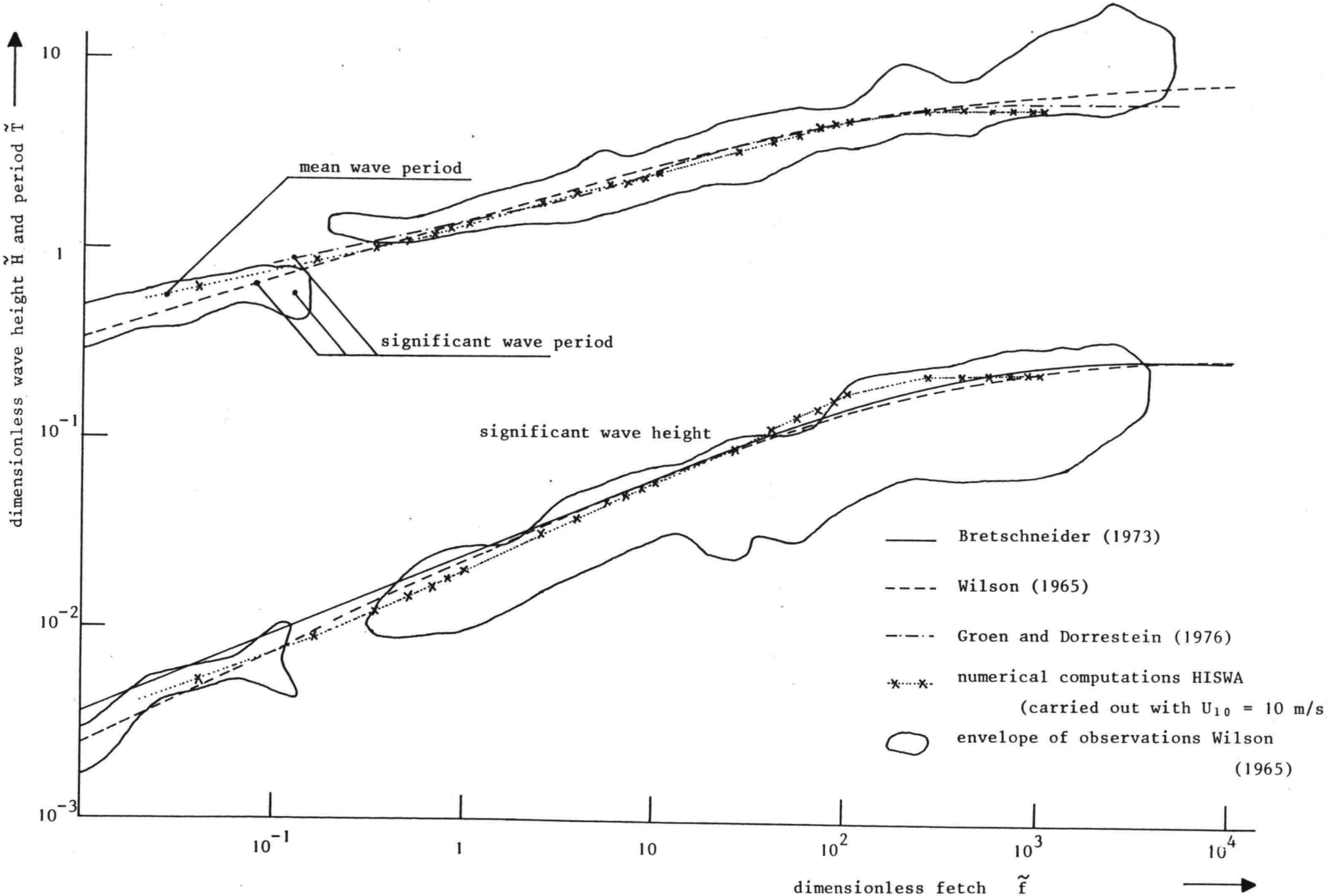


Fig. 5 Wave growth in deep water

and empirical data from Wilson (1965, quoted from Holthuijsen, 1980). The wave periods computed by HISWA are mean wave periods unlike the included results from literature which deal with significant wave periods. Therefore, the computed periods should be somewhat smaller (about 15%) than the significant periods.

The overall agreement is satisfactory although for small fetches ($\tilde{F} = F\, g/U_{10}^2 = 0(10 - 10^3)$) the predicted wave heights are slightly below the growth curves from literature (which are all in the upper part of the envelope of observations).

4.2 Bottom dissipation

The formulation of bottom dissipation, implemented in HISWA, deviates slightly from the one used by Hasselmann and Collins (1968) and described in Holthuijsen and Booij (1983). The reason for this is the fact that this formulation contains a bottom friction coefficient which is not dimensionless. In addition its value and relation to wave and bottom conditions is not very well known. However, the basic assumptions underlying the formulation of Hasselmann and Collins (1968) are maintained. This basis is the quadratic friction law:

$$\underline{\tau} = c_f\, \rho\, \underline{u}|\underline{u}| \tag{9}$$

in which τ is the shear stress vector at the bottom, ρ is the density of water and $\underline{u}$ is the velocity vector at the bottom. The rate of dissipation of the total energy D_f is then given by:

$$D_f = \underline{\tau}\cdot\underline{u} \tag{10}$$

or

$$D_f = c_f\, \rho\, |\underline{u}|^3 \tag{11}$$

Substituting the orbital motion at the bottom for a uni-directional monochromatic wave component with height H and frequency ω from the linear theory of gravity waves gives:

$$D_f = \frac{1}{6\pi}\, c_f \rho\, \frac{\omega^3}{\sinh^3 kd}\, H^3 \tag{12}$$

in which k is the wave number and d is the water depth.

A commonly used value for the dimensionless coefficient c_f is 0.01. This

expression can be extended to a uni-directional random wave field (Dingemans, 1983) by considering the joint distribution of H and T: $p(H,T)$.

$$D_f = \frac{4}{3}\ \pi^2 \rho c_f \int_0^\infty \int_0^\infty \left(\frac{H}{T \sinh kd}\right)^3 dp(H,T) \qquad (13)$$

A realistic function $p(H,T)$ has a complicated form. To simplify the analyzing $p(H,T)$ is restricted to one wave period and a Rayleigh distribution of wave heights. This approach is rather crude but it is deemed justified regarding the uncertainties in the validity of eq. 13 and the value of c_f. The result is:

$$D_f = \frac{1}{8\sqrt{\pi}}\ \rho\ c_f\ \frac{\omega^3}{\sinh^3 kd}\ H_{rms}^3 \qquad (14)$$

A comparison of eq. 12 and eq. 14 shows that the dissipation rates of a regular wave train and a random wave field are approximately equal for $H \simeq 1.1\ H_{rms} \simeq 0.78\ H_s$.

Expressed in terms of wave energy density E eq. 14 becomes:

$$\left(\frac{dE}{dt}\right)_{\text{bottom dissipation}} = -2\sqrt{\frac{2}{\pi}}\ c_f\ \frac{\omega^2}{g \sinh^2 kd}\ \langle u \rangle\ E \qquad (15)$$

in which $\langle u \rangle$ is the orbital velocity at the bottom (linear wave theory).

$$\langle u \rangle = \frac{\omega}{\sinh kd}\ E^{1/2} \qquad (16)$$

In order to apply eq. 15 to a directionally decoupled model the dissipation rate is assumed to be distributed proportional to the wave energy density in a direction:

$$\frac{dE_o}{dt} = B\ \frac{dE}{dt},\ E_o = BE \qquad (17)$$

in which B is the directional distribution of wave energy.

Accordingly the dissipation rate $\frac{dE_o}{dt}$ is expressed by:

$$\left(\frac{dE_o}{dt}\right)_{\text{bottom dissipation}} = -2\sqrt{\frac{2}{\pi}}\ c_f\ \frac{\omega_o^2}{\sinh^2 k_o d}\ \langle u_o \rangle E_o \qquad (18)$$

The orbital velocity at the bottom $\langle u_o \rangle$ is approximated by (Collins, 1972):

$$\langle u_o \rangle = \left\{ \int_0^{2\pi} \frac{\omega_o^2}{\sinh^2 k_o d}\ E_o\, d\theta \right\}^{\frac{1}{2}} \qquad (19)$$

The effect of a mean current on bottom dissipation is included in the same way as the effects of orbital velocities. The equivalent set of equations (frequency ω_o is replaced by relative frequency σ_o) is:

$$\left(\frac{dE_o}{dt}\right)_{\text{bottom dissipation}} = -2\sqrt{\frac{2}{\pi}}\ \frac{\sigma_o^2}{g\sinh^2 k_o d}\ \{c_{fw}\langle u_o\rangle + c_{fc}\langle u_c\rangle\} \qquad (20)$$

$$\langle u_o\rangle = \left\{\int_o^{2\pi} \frac{\sigma_o^2}{\sinh^2 k_o d}\, E_o\, d\theta\right\}^{\frac{1}{2}} \qquad (21)$$

$$\langle u_c\rangle = |u_x \cos\theta + u_y \sin\theta| \qquad (22)$$

The value of the coefficient c_{fc} has to be determined empirically (in HISWA values of c_{fw} and c_{fc} of 0.01 respectively 0.005 are assumed).
Further the influence of bottom friction on ω_o is expressed by:

$$\left(\frac{d\omega_o}{dt}\right)_{\text{bottom dissipation}} = \omega_o^2\ a_3\ \{-g^{-2}\omega_o^3\ \left(\frac{dE_o}{dt}\right)\}^{b_3}_{\text{bottom dissipation}} \qquad (23)$$

Since there is practically no information on the coefficients a_3 and b_3 this term is assumed to be 0.0 in the tests presented in this report.

The decay of waves due to bottom friction was tested in shallow water (constant depth) in absence of currents (fig. 6a). The computed decrease of H_s plotted versus distance x is given in fig. 6b. A comparison with the analytical formulation can be obtained as follows. For a monochromatic uni-directional wave train in water of constant depth the ratio $-\frac{dH}{dx}/H^2$ is constant:

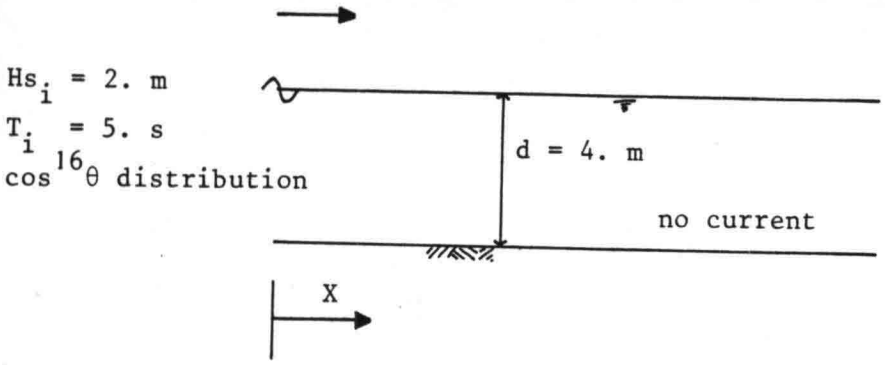
$$D_f = -\frac{d}{dx}(c_g \rho g E) = -\frac{1}{4} c_g g H \frac{dH}{dx} \qquad (24)$$

Substitution of eq. 12 in eq. 24 yields:

$$-\frac{dH}{dx} = \beta H^2,\ \beta = \frac{2}{3\pi} c_f \frac{k^2}{n\cosh kd \sinh^2 kd},\ n = \frac{c_g}{c} \qquad (25)$$

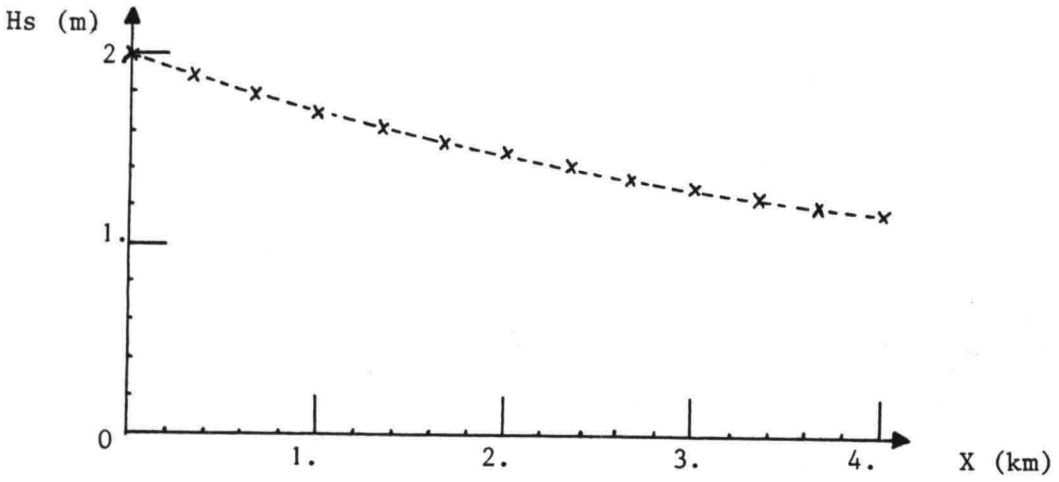
In fig. 6c this analytical expression (the parameters c_g, c and k are evaluated from linear wave theory) is given together with numerical results of HISWA. The agreement is good.

Empirical information on bottom dissipation is indirectly available in observations of wave growth in shallow water. Characteristic for these observations

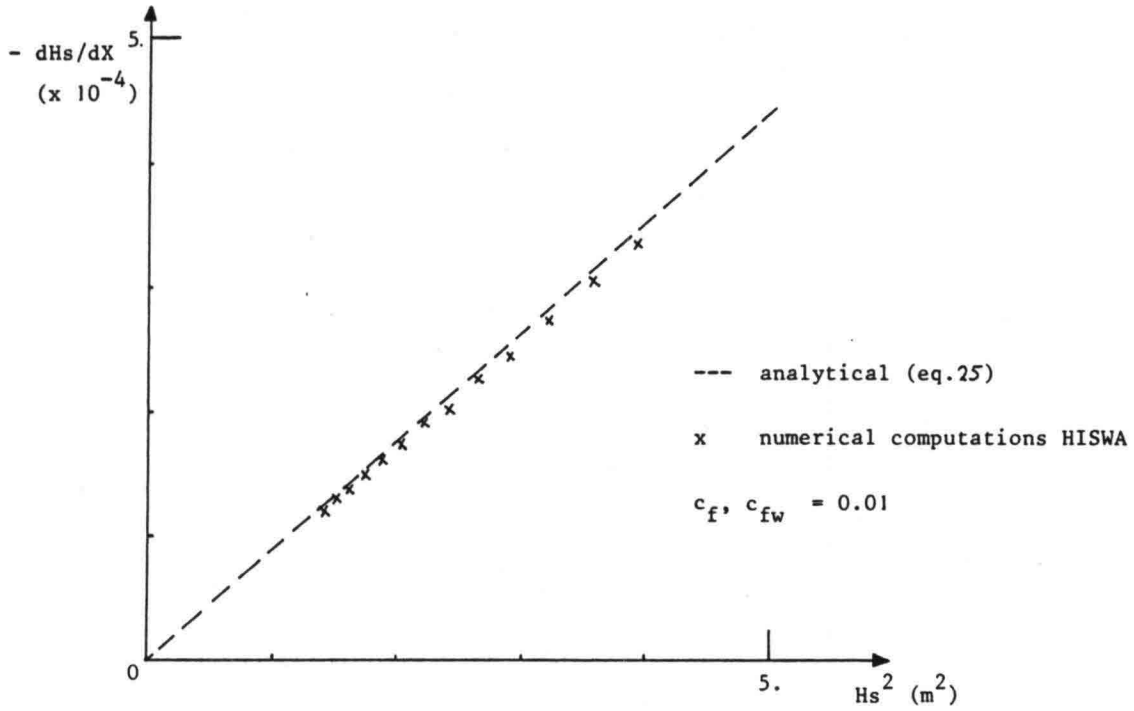


area 4000 x 8000 m
grid 50 x 25 x 17 points
minimum direction of propagation - 45°
maximum direction of propagation + 45°
mesh size 83.3 m x 333.3 m x 5.6°

a) situation of test.



b) variation of significant wave height.



c) a comparison of HISWA results with the analytical solution (eq. 25).

Fig. 6 Bottom dissipation in water of constant depth.

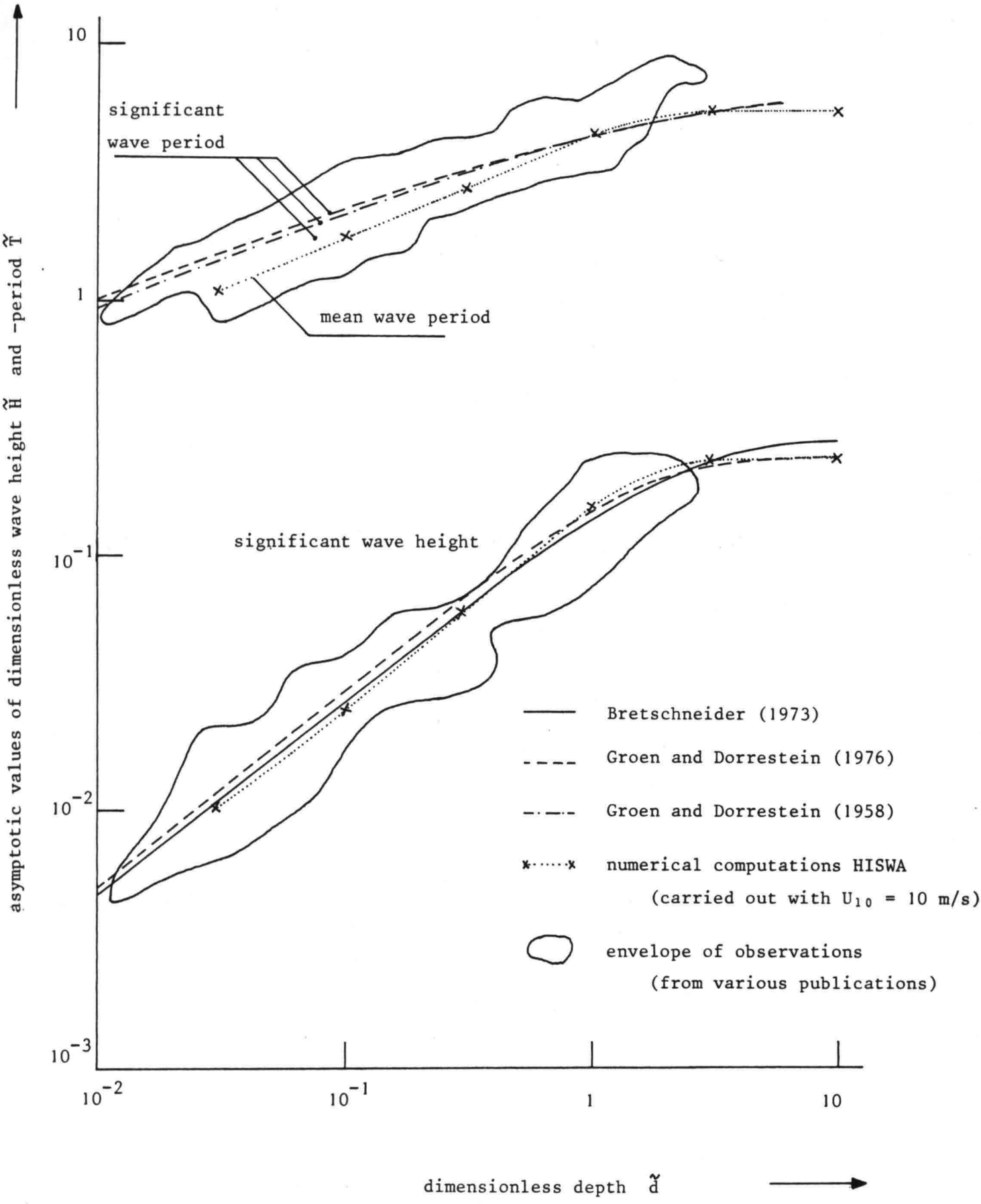


Fig. 7 Wave growth in shallow water.

is the value of the total wave energy (or the significant wave height H_s) and of the significant period in a stationary situation for a limitless water basin and a homogeneous stationary wind field. Fig. 7 gives the HISWA results in dimensionless form (determined with wind speed U_{10} = 10 m/s) and the review results of Holthuijsen (1980). The agreement is good but the computed (mean) wave periods are not asymptotic values since periods will continue to grow when growth and dissipation of energy balance each other. The indicated results are based on the value of the wave period at the moment when the energy attained its maximum value. A refinement in the model to stop the development of the period in these conditions should be considered.

4.3 Surf dissipation

The source terms for energy dissipation due to wave breaking in the surf zone and in strong adverse currents implemented in HISWA (based on Battjes and Janssen, 1978) are:

$$\left(\frac{dE_o}{dt}\right)_{\text{surf dissipation}} = -\alpha_1 \frac{1}{8\pi} Q_b \omega_o H_m^2 \frac{E_o}{E} \qquad (26)$$

$$H_m = 0.88\, k_o^{-1} \tanh(\gamma\, k_o d/0.88) \qquad (27)$$

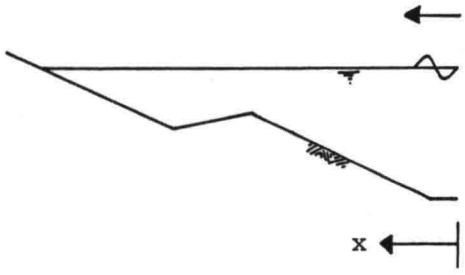
$$\frac{1-Q_b}{\ln Q_b} = -8 \frac{E}{H_m^2} \qquad (28)$$

$$\left(\frac{d\omega_o}{dt}\right)_{\text{surf dissipation}} = \omega_o^2\, a_4 \{-g^{-2}\, \omega_o^3 \left(\frac{dE_o}{dt}\right)\}^{b_4}_{\text{surf dissipation}} \qquad (29)$$

A test was carried out with waves perpendicular to a beach having a bar-through profile (fig. 8a). Wave conditions and dimensions of the beach profile were taken from Battjes and Janssen (1978).

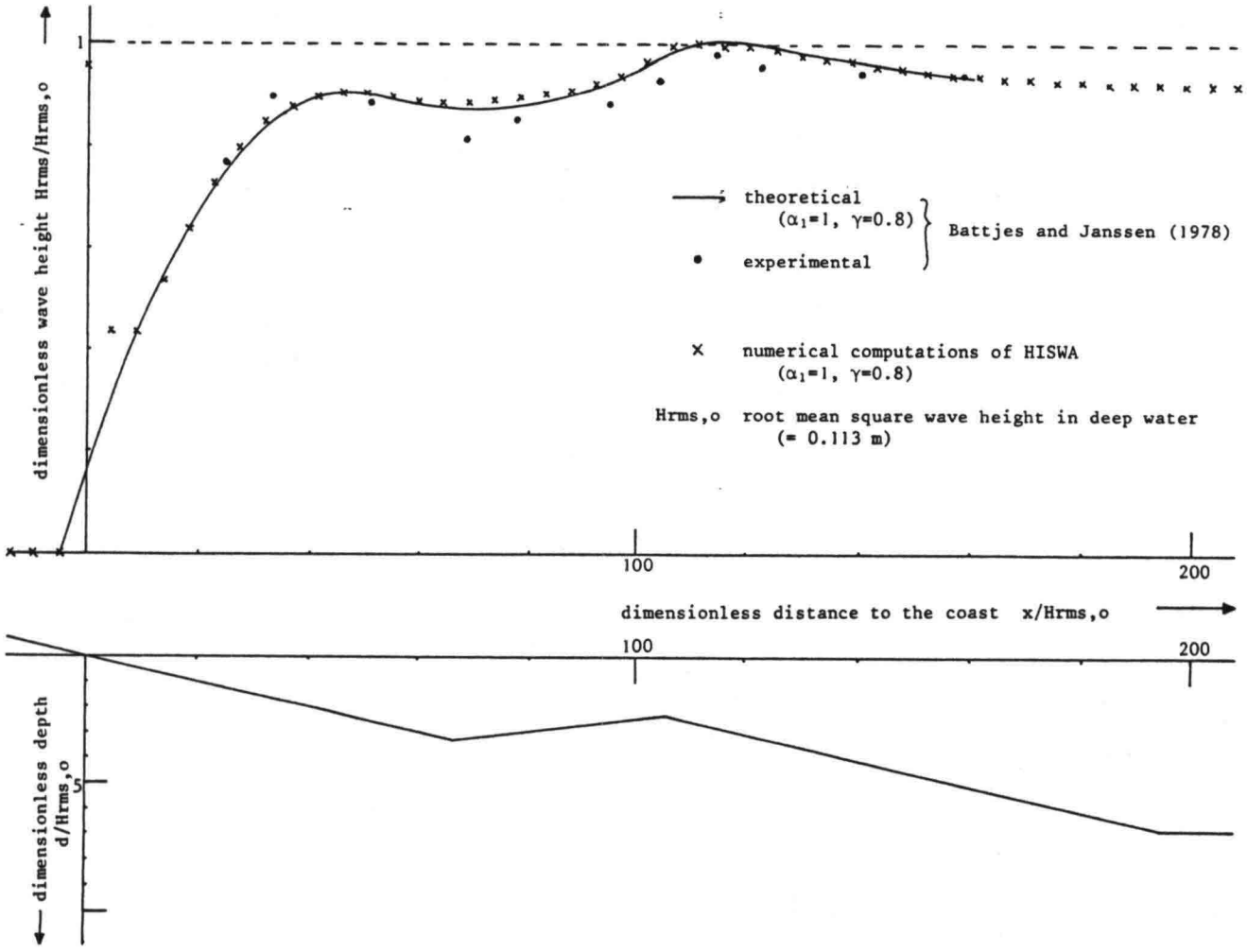
These authors checked their theoretical model with a laboratory experiment. The values for α_1 = 1.0 and γ = 0.8 are used in this test. For this test $(d\omega_o/dt)_{surf}$ is assumed 0.0 since there is hardly any information on the values of a_4 and b_4.

In fig. 8b the computed variation of wave height is plotted versus distance to the coast for both the HISWA results and the theoretical and laboratory results of Battjes and Janssen. The agreement is good except for very small distances to the coast where predicted wave heights are too large. This is at least partly due to the absence of wave set-up in HISWA.



area 25 x 50 m
grid 49 x 25 x 17 points
minimum direction of propagation -45°
maximum direction of propagation +45°
mesh size 0.52m x 2.08m x 5.6°

a) situation test.



b) results.

Fig. 8 Surf dissipation

4.4 Dissipation due to currents

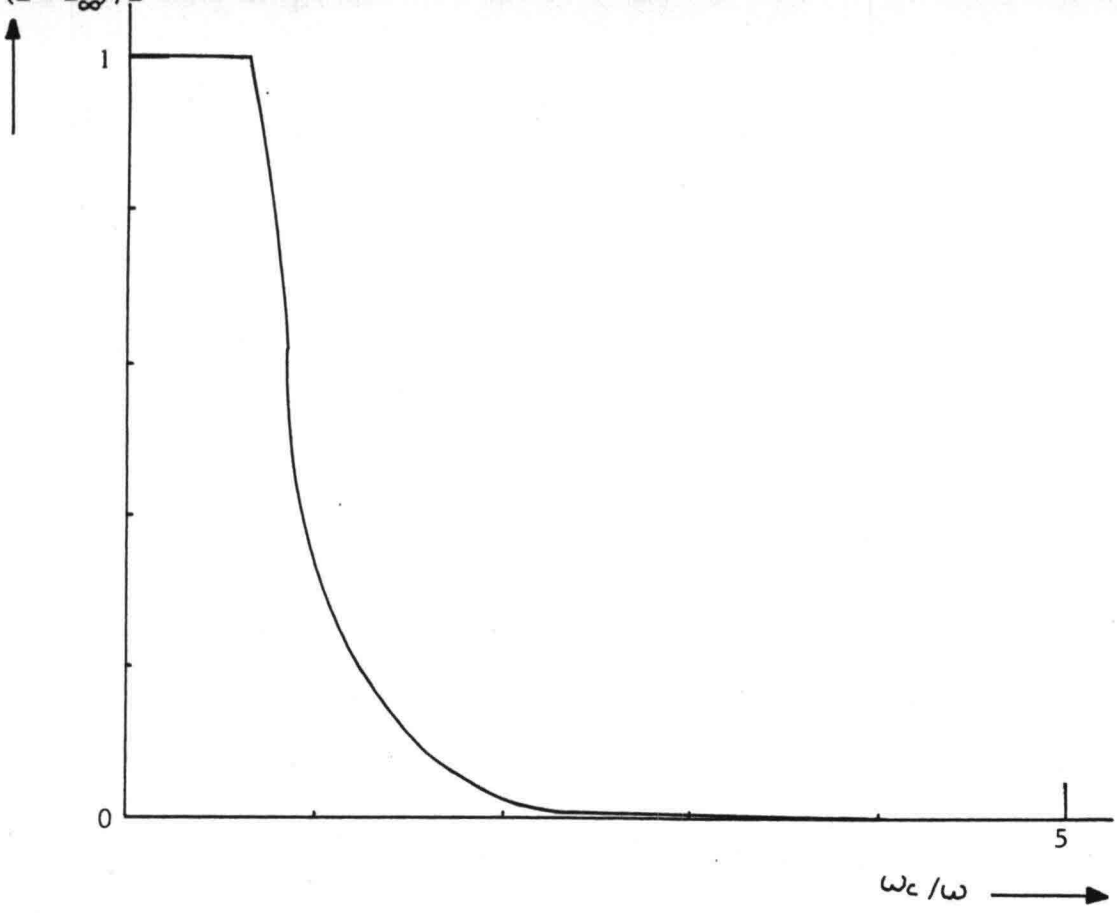
In addition to the surf dissipation mechanism (which operates also for strong adverse currents, some dissipation is assumed for energy at high frequencies in situations with strong opposing currents. This model is described in Booij et al. (1984). The resulting source terms are:

$$\left(\frac{dE_o}{dt}\right)_{currents} = -\frac{1}{\tau_1}(E_o - E_\infty) \qquad (30)$$

$$\left(\frac{d\omega_o}{dt}\right)_{currents} = -\frac{1}{\tau_2}(\omega_o - \omega_\infty) \qquad (31)$$

The relations between $(E_o - E_\infty)$ and $(\omega_o - \omega_\infty)$ and ω_c/ω_o (ω_c is a critical frequency above which wave components loose energy) is shown in fig. 9. The time scales τ_1 and τ_2 are assumed to be equal to the mean wave period.

The above source term is tested only in the following hypothetical situation. In deep water a narrow stream is assumed with high current velocities and strong gradients (fig. 10a). The direction of incidence of the waves is 60° relative to the main direction of the stream. In fig. 10b through 10d the variation of wave height, -period and -direction in a section perpendicular to the channel is shown with and without current induced dissipation.
The decrease of wave amplitude and -frequency and shift in wave direction seem realistic although there is at the moment no empirical data to support these results.



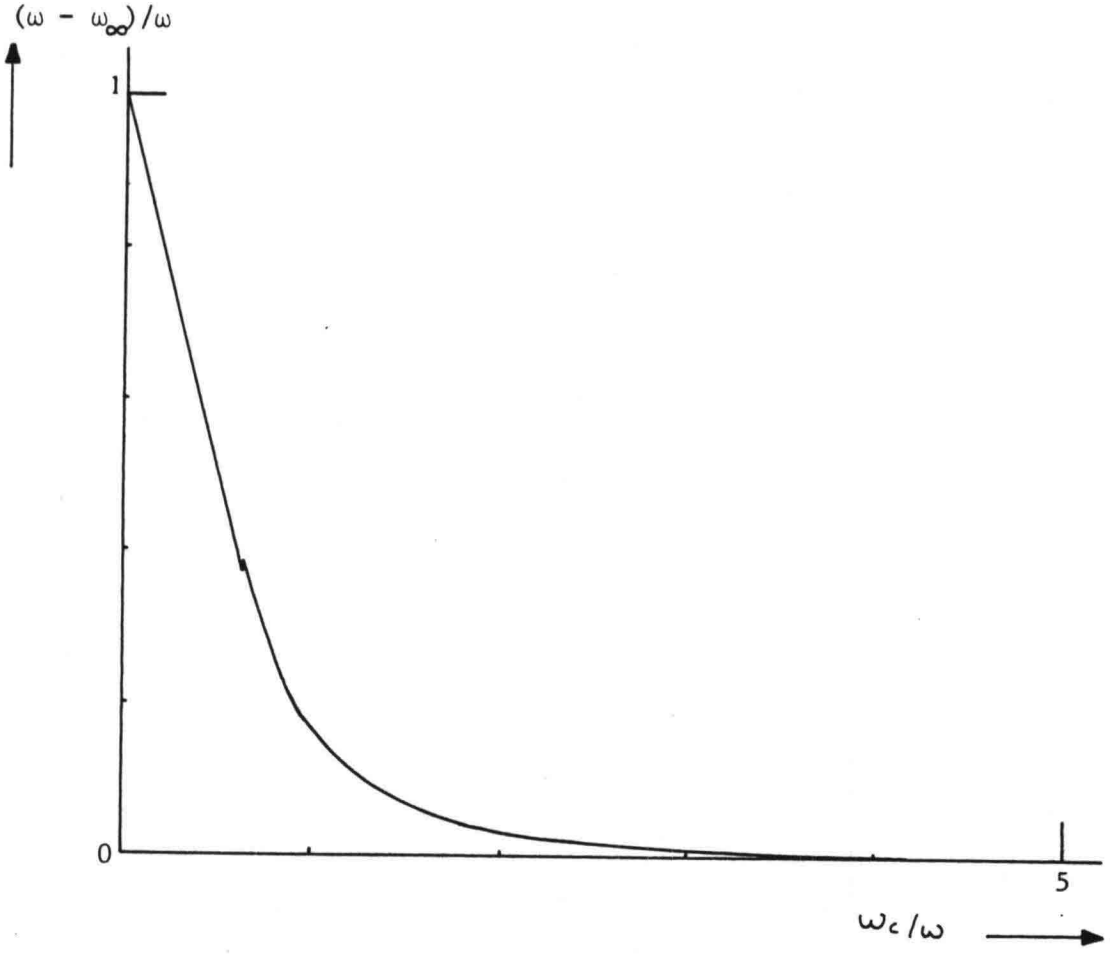
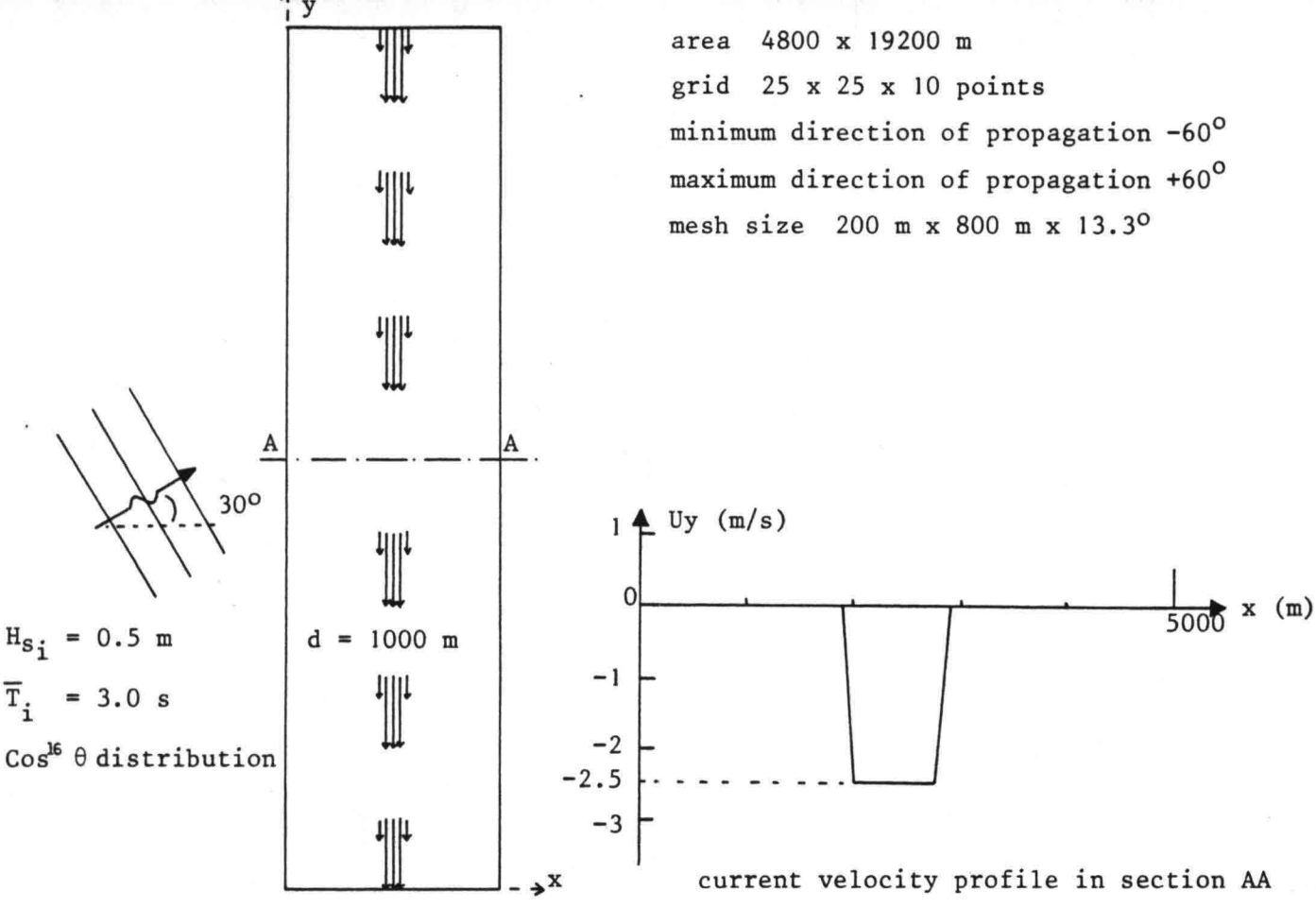
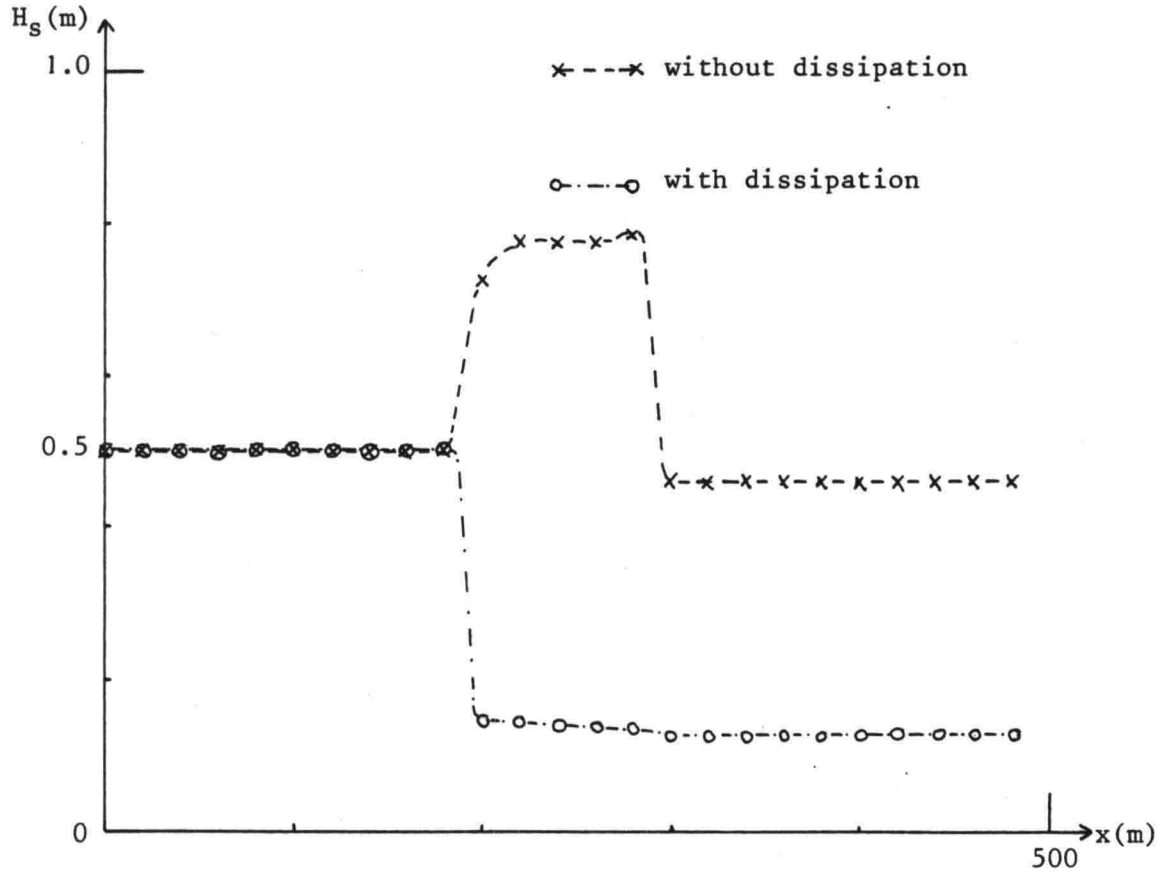


Fig. 9 Decay of wave energy and -frequency in adverse currents.

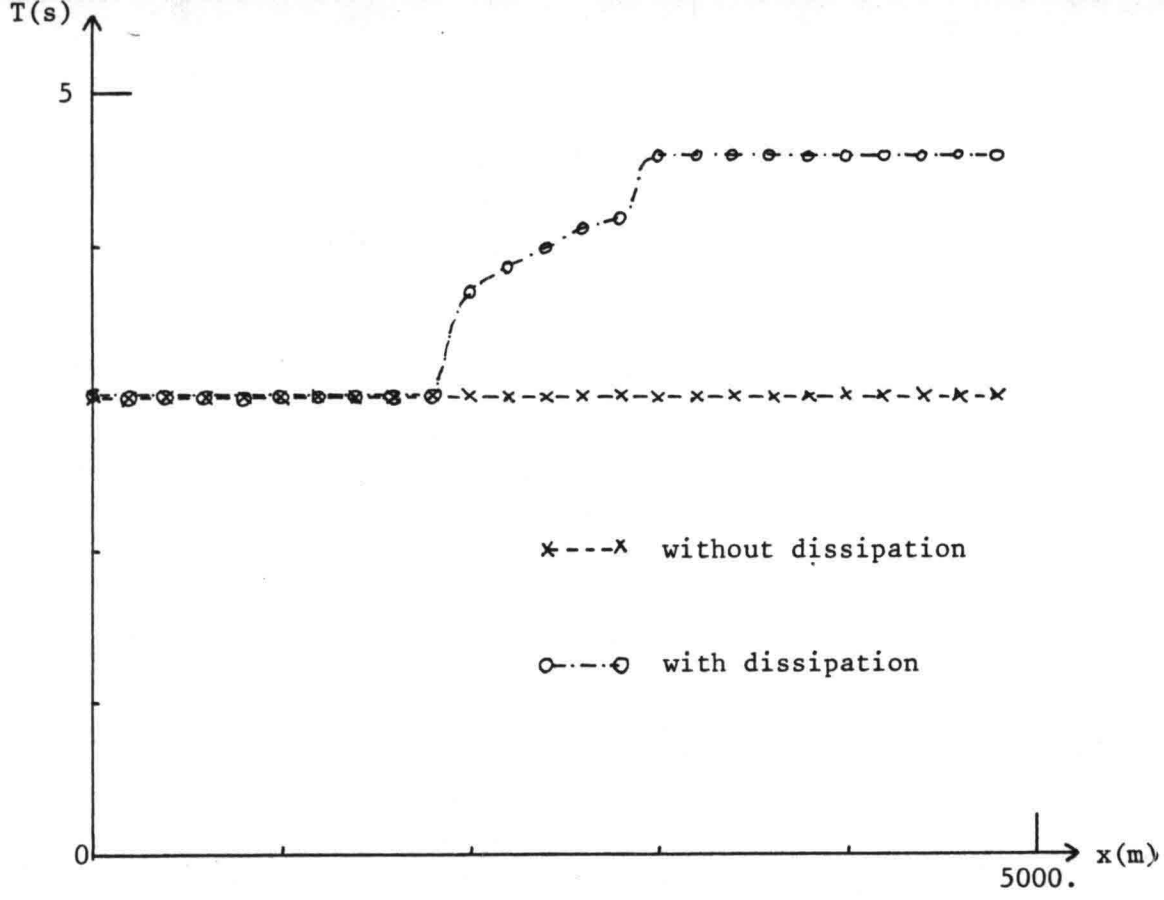


a) situation test.

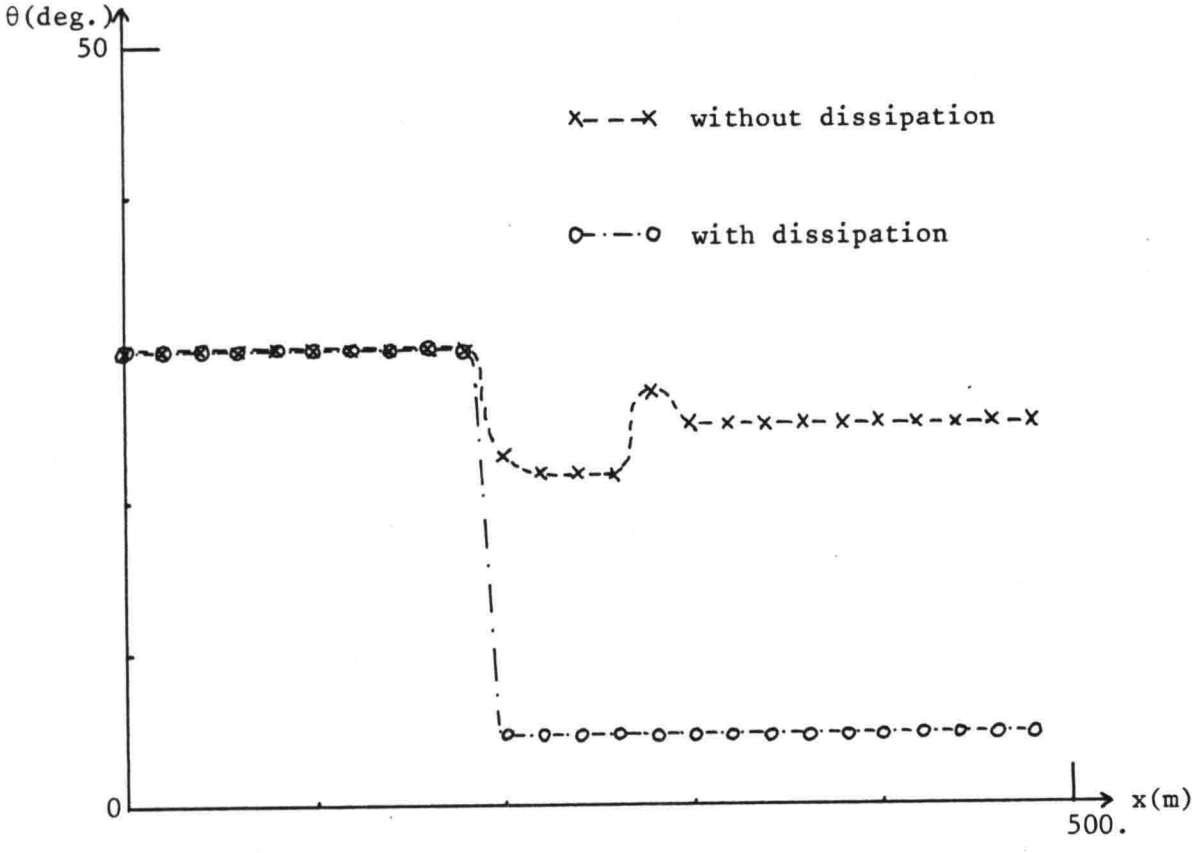


b) variation of significant wave height in section AA

Fig. 10 Dissipation in currents.



c) Variation of mean wave period in section AA.



d) Variation of mean wave direction in section AA.

Fig. 10 (continued).

5. CONCLUSIONS

The implementation of the numerical model HISWA is almost finished. Following the tests of wave propagation (Booij et al., 1984) a number of tests is carried out to check the representation in HISWA of diffraction effects, wind generation and dissipation processes.

The computation of diffraction-like behaviour in HISWA seems only practicable in situations in which the distance between the grid points is a very small fraction of the wavelength due to numerical instabilities. We feel, however, that this is not a significant limitation in the applicability of HISWA.

The results of tests for wind generation and dissipation processes agree well with empirical data and analytical solutions in sofar as these were found in relevant literature.

REFERENCES

1. Battjes, J.A. and Janssen, J.P.F.M., 1978, Energy loss and set-up due to breaking of random waves, proc. 16th Coastal Engineering Conference, Hamburg, ASCE New York, pp. 569-587.
2. Booij, N., Herbers, T.H.C. and Holthuijsen, L.H., 1984, Implementation (I) of the numerical shallow water hindcast model HISWA, Delft University of Technology, Report No. 6-84.
3. Bretschneider, C.L., 1973, see CERC, 1973.
4. CERC, 1973, Shore Protection Manual, U.S. Army Coastal Engineering Research Center, Corps of Engineers.
5. Dingemans, M.W., 1983, Verification of numerical wave equation models with field measurements. CREDIZ verification Haringvliet, Delft Hydraulics Laboratory, Report No. W 488.
6. Günther, H., 1981, A parametric surface wave model and the statistics of the prediction parameters, Hamburger Geophysikalische Einzelschriften, Reihe A, heft 55.
7. Hasselmann, K. and Collins, J.I., 1968, Spectral dissipation of finite depth gravity waves due to bottom friction, Journal of Marine Research, Vol. 26, No. 1, pp. 1-12.
8. Holthuijsen, L.H., 1980, Methoden voor golfvoorspelling, Technische adviescommissie voor de waterkeringen.
9. Holthuijsen, L.H. and Booij, N., 1983, Selection and formulation of a numerical shallow water wave hindcast model, Delft University of Technology, Report No. 17-83.
10. SWAMP, 1981, The Sea Wave Modelling Project, draft for the Proceedings, Conference on Radio Probing and Wave Dynamics of the Ocean, Miami.
11. Wiegel, R.L., 1964, Oceanographical Engineering, Prentice-Hall, Englewood Cliffs, N.J.

Appendix I User's Manual of HISWA

HISWA user manual

preliminary version

by N. Booij
T.H.C. Herbers
L.H. Holthuijsen

date of printing: September 19, 1984

Contents

1. PHYSICAL AND NUMERICAL BACKGROUND

1.1. Introduction

The model HISWA is a numerical model to obtain realistic estimates of wave parameters in coastal areas, lakes and estuaries for given stationary wind-, bottom-, and current conditions. The basis of the model is a parameterized version of the energy balance of the waves (or action balance in the presence of currents).

The notion of energy (*) balance is to equate for each spectral wave component the rate of change of its energy density to the rate of energy transfer due to such physical processes as wind growth, bottom dissipation etc.. In many conventional spectral wave models each such wav component is followed across the area of interest. During this journey the effects of wind, bottom etc. are determined and accumulated until th wave component arrives at the location of interest. This process is repeated for all spectral components (all frequencies and all directions) so that eventually the full two-dimensional spectrum is determined at the location of interest.

The technique which is used in HISWA is similar to the above described technique but it differs from it in that per spectral direction the spectrum is parameterized in frequency domain with one (variable) frequency and a (variable) directional energy density (hence the characteristic "parameteric in frequency" of the HISWA model). The wave component is thus characterized by its direction only; its energy density and its frequency (equal to the mean frequency of the waves from that direction) are computed in HISWA by integrating the local effects of wind, bottom and currents while propagating the waves over the area of interest. These computations are carried out for each direction separately and virtually independently from each other (hence the characteristic "directionally decoupled" of the HISWA model). For this HISWA uses two equations for each spectral direction, both obtained by integrating the spectral energy balance of the waves over th frequencies :

- the energy balance equation for the energy density of all wave components from the direction under consideration

- an evolution equation to determine the mean frequency of

---

* In this manual the term "energy" stands for the more appropriate term "variance", it stands for "action" whenever a mean current is present.

all wave components from the same direction.

The physical phenomena which are accounted for in these equations are :

a) refraction

b) diffraction

c) wind growth

d) bottom dissipation

e) surf dissipation

f) current dissipation

These phenomena are addressed separately below.

1.2 Refraction

Refraction which is commonly defined for one spectral wave component as the change in propagation direction due to variations along the wave crest of phase speed, is usuallly computed with the so-called ray-technique.From an arbitrarily chosen location (usually on the up-wav boundary of the area of interect) the path of the wave components under consideration is computed from the initial direction and frequency on the basis of local depth and current information. The wave direction is thus a dependent variable as the computations progress along the ray. During the journey alon, the ray such a model can take into account the local effects of wave growth and dissipation. This is essentially a Lagrangian approach : the changes are considered while travelling with the component.

In HISWA the approach is basically Eulerian : the changes in the wave component are considered at one fixed location and at fixed spectral directions. This different approach requires that the wave energy density and frequency are considered as a function of direction at the fixed location. Refraction is then accounted for by transferring energy across the directions. The mathematical formulation for refractio that is used in HISWA can therefore be considered as an Eulerian formulation of the conventional refraction theor,. It is applicable to both bottom- and current refraction.

The above Eulerian formulation is used in HISWA in the energy ailance equation and in the evolution equation for the determination of the main frequency. This implies that in contrast to conventional refraction computations, these refraction computations are carried out for a wave frequency

which varies due to wave growth and decay. This frequency is constant when growth and decay are absent.

2.2. Diffraction

Diffraction is commonly defined for one spectral wave component as the change in propagation properties due to variations along the wave crest in wave energy. Its effect is to redistribute wave energy over the area of interest, away from areas of high wave energy towards areas of low wave energy, and to change the wave direction correspondingly. The computation of diffraction in arbitrary geophysical conditions is rather complex and requires considerable computing effort. To avoid this the theoretical correct approach is not used in HISWA. Instead a diffraction-like behaviour of the waves is modelled in the sense that the same qualitative behaviour of spatial redistribution and changes in wave direction is obtained. This is achieved by transferring wave energy across the spectral directions (at the fixed location under consideration). The rate of transfer depend on the local gradient along the wave crest of the energy density of the wave component considered. The formulation is such that currents affect this diffraction-like behaviour.

The above diffraction simulation is used in HISWA in the energy balance and in the evolution equation for the determination of the mean frequency. This implies that, in contrast to conventional diffraction computations, the diffraction-like computations are carried out for a wave frequency which varies due to wave growth and decay. The frequency is constant if wave growth and decay are absent.

2.3. Wind growth

The growth of the wave energy is considered for each spectral direction separately and independently from other spectral directions. The growth is taken equal to the growth for the considered direction in an idealize situation (unlimited ocean with homogeneous and stationary wind) as published in an extensive study of advanced spectral wave models (Ris, 1991), under the assumption that in that situation the directional energy distribution has a universal shape.

The evolution of the mean frequency is also considered separately and independently for each direction. It is directly related to the growth of the wave energy in the considered direction. This evolution of the mean frequency is based on an assumed universal relationship between the wave energy and the mean frequency (in dimensionless form). When the JONSWAP field does not conform to this universal

relationship (e.g. by rapid changes in the bottom or current pattern or through user-imposed boundary conditions), the evolution of the mean frequency is adapted to force the wave field towards this relationship.

The effect of currents is included by using the apparent wind speed and direction in the growth and evolution expressions rather than the real wind speed and direction.

2.3 Bottom dissipation

In shallow water the wave energy in each spectral direction is dissipated in HISWA by bottom friction. This dissipation is determined with a fairly conventional nonlinear bottom friction model including the effect of a mean current. It is based on a conventional formulation for periodic waves (quadratic friction law) with the appropriate parameters adapted to suit a random wave field as considered in HISWA.

The mean wave frequency is only indirectly affected by bottom friction in HISWA, virtually by an assumed relationship with the wave energy dissipation due to bottom friction.

2.5. Surf dissipation

In extremely shallow water the waves break in a surf zone. The corresponding energy dissipation is determined in HISWA with a bore-model for the waves which are higher than some treshold value. Only the total rate of energy dissipation is thus determined (i.e. integrated over all directions). This total rate is distributed over the directions proportional to the wave energy in a direction.

The mean wave frequency is affected only indirectly by surf dissipation in HISWA, virtually by an assumed relationship with the wave energy dissipation due to surf breaking.

The effect of a mean current on surf breaking is not taken into account.

2.6 Current dissipation

In a strong adverse current some wave energy is carried away by the current. This is energy which in a full spectral model (all directions and frequencies included) is carried by wave components that cannot travel against the current (propagation speed less than current speed). This energy is removed from the wave field in the HISWA model through high frequency dissipation in an assumed standard frequency spectrum.

4. DESCRIPTION OF COMMANDS

4.1. List of available commands

The following commands are available to users of HISWA:

General commands

PROJECT	Title of the problem to be computed
POOL	Assign space for dynamic data pool
TEST	requests the output of intermediate results for testing purposes.
STOP	End of user's input. Starts a computation.
RESTORE	Brings data of old run back into memory.

Commands for model description

SET	sets values of certain general parameters
GRID	position and size of computational grid
BOTTOM	position and size of bottom grid
READ	read bottom and/or current velocity values
INC	defines incident wave field
BOUND	boundary conditions at lateral sides of the computational grid
WIND	wind influence on wave field
BREAK	breaking of waves
FRIC	bottom friction
DIF	diffraction

Output commands

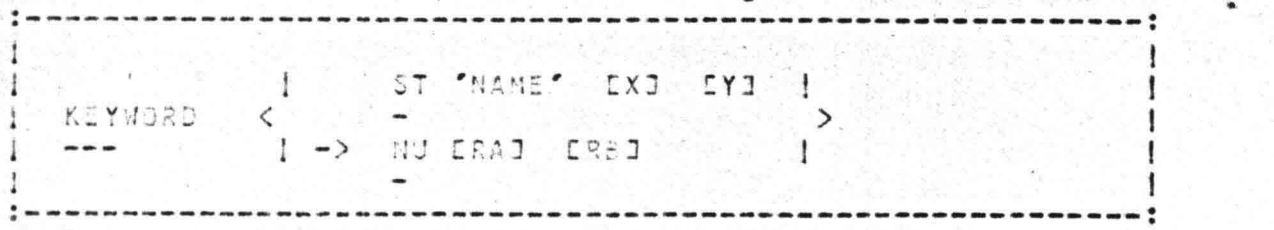
FRAME	defines an output frame
CURVE	defines an output curve
POINTS	defines a set of output points
BLOCK	requests a block print
TABLE	requests printing of a table
PAPER	data concerning plotting paper
PLOT	requests plotting of a figure
SHOW	requests a verification plot

Format of the input

The input for HISWA is organized in the form of commands. Each command is designated by a keyword consisting of letters and (sometimes) digits, but always beginning with a letter. After this keyword usually other data appear, such as real or integer numbers, or character strings. Character strings must always be enclosed in quotes; keywords are not in quotes. Strings and keywords have an entirely different meaning to the program; a string is a variable, a keyword has a fixed meaning, instructing the program to perform certain actions.

It is not always necessary to actually write down all the data required by the program. In many cases the program will assume reasonable values for variables that do not appear in the command. The command description will mention whether an initial value or a default value is assigned to a variable. An initial value is assigned by the program at the very start of the job, whereas a default value is assigned at the moment that the command is executed. Often an initial value and a default value are different.

Commands are described in the following form:



The following rules apply for the command description:

- Keywords are not enclosed by square brackets or quotes, the letters that are underlined must be copied literally; other letters or digits may be added, as well as the characters - and _. So in the command outlined above one may write: KEY or KEYW or KEY-WORD or KEYHOLE etc.
  The first keyword in the command scheme is also the command name.
- A name between square brackets is to be replaced by a (real or integer) number; a name between quotes is to be replaced by a string, also enclosed in quotes. In the command description one can find what the program does if a variable is not assigned a value by the user. The description also should make clear whenter a real or integer number is expected.
  Note: a decimal point is not permitted in an integer number.
  The data must be given in the same order as they appear in the description. If one wants to assign the value 6 to the variable [RA], one writes 6 or 6., or RA=6.
- If one line of input is not large enough to hold the data

for one command, the command can be continued on the next line, if the last one has a continuation mark as its last item. The following continuation marks can be used: & or _ (the underscore, not the minus sign). In the command descriptions the & also signifies a continuation mark.

- A group of data between parentheses ( ) is optional; the command description tells what happens if the group of data does not appear int the user's command.
- A group of data within angle brackets < > can be given repeatedly. In the user's input the end of the repetition is indicated by the end of the line, by the appearance of a keyword, or by one of the following characters: / or]. The group of data must be given at least once, unless it is also surrounded by parentheses.
- Alternative options in the command are written between braces of the following form:

```
    ! ....... !
  <             >
    ! ....... !
```

Each alternative is characterized by a keyword (in the above example: ST and NU). If an arrow appears before one of the alternatives (->), this alternative is chosen if none of these keywords appears.

- Data are separated by blanks and/or commas. A keyword is closed by a blank or one of the following characters: = or :. An empty data field in a series of data fields is recognized only if it is surrounded by commas. Also the program will assume that a data field is empty is it finds a keyword where a data item is expected.
- If the user wants to write a set of identical data fields, he can use the repetition factor, e.g.

```
    6 * 'I.C.U.', 3*0.,
```

Note: The repetition factor cannot be used in front of a coefficient that may appear as the first on a new line of input, nor can it be used to give a number of empty data fields.

- To clarify the meaning of the input, one can insert comment. Comment must be enclosed in $ signs. If there is only one $ sign on a line it is assumed that the end of the line is the end of the comment; the next line is considered again as ordinary input.

An example is found in chapter 6 of this guide.

General commands

```
:-------------------------------------------------------------:
|                                                             |
|  PROJECT  'TITLE1'                                          |
|  ----                                                       |
|                                                             |
|           'TITLE2'                                          |
|                                                             |
|           'TITLE3'                                          |
|                                                             |
:-------------------------------------------------------------:
```

A description of the run is given in the 3 lines 'TITLE1', 'TITLE2' and 'TITLE3'. Each of these is max. 56 characters long. The lines will be reproduced in the output by the program. Initially all three lines are empty.

```
:-------------------------------------------------------------:
|                                                             |
|  POOL  [NPOOL]                                              |
|  ----                                                       |
:-------------------------------------------------------------:
```

HISWA stores most numbers in a dynamic data pool, which must be large enough to hold all this information. The command POOL is used to give the pool a proper size. [NPOOL] is the size of the pool in blocks of ... numbers. The initial value of [NPOOL] is 12, which is sufficient for smaller problems.

Important: if the command POOL is used, it must be the first command in the list of commands given by the user.

```
:-------------------------------------------------------------:
|                                                             |
|  STOP    ( NORUN )    ( SAVE [IWF] )                        |
|  ----      ---          ---                                 |
:-------------------------------------------------------------:
```

Marks the end of the users input.
After this command the program will start to write the model description and other computational instructions to file INSTR (computational instructions), unless the keyword NORUN is present.
Furthermore it will write the output parameters and output requests to file INSTO (output instructions).
If the keyword SAVE is present, the model description, output data and output requests are written to the file with reference number IWF. This file is called the Workfile, it can be used if one wants to carry out a series of computations without being forced to repeat the model description etc. every time. The SAVED data can be restored in the next run by the command RESTORE. Default: IWF=11

Note: the SAVE facility is not yet available

RESTORE [WF]

This command RESTORES the data that were SAVED by the command STOP in a previous run. Init: [WF]=12
If this command is used, it should be the first in that run. The reason is that it will destroy all information entered by previous commands in the same run, and will replace it by information from the run in which the SAVE was done.
One can do in the same run a RESTORE and a SAVE on the same file. In that case the information from the previous run is lost. If one wants to retain the information from the previous run, the value for [WF] in command RESTORE must differ from the one in the command SAVE.

After RESTORE one has to enter only the commands needed to change the model, or the output.

Note: facility not yet available.

TEST [ITEST] [ITRACE]

If the program produces unexpected results, this command can be used to instruct the program to produce intermediate results during the computation process.

Parameters:
[ITEST] instructs HISWA to produce test output; the larger the value of [ITEST], the more output is produced. For values under 100 the amount is usually reasonable, for values above 200 it can be huge.
Init: [ITEST]=0, so no test output is made;
Default: [ITEST]=30, test output is made that can be interpreted by the user. For higher values of [ITEST] output is made, that can only be interpreted by those who have the program source listing at their disposal.
[ITRACE] instructs HISWA to produce a message if subroutines are entered.
[ITRACE]=1: only the first entry is signalled;
[ITRACE]=2: every entry is signalled;
Init: [ITRACE]=0 (no messages).

Commands for model description

```
:-------------------------------------------------------------:
|                                                             |
|  SET  [LEVEL]  [MAXERR]  [GRAV]                             |
|  ---                                                        |
:-------------------------------------------------------------:
```

Assigns values to various general parameters.

[LEVEL] is the (constant) water level in the region (in m). The depths used in the wave propagation computation are equal to the sum of the water level defined by this command, and the bottom level read by the command READ. Init: [LEVEL]=0.

[MAXERR] indicates at which error level a computation may be started. The error level is coded as follows: 1: Warnings, 2: (possibly repairable) errors, 3: Severe errors. Init: [MAXERR]=1

[GRAV] is the gravitational acceleration (in m/s2). Init: [GRAV]=9.81

```
:-------------------------------------------------------------:
|                                                             |
|  GRID  [XLEN]  [YLEN]  [SECTOR]  [MX]  [MY]  [MD]  &        |
|  ----                                                       |
|                                                             |
|        |  FIXED     [XPC]  [YPC]  [ALPC]          |         |
|        |  -                                       |         |
|      <                                             >        |
|        |  ROT   [XPR]  [YPR]  [XCR]  [YCR]        |         |
|        |  -                                       |         |
|                                                             |
:-------------------------------------------------------------:
```

Defines the position and size of the computational grid. The X-axis of this grid is the computational direction, the Y-axis is normal to this direction. The X-axis should be roughly in the mean wave propagation direction. The orientation of the grid can either be FIXED, or ROTATING. In the latter case the program chooses the X-axis to be coincident with the average direction of the incoming waves given by the command INC; this is allowed only for parametric incident waves (see command INC PARA ...).

Meaning of the parameters:
[XLEN] length of the grid in X-direction (in m).
[YLEN] length of the grid in Y-direction (in m).
[SECTOR] directional interval of propagation directions for which the wave energy density will be determined (in degrees). This sector must be smaller than 180 degrees. The X-direction shall be in the middle of

the sector.
[MX] number of meshes in X-direction. In view of numerical stability the program will check whether [XLEN]/[MX] is smaller than [YLEN]/[MY] divided by tan(0.5*[SECTOR]). If not, an error message is printed.
[MY] number of meshes in Y-direction.
[MD] number of subdivisions of the directional interval, so [SECTOR]/[MD] is the spectral directional resolution.
[XPC] Fixed grid: position of the origin of the computational grid in terms of user coordinates (X-coordinate, in m).
[YPC] ,, (Y-coordinate, in m).
[ALPC] Fixed grid: orientation of the computational grid with respect to the X-axis of the user coordinate system (angle in degrees, measured counterclockwise).
[XPR] Rotating grid: position of the rotation point in user coordinates (in m).
[YPR] ,,
[XCR] Rotating grid: position of the rotation point in computational grid coordinates (in m).
[YCR] ,,

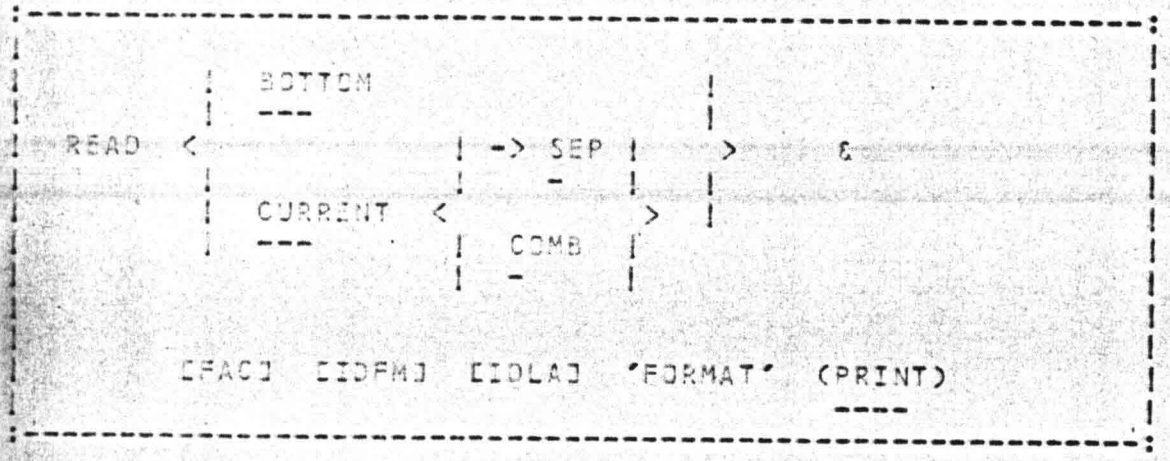
```
:-------------------------------------------------------------:
|                                                             |
|  BOTTOM  [XUB] [YUB] [ALUB] [MX] [MY]   &                   |
|  ---                                                        |
|                                                             |
|        [DX] [DY] [DEPX]                                     |
|                                                             |
:-------------------------------------------------------------:
```

Defines the position and size of the bottom grid. If a current field is present the current has to be given at the same points as the bottom.

Parameters:
[XUB] position of the origin of the bottom grid in user coordinates (in m).
[YUB] ,,
[ALUB] orientation of the bottom grid with respect to the positive X-axis of the user coordinate system (angle in degrees, measured counterclockwise).
[MX] number of meshes in X-direction of the bottom grid.
[MY] number of meshes in Y-direction of the bottom grid.
[DX] mesh size in the bottom grid (in m).
[DY] mesh size in the bottom grid (in m); default: equal to [DX].
[DEPX] depth outside the bottom grid (in m); if the computational region contains points outside the bottom grid, in these points the depth will be taken

equal to [DEPX]. Default: [DEPX]=1000.

The bottom levels themselves are read by the command READ (see next command). These are taken positively downward, measured from a datum chosen by the user. A certain water level (read by SET LEVEL) constant over the region can be added. If the water level is not horizontal, one should input the actual depth, i.e. the bottom level measured from the water surface, instead of the depth relative to the chosen datum.



With this command the bottom configuration and/or the current field is read by the program.

The bottom levels or the current velocity components are read from the file according to lay-out index [IDLA], and multiplied by [FAC]. Defaults: [FAC]=1. and [IDLA]=1. [IDLA]=1 means that the numbers appear line by line (a line is a line in X-direction), with each line starting on a new input line. [IDLA]=2 means the same order, however, a new line must not necessarily appear on a new input line. [IDLA]=3 means that the numbers appear column by column (with a column is meant a line in Y-direction), with a new column starting on a new input line, =4 means the same order, however, a new column must not necessarily start on a new input line.

The format in which the numbers are written must also be given. The format number [IDFM] is interpreted as follows: 5: format(10F5) or 20 fields of 5 places, 6: (12F6), 8: (10F8), 2: FORMAT must be provided by the user, such as: '(10X,12F5)'.
Regarding the input on the file the rules for ordinary Fortran input apply; i.e. continuation marks are not allowed. Simply continue on next line if a line is not long enough to hold all the numbers.

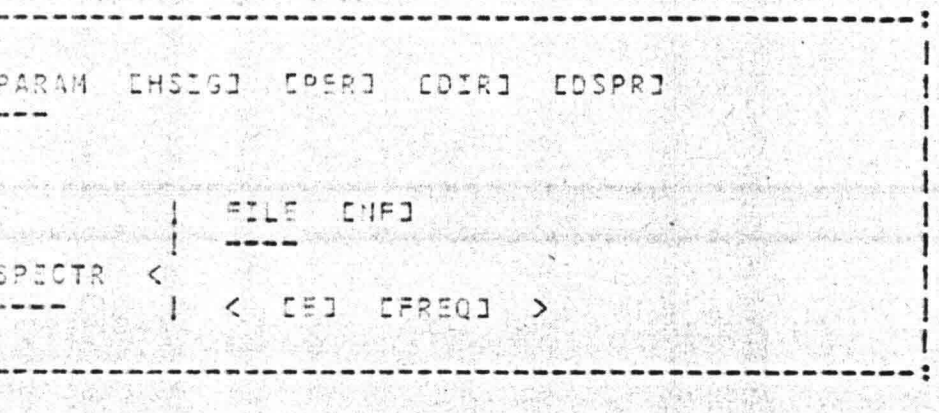
Remark: HISW4 uses the level of the bottom with respect to a common datum. Therefore a positive depth will usually mean

a negative bottom level. If necessary, the sign of the value can be corrected by assigning a negative value to the number [FAC].

One can obtain a contour plot of the bottom levels by the following command:

SHOW BOTTOM

for details see command SHOW.



This command defines the incident wave field. This command in not compulsory; if the command is absent it is assumed that there are no incident waves.

The incident waves can be given either in PARAMETRIC or in SPECTRAL form.

In the PARAMETRIC case the user prescribes: [HSIG], significat wave height, [PER], the mean wave period, [DIR], the mean direction of the waves, and [DSPR], which is a parameter for the directional spread, defined as follows. The program assumes that the incident waves are distributed over the directions according to the function (Cos(Theta/2))**[DSPR]. The default is: [DSPR]=2.

In the SPECTRAL case the spectrum is prescribed either by giving a set of values for energy density and mean period, as function of the direction Theta, in which case the same spectrum is assumed for the whole boundary x=0; or by reading the values from a file, in which case the spectrum must be given for each point of this boundary.

The numbers [E] and [FREQ] must be given for each spectral direction, starting with the lowest value of Theta, so [MDC]+1 occurrences of this pair must be present. The command GRID must be given prior to INC in this case. [E] is the spectral energy density for that direction, and [FREQ] is the mean frequency for the same direction.

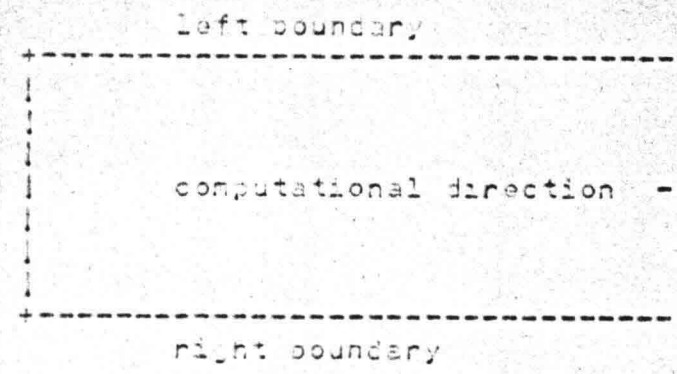
With the option FILE [NF] the spectral values are read from a file; the file reference number [NF] must be given. For

each point on the boundary x=0, starting with the point y=0 the whole spectrum must be present on the file.

```
:-------------------------------------------------------:
!                                                       !
!             ! LEFT  !     ! INC                       !
!             ! --    !     ! ---                       !
! BOUNDARY  <           >   <                           !
! ---         ! RIGHT !     ! FILE [NF]                 !
!             ! --    !     ! ----                      !
:-------------------------------------------------------:
```

This command determines the boundary condition at the lateral side of the computational region, i.e. the wave that are incident over that particular boundary. The command has no effect on the waves leaving the computational region; these are always absorbed by the boundary. If the command BOU is absent it is assumed that no waves enter the region from outside.

The RIGHT boundary is the boundary y=0, the LEFT boundary is y=[MY]*[DY].



INC: the wave spectrum at the boundary (only the incoming waves) is the same as that of the wave field entering through x=0, which is defined by the command INC. Applicable only in the case of INC PARAMETRIC.

FILE [NF]: the spectra are read from a file. For each point of the boundary the whole spectrum must be present on the file; so also values for wave directions pointing out of the region must be available on the file, although these do not have any effect on the computation.

Note: facility not yet available.

```
:-------------------------------------------------------:
!                                                       !
!  WIND [VEL] [DIR] [A] [B] [C] [D] [A2] [B2]           !
!  ----                                                 !
:-------------------------------------------------------:
```

Upon this command a source term due to wind influence is added to the energy balance equation. Only the values for [VEL], the wind velocity (at an elevation of 10 m, unit: m/s) and [DIR] the direction in which the wind blows, with respect to the user coordinate system (in degrees), are required. For the other parameters reasonable values are assumed by the program. These values are empirical. Ref: ...

```
:-------------------------------------------------------------:
|                                                             |
|  BREAKING [GAMMA] [ALFA]     &                              |
|  ---                                                        |
|         ( FREQ  [A4]  [B4] )                                |
|           ----                                              |
:-------------------------------------------------------------:
```

With this command a source term due to breaking can be specified. For each coefficient in the formula for the source term a default value is assumed by the program. Initially the breaking term is already present, in contrast with the wind and bottom friction terms, which must explicitly switched on using the commands WIND and FRICTION. Initially no influence of the breaking process on the frequency is assumed. The frequency influence can be switched on by: BRE FREQ .

If the keyword FREQ is not present, the program assumes that the breaking process does not influence the frequency of the waves. If FREQ is given, the change of frequency is related to the change of wave energy, by means of a formula described in ref ...

```
:-------------------------------------------------------------:
|                                                             |
|  FRICTION  [CFWI]  [CFCU]   ( FREQ  [A3]  [B3] )            |
|  ----                        ----                           |
:-------------------------------------------------------------:
```

Upon this command a source term due to bottom friction is added. For each coefficient a default value is assumed by the program; usually it suffices to give the command: FRIC or: FRIC FREQ . Ref: ...

Meaning of parameters:

[CFWI]	Coefficient for bottom friction due to waves,
[CFCU]	Coefficient for bottom friction due to the current,
FREQ	If this keyword is present, it is assumed that the wave frequency is influenced by the bottom friction.
[A3]	Factor in rate of change of frequency
[B3]	Power in rate of change of frequency.

DIFFRACTION [ALFA]

Upon this command diffraction terms are added to the model. It is stressed that diffraction is approximated rather roughly.

A likely value of [ALFA] cannot be given, since it is restricted by the forward step size. A value of [ALFA] which is too large, may cause numerical instabilities.

Output requests

There are different kinds of output commands:
1. commands defining sets of points for which one will want output.
   The following types of point sets exist: FRAME, a rectangular array of points, CURV, a set of points along a curve, POINT, a set of isolated points.
2. names and lines that can be plotted in figures.
   Names of towns or regions that must appear in the plots are defined by the command PLACE, and lines (for instance giving the contour of a certain object) are defines by the command LINES.
3. commands requesting certain output actions.
   Types of output available in HISWA are: BLOCK, a wave variable is printed for a rectangular array of points; TABLE, for each point of a set of points a number of variables is printed; PLOT, a figure is plotted for a rectangular region, containing iso-lines of a scalar quantity and/or a vector plot for a vector-type quantity.

In addition the command SHOW provides the possibility to verify the location of output frames, curves and points. The command SHOW also has options to verify the depths and currents velocities. The verification plots are made before the computation starts.

```
:-------------------------------------------------------------:
|                                                             |
|  FRAME  'SNAME'  [XLEN]  [YLEN]  [XP]  [YP]  &              |
|  ---                                                        |
|             [ALP]  [MX]  [MY]  [SCALE]                      |
|                                                             |
:-------------------------------------------------------------:
```

An output region with rectangular shape is defined. The name of the point set is 'SNAME'.
Its location is defined in terms of user coordinates; [XP] and [YP] are the coordinates of the origin of the frame (in m.); and [ALP] is the angle in degrees of the x-axis of the frame with respect to the x-axis of the user coordinate system.
[XLEN] and [YLEN] are the dimensions of the frame (in m.); these values are required. [MX] and [MY] are the number of meshes along each of the sides of the frame. Default: [MX]=20, [MY]=20. Restriction: if the frame is used in a PLOT or SHOW command, [MY] must be smaller than 74.
[SCALE] is the scale to which the region must be plotted (if plotting is requested), in cm on paper per m in reality. The default is such that a picture with a width (in Y-direction) of 15 cm results.

Each output point set is characterized by its name, denoted in this manual as SNAME. If one gives two definitions for the same set name, the first definition is lost.

```
CURVE 'SNAME'
---

      < [XP1] [YP1]  < [INT] [XP] [YP] > >
```

This command defines a set of output points along a curve. Actually this curve is a broken line, defined by meand of its corner points.
The command can be used to define more than one curve.

'SNAME' name of the point set, max. 8 characters.
[XP1] [YP1] user coordinates of the first point of a curve (in m.).
[INT] number of sub-intervals (integer) between two consecutive between two corners of the curve; the program will generate [INT]-1 intermediate output points.
[XP] [YP] user coordinates of a corner point of the curve.

```
RAY   'RNAME'      &
---
      [XP1] [YP1] [XQ1] [YQ1]     &

      < [INT] [XP] [YP] [XQ] [YQ] >
```

Together with the command DEP (see next command) this command serves to define a set of output curves that follow a certain depth contour.

The command RAY defines a set of straight rays along which the program will attempt to find a point with a certain depth. Such a point will be added to the set of output points. Each of these rays is characterized by its end points ([XP], [YP]) and ([XQ], [YQ]). Between two rays defined by the user the program will generate [INT]-1 intermediate rays.

'RNAME' name of the set of rays. The command DEP will refer to this ray set name.
[XP1] [YP1] [XQ1] [YQ1] user coordinates of the end points of the first ray.
[INT] number of subdivisions; the program will generate

[INT]-1 intermediate rays.
[XP] [YP] [XQ] [YQ] user coordinates of the end points of a ray.

```
DEPTH  'SNAME'  'RNAME'   < [DEP] >
---
```

Together with the command RAYS (see previous command) this command serves to define a set of output curves that follow a certain depth contour.

'SNAME' name of the output curve, or the set of output curves.
'RNAME' name of the set of rays; the program will search points with the depth [DEP] along each ray in this set.
[DEP] the depth on one of the output curves. More than one values may be given. Each value leads to an output curve.

```
POINTS  'SNAME'   < [XP] [YP] >
----
```

This command defines a set of individual output points.

'SNAME' name of the point set
[XP] [YP] user coordinates of one output point.

```
PLACE
---
      < 'PNAME'  [XP]  [YP]  [SIZE]  [SIT]  >
```

Defines name of towns, regions etc. that can be plotted in the figures produced by the program. The command PLACE can be entered more than once. Each time new names are added to the set of names.
'PNAME' is the name of a town or region within the problem area. It can be max. 16 characters long.
[XP], [YP] are the coordinates of the point of reference (in m) where the name must be plotted.
[SIZE] is the size of the characters on the plot (in cm).

Default: 0.28 cm.

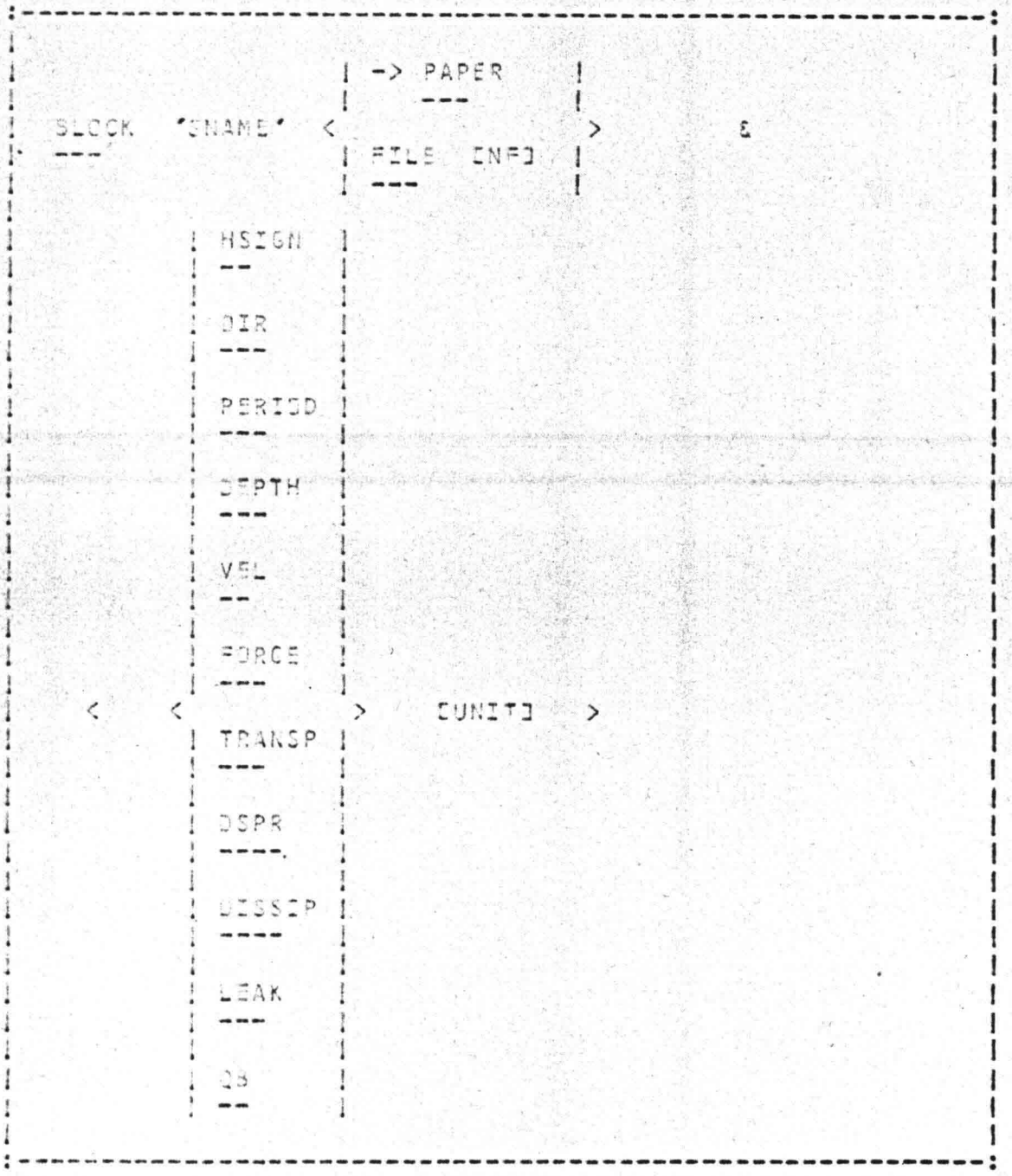
[SITI] an integer number indicating how the name must be placed on the plot.
1 (default): the first letter is placed at the point of reference (if the space allows). This is intended for names of towns etc.
2: the name is plotted with the point of reference in the middle (if the space allows). This is intended for names of regions, islands etc.
If the point of reference is near one of the edges of the frame, the name will be shifted as far as necessary.

LINE

< [LINTYP] < [XP] [YP] > >

Defines one or more lines that can be plotted in the figures produced by the program. The lines consist of straight rods, of which the end points must be given in the command. The command LINES can be entered more than once. Each time new lines are added to the set of lines.
The command is provided to facilitate orientation on the plots. One can indicate coastlines, contours of certain landmarks etc. For each line the following data must be entered:
[LINTYP] indicates the type of line;
0: Heavy continuous line,
1: Thin continuous line,
2: Continuous line with cross-hatches,
4: Thin broken line.
[XP], [YP] coordinates of a corner point of the line. The number of corner points is free.

For each new line (except the first) the number [LINTYP] must appear on a new input line.



This command instructs the output program of HISWA to produce a block print or a number of block prints. The block can either be made on paper, or on a file for further processing. In the latter case [NF] is the file reference number.

The block print is made for the set of output points denoted by the point set name 'SNAME'. This point set must be of the type FIXED.

In the block print on paper only integer numbers are

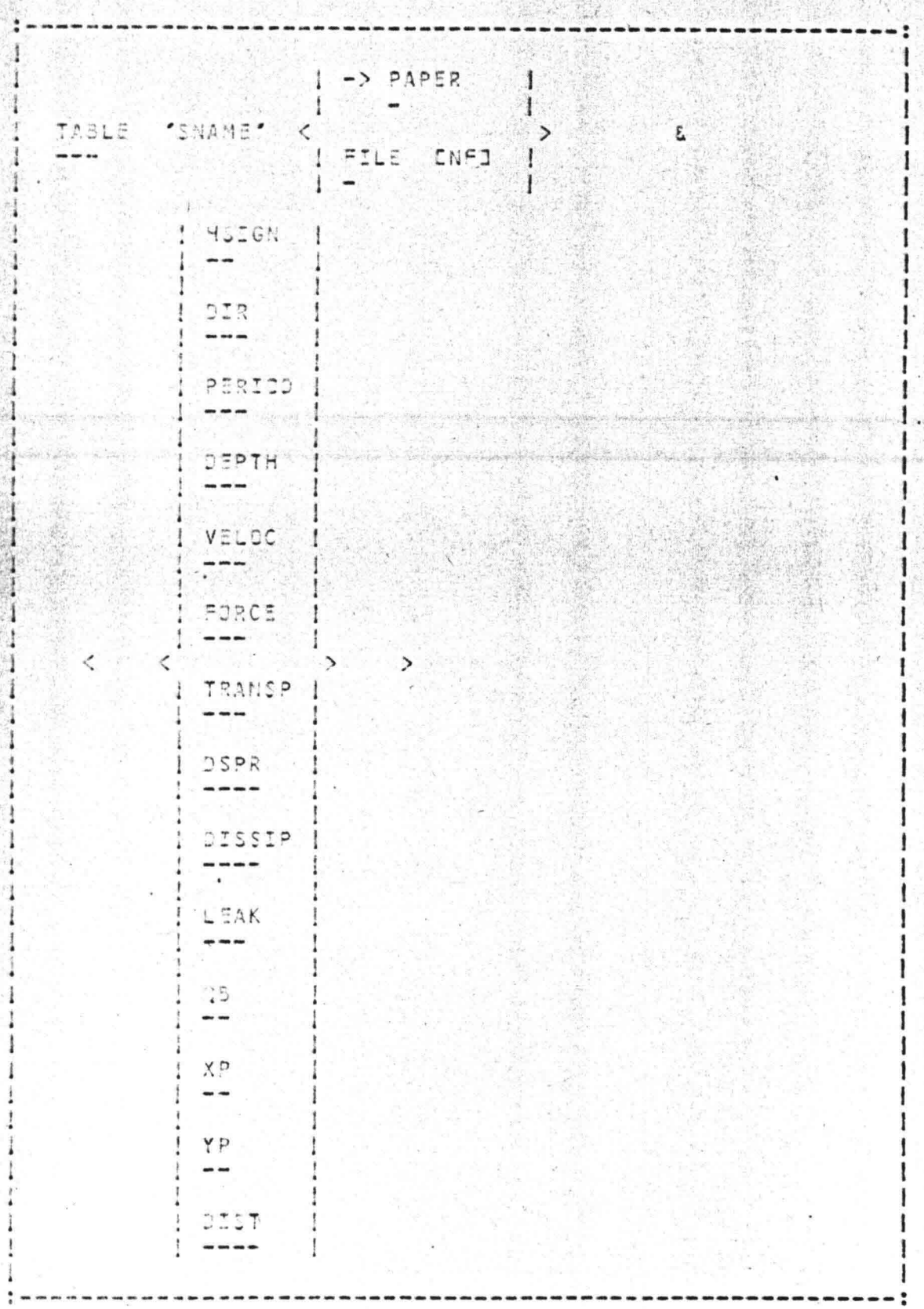
printed. The number printed is the value of the variable divided by [UNIT]. Choose [UNIT] small enough so that a sufficient number of significant digits remains. For each integer 5 places are available. The above does not apply in the case of output to a file. In that case the number are written to the file in real format. Default: [UNIT]=1.

Alternatief: Als [UNIT] ontbreekt, stelt het programma zelf een waarde vast.

The block print can be made for several different variables:

HSIGN the significant wave height,
PERIOD the mean wave period
DIR the mean wave direction (in degrees, measured counterclockwise from the x-axis of the output frame); this is the direction normal to the wave crests; note that, if currents are present, this direction is different from the energy transport direction.
DSPR the directional spread of the waves (in degrees),
DEPTH the depth,
VEL the current velocity, both the x- and the y-component with respect to the frame coordinate system are printed.
FORCE the radiation stress gradient, which is equal to the resulting force exerted by the waves per unit surface; both the x- and the y-component with respect to the frame coordinate system are printed.
TRANSP the energy transport vector, both the x- and the y-component with respect to the frame coordinate system are printed.
TRY the energy transport in y-direction,
DISSIP the energy dissipation,
LEAK the leakage of energy over the sector boundaries,
QB the fraction of breaking waves (a parameter in the surf breaking formula).

Alternatief: vectoriele grootheden niet weergeven t.o.v. frame-coordinaten, maar t.o.v. user coordinates.



The output module of HISWA is instructed to produce a table. The table can either be made on paper, or on a file for

further processing. In the latter case [NF] is the file reference number.

The table is made for the set of output points denoted by the point set name 'SNAME'. This point set can be of any type. For each output point all different variables are written in a row.

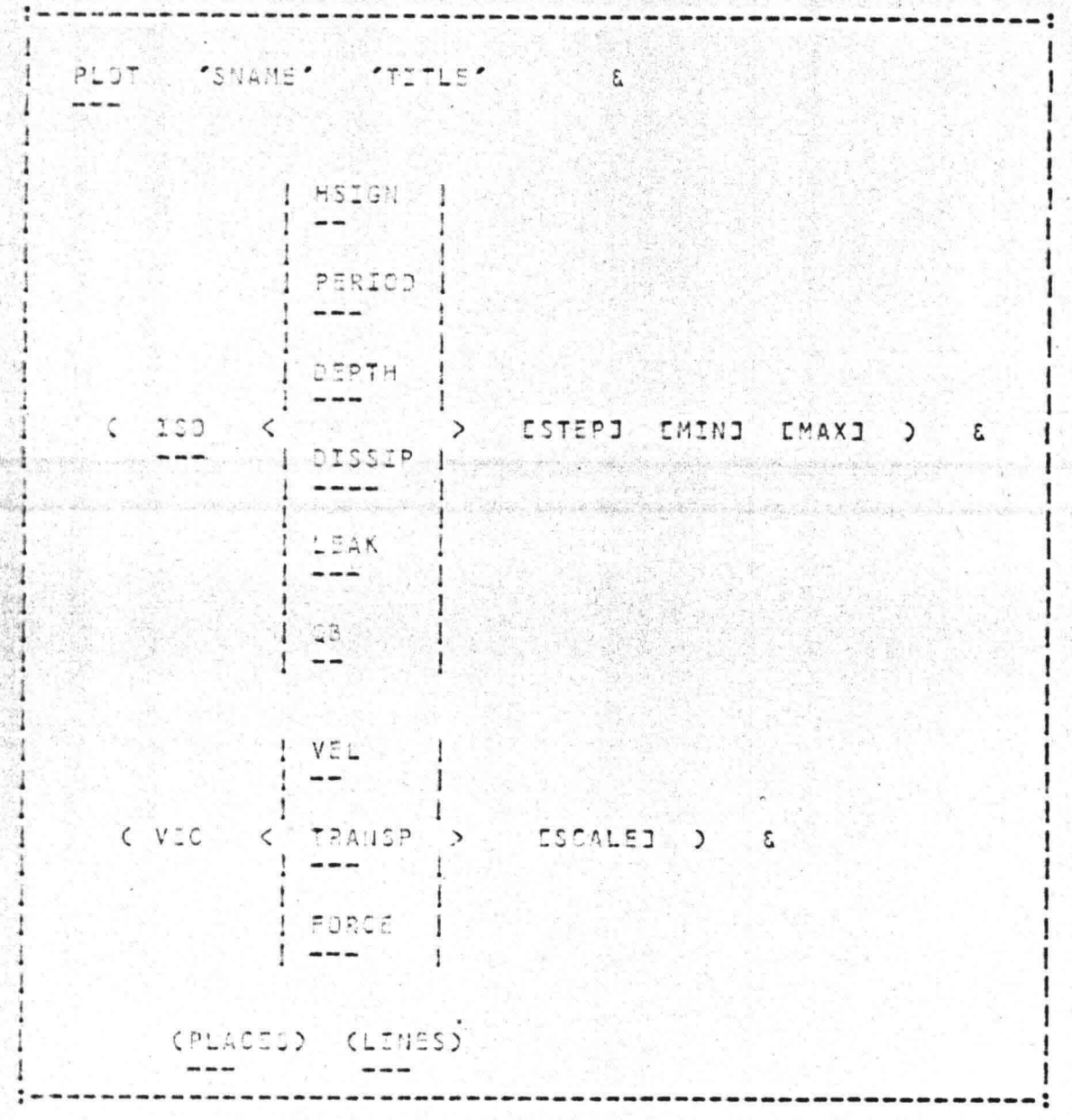
The table can be made for several different variables; most of these are already described with the command BLOCK. The vectorial quantities VELOCITY, FORCE and TRANSPORT are now given with respect to the user coordinate system.
Some variables are added, which indicate the output locations, and which might be needed for further processing, viz.:

XP problem x-coordinate of the output point
YP problem y-coordinate of the output point
DIST distance along an output curve (only useful if the point set is of type CURVE). For the first point the distance is 0.

PAPER [FAC] [WIDTH]

This command provides information about the plotting paper to be used.

[FAC] is an enlargement factor for the figures made by the program. In the preparation phase one can choose [FAC] smaller than 1 in order to save plot and computer cost. Init: [FAC]=1.
[WIDTH] is the paper width to be used for the figures.



The output module of HISWA is instructed to produce a figure containing a contour line plot and/or a vector plot.

The plot is made for the set of output points denoted by the point set name 'SNAME'. This point set must be of the type FRAME. Default: SNAME='BOTTGRID'.

The 'TITLE' is plotted with the figure. If no title is given by the user, the program will generate a title.

For one scalar variable a contour plot is made, if the part ISO ... is present. The variable is characterized by a keyword (HSIGN, PERIOD, DEPTH etc.). In the description of the command BLOCK it is described which keyword indicates which physical variable.

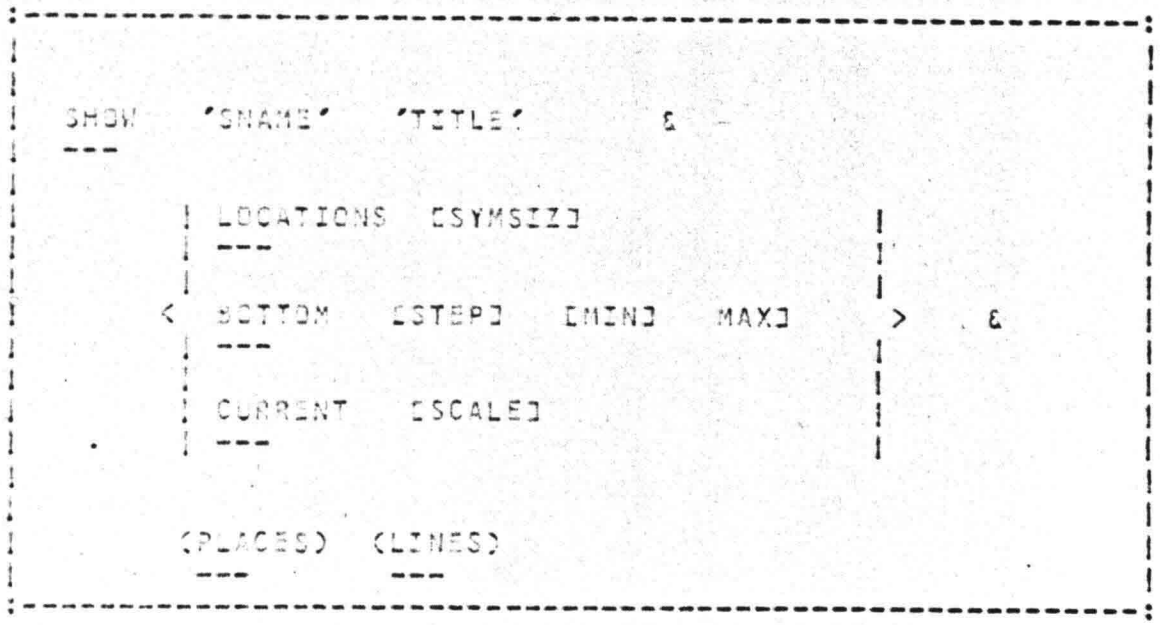
For the contour plot one should prescribe: [STEP], the difference in value for two neighbouring contours (default: 1.), [MIN], the minimum value for which a contour is made (default: 0.), and [MAX], the maximum value for which a contour is drawn (default: 10*[STEP]).

Alternatief: als [STEP] niet gespecificeerd is, dan zal het programma voor [MIN] de minimale functiewaarde neme, als deze ook niet gespecificeerd is, en voor [MAX] de maximale functiewaarde, eveneens indien die niet gespecificeerd is. Vor [STEP] zal het programma vervolgens nemen: ([MAX]-[MIN])/10.

A vector plot can be made for one of the following variables: current velocity (keyword VEL), energy transport (TRANSP), or the radiation stress gradient (FORCE).
The user must prescribe [SCALE], this is the scale to which the vectors are plotted. [SCALE] is in cm (on paper) per unit of the physical variable (for instance m/s in the case VEL).

Alternatief: als [SCALE] niet gespecificeerd is, maakt het programma de waarde zo dat de pijltjes niet langer worden dan de roosterafstand.

If the keyword PLA is present, names of places (defined in the command PLA) will be plotted in the figure. If the keyword LIN is present, lines (defined in the command LIN) will be plotted in the figure.



This command produces figures before the computation is carried out. These figures serve to facilitate the verification of input data.

The plot is made for the set of output points denoted by the point set name 'SNAME'. This point set must be of the type FRAME. Default: SNAME='BOTTGRID'.

The 'TITLE' is plotted with the figure. If no title is given by the user, the program will generate a title.

SHOW LOCATIONS : the location of output frames, curves and points is shown in the figure. [SYMSIZ] is the size of the symbols on the plot, in cm. Default: SYMSIZ=0.28 .

SHOW BOTTOM : iso-lines of the bottom level are shown; for [STEP], [MIN] and [MAX] see command PLOT.

SHOW CURRENT : a vector plot of the current velocity is made; for [SCALE] see command PLOT.

PLACES, LINES: see command PLOT.

```
:-------------------------------------------------------------:
!                                                             !
!  SPEC 'SNAME' [NF]                                          !
!  ----                                                       !
!                                                             !
:-------------------------------------------------------------:
```

Spectra are written onto the file with reference number [NF]. These spectra can be used as input with the command INC (incident waves) or BOU (boundary conditions). This facility is useful in the case of nesting of models, or if one wants to restart a job.

Note: this facility is not yet available.

Appendix II List of subroutines.

Name	used by program(s)		
	PREP	COMPU	OUTP
ADDURI	—		
ADDURR	—		
BRTRAF	—		
COPYS	—		
GETKAR	—		
INBOPD	—		
INCHAR	—		
ININT	—		
INKEY	—		
INREAL	—		
INREKP	—		
INUPT	—		
IRAAI	—		
LEESEL	—		
NEWUR	—		
NWLINE	—		
ODALPR	—		
ODAPT	—		
ODDELP	—		
ODDELR	—		
ODEXP	—		
ODSHIF	—		
PLODEP	—		
PLOSIT	—		
PLOTX	—		
PLSTRM	—		
PLTKOP	—		
PLYN	—		
PNAMES	—		
PRCON	—		
PROUT	—		
REDEP	—		
REINVA	—		
SRTVAR	—		
STARTW	—		

BOCUR		—	
DIFT		—	
DISPA		—	
FRABRE		—	
ITWN		—	
NUMSC		—	
PREDT		—	
SBOT		—	
SDBR		—	
SSURF		—	
STARTB		—	
SWIND		—	
SUMDE		—	
TERMD		—	
TRSY		—	
TRST		—	
VWPRO		—	
WAVPA		—	
WRIRE		—	
BLKHFD			—
BLKPT			—
BLOKX			—
G1			—
G2			—
ISOVEC			—
IXPOL			—
ODGETR			—
PSET			—
TABPRT			—
TOGPCP			—
UVDEP			—
UVDIR			—
UVDISS			—
UVDIST			—
UVDSPR			—
UVENX			—
UVENY			—
UVFRX			—
UVFRY			—
UVHSIG			—

UVIPOL			—
UVLEAK			—
UVPER			—
UVQB			—
UVUX			—
UVUY			—
UVXP			—
UVYP			—
ADPOOL	—	—	—
COPYCH	—	—	—
DISTR	—	—	—
DIVKOP	—	—	—
INKAD	—	—	—
INPDC	—	—	—
INPOOL	—	—	—
KARSEL	—	—	—
KOMPAR	—	—	—
MSGERR	—	—	—
ODGET	—	—	—
ODLOC	—	—	—
OPENF	—	—	—
PCOAST	—	—	—
PENTO	—	—	—
PLNAME	—	—	—
SNYPT1	—	—	—
SNYPT2	—	—	—
STRACE	—	—	—
VERSIE	—	—	—
WRCOM	—	—	—
WRCOMX	—	—	—
WRDUMP	—	—	—
WRPOOL	—	—	—

Appendix III System Documentation of COMPU

HISWA system documentation

program COMPU

date of printing: 19 September 1984

by : T.H.C. Herbers
N. Booij
L.H. Holthuijsen

TABLE OF CONTENTS

1. INTRODUCTION

1.1 General characteristics of the model

In this document the system documentation of a numerical shallow water waves hindcast model named HISWA is presented. This model is expected to provide realistic estimates of the wave conditions in the Oosterschelde. It is a directionally decoupled parametric model containing bottom refraction, wave growth, dissipation due to wave breaking (surf zone) and bottom friction as well as a simple representation of diffraction effects. Further the effects of currents on refraction, wind generation, bottom friction and wave breaking in deep water is included.
For the mathematical formulation of this model reference is made to Holthuijsen and Booij (1983).
Two balance equations in the parameters A0 (frequency integrated wave action) and W0 (mean wave action frequency), containing gradients in three dimensions x, y and θ (wave direction), are solved :

$$\frac{\partial}{\partial X}(CX0 \cdot A0) + \frac{\partial}{\partial Y}(CY0 \cdot A0) + \frac{\partial}{\partial \theta}(C\theta 0 \cdot A0 + Cdif(CY0 \cdot \frac{\partial}{\partial X} A0 - CX0 \cdot \frac{\partial}{\partial Y} A0)) = \frac{S0}{\sigma 0} - \frac{A0}{W0} \cdot \frac{d}{dT} W0 \quad (1)$$

$$\frac{\partial}{\partial X}(CX0 \cdot W0 \cdot A0) + \frac{\partial}{\partial Y}(CY0 \cdot W0 \cdot A0) + \frac{\partial}{\partial \theta}(C\theta 0 \cdot W0 \cdot A0 + Cdif(CY0 \cdot \frac{\partial}{\partial X}(W0 \cdot A0) - CX0 \cdot \frac{\partial}{\partial Y}(W0 \cdot A0))) = \frac{W0 \cdot S0}{\sigma 0} \quad (2)$$

in eq. 1 and 2 :

CX0, CY0, Cθ0 are the components in X, Y resp. θ direction of wave action transport velocity

σ0 is the relative frequency

S0 is the source term including wind generation, bottom friction and wave breaking (surf zone and deep water)

Cdif is a diffraction coefficient

(expressions for these terms are given in chapter 3)

A numerical grid is defined in three dimensions x, y and θ (fig. 1). The direction of wave propagation θ is defined as the angle between the wave number vector and the positive x-axis, measured counter-clockwise.

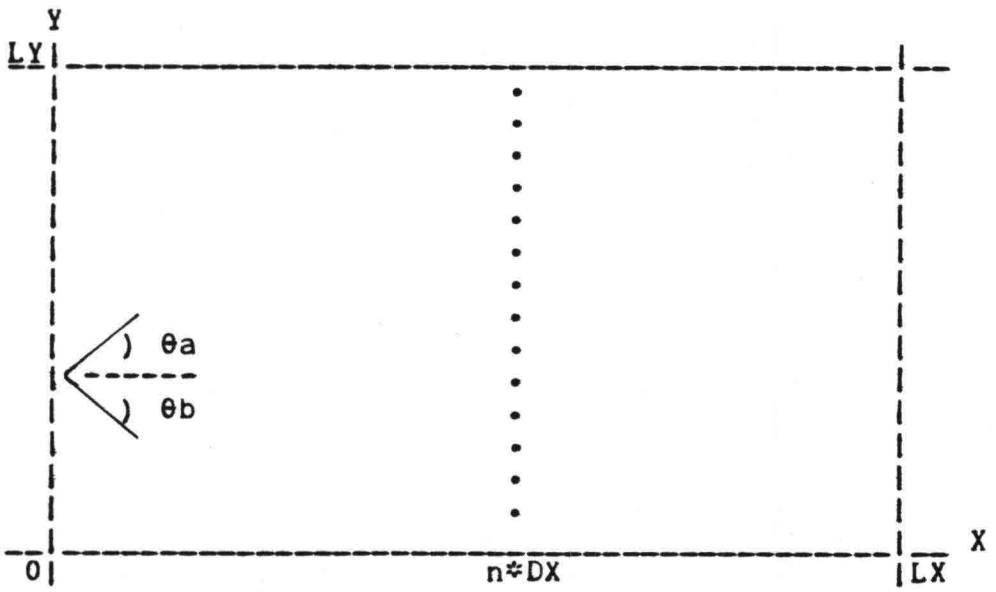


fig. 1 the computational region

The computation progresses in the positive x-direction and propagation of wave energy is limited to a sector defined by θa and θb around the direction of the x-axis. The computations are carried out line by line with an explicit predictor-corrector scheme. The number of corrector steps is free but two steps are sufficient to obtain a stable scheme. Lines are defined parallel to the y and θ axis.

Beside the computational grid described above two other grids are used in the model HISWA :

- a problem grid in which the user defines his problem (in x-y plane)
- a bottom grid containing the bottom topography and current field (in x-y plane)

1.2 Computer programs

The model HISWA consists of three computer programs :

PREP input preparation and control part
COMPU computational part
OUTP output of results

Here the the computational part COMPU will be considered. The programs PREP and OUTP are adjusted versions of the programs PREP and UITV of the refraction/diffraction model CREDIZ of Rijkswaterstaat.

INPUT :
In the program COMPU instructions, definitions of grids, coefficients etc. (formulated in PREP) and arrays containing bottom topography and current field are read from a file.
PROCEDURE :
First the values of the frequency integrated action A0 and

mean action frequency W0 are determined at the boundary x=0. Further depths, currents, wave numbers and wave propagation velocity components are computed at this boundary. For every new line (x=n*dx) A0 and W0 are obtained through the application of the numerical scheme described in section 1.1.
OUTPUT :
For every line the following results are written to a file that will be read by the program OUTP :

wave action, -frequency, relative frequency, group velocity, wave number and components of wave action transport velocity (in every grid point in the y-θ plane)

leakage of energy through the θ boundary (for every value of y)

dissipation of energy due to bottom friction, surf- and deep water breaking (for every value of y)

the fraction of breaking waves (for every value of y)

1.3 Documentation

A description of the computer program COMPU is given in chapters 2 through 4.
In chapter 2 the structure of the program COMPU is explained. The relations between the subroutines in COMPU are shown in block diagrams. An example of a block diagram is given below.

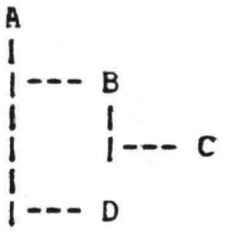


fig. 2 block diagram

In the program or subrutine A, subroutines B and D are called. Further subroutine C is called in subroutine B.
The structure of a program or subroutine is presented in Nassi-Schneidermann diagrams. For convenience the conventional construction in the left part of fig. 3 is replaced by the one on the right.

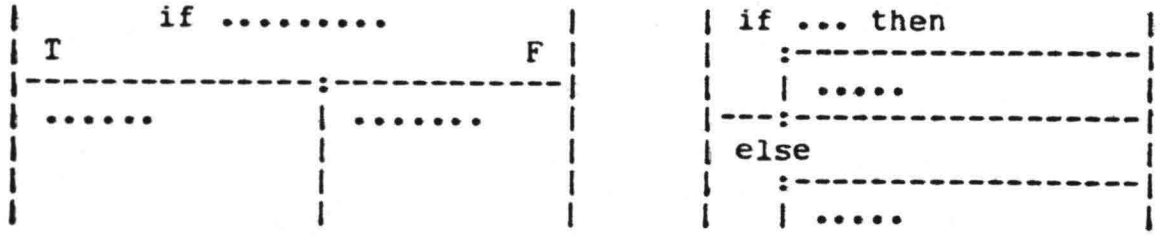


fig. 3 representation of a conditional statement

Descriptions of the various subroutines are given in chapter 3. The sequence in which the subroutines are discussed corresponds with the place in the program COMPU at which the subroutine is called for the first time. Parameter lists of the subroutines are described in which input- and output parameters are denoted by (I) resp. (O).
In chapter 4 the storage of variables and arrays in common blocks and files is described. A flexible handling of computer storage, necessary for the considerable number of arrays in COMPU is obtained through the application of a Dynamic Data Pool.
In this report reference is made to the system documentation of CREDIZ for a detailed description, in so far as subroutines and other facilities of the model CREDIZ are implemented in the present model.

2. STRUCTURE OF THE PROGRAM COMPU

The computational part COMPU of the model HISWA is a FORTRAN program consisting of a main program and various subroutines. Input is read from a file named INSTR and output is written to a file named REKRES (section 4.3). Fig. 1 shows a diagram of the main program.

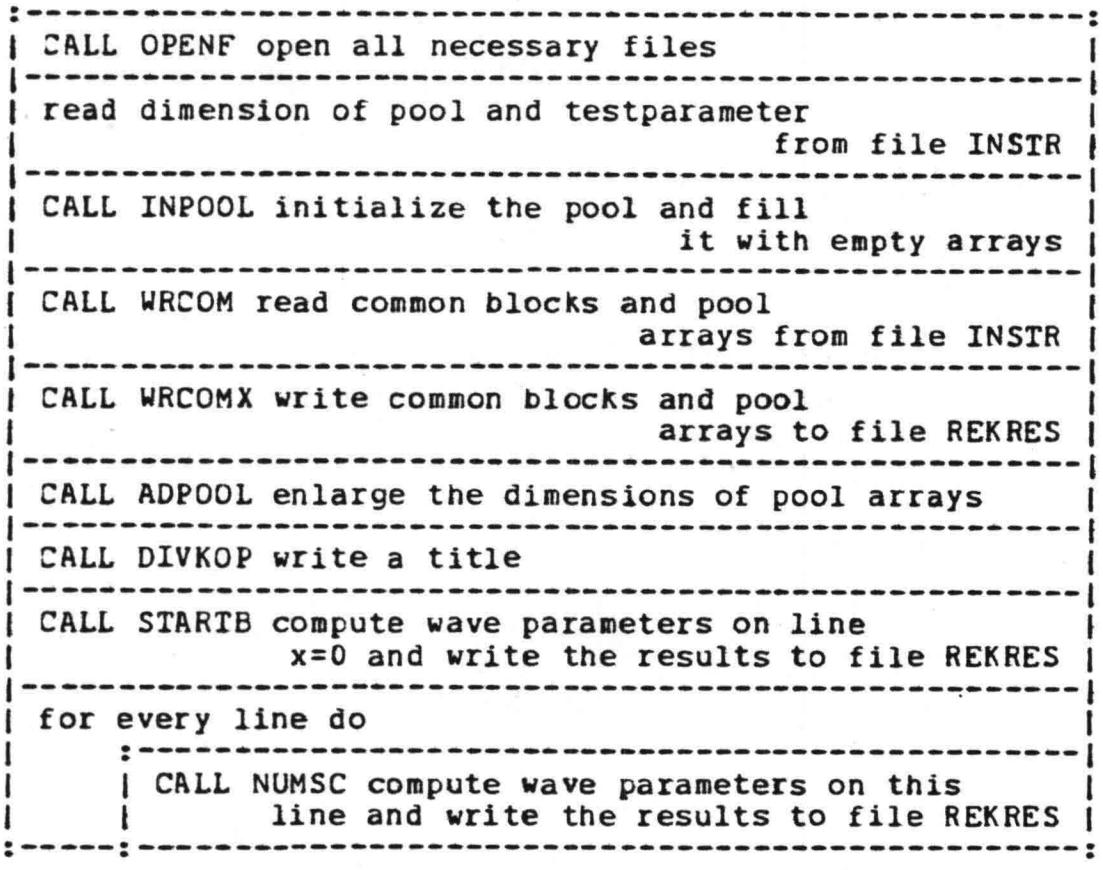


fig. 1 diagram of the main program

A block diagram showing the relations between the various subroutines of COMPU is given in fig. 2. Separate diagrams for the subroutines WRCOM, WRCOMX, WAVPA and TERMD are included in fig. 3 through 6. The subroutines STRACE, COPYCH and MSGERR are called in various parts of the program.

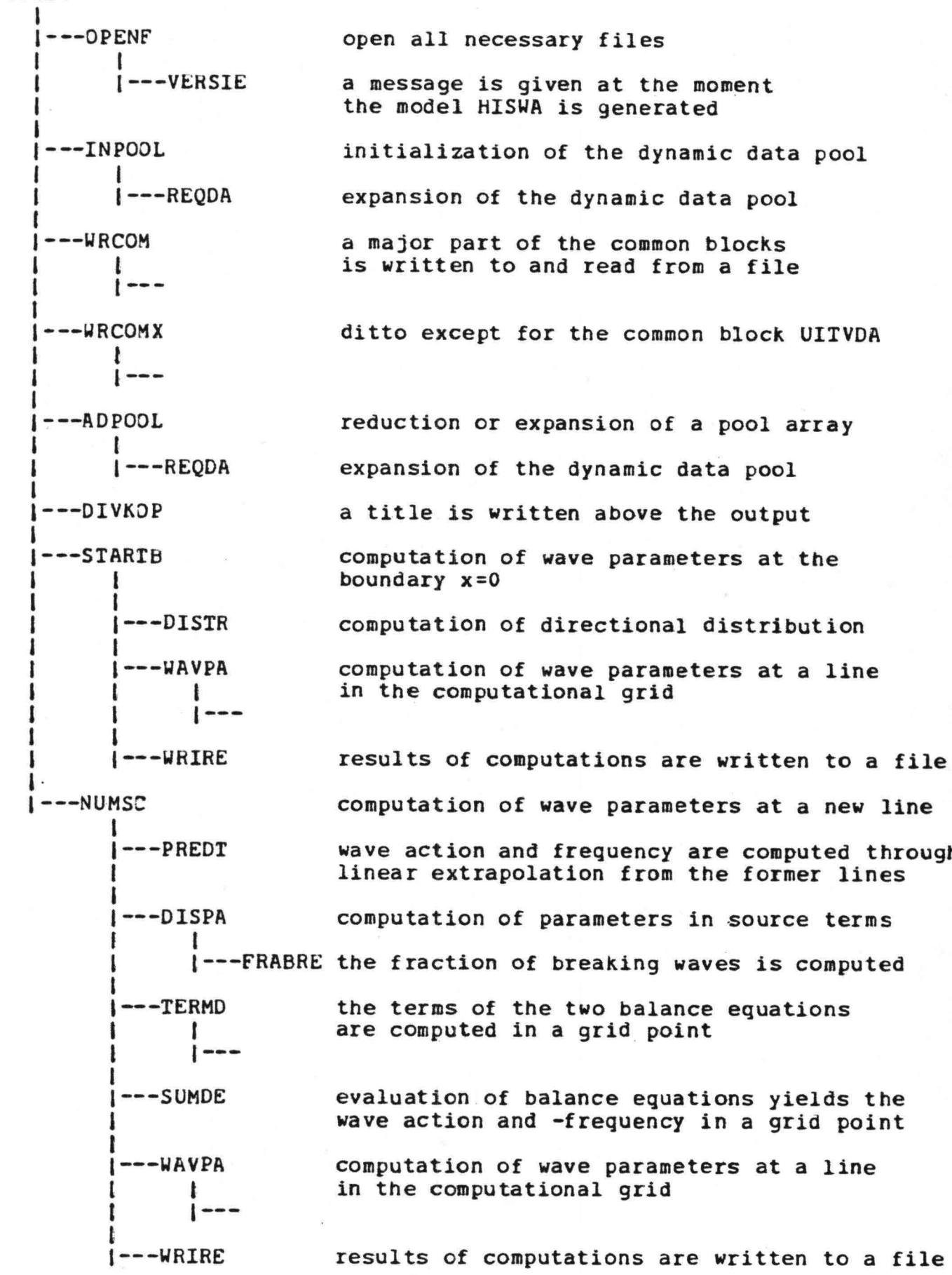


fig. 2 relations between subroutines in COMPU

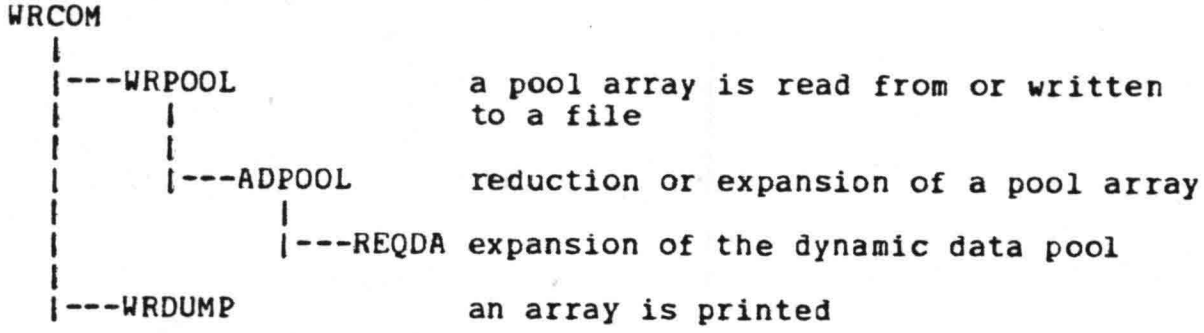


fig. 3 relations between subroutines in WRCOM

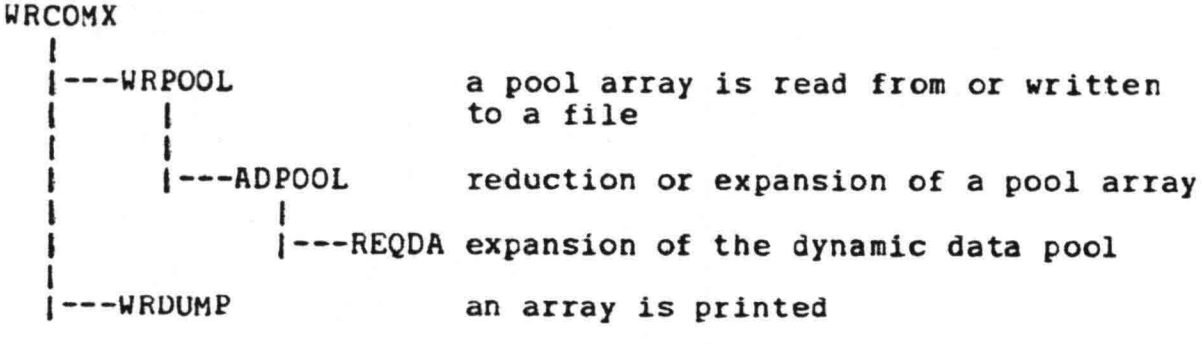


fig. 4 relations between subroutines in WRCOMX

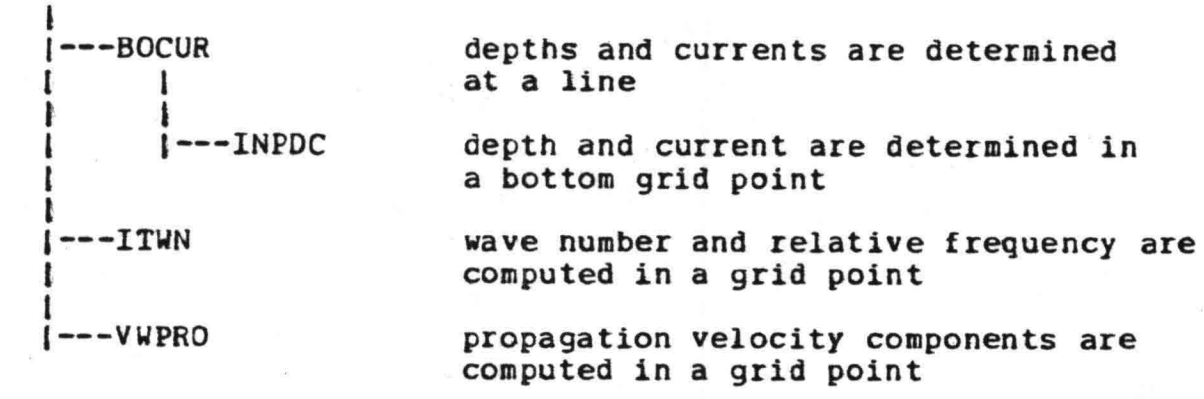


fig. 5 relations between subroutines in WAVPA

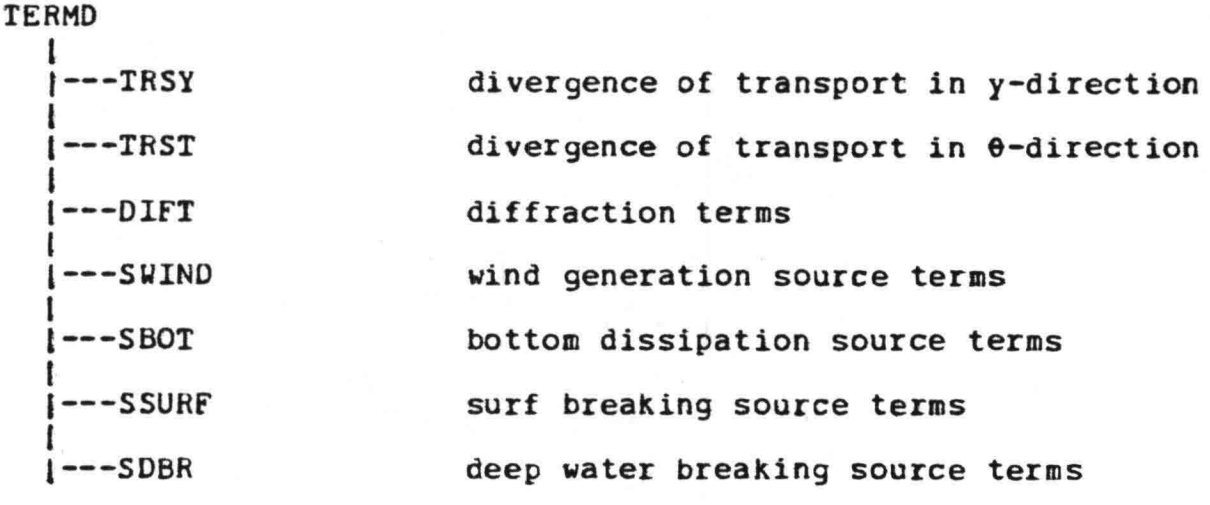


fig. 6 relations between subroutines in TERMDE

3. DESCRIPTION OF SUBROUTINES

In this chapter the subroutines of the program COMPU are described. A number of these subroutines is copied from the model CREDIZ with a few adjustments in the source text. Only a short description is given of the function of these subroutines and reference is made to the documentation of CREDIZ.

3.1 Subroutine OPENF

In this subroutine all necessary files are opened in order to reserve input/output buffers. This action is taken in connection to repeated calls of the standard routine REQDA. This subroutine is copied from CREDIZ.

3.2 Subroutine VERSIE

A message is printed at the moment (time and date) the model HISWA is generated.
This subroutine is copied from CREDIZ.

3.3 Subroutine INPOOL

The dynamic data pool is initialized by this subroutine.The dimension of the pool is determined from the common variable NPOOL (NPOOL * 1024). A number of empty arrays is initiated (50 in he program COMPU).
INPOOL is copied from CREDIZ with minor adjustments.

3.4 Subroutine REQDA

The standard routine REQDA,copied from CREDIZ, is used for expansion of the dynamic data pool.

3.5 Subroutines WRCOM and WRCOMX

A major part of the common blocks is written to and read from a file by the subroutines WRCOM and WRCOMX. This is necessary for the communication between the programs PREP, COMPU and OUTP. The difference between WRCOM and WRCOMX is the fact that WRCOMX doesn't read or write the common block UITVDA, containing instructions and information for the program OUTP.
WRCOM and WRCOMX are copied from CREDIZ with minor adjustments.

3.6 Subroutine WRPOOL

The subroutine WRPOOL reads or writes a pool array (unformatted) from resp. to a file.
WRPOOL is copied from CREDIZ with minor adjustments.

3.7 Subroutine ADPOOL

This routine is called by WRCOM,WRCOMX and in the main program for shrinking or expansion of an array in the dynamic data pool.
Subroutine ADPOOL is copied from CREDIZ with a few adjustments.

3.8 Subroutine WRDUMP

The contents of an array is printed by the subroutine WRDUMP.
WRDUMP is copied from CREDIZ with minor adjustments.

3.9 Subroutine DIVKOP

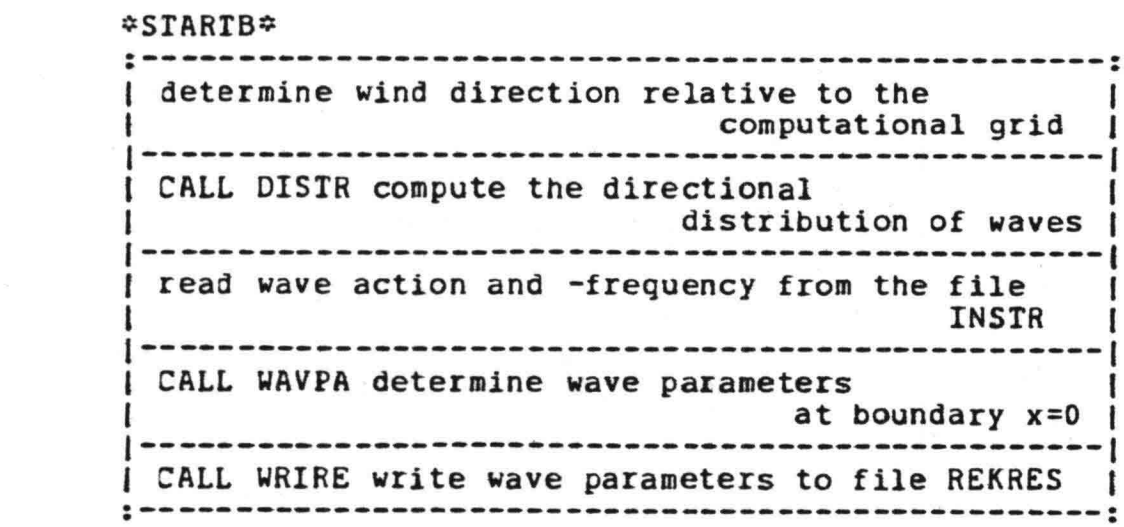
Function:
A title is printed above the output of the program COMPU.
DIVKOP is copied from CREDIZ.

3.10 Subroutine STARTB

Function :
In the subroutine STARTB the wave conditions at the boundary x=0 are determined. Further the directional distribution of waves in an ideal wind field with given wind direction is determined.

Method :
The wave action A0(y,θ) and -frequency W0(y,θ) are read from the file INSTR.
Depths and currents are determined at the boundary x=0. Wave number, relative frequency, group velocity and components of propagation velocities are computed in every grid point in the Y-θ plane.
Next these parameters are written to the file REKRES.

Structure :



The call of this subroutine is :

```
CALL STARTB
```

3.11 Subroutine DISTR

Function :
In subroutine DISTR the directional distribution of waves is computed.

Method :
For the directional distribution a cos**COEF distribution is chosen. The parameter COEF is supplied by the user.

$$B(\theta) = c.\cos^{COEF}(\theta - DIR).A(Y) \quad \text{for } |\theta - v| < 90 \text{ deg.}$$
$$= 0 \quad \text{for } |\theta - v| \geq 90 \text{ deg.} \qquad (1)$$

(DIR is the mean wave direction)

The call of this subroutine is :
CALL DISTR(DIR,COEF,NORM)

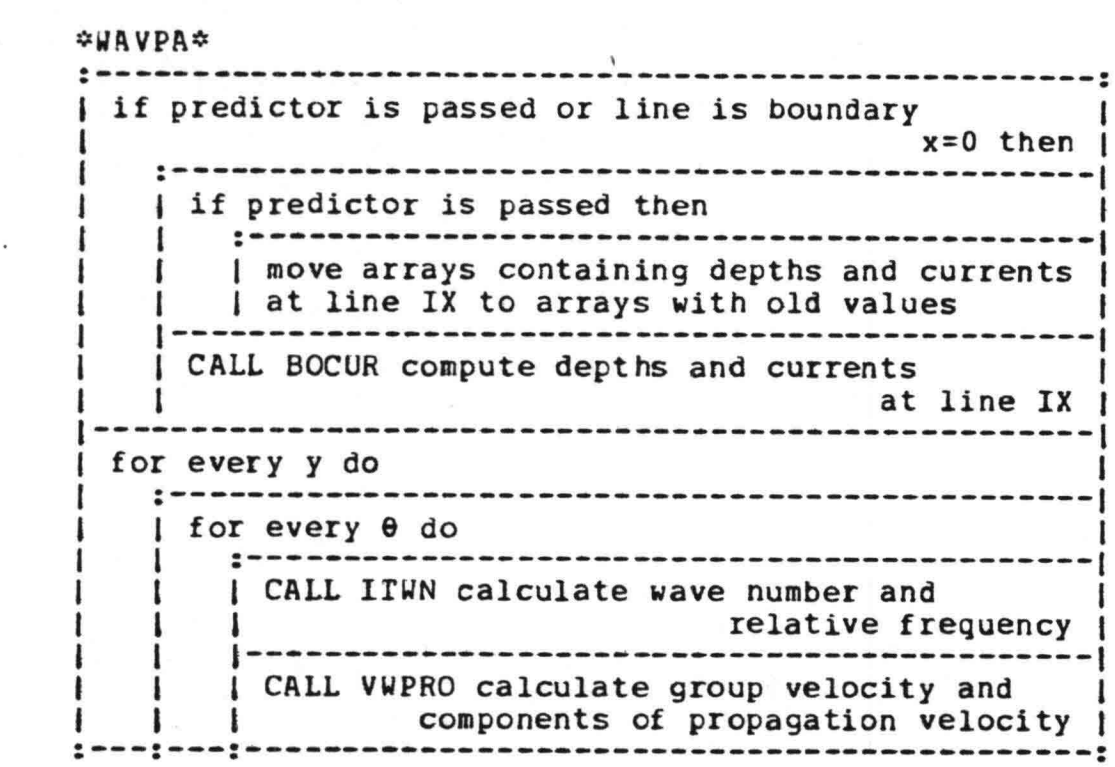
Parameters :
DIR (I) mean wave direction
COEF (I) parameter of directional distribution
NORM (I) = 1 the normalization coeff. c is computed
= 0 c is assumed to be known

3.12 Subroutine WAVPA

Function :
In this subroutine wave numbers K0, relative frequency 00, group velocity CG0 and propagation velocity components CX0, CY0 and Cθ0 are computed at a line IX in the computational grid.

Method :
In order to evaluate these parameters first the depths D and current velocity components UX, UY at line IX are determined.

Structure :



The call of this subroutine is :

```
CALL WAVPA(IX)
```

parameter :

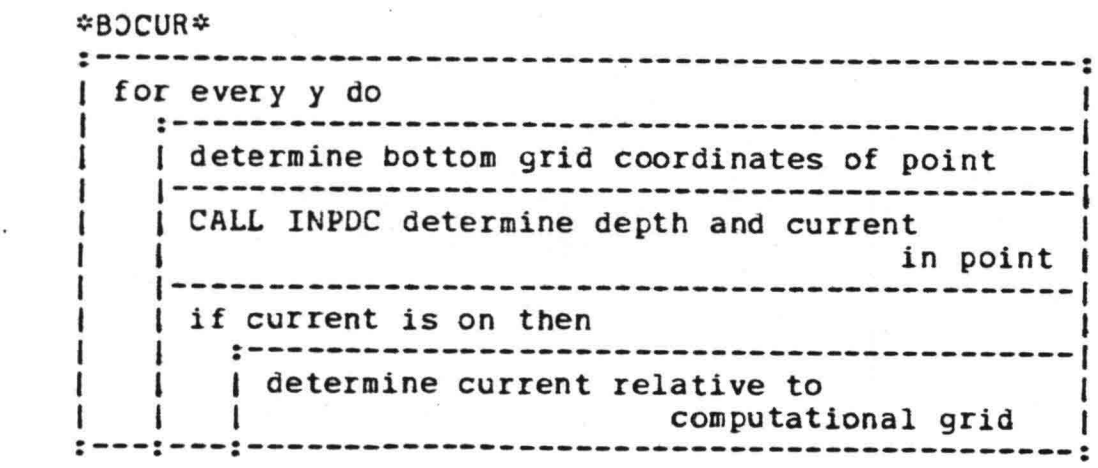
IX (I) line in computational grid at which wave parameters are computed (X=(IX-1)DX)

3.13 Subroutine BOCUR

Function :
In this subroutine depths D(Y) and current velocity components UX(Y), UY(Y) are determined at a line in the computational grid.

Method :
For every grid point, bottom grid coordinates are computed and depth and current velocity components are determined through bilinear interpolation.

Structure :



The call of this subroutine is :
CALL BOCUR(IX)

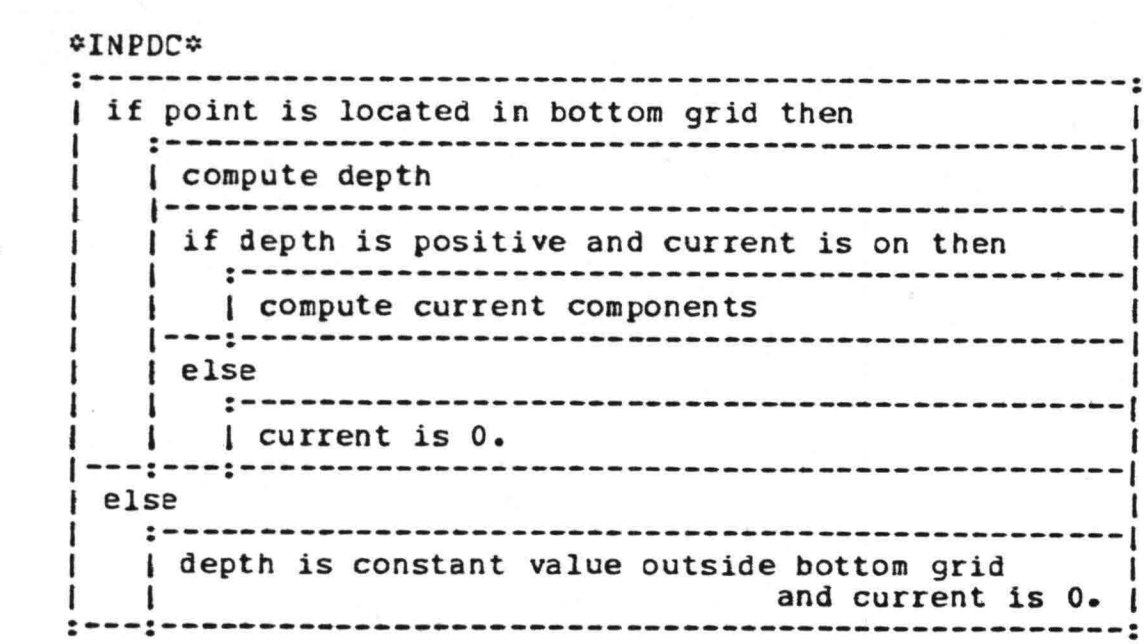
parameter :
IX (I) line in computational grid at which depths and currents are computed (X=(IX-1).DX)

3.14 Subroutine INPDC

Function :
Depth and current velocity components are computed in a point given in bottom grid coordinates (IB,JB).

Method :
A bilinear interpolation is carried out with the surrounding points in the bottom grid. If point (IB,JB) is located outside the bottom grid then a constant depth and no current is assumed.

Structure :



The call of this subroutine is :
CALL INPDC

3.15 Subroutine ITWN

Function :
In ITWN the wave number K0 and relative frequency 00 in a grid point (IX,IY,IT) is determined.

Method :
The current component in the direction of wave propagation is determined.

$$U = Ux.\cos\theta + Uy.\sin\theta \qquad (1)$$

If U > 0 then the wave number K0 is computed through a Newton-Raphson iteration process, applied to eq. 2.

$$F \equiv W0 - K0(UX.\cos\theta + UY.\sin\theta) - (g.K0.\tanh(K0.D))^{1/2} = 0 \qquad (2)$$

This procedure requires an estimate of the wave number K0 as a start value. Here the value of K0 on line IX-1 is used as an estimate for K0. At the boundary x=0 the following approximation of K0 is applied.

$$\frac{g.K0}{W0^2} = \frac{1}{(\tanh(W0^2.D/g))^{1/2}} \qquad (3)$$

The relative frequency $\sigma 0$ is calculated with eq. 4.

$$\sigma 0 = (g.K0.\tanh(K0.D))^{1/2} \qquad (4)$$

If $U < 0$ first the frequency Wc is determined which is the highest frequency capable to transport wave energy against the current U. For this purpose eq. 5 is solved through a Newton-Raphson iteration process.

$$G \equiv U + \sigma 1.\left(\frac{1}{2.K1} + \frac{D}{\sinh(2.K1.D)} \right) = 0 \qquad (5)$$

with

$$\sigma 1 = (g.K1.\tanh(K1.D))^{1/2} \qquad (6)$$

Wc is given by

$$Wc = K1.U + \sigma 1 \qquad (7)$$

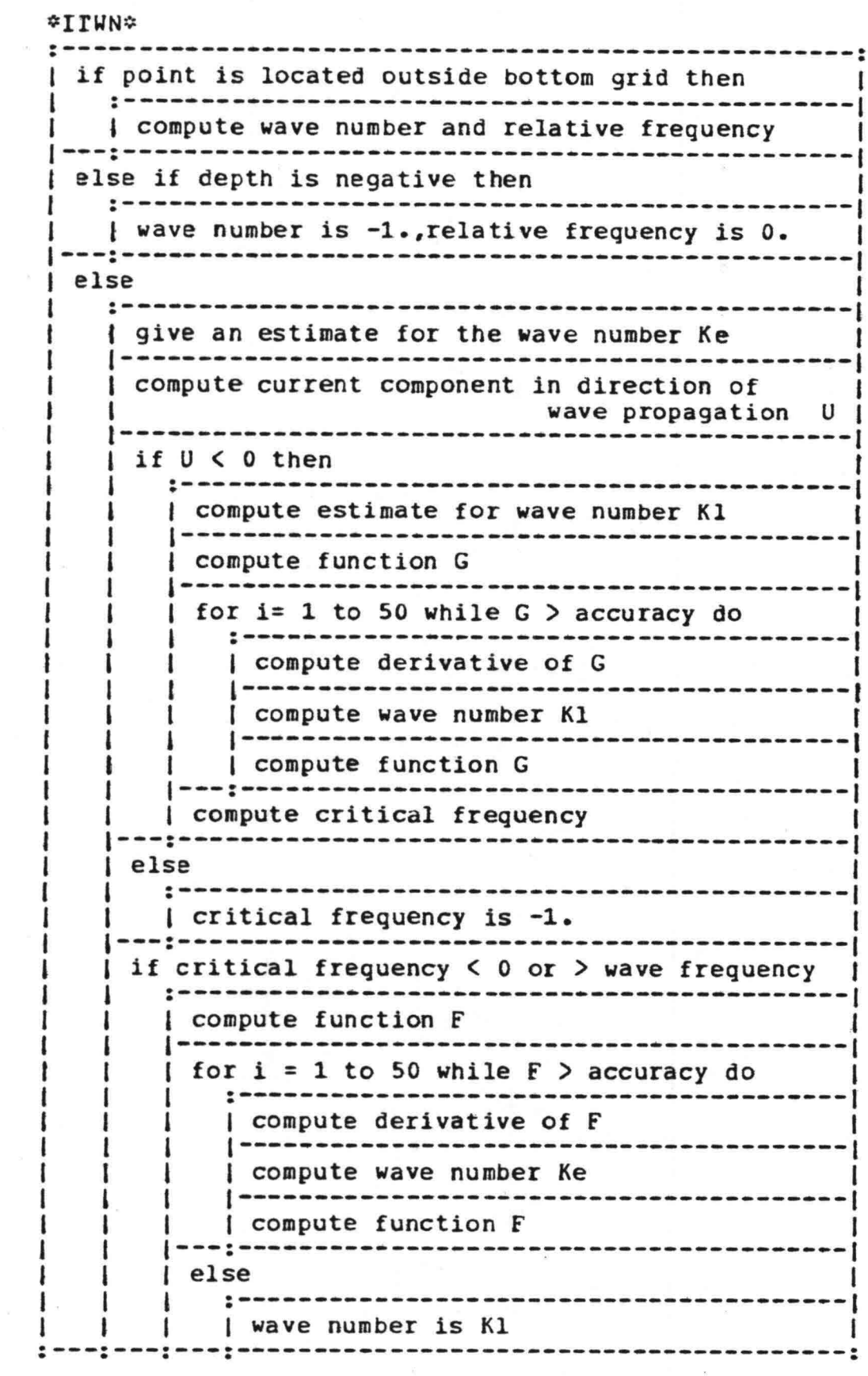
If $W0 < Wc$ (eq. 2 has a solution) then K0 and $\sigma 0$ are computed with the procedure described above. If $W0 > Wc$ (eq. 2 has no solution) then K1 and $\sigma 1$ are used as estimates of K0 resp. $\sigma 0$.

A source term Sdbr is introduced to dissipate wave components with frequency > Wc (section 3.28).
For points outside the bottom grid K0 is computed with eq. 8.

$$K0 = W0^2/g \qquad (8)$$

If a negative depth is encountered then K0 and $\sigma 0$ are given the values -1.0 resp. 0.0.

Structure :


ITWN
if point is located outside bottom grid then
compute wave number and relative frequency
else if depth is negative then
wave number is -1.,relative frequency is 0.
else
give an estimate for the wave number Ke
compute current component in direction of wave propagation U
if U < 0 then
compute estimate for wave number K1
compute function G
for i= 1 to 50 while G > accuracy do
compute derivative of G
compute wave number K1
compute function G
compute critical frequency
else
critical frequency is -1.
if critical frequency < 0 or > wave frequency
compute function F
for i = 1 to 50 while F > accuracy do
compute derivative of F
compute wave number Ke
compute function F
else
wave number is K1

The call of this subroutine is :

```
CALL ITWN(IX,IY,IT)
```

Parameters :

IX (I)
IY (I)) coordinates of point in computational grid
IT (I)

3.16 Subroutine VWPRO

Function :
In subroutine VWPRO the group velocity CG0 and the components of wave action transport velocity CX0, CY0 and Cθ0 are determined in a point (IX,IY,IT) in the computational grid.

Method :
The relations for the parameters mentioned above used in this model are :

$$CG0 = \sigma 0\left(\frac{1}{2.K0} + \frac{D}{\sinh(2.K0.D)}\right) \qquad (1)$$

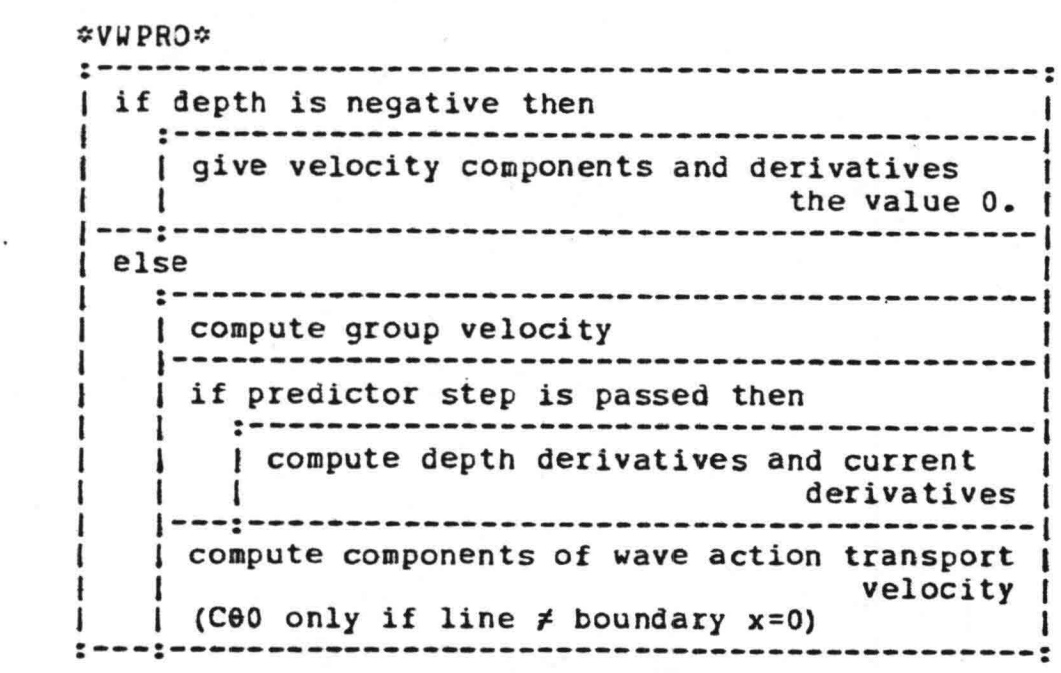
$$CX0 = CG0.\cos\theta + UX \qquad (2)$$

$$CY0 = CG0.\sin\theta + UY \qquad (3)$$

$$C\theta 0 = -\frac{\sigma 0}{\sinh(2.K0.D)}\left(-\sin\theta\frac{\partial D}{\partial X} + \cos\theta\frac{\partial D}{\partial Y}\right) - \cos\theta\left(-\sin\theta\frac{\partial UX}{\partial X} + \cos\theta\frac{\partial UX}{\partial Y}\right) - \sin\theta\left(-\sin\theta\frac{\partial UY}{\partial X} + \cos\theta\frac{\partial UY}{\partial Y}\right) \qquad (4)$$

The terms containing current velocity components UX and UY in eq. 2 through 4 are omitted if no current is present.
The term Cθ0 is evaluated intermediate lines IX and IX+1.
Derivatives of depth and current are determined through a central difference scheme (after the predictor step).
If negative depths are encountered then all velocity components are given the value 0..

Structure :



The call of this subroutine is :
CALL VWPRO(IX,IY,IT)

Parameters :
IX (I)
IY (I) coordinates of point in computational grid
IT (I)

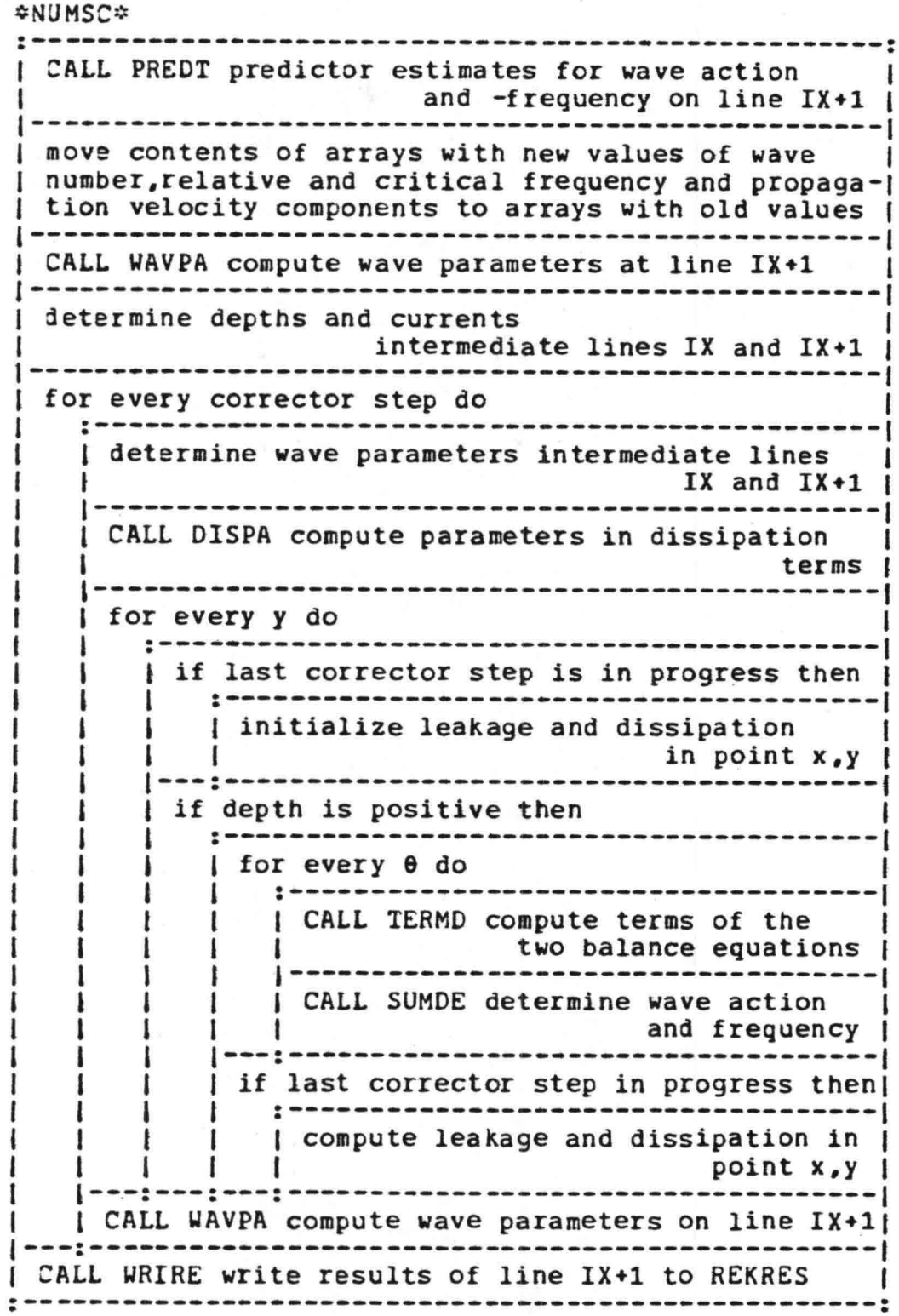
3.17 Subroutine NUMSC

Function :
In the subroutine NUMSC wave parameters are computed at a new line IX+1 in the computational grid.

Method :
The following numerical scheme is applied :
Estimates for the wave action A0 and -frequency W0 at line IX+1 are obtained through a linear extrapolation from the lines IX-1 and IX (predictor step). With these estimates the other wave parameters at line IX+1 can be determined. Linear interpolation between the lines IX and IX+1 yields the wave parameters at line IX+1/2, necessary for the corrector step. The corrector step (which can be repeated several times) consists of an explicit differential scheme,applied to the two balance equations described in section 1.1.
The amount of energy lost through dissipation (FD) and leakage of energy through the boundaries θa and θb (FL) is kept.

Results of the computations are written to the file REKRES.

Structure :



The call of this subroutine is :

```
CALL NUMSC(IX)
```

Parameter :

IX (I) wave parameters are determined at line IX+1 in the computational grid (X=IX.DX)

3.17 Subroutine PREDT

Function :
Estimates for the wave action A0 and -frequency W0 at line IX+1 (predictor step) are determined in this subroutine.

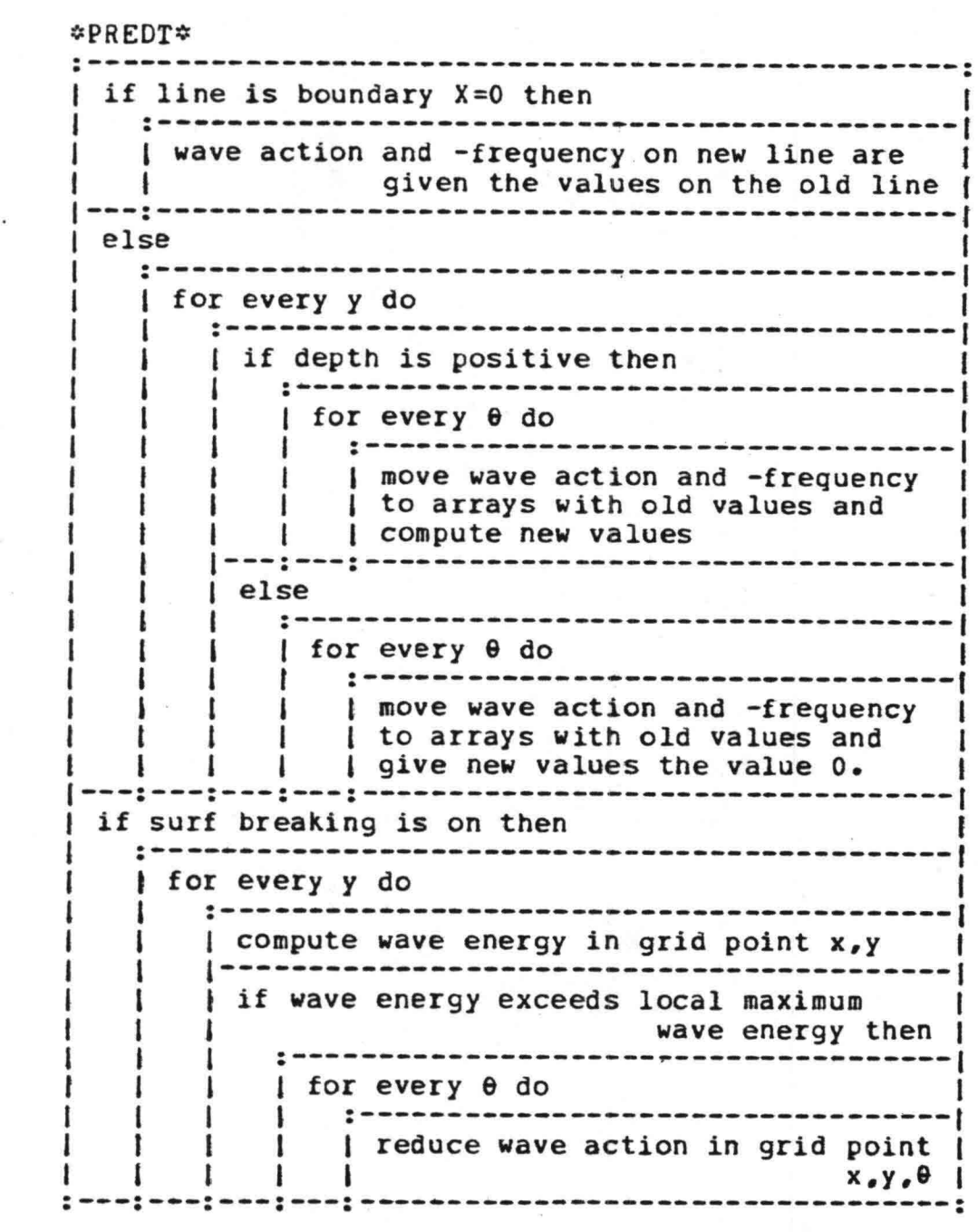
Method :
The predictor is a simple extrapolation procedure. A0 and W0 at line IX+1 are determined as follows :

$$A0^{IX+1} = 2.A0^{IX} - A0^{IX-1} \qquad (1)$$

$$W0^{IX+1} = 2.W0^{IX} - W0^{IX-1} \qquad (2)$$

If a negative depth is encountered then A0 and W0 are given the value 0..
If the estimate of the wave height in a grid point (x,y), obtained thus, exceeds the local maximum wave height (gamma*d) then the predictor estimates of A0 in this point are reduced accordingly.

Structure :



The call of this subroutine is :

```
CALL PREDT(IX)
```

Parameter :

IX (I) wave parameters are determined at line IX+1 in the computational grid (X=IX.DX)

3.18 Subroutine DISPA

Function :
In this subroutine parameters at line IX+1/2 are determined, necessary for the evaluation of the dissipation terms in the two balance equations.

Method :
The following parameters are determined :
- orbital velocity at the bottom Ubot
- current velocity at the bottom Ucur
- wave energy density Et
- local maximum wave height Hm
- the fraction of breaking waves Qb

For these parameters the following relations are used :

$$Ubot = \left(D\theta . \overline{\sum_{\theta}} \frac{\sigma 0^3 . A0}{\sinh^2 (K0.D)} \right)^{1/2} \qquad (1)$$

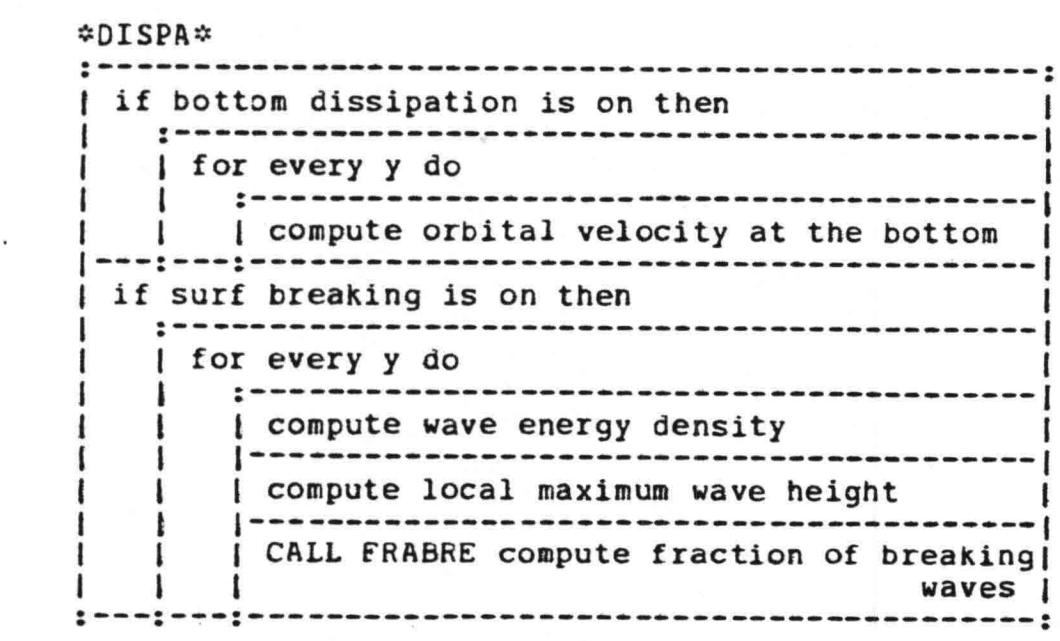
$$Et = D\theta . \overline{\sum_{\theta}} \sigma 0 . A0 \qquad (2)$$

$$Hm = 0.88 . \overline{\overline{K0}}^{-1} . \tanh(\text{gamma} . \overline{\overline{K0}} . D / 0.88) \quad , \quad \overline{\overline{K0}} = \frac{1}{N\theta} . \overline{\sum_{\theta}} K0 \qquad (3)$$

(the coefficient gamma is given by the user)

$$\frac{1-Qb}{\ln Qb} = -8 . \frac{Et}{Hm^2} \quad \text{(evaluated in FRABRE)} \qquad (4)$$

Structure :



The call of this subroutine is :
CALL DISPA

3.19 Subroutine FRABRE

Function :
In this subroutine the fraction of breaking waves in a point x,y in the computational grid (Qb) is computed.

Method :
The fraction of breaking waves is given by the implicit relation :

$$F \equiv 1 - Qb + 8 \cdot \frac{Et}{Hm^2} \cdot \ln Qb = 0 \qquad (1)$$

The following approximation is applied (Dingemans, 1983) :

$$b = (8Et/Hm^2)^{1/2} \qquad (2)$$

$$Q0 = (2b - 1)^2 \qquad 0.5 < b < 1 \qquad (3)$$

$$= 0 \qquad b < 0.5$$

$$Qb = 0 \qquad b < 0.3 \qquad (4)$$

$$= Qb - b^2 \cdot \frac{Qb - \exp\left((Qb-1)/b^2\right)}{b^2 - \exp\left((Qb-1)/b^2\right)} \qquad 0.3 < b < 0.9$$

$$= Qb \qquad 0.9 < b < 1.0$$

The parameters Et and Hm (in eq. 1) are determined in subroutine DISPA.

The call of this subroutine is :

```
CALL FRABRE(IY)
```

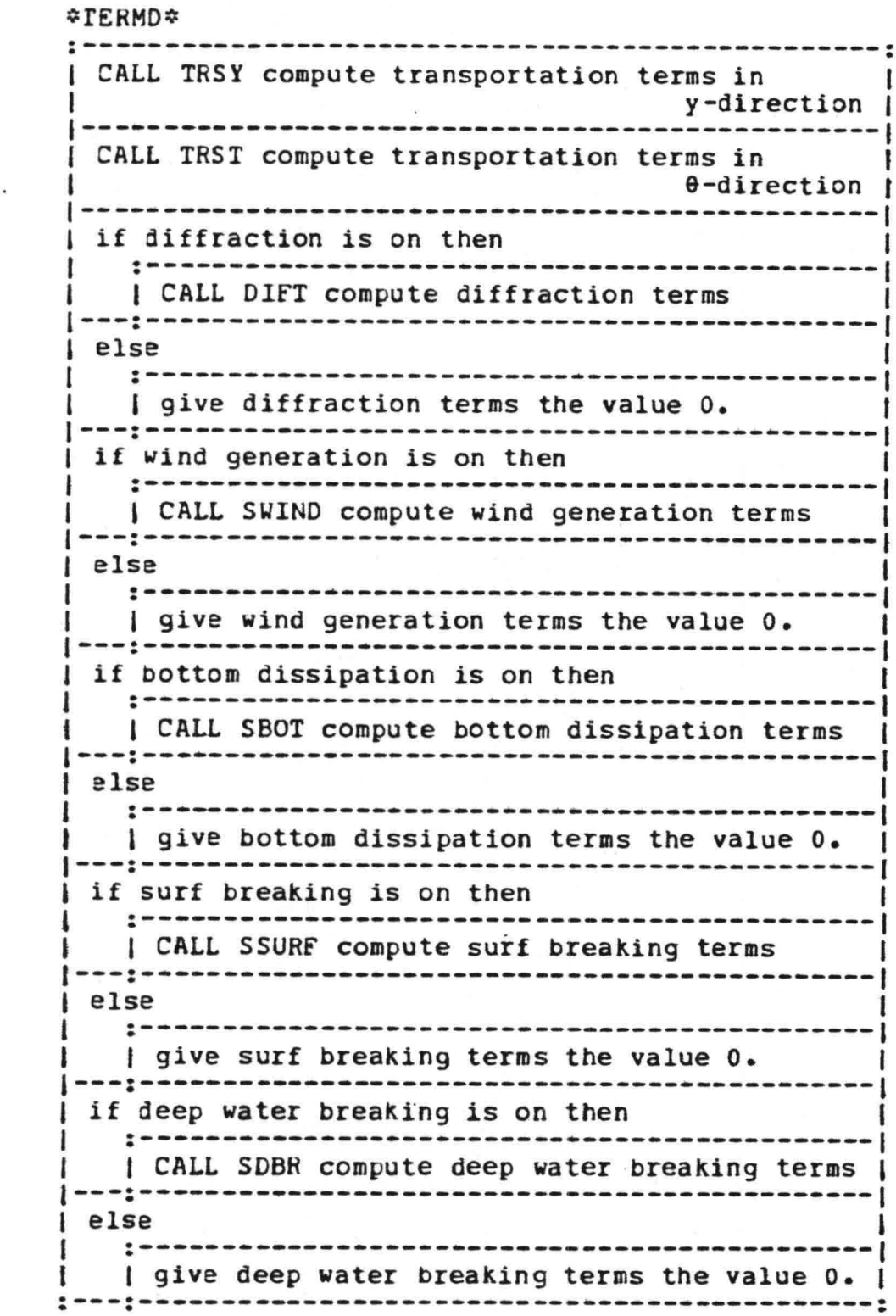
Parameter :

IY (I) y-coordinate of point in which the fraction of breaking waves Qb is computed

3.20 Subroutine TERMD

Function :
In this subroutine the terms of the two balans equations are evaluated in the point IX+1/2,IY,IT. The source terms S0 and dW0/dT are split up in components of wind generation, bottom friction, surf breaking and dissipation in adverse currents.

Structure :



The call of this subroutine is :

```
CALL TERMD(IY,IT)
```

Parameters :
IY (I)
) coordinates of point in computational grid
IT (I)

3.21 Subroutine TRSY

Function :
The transportation terms of the two balance equations in y-direction :

$$\frac{\partial}{\partial Y}(CY0.A0) \text{ and } \frac{\partial}{\partial Y}(CY0.W0.A0)$$

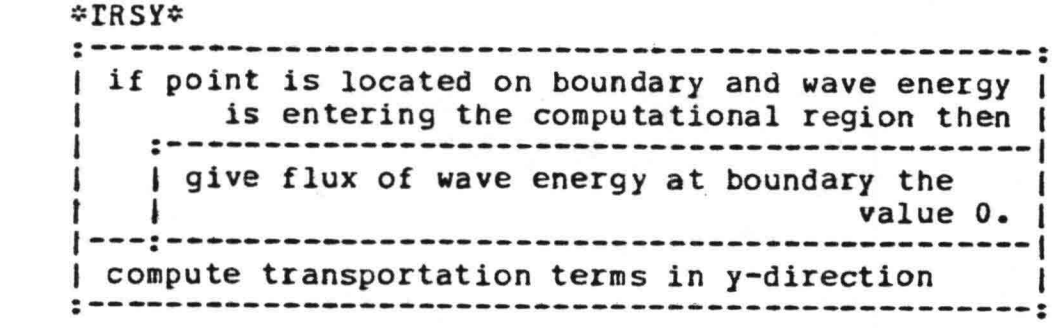
are determined in this subroutine.

Method :
A conservative central difference scheme is applied :

$$\frac{\partial f}{\partial Y} = \frac{f_{IX+1/2,IY+1,IT} - f_{IX+1/2,IY-1,IT}}{2.dY} \qquad (1)$$

Energy entering the computational region through the boundaries Y=0 and Y=LY is not taken into account. At these boundaries somewhat different schemes are used.

Structure :



The call of this subroutine is :
CALL TRSY(IY,IT)

Parameters :
IY (I)
) coordinates of point in computational grid
IT (I)

3.22 Subroutine TRST

Function :

The transportation terms of the two balance equations in θ-direction :

$$\frac{\partial}{\partial\theta}(C\theta 0.A0) \text{ and } \frac{\partial}{\partial\theta}(C\theta 0.W0.A0)$$

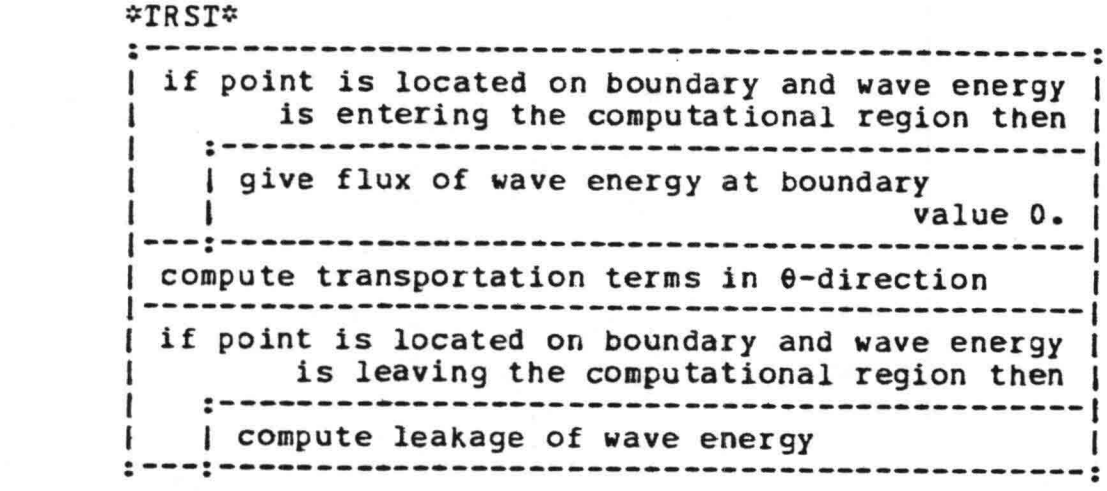
are determined in this subroutine.

Method :
A conservative central difference scheme is applied :

$$\frac{\partial f}{\partial\theta} = \frac{f^{IX+1/2,IY,IT+1} - f^{IX+1/2,IY,IT-1}}{2.d\theta} \qquad (1)$$

Energy entering the computational region through the boundaries θ=θa and θ=θb is not taken into account. At these boundaries somewhat different schemes are used. The leakage through these boundaries |Cθ0|.A0.00 is kept.

Structure :



The call of this subroutine is :

```
CALL TRST(IY,IT)
```

Parameters :

IY (I)
IT (I)) coordinates of point in computational grid

3.23 Subroutine DIFT

Function :
The diffraction terms in the two balance equations :

$$\frac{\partial}{\partial\theta}(Cdif(CY0.\frac{\partial}{\partial X}A0 - CX0.\frac{\partial}{\partial Y}A0)) \quad \text{and}$$

$$\frac{\partial}{\partial\theta}(Cdif(CY0.\frac{\partial}{\partial X}(W0.A0) - CX0.\frac{\partial}{\partial Y}(W0.A0)))$$

are determined in subroutine DIFT.

Method :
Derivatives in x,y and θ direction are approximated by central difference schemes :

$$\frac{\partial f}{\partial X} = \frac{f^{IX+1,IY,IT} - f^{IX,IY,IT}}{dX} \qquad (1)$$

$$\frac{\partial f}{\partial Y} = \frac{f^{IX+1/2,IY+1,IT} - f^{IX+1/2,IY-1,IT}}{2.dY} \qquad (2)$$

$$\frac{\partial f}{\partial \theta} = \frac{f^{IX+1/2,IY,IT+1} - f^{IX+1/2,IY,IT-1}}{2.d\theta} \qquad (3)$$

At the boundaries in the y-θ plane somewhat different schemes are applied.

The call of this subroutine is :
```
CALL DIFT(IY,IT)
```

Parameters :
```
IY (I)
    ) coordinates of point in computational grid
IT (I)
```

3.24 Subroutine SWIND

Function :
In this subroutine the wind generation components

$$S0_{wind} \quad \text{and} \quad \frac{d\_W0}{dt}_{wind}$$

of the source terms in the two balance equations are determined.

Method :
The following relations are used for the terms mentioned above :

$$S0_{wind} = \frac{U10^3}{g} Babcd \left\langle \left|\frac{E/B}{a}\right|^{\frac{d-1}{d}} \left\langle 1 - \left(\frac{E/B}{a}\right)^{\frac{2}{d}} \right\rangle \left\langle \frac{1}{b} - \operatorname{atanh}\left(\left(\frac{E/B}{a}\right)^{\frac{1}{d}}\right)^{\frac{c-1}{c}} \right\rangle \right. \quad (1)$$

$$\frac{dW0}{dt}_{wind} = \frac{g^3}{U10^5} 2PI.a2 \left(b2\left(\frac{W}{2PI}\right)^{\frac{(b2-1)}{b2}} - \frac{1}{B} S0_{wind}\right) \left\langle \left| \frac{W}{2PI.a2(E/B)^{b2}} \right|^5 \right\rangle \quad (2)$$

with

E is the dimensionless wave energy density $E0.\frac{g^2}{U10^4}$, $E0=\sigma 0.A0$

W is the dimensionless mean action frequency $W0.\frac{U10}{g}$

B is the directional distribution of waves, $E0(Y,0)=B(\theta).Et(Y)$

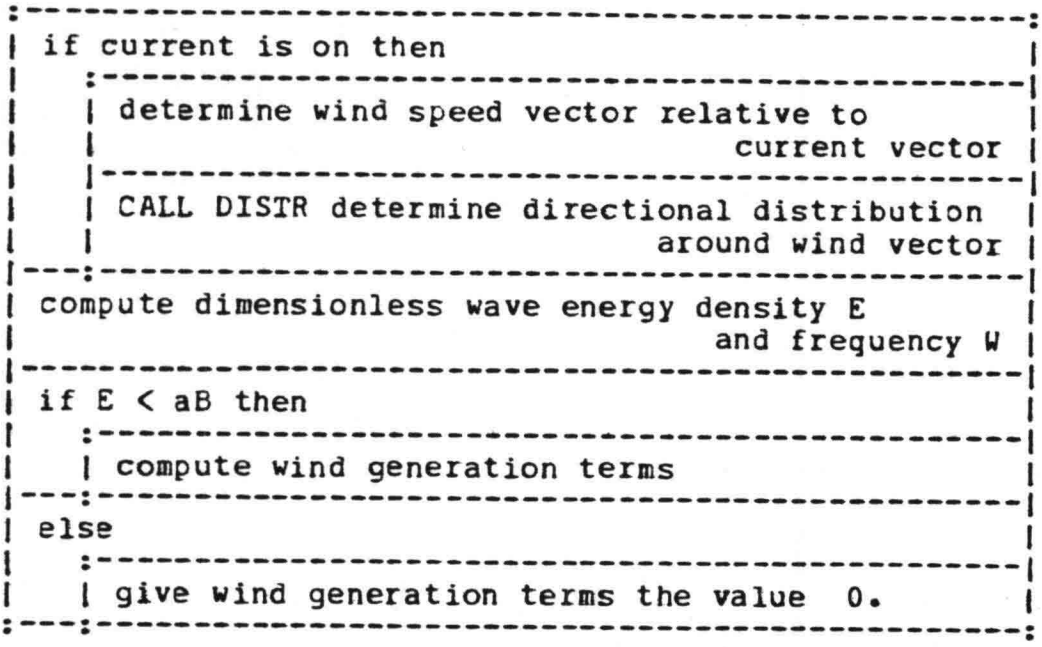
$U10$ is the wind velocity at 10m elevation relative to the current velocity

a, b, c and d are coefficients derived from literature

Eq. 1 and 2 hold only for growing waves ($E<aB$). In the case $E>aB$ then the wind generation terms are assumed to be 0.

Structure :

SWIND



The call of this subroutine is :
CALL SWIND(IY,IT)

Parameters :
IY (I)
) coordinates of point in computational grid
IT (I)

3.25 Subroutine SBOT

Function :
The bottom dissipation terms

$$S0_{bot} \quad \text{and} \quad \frac{d\_W0}{dt}_{bot}$$

are computed in this subroutine.

Method :
The following relations are applied :

$$S0_{bot} = -W_{bot} \cdot \sigma 0 \cdot A0 \qquad (1)$$

$$\frac{dW0}{dt}_{bot} = W0 \cdot a3 (g^{-2} \cdot W0^{3} \cdot W_{bot}^{b3} \cdot \sigma 0 \cdot A0) \qquad (2)$$

with

$$W_{bot} = (8/\pi)^{1/2} \frac{\sigma 0^2}{g.\sinh^2(K0.D)}.(Cfw.Ubot+Cfc.Ucur) \quad (3)$$

The second term in eq. 3 is omitted if no current is present. In this formulation the effects of currents on bottom dissipation are included in the same way dissipation due to wave orbital velocities is determined. The same procedure with somewhat different relations is applied in the model CREDIZ.
The terms

$$W_{bot} \quad \text{and} \quad \underline{d}\_W0_{dt\ bot}$$

are determined in point IX+1/2,IY,IT in the computational grid. The wave action A0 in the linear term S0bot is included implicitly in the two balance equations (section 3.27).
The coefficients Cfw,Cfc,a3 and b3 have to be determined, by the user, empirically.

The call of this subroutine is :

```
CALL SBOT(IY,IT)
```

Parameters :

IY (I)
IT (I)) coordinates of point in computational grid

3.26 Subroutine SSURF

Function :
The terms representing dissipation of wave energy due to surf breaking

$$S0_{surf} \quad \text{and} \quad \underline{d}\_W0_{dt\ surf}$$

are determined in this subroutine.

Method :
Relations for these terms applied in this model are :

$$S0_{surf} = -W_{surf}.\sigma 0.A0 \quad (1)$$

$$\frac{dW0}{dt}_{surf} = W0.a4.(g^{-2}.W0^{3}.W_{surf}^{b4}.\sigma 0.A0) \quad (2)$$

with

$$W_{surf} = alfa1 \cdot \frac{1}{8pi} \cdot Qb \cdot W0 \cdot \frac{Hm^2}{Et} \qquad (3)$$

The terms

$$W_{surf} \quad \text{and} \quad \frac{d\_W0}{dt}_{surf}$$

are determined in point IX+1/2,IY,IT in the computational grid. The wave action A0 in the linear term S0surf is included implicitly in the two balance equations (section 3.27).
The coefficient alfa1 is of order 1 while the coefficients a4 and b4 should be determined empirically.

The call of this subroutine is :

```
CALL SSURF(IY,IT)
```

Parameters :

IY (I)
IT (I)) coordinates of point in computational grid

3.27 Subroutine SDBR

Function :
The terms representing dissipation of wave energy in currents with direction opposite to the direction of wave propagation

$$S0_{dbr} \quad \text{and} \quad \frac{d\_W0}{dt}_{dbr}$$

are determined in this subroutine.

Method :
Relations for these terms applied in this model are :

$$S0_{dbr} = -W_{dbr} \cdot \sigma 0 \cdot A0, \quad W_{dbr} = \frac{f}{tau1} \qquad (1)$$

$$\frac{dW0}{dt}_{dbr} = -\frac{g}{tau2} \cdot W0 \qquad (2)$$

with

tau1, tau2 are time constants to be determined empirically
(initially values $W0^{-1}$ are assumed)

$$f = 0.3326\ (Wc/W0)^{-4} \qquad \text{for } Wc > 0.8547\ W0$$

$$= -5.712\ (Wc/W0)^{2} + 6.830\ (Wc/W0) - 1.042 \qquad \text{for } 0.5979\ W0 < Wc < 0.8547\ W0$$

$$= 1. \qquad \text{for } Wc < 0.5979\ W0$$

$$g = 1 - \frac{1 - 0.4434\ (Wc/W0)^{-3}}{1 - 0.3326\ (Wc/W0)^{-4}} \qquad \text{for } Wc > 0.8547\ W0$$

$$= 1 - \frac{0.66667\ (Wc/W0)^{3} - 0.59788\ (Wc/W0)^{2} + 0.071238}{(Wc/W0 - 0.59788)^{2}} \qquad \text{for } 0.62\ W0 < Wc < 0.8547\ W0$$

$$= 1 - Wc/W0 \qquad \text{for } Wc < 0.62\ W0$$

If E0 = 0. then g is given the value 0..

The call of this subroutine is :

```
CALL SDBR(IY,IT)
```

Parameters :

IY (I)
IT (I)) coordinates of point in computational grid

3.28 Subroutine SUMDE

Function :
In this subroutine the wave action A0 and -frequency W0 in the grid point IX+1,IY,IT are determined.

Method :
The two balance equations 1 and 2 are solved.

$$\frac{\partial}{\partial X}(CX0.A0) = -(\text{transportation+diffraction terms}) + \frac{S0_{wind}}{\sigma 0} - (W_{bot} + W_{surf} + W_{dbr} - \frac{1}{W0}\frac{dW0}{dt})A0 \quad (1)$$

$$\frac{\partial}{\partial X}(CX0.W0.A0) = -(\text{transportation+diffraction terms}) + \frac{W0.S0_{wind}}{\sigma 0} - (W_{bot} + W_{surf} + W_{dbr})W0.A0 \quad (2)$$

For brevity the transportation- and diffraction terms as well as the source term dW0/dt have not been written in full in eq. 1 and 2.
The numerical scheme applied to these equations is:

$$\frac{CX0_{IX+1,IY,IT}\ f_{IX+1,IY,IT} - CX0_{IX,IY,IT}\ f_{IX,IY,IT}}{dX} = G - \frac{H}{2}(f_{IX+1,IY,IT} + f_{IX,IY,IT}) \quad (3)$$

with :
f represents the terms A0 (eq.1) or W0.A0 (eq.2)
G contains the non-linear terms of eq. 1 and 2
H contains the linear terms of eq. 1 and 2
Further the dissipation of wave energy $(W_{bot}+W_{surf}+W_{dbr})A0.\sigma 0.d\theta$ is determined in point IX+1/2,IY,IT in the computational grid.

Structure :

```
*SUMDE*
:-------------------------------------------------:
| determine wave action and -frequency            |
|-------------------------------------------------|
| if last corrector step is in progress then      |
|   :---------------------------------------------|
|   | compute dissipation of wave energy          |
:---:---------------------------------------------:
```

The call of this subroutine is :
CALL SUMDE(IY,IT)

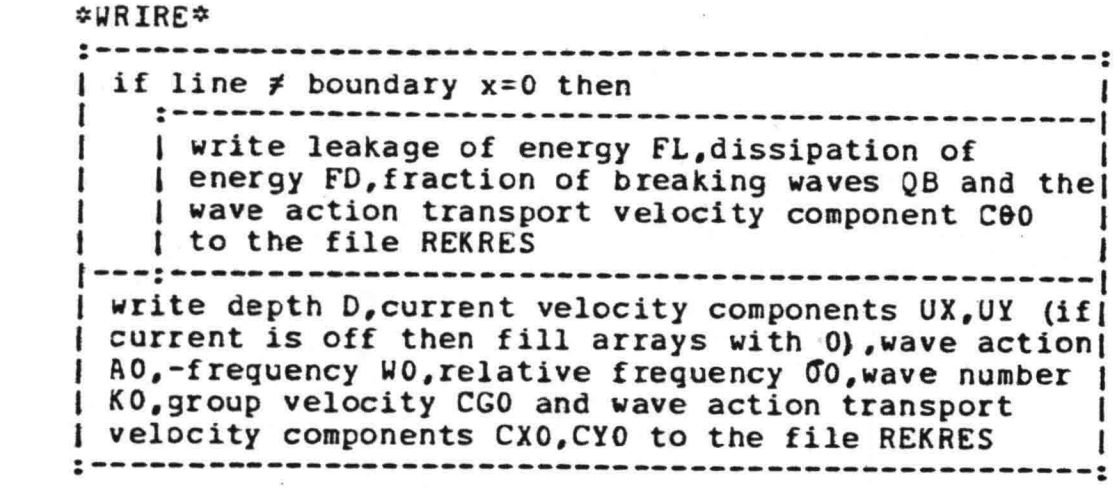
Parameters :

IY (I)
IT (I)) coordinates of point in computational grid

3.29 Subroutine WRIRE

Function :
In the subroutine WRIRE arrays containing wave parameters at a line in the computational grid are written to the file REKRES.

Structure :



The call of this subroutine is :

```
CALL WRIRE(IX)
```

Parameter :

IX (I) results of line IX in the computational grid are written to the file REKRES

3.30 Subroutine STRACE

This subroutine, called at the start of every subroutine and the main program, provides a message of the entry of this subroutine resp. program.
STRACE is copied from the model CREDIZ.

3.31 Subroutine MSGERR

The subroutine MSGERR, called when an error is encountered during the execution of the program COMPU, provides an error message.
MSGERR is copied from CREDIZ.

3.31 Subroutine COPYCH

In subroutine COPYCH character strings are copied to real variables and back. It is copied from CREDIZ.

4. STORAGE OF DATA

4.1 Dynamic data pool

As in CREDIZ a dynamic data pool is used to obtain an efficient and flexible storage of arrays. With the subroutine ADPOOL the dimension of an array can be extended or reduced. The structure of the pool is the same as in CREDIZ.
An element of an array A is found by :
A(I)=POOL(IA+I), IA is the adres of array A
A two-dimensional array is stored row by row :
eg. : array B(1:n,1:m)
storage in pool :
B(1,1),...,B(n,1),B(1,2),...,B(n,2),B(1,m),...,B(n,m)
thus B(k,1)=POOL(IB+(1-1)n+k), IB is the adres of array B
For more detailed information on the structure of the dynamic data pool, reference is made to the system documentation of CREDIZ.
The following arrays of the program COMPU are included in the pool :

number	name	adres	description
1	DEP	IDEP	depths
2	VX	IVX	x-component current velocity
3	VY	IVY	y-component current velocity
4	WAO	IWAO	wave action (old line)
5	WFO	IWFO	wave frequency (old line)
6	WKO	IWKO	wave number (old line)
7	RFO	IRFO	relative frequency (old line)
8	CGO	ICGO	group velocity (old line)
9	CXO	ICXO	x-component wave action transport velocity (old line)
10	CYO	ICYO	y-component wave action transport velocity (old line)
11	WA	IWA	wave action (between old and new line)
12	WF	IWF	wave frequency (between old and new line)
13	WK	IWK	wave number (between old and new line)
14	RF	IRF	relative frequency (between old and new line)
15	CG	ICG	group velocity (between old and new line)
16	CX	ICX	x-component wave action transport velocity (between old and new line)
17	CY	ICY	y-component wave action transport velocity (between old and new line)
18	CT	ICT	θ-component wave action transport velocity (between old and new line)
19	WAN	IWAN	wave action (new line)
20	WFN	IWFN	wave frequency (new line)
21	WKN	IWKN	wave number (new line)
22	RFN	IRFN	relative frequency (new line)
23	CGN	ICGN	group velocity (new line)

24	CXN	ICXN	x-component wave action transport velocity (new line)
25	CYN	ICYN	y-component wave action transport velocity (new line)
26	DD	IDD	derivatives in bottom geometry
27	DC	IDC	derivatives in current field
28	HM	IHM	local maximum wave height
29	QB	IQB	fraction of breaking waves
30	ET	IET	directionally integrated wave energy
31	FD	IFD	dissipation of wave energy
32	FL	IFL	leakage of wave energy
33	CRN	ICRN	maximum frequency that is able to transport wave energy against current (new line)
34	UCUR	IUCUR	current velocity near the bottom
35	DEO	IDEO	depths (old line)
36	UXO	IUXO	x-component current velocity (old line)
37	UYO	IUYO	y-component current velocity (old line)
38	DEM	IDEM	depths (between old and new line)
39	UXM	IUXM	x-component current velocity (between old and new line)
40	UYM	IUYM	y-component current velocity (between old and new line)
41	DEN	IDEN	depths (new line)
42	UXN	IUXN	x-component current velocity (new line)
43	UYN	IUYN	y-component current velocity (new line)
44	CRO	ICRO	maximum frequency that is able to transport wave energy against current (old line)
45	CR	ICR	maximum frequency that is able to transport wave energy against current (between old and new line)

table 1 arrays in dynamic data pool

4.2 Common blocks A number of common blocks are defined in which principal data for the model HISWA is included.Each of these blocks contains a certain category of information :

name	description
TITEL	title
DEPROS	location and dimensions of bottom grid
REKROS	location and dimensions of computational grid
TRANSF	transformation coefficients between different grids
NUMS	information on the numerical scheme
TERMDE	terms of the balance equations
TESTDA	information for output control (especially the tracing of errors)
FYSPAR	physical parameters
UITVDA	information for the program OUTP (not used by the program COMPU)
REFNRS	data set reference numbers
LEESDA	information for reading data
POOL	references to arrays in the dynamic data pool

table 1 description of common blocks

Most of the common blocks are copied from the model CREDIZ. In the following only the changes with regard to the common blocks in CREDIZ are discussed. For more information reference is made to the system documentation of CREDIZ.

TITEL as in CREDIZ

DEPROS the elements AKX,CCGX and WKCX are omitted

REKROS the following elements are added :
MTR number of grid points in θ-direction
TETAA
) boundaries of computational grid in θ-direction
TETAB
DT grid size in θ-direction

TRANSF as in CREDIZ

NUMS contains the following elements :
NCOR number of corrector steps
IPRE indicator of predictor step
ICOR indicator of corrector step
IOBW option for the representation of the boundary condition at x=0
ICUR switch for the introduction of current
IDIF switch for the introduction of diffraction
IWIND switch for the introduction of wind
IBOT switch for the introduction of bottom dissipation
ISURF switch for the introduction of surf breaking
IDBR switch for the introduction of deep water breaking

WDIP wind direction relative to problem grid
WDIC wind direction relative to computational grid
U10 wind velocity at 10 m. elevation
U10C wind velocity at 10 m. elevation relative to current
ADIR coefficient for directional distribution of waves
AT(300) wave action at boundary x=0
WT(300) wave frequency at boundary x=0
BDIR(30) directional distribution of waves
CDIF diffraction coefficient
PWIND(10) parameters of wind generation (a,b,c,d,a2,b2)
PBOT(5) parameters of bottom dissipation (Cfw,Cfc,a3,b3)
PSURF(5) parameters of surf breaking (alfa1,gamma,a4,b4)
PDBR(5) parameters of deep water breaking terms (tau1,tau2)
CNORM normalization coefficient of directional distribution

TERMDE contains the following elements :
FYA
) transportation terms in y-direction
FYF
FTA
) transportation terms in θ-direction
FTF
DIFA
) diffraction terms
DIFF
WINDA
) wind generation terms
WINDF
WBOT
) bottom dissipation terms
BOTF
WSURF
) surf breaking terms
SURFF
WDBR
) deep water breaking terms
DBRF
TDIS dissipation of wave energy
TLEAK leakage of energy

TESTDA as in CREDIZ

FYSPAR the following elements are omitted :
IM,DEP1,DEP2,DEP3,UX1,UX2,UX3,UY1,UY2,UY3,
AK1,AK2,AK3,CCG1,CCG2,CCG3,WKC1,WKC2,WKC3,
ISTA1,ISTA2,ISTA3,SIGMA1,SIGMA2,SIGMA3,
SINH1,SINH2,SINH3,AMPL1,AMPL2,AMPL3
the following elements are added :
DEPTH depth in a point in the bottom grid
UXC
) current velocity components relative to computational grid
UYC
WNU wave number

UITVDA as in CREDIZ (for the time being)

REFNRS the elements HULPF1 and HULPF2 are omitted

LEESDA as in CREDIZ

POOL this block will be adjusted to the new construction of the dynamic data pool (section 4.1)

4.3 Files

In the model HISWA a number of files are used that serve as communication tools between the programs PREP, COMPU and OUTP. Two of these files are used by the program COMPU :

a) INSTR

This file contains instructions formulated in PREP and data necessary for the computations carried out in the program COMPU.

contents :

- NPOOL dimension of pool
- ITEST test parameter
- common blocks :
  - TITEL, DEPROS, REKROS, TRANSF,
  - NUMS, FYSPAR, REFNRS, UITVDA, TESTDA
- arrays :
  - DEP,VX,VY,OUTREQ,OUTDA

b) REKRES

In file REKRES the results of the computations carried out in the program COMPU are stored. This file will be read by the program OUTP.

contents :

- common blocks :
  - TITEL, DEPROS, REKROS, TRANSF,
  - NUMS, FYSPAR, REFNRS
- arrays :
  - DEP, VX, VY
  - DEN, UXN, UYN, WAN, WFN,) line x=0
  - RFN, WKN, CGN, CXN, CYN
  - FL, FD, QB, CT) line x=(i-1/2).dx) i=1,MXR-1
  - DEN, UXN, UYN, WAN, WFN,) line x=i.dx) i=1,MXR-1
  - RFN, WKN, CGN, CXN, CYN

Each array in the files described above is preceded by the dimension of the array. The same conventions as described under 4.1 are applied with regard to he storage of two-dimensional arrays. Files are read and written unformatted.

REFERENCES

Abramowitz,M. and Stegun,I.A.,1968,
Handbook of mathematical functions,Dover publications

CREDIZ01 system documentation,1984 (in Dutch)

Holthuijsen,L.H. and Booij,N.,1983,Selection and formulation of a numerical shallow water wave hindcast model,Delft University of Technology,Report no 1783

Appendix IV System Documentation of OUTP

Output Program HISWA/OUTP

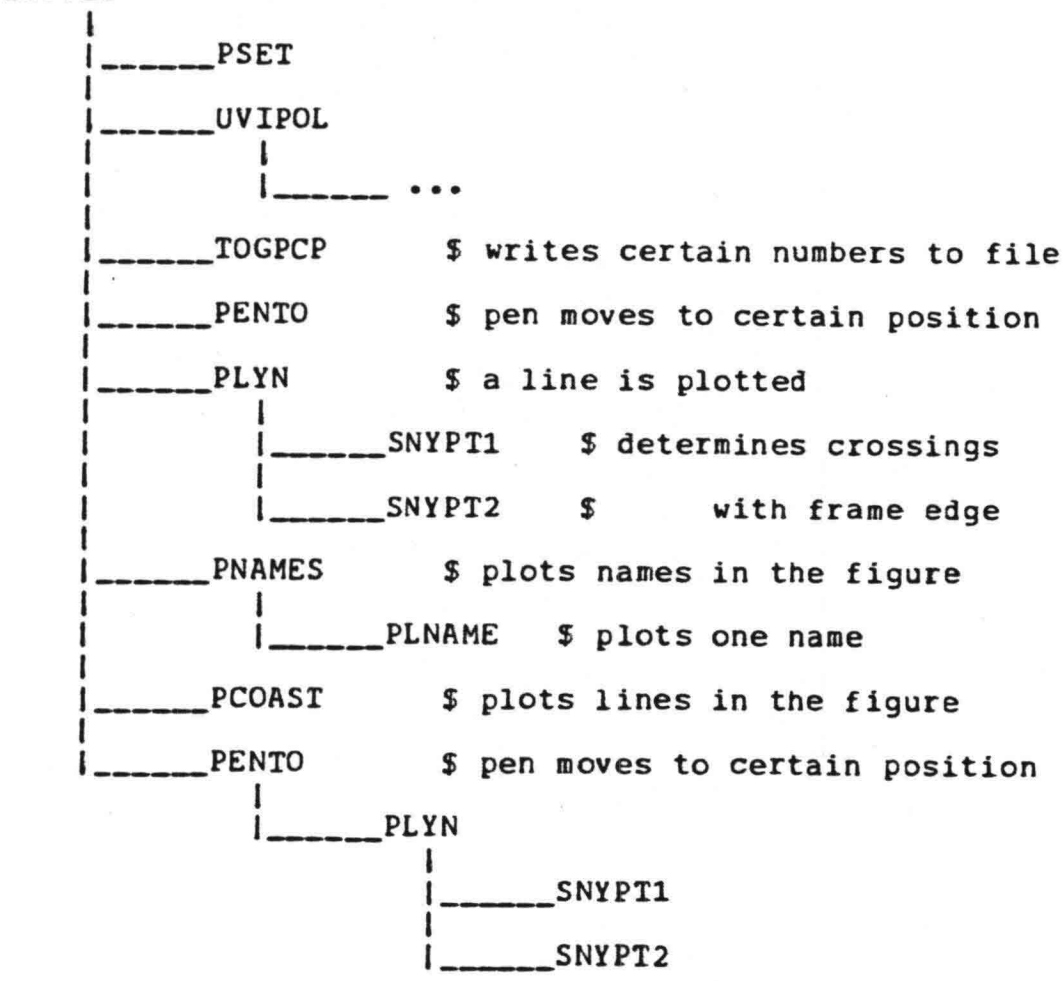
The program HISWA/OUTP processes the results computed by the program HISWA/COMPU. The results are stored in the file 'COMPRES'. The processing is controlled by the output requests encoded in the pool array OUTREQ, and using data in the array OUTDA.

HISWA/OUTP starts reading the output requests and output data. For each output request the file COMPRES is read, for the output points values are interpolated from the data in that file.

linkage diagram of Hiswa/Outp

```
Outp.MAIN
    |
    |_____WRCOMX etc.     $ read output instructions
    |
    |_____BLOKX           $ prints a block table
    |         |
    |         |_____PSET       $ determines output point set
    |         |
    |         |_____UVIPOL     $ reads results and interpolates
    |         |         |
    |         |         |_____IXPOL
    |         |                   |
    |         |                   |_____G1      $ interpolation
    |         |                   |
    |         |                   |_____G2      $     functions
    |         |
    |         |_____BLKHFD     $ block heading
    |         |
    |         |_____BLKPT      $ prints numbers
    |                   |
    |                   |_____UVHSIG etc.
    |
    |_____TABPRT          $ prints a table
    |         |
    |         |_____PSET       $ determines output point set
    |         |
    |         |_____UVIPOL     $ reads results and interpolates
    |         |         |
    |         |         |_____IXPOL
    |         |                   |
    |         |                   |_____ ...    $ see above
    |         |
    |         |_____UVHSIG etc.   $ determines values
    |
    |_____ISOVEC          $ produces a plot
    |         |
    |         |_____ ...    $ see next page
    |
    |_____WSPEC           $ spectral output to disk
```

linkage scheme of plot routines



Description of subroutines

MAIN program

The main program reads the output instructions that have been composed by the Reading program of HISWA. Furthermore it reads some general dat concerning the computation from the file containing the computational results.
Depending on the type of output request MAIN branches to the subroutin that produces that particular output.
After this subroutine has finished its task, MAIN will process the nex output intstruction.

HISWA/OUTP Main program

```
Read from file INSTU: some general output parameters and
pool arrays IOUTR (output requests) and IOUTD (output data).

Call WRCOMX    $ reads data concerning the computation from the
               $ file REKRES

For each output request do
    Obtain from array OUTREQ:
    - pointer to next request
    - type of output request RTYPE
    ------------------------------------------------------
    Case RTYPE =
    = 'BLK' : Call BLOKX      $ block output
    = 'TAB' : Call TABPRT     $ tabular output
    = 'PLOT' : Call ISOVEC    $ plot
    = 'SPEC' : Call WSPEC     $ spectral output
-------------------------------------------------------------
```

BLOKX, TABPRT and ISOVEC

The output routines BLOKX, TABPRT and ISOVEC have to a large extent the same structure. Each of these works in the following way: It obtains from array IOUTR the necessary parameters for the output at hand. It calls PSET, which determines the type of the output point set, and which stores the user coordinates of the output points in the arrays IXP and IYP. It calls UVIPOL, which reads the computational results and interpolates these to the output points. It calls UVHSIG or another routine which calculates the requested variable, which it then prints, writes to disk, or plots, as the case may be.

Parameters are:

RTYPE	type of output request
IUREC	place where to find the output instruction in array IOUTR

BLOKX, TABPRT and ISOVEC

```
-----------------------------------------------------
Obtain from array IOUTR:
   KNAAM      $ name of output point set
Call PSET     $ determines KTYPE, type of the point set,
```

```
            $ MIP, number of points in the set.
(in the case of BLOKX and ISOVEC:)
If KTYPE is not 'KADR' (frame)
Then Error message
-----------------------------------------------------------
Obtain NVAR, number of variables that must be output.
Determine IUR1, and IUR2, indicating where to find ISOORT,
which indicates the type of variable wanted for output.

Call UVIPOL  $ reads computational results and
             $ interpolates to output points

For JVAR from 1 to NVAR
    Determine ISOORT $ type of variable
    Depending on value of ISOORT,
    output UVHSIG, UVDIR, UVPER etc.
    $ see section on coding and processing of variables
-----------------------------------------------------------
```

Pool arrays in output module

1	IDEP	depths in the bottom grid
2	IVX	x-velocities in the bottom grid
3	IVY	y-velocities in the bottom grid
4	IOUTR	output requests
5	IOUTD	output data
6	IFF	unused
7	IWAN	wave action density
8	IWFN	mean frequency
9	IRFN	mean relative frequency
10	IWKN	wave number
11	ICGN	group velocity
12	ICXN	x-comp. of group velocity
13	ICYN	y-comp. of group velocity
14	IFL	energy leakage on a line halfway between two lines in comp. grid
15	IFD	dissipation
16	IQB	fraction of breaking waves
17	ICT	transfer velocity in Theta-dir.
18	IXP	the set of output points in problem coordinates
19	IYP	,,
20	IXC	the set of output points in computational grid coordinates
21	IYC	,,
22	IDEN	depths on a line of the computational grid
23	IUXN	x-velocities
24	IUYN	y-velocities
25	INF	indicates for each variable JVAR where data can be found in array IFOP
26	IFOP	function values in the output points
27	IA1	auxiliary array
28	IA2	,,
29	IA3	,,

Add to common area /UITVDA/: XPQ coord. of frame base point in user coordinates
YPQ
UCOS coeff. to transform from computational to
USIN frame coordinates

Coding of output requests

The output requests are coded in array OUTREQ (pool array with pointer IOUTR). The coding of the output requests as given by the user is carried out by the subroutine PROUT in the reading program. Each output request is represented by one record in the array.

The structure of the array OUTREQ is as follows:
- OUTREQ(1) number of places occupied in the array.
- OUTREQ(2) place in the array where the last record starts.

for each record the following data:
- NEXT pointer to next record in the array
- TYPE type of output request
- SNAME name of point set for which output is requested,
  occupying places in the array.
- other data specifying the output request (see below).

The following TYPEs of output request occur:
'BLKP' or 'BLKD' block type print or output to file
'PLOT' plot of iso-lines and/or vectors
'SPEC' output of spectra to file.

The data in the record depend on the type of the output request. For each type they are described in detail as follows:

Block output

- 'BLKP' or 'BLKD' type of output
- SNAME output point set
- NREF file reference number of destination
- NVAR number of variables which is to be written.

For each of the NVAR variables:
- ISOORT type of physical variable
- DFAC factor with which each value is multiplied.

Plot

- 'PLOT' type of output
- SNAME output point set
- PTI title of the plot
- ISOORT type of physical variable in iso-line plot
- FSTEP step of function
- FMIN minimum function value
- FMAX maximum function value.
- ISOORT type of physical variable in vector plot
- SCALE vector scale.
- IPLAC flag for plotting names of places
- ILINS flag for plotting lines

Spectral output

- 'SPEC'
- SNAME
- NREF destination of the output (unit reference number)

Coding and processing of variable types

In the output requests the type of variable that is wanted on output is encoded by the number ISOORT. With each type of variable a different procedure is carried out by the subroutine UVIPOL.

In the following table one finds for each value of ISOORT: code of the type in the user's output request, description of the variable type, procedure carried out by the subroutine UVIPOL. In the procedure description Sum(..) means the sum over the directions.

1. HSIG significant wave height
   Proc. UVIPOL: Sum(RFN*WAN) into aux. array A1
   Into array IFOP: interpolated value of A1
   Proc. UVHSIG: 4*Sqrt(F1)
2. DIR dominant wave direction
   Proc. UVIPOL: Sum(Theta*WAN) into A1, Sum(WAN) into A2
   Into IFOP: interp. A1, interp. A2
   Proc. UVDIR: F1/F2
3. PER mean wave period
   Proc. UVIPOL: Sum(WFN*WAN) into A1, Sum(WAN) into A2
   Into IFOP: interp. A1, interp. A2
   Proc. UVPER: 2*PI*F2/F1
4. DEPT depth
   Proc. UVIPOL: into IFOP: interpolated value of DEN
   Proc. UVDEP: F1
5. VEL current velocity
   Proc. UVIPOL: into IFOP: interp. UX, interp. UY
   Proc. UVUX: F1
   Proc. UVUY: F2
6. FORC resulting force exerted by the waves
   Proc. UVIPOL: n=.., cost=cos(theta), sint=sin(theta)
   Sum ((n*cost**2 + (n-.5)) * WAN) into A1
   Sum (n*cost*sint * WAN) into A2
   Sum ((n*sint**2 + (n-.5)) * WAN) into A3
   Into IFOP: x-derivative of A1, y-der. of A2,
   x-der. of A2, y-der. of A3
   Proc. UVFRX: F1+F2
   Proc. UVFRY: F3+F4
7. TRAN energy transport
   Proc. UVIPOL: Sum(CXN*RFN*WAN) into A1,
   Sum(CYN*RFN*WAN) into A2;
   Into IFOP: interp. A1, interp. A2
   Proc. UVENX: F1
   Proc. UVENY: F2
8. DSPR directional spread
   Proc. UVIPOL: Sum(WAN) into A1, Sum(Theta*WAN) into A2,
   Sum (Theta**2 * WAN) into A3;
   Into IFOP: interp. A1, interp. A2, interp.
   A3.
   Proc. UVDSPR: Sqrt(F3/F1-(F2/F1)**2)
9. DISS dissipated energy
   Proc. UVIPOL: Into IFOP: interp. FD
   Proc. UVDISS: F1
10. LEAK energy leak over sector boundary
   Proc. UVIPOL: Into IFOP: interp. FL
   Proc. UVLEAK: F1
11. QB fraction breaking waves

Proc. UVIPOL: Into IFOP: interp. QB
Proc. UVQB: F1

12. XP x-coordinate of output point
Proc. UVXP: Pool(IXP+IP)

13. YP y-coordinate of output point
Proc. UVYP: Pool(IYP+IP)

14. DIST distance of output point along curve
Proc. UVDIST: 0, if IP=1;
Value in IP-1 + distance to point IP, otherwise