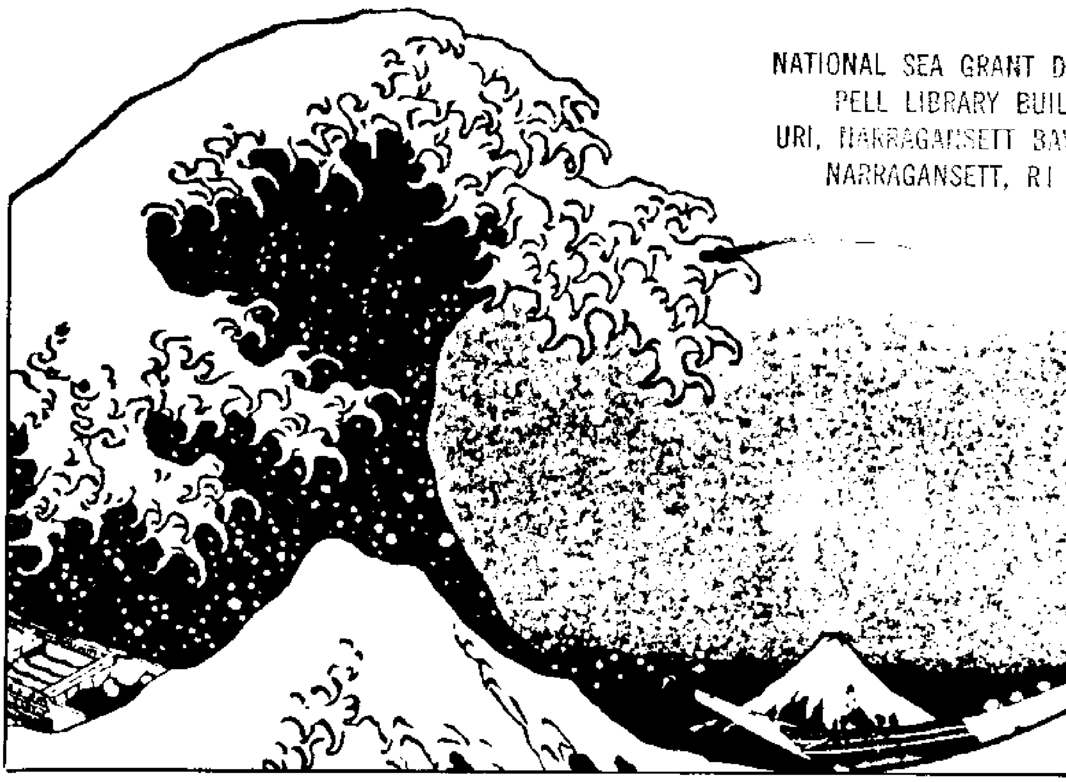
ESTIMATION of DESIGN WAVE HEIGHTS in the GULF of MAINE

Bryan Pearce
Vijay Panchang
University of Maine at Orono



University of Maine/University of New Hampshire
Sea Grant College Program

ESTIMATION of DESIGN WAVE HEIGHTS in the GULF of MAINE

Bryan Pearce and Vijay Panchang
Department of Civil Engineering
University of Maine at Orono
November 1, 1983

Published by the Communications Office at the Sea Grant Marine Advisory Program, University of Maine, Orono; supported by Grant #NA-81AA-D-00035 to Maine/New Hampshire Sea Grant College Program by the National Oceanic and Atmospheric Administration.

TR-MSG-83-4 200

ESTIMATION OF DESIGN WAVE HEIGHTS IN THE GULF OF MAINE

The purpose of this study was to generate wave height information for the Gulf of Maine by means of a model which calculates synoptic wind velocities for Northeast storms and other numerical wave hindcast models. A parametric type wave model was used with a set of the 22 strongest Northeast storms from a 32 year record, and the highest wave height at each grid point of the model for each storm was used to generate the extreme wave statistics. A discussion of the results is presented along with scope for future work in this area. The extensive appendices indicate the method of application of the different computer programs in computer wave heights.

ACKNOWLEDGEMENTS

We wish to thank the Maine/New Hampshire Sea Grant College Program, whose support made this study possible, and Stone and Webster Engineering Corporation, whose assistance and cooperation greatly facilitated this project.

Assistance rendered by Mr. David Kleinschmidt with computer operations proved extremely valuable, and we are grateful to him. We would also like to thank Maureen Rhodes for her meticulous efficiency in typing this report, and, more important, for bearing with us so cheerfully.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE PAGE

CHAPTER I

INTRODUCTION

Quantitative wave data in the coastal and offshore regions of the Gulf of Maine (Figure 1) are almost non-existent. In this part of the Atlantic Ocean, which is viewed as an economically developing area, a study of the wave climate is desirable, and as such, information about the wave climate in the Gulf of Maine is sought.

In engineering studies concerned with the design and certification of offshore structures it is essential to have probability estimates of most extreme wave conditions expected during the lifetime of the structure. Similar data are needed for problems of coastal engineering design and sediment transport. Reliable estimates of extreme wave statistics require a long time base of statistical observations from which to extrapolate to return periods of 50 or 100 years. The only method available at the present time for constructing wave statistics therefore has to be based on a wave hindcast approach using historical wind fields, and in view of the long periods and high cost involved in gathering wave measurements, a numerical model offers the only way of establishing a statistically valid data base.

The purpose of this study was to generate long term extreme wave statistics of the Gulf of Maine using hindcast wave data. Large ocean waves are generated predominantly by strong winds accompanying a storm; consequently, wave data were hindcast from severe "Northeast" storms occurring over a 32-year period from 1944 to 1976. The entire region was covered by a mesh, and wind velocities at each grid-point were obtained by feeding certain characteristics of the storm, described in detail in the following pages, to a "Northeaster" wind field model. The wind field so generated was utilized by a deep water, hybrid parametric wave model (Hydraulics Research Station, 1977; Gunther, et al., 1979) whence wave-heights at each grid-point were obtained. The wave model has been verified against wave measurements near the British Isles (Ewing, et al., 1979) and wave measurements for hurricane Camille in the Gulf of Mexico (Puri and Pearce, 1981). Finally, a distribution of the Weibull family, to which, incidentally, the Rayleigh distribution belongs, was used to estimate the long-term wave statistics, vis., the 50-year and 100-year wave heights.

A method to obtain wind velocities from known synoptic pressures is described in Chapter II. The pressure field of a Northeast storm is simulated, and wind velocities are calculated by application of the geostrophic equations to the simulated pressure pattern.

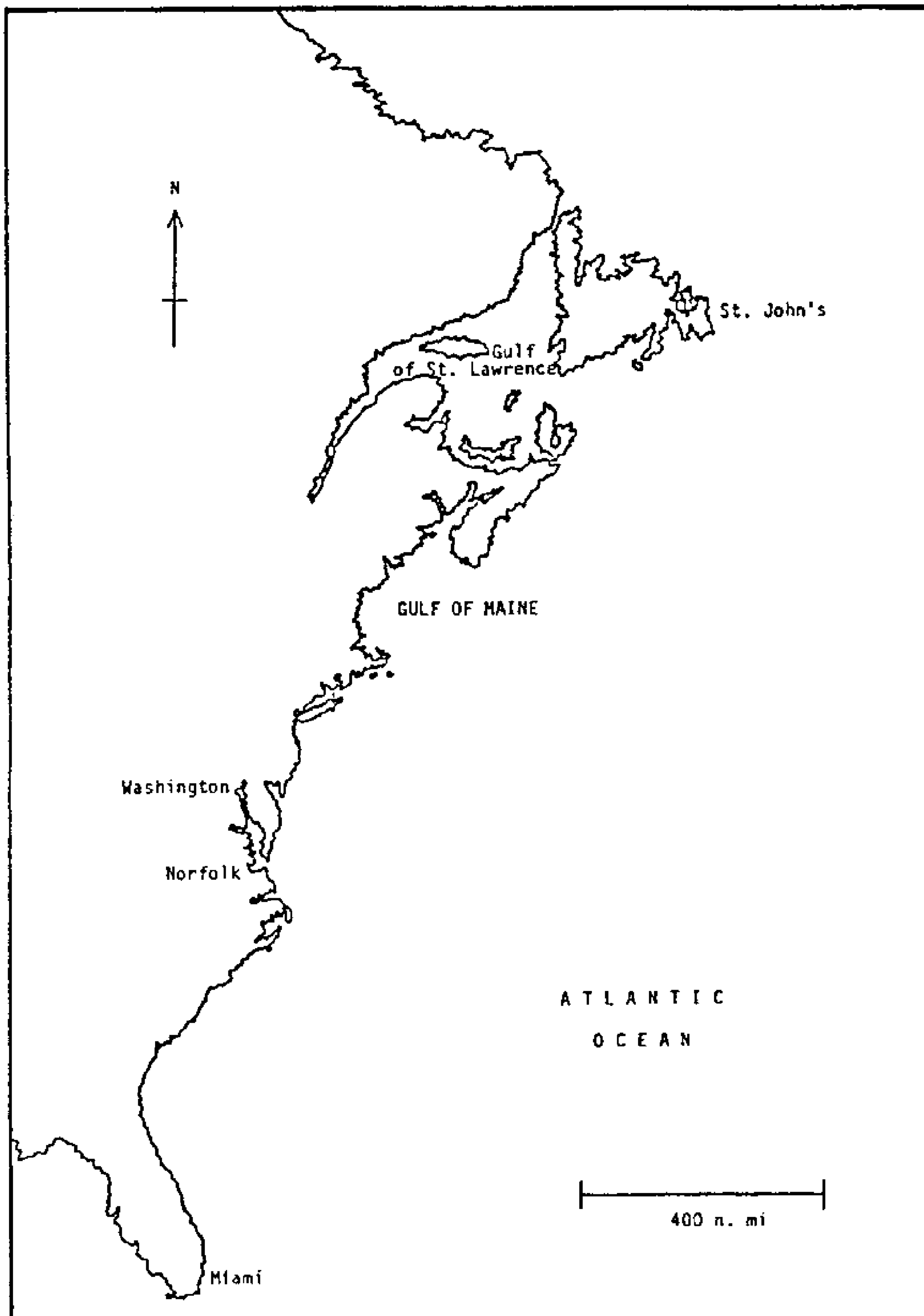


Figure 1. The East Coast of the United States

Chapter III deals with spectral and parametric wave models. The choice of the parametric numerical technique, and the incorporation of swell into it is described. The method suggested by Petruaskas and Aagard (10=970) for fitting a Weibull distribution to the calculated wave heights at each grid point is discussed in Chapter IV. Extrapolations to return periods of 50 and 100 years is explained at some length.

Chapter V highlights the role of the boundary conditions of the wave model in the choice of the grid system for discretizing the equation representing the conservation of energy. The 50-year and 100-year significant wave heights in the Gulf of Maine are presented as results. The validity of the model in certain regions and the scope of future work are discussed in Chapter VI.

CHAPTER II

COMPUTATION OF THE WIND FIELD

Winter extratropical storms or Northeast storms, or "Northeasters" as they are generally known, are responsible for the extreme wave conditions in the Gulf of Maine. While hurricane activity is intense south of Cape Cod, Northeast storms reach their greatest intensity as they pass New England. Characteristic parameters of 55 severe storms responsible for high storm tide elevations along the New England Coast are shown in Figure 2. Only one of these storms is a hurricane, and in light of its considerable distance from the study area and relatively smaller length scale, hurricanes were not chosen for wave hindcasting.

The northeaster presents, in general, a more complicated event than a hurricane. Whereas hurricane isobars can be acceptably modelled as circular, asymmetrical pressure patterns and large storm diameters (600-1500 nautical miles) are typical of Northeasters. Nevertheless, it is possible to define certain synoptic parameters like pressures, maximum radius, storm track, etc., from which surface wind velocities, needed as input to the wave model, can be calculated. This is accomplished with the aid of a Northeaster model developed by Stone & Webster Engineering Corporation (1978a).

A typical synoptic surface pressure chart with a Northeast storm is shown in Figure 3. The wind model must somehow approximate or otherwise describe this intricate and rather chaotic pattern of the pressure field, use the momentum equations, calculate wind speed, and finally deal with the atmospheric boundary layer in an appropriate manner.

Synthetic Pressure Pattern

It is obvious from Figure 3 that a wind field can be derived only if a mathematical relationship to describe the pressure distribution within the storm is developed. To this end, a group of 8 "typical" Northeast storms were chosen by Stone & Webster Engineering Corporation and analyzed. Figure 4 shows a storm schematic where

P_o = central pressure of the storm
P_p = peripheral pressure of the storm
R_{max} = maximum radius to storm periphery
r = radius to point within the storm
R_0 = radius to periphery at 0 from R_{max}.

Storm Number	Storm Location and Parameter at Starting Time: Time (EST)	Location: Lat (°N)	Location: Long (°W)	P (mb)	Storm Location and Parameters at Maturity: Time (EST)	Location: Lat (°N)	Location: Long (°W)	P (mb)	R_{max} (nm)	T (knot)
1	0030,11/30/44	36	75	14	1930,11/30/44	50	71	36	755	48
2	1530,11/19/45	40	77	10	1930,11/20/45	45	66	21	781	26
3	0330,03/02/47	36	71	10	0730,03/03/47	45	72	37	1048	13
4	1830,11/24/47	41	73	13	0730,11/25/47	48	68	30	800	39
5	1630,02/07/47	37	78	7	0130,02/09/47	46	70	37	960	22
6	1930,12/30/48	37	75	12	1930,12/31/48	42	72	33	1167	12
7	0630,11/27/48	39	76	6	1930,11/27/48	46	63	22	960	59
8	1830,04/13/49	39	78	9	1330,04/14/49	43	64	13	635	30
9	2130,04/05/49	36	79	6	0130,04/07/49	45	70	27	534	25
10	0930,12/11/50	33	69	11	0730,12/12/50	47	65	18	745	40
11	1230,01/07/51	38	79	8	0730,01/08/51	43	66	24	1281	32
12	1830,11/02/51	37	75	0	1930,11/03/51	48	70	16	382	29
13	0330,03/11/52	39	80	29	1330,03/12/52	48	64	35	961	28
14	0630,02/27/52	35	72	22	0130,02/28/52	42	66	42	840	24
15	1830,04/12/53	37	76	6	0730,04/13/53	42	67	15	1015	26
16	0630,10/24/53	39	69	11	0730,10/25/53	41	66	11	993	5
17	0930,12/14/54	36	77	5	1630,12/15/54	47	67	25	850	27
18	0000,08/31/54	37	74	52	0000,08/31/54	37	74	52	22*	33
19	0930,11/19/55	41	81	9	0730,11/20/55	46	69	37	951	31
20	1230,02/11/55	41	73	13	0430,02/12/55	46	68	45	841	25
21	1830,02/06/55	40	79	0	1030,02/07/55	46	65	16	402	47
22	0930,03/16/56	35	80	9	0430,03/17/56	43	66	37	608	45
23	0630,12/10/57	35	75	13	0430,12/11/57	47	68	33	1365	37
24	0030,01/16/57	34	75	1	1630,01/16/57	41	66	21	431	36
25	1830,02/14/57	36	68	6	1630,02/15/57	43	59	17	831	28
26	1830,04/06/58	39	80	13	1630,04/07/58	42	69	17	919	26
27	1230,02/07/58	37	79	7	0430,02/09/58	48	69	33	1157	21
28	1230,01/07/58	33	76	16	0730,01/08/58	45	67	49	1640	46
29	1800,12/28/59	41	80	14	1300,12/29/59	41	69	17	834	27
30	0000,12/21/60	39	81	15	0400,12/22/60	47	67	35	785	29
31	1800,01/19/61	37	78	20	1300,01/20/61	42	66	46	963	32
32	1500,03/06/62	33	69	17	0700,03/07/62	38	68	36	831	16
33	1800,11/29/63	40	77	36	1300,11/30/63	47	71	50	1629	27
34	1800,11/01/63	35	80	8	0100,11/03/63	47	70	32	1264	28
35	2100,11/30/64	37	67	28	0700,12/01/64	42	66	53	1031	29
36	1800,11/19/64	40	61	11	1000,11/20/64	50	74	20	660	40
37	2100,01/07/66	43	76	10	2200,01/08/66	43	67	24	660	21
38	0900,12/24/66	36	74	22	0400,12/25/66	42	70	38	905	22
39	0000,03/05/66	39	79	17	2200,03/05/66	48	75	24	1383	22
40	0300,04/27/67	36	74	11	0400,04/28/67	39	68	32	677	11
41	1800,05/24/67	33	75	3	1900,05/26/67	40	66	21	518	13
42	1500,02/29/68	36	74	20	0700,03/01/68	40	70	47	631	15
43	0000,12/23/68	46	87	26	0400,12/24/68	52	75	34	1239	22
44	0300,12/26/69	38	76	17	1900,12/27/69	43	68	35	1108	10
45	0900,02/03/69	40	76	9	0400,02/04/69	47	67	40	1046	31
46	2100,12/10/69	38	78	15	1000,12/11/69	45	77	25	699	30
47	0900,02/10/70	39	76	15	1300,02/11/70	43	68	46	1212	14
48	2100,03/26/71	35	73	18	1600,03/27/71	38	67	28	923	15
49	0300,02/19/72	38	75	25	1600,02/19/72	41	73	39	835	16
50	0300,12/17/73	36	74	24	1300,12/17/73	39	72	24	632	36
51	1800,12/01/74	38	77	22	1000,12/03/74	46	64	38	1443	18
52	0300,04/03/75	43	80	27	1900,04/03/75	43	72	42	1004	16
53	2100,12/17/75	37	74	9	1000,12/18/75	42	67	27	764	37
54	1800,12/21/75	36	71	13	1300,12/22/75	41	68	20	961	19
55	1500,03/16/76	40	76	19	0700,03/17/76	44	66	46	734	34

NOTE:

* Radius to the maximum wind for Hurricane Carol (1954)

Figure 2. Characteristic Parameters of 55 Selected Storms

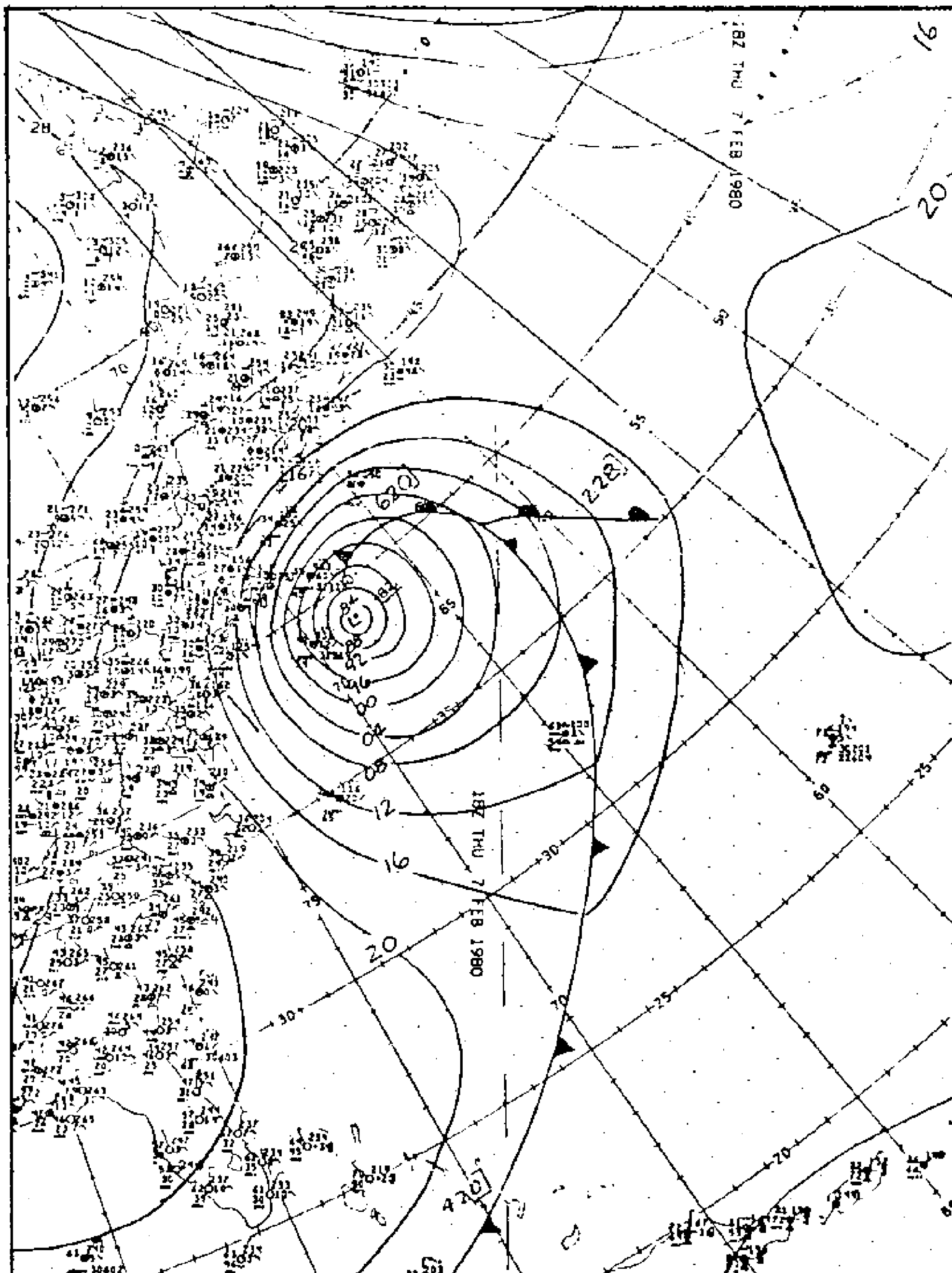


Figure 3. A Surface Pressure Chart Depicting Isobars of a Typical Northeast Storm

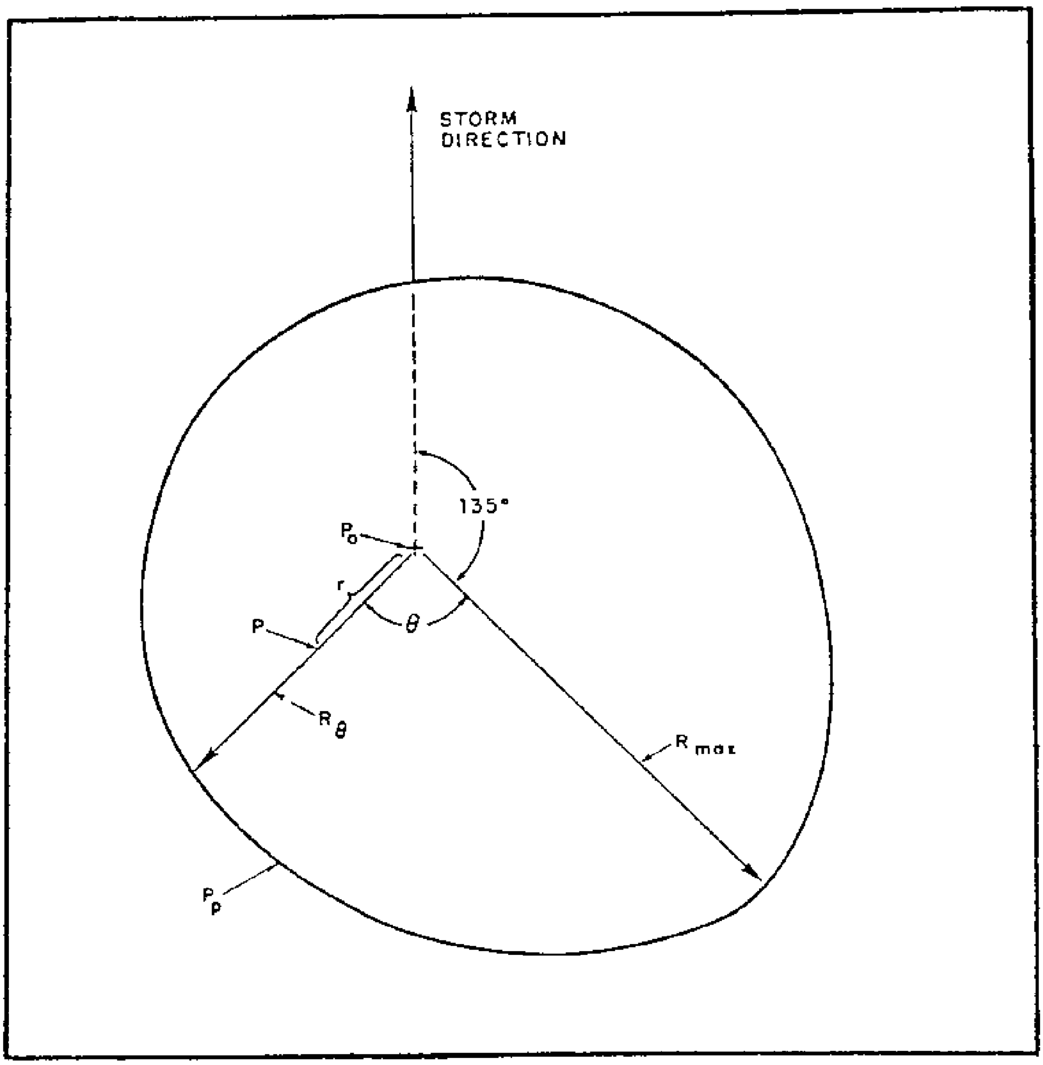


Figure 4. Definition Sketch for Northeast storm model (From Stone & Webster Engineering Corp., 1978)

The angle 135° between R_{max} and the storm direction has been chosen as an average for these storms and is constant in the model. After examining the 8 storms, average values were obtained for certain parameters, leading to the following empirical description of the synthetic pressure field within the Northeaster:

$$\frac{P-P_0}{P_p - P_o} = \exp[-a \ (\frac{R_\theta}{r} - 1] \qquad (1)$$

where:

P is the local pressure at r

r is local radial distance of a point where pressure is calculated

a is an empirically determined coefficient

$$a = 0.07254+0.03806\beta-0.1334\beta^2+0.03862\beta^3-0.002919\beta^4 \qquad (2)$$

in which

β = angle (clockwise) in radians from one storm track direction

The peripheral isobars can be described by the equation of the form

$$\frac{R_\theta}{R_{max}} = A + B\theta + C\theta^2 \qquad (3)$$

The coefficients A, B, & C were determined by a least squares fit in the analysis of the 8 storms, and the following equation resulted

$$\frac{R_\theta}{R_{max}} = 1-0.3143\theta + 0.050003\theta^2$$

where - is measured clockwise from R_{max}.

If the central pressure, peripheral pressure and maximum storm radium R_{max} are known as initio, the entire synthetic pressure field of a mature Northeast storm can be described in equations 1, 2, and 3. For instance, an observed isobaric pattern shown in Figure 5 can be modeled to give a more orderly appearance of isobars (Figure 6).

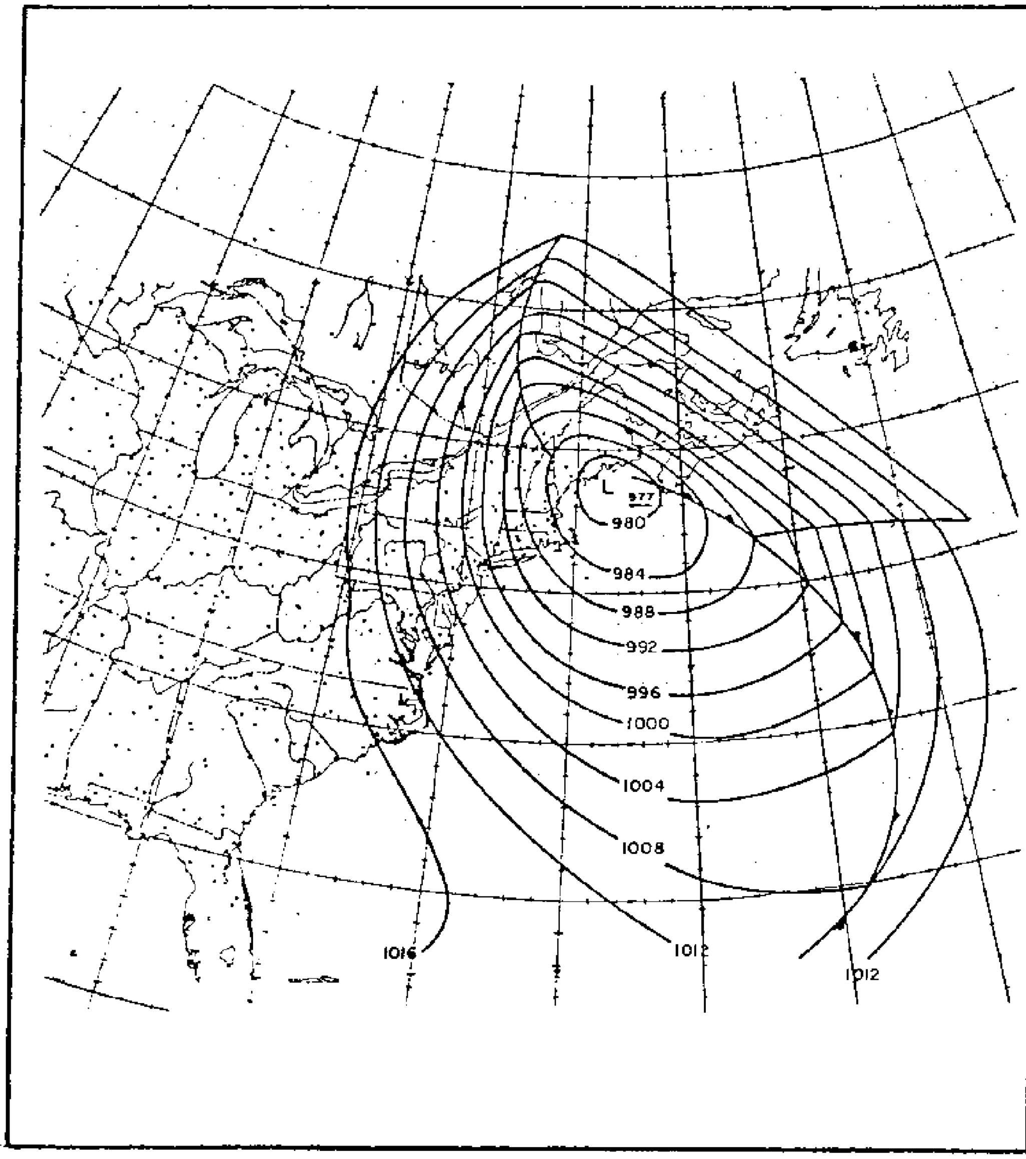


Figure 5. Observed Pressure Field (mb) for the Northeaster of December 28, 1969. (From Stone & Webster Engineering Corp., 1978a)

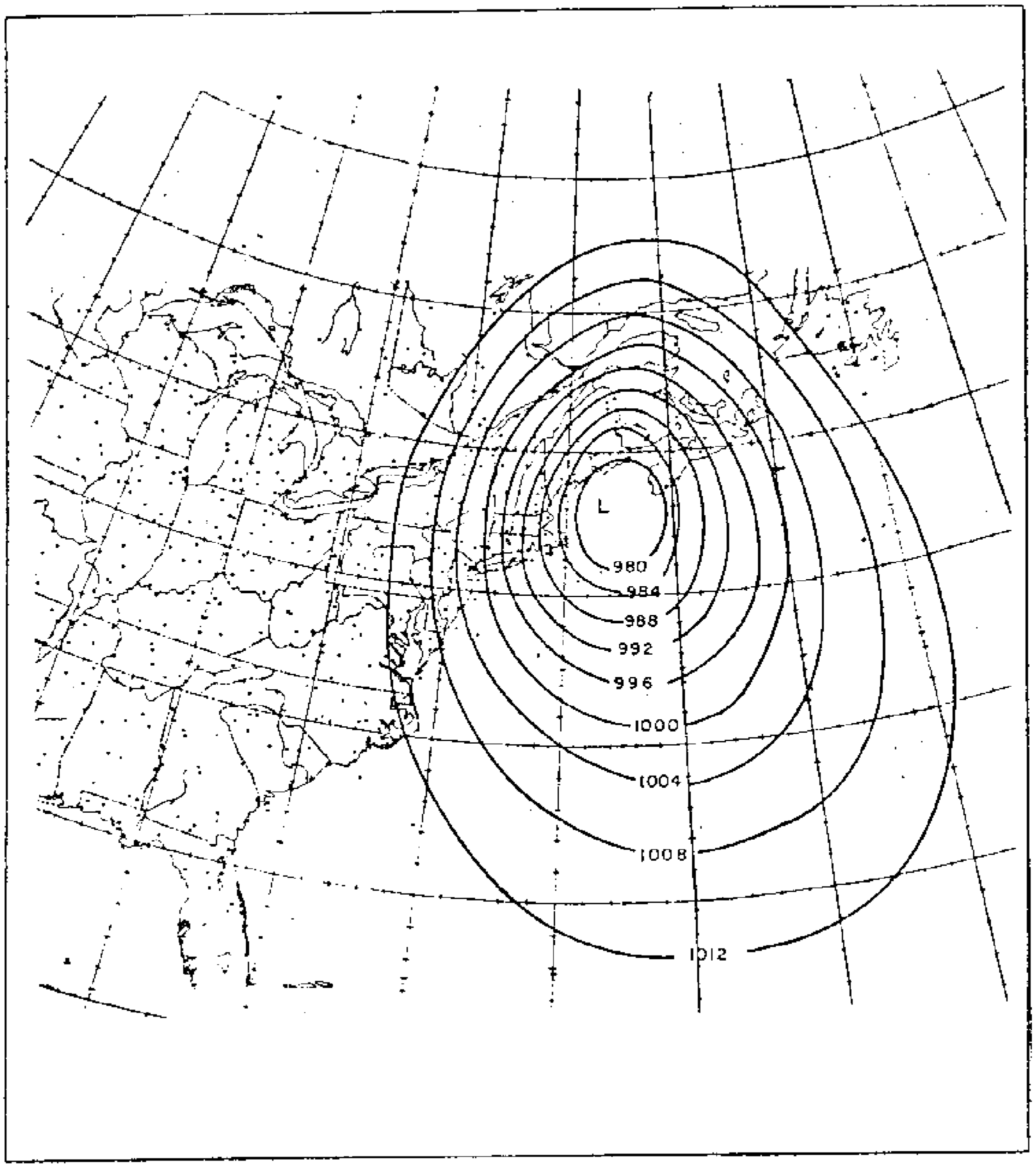


Figure 6. Synthetic Pressure Field (mb) for the Northeaster of December 28, 1969. (From Stone & Webster Engineering Corp., 1978a)

Synthetic Wind Field

Once a set of equations has been fitted to the pressure distribution of a Northeast storm, wind velocities must be computed. The factors that determine local surface wind speed include the Coriolis force, the pressure gradient force, the centripetal force, and surface friction. Two wind models commonly used are the geostrophic wind model and the equilibrium wind model. Though the latter incorporates more of the pertinent physical factors in the computation of wind speeds, the geostrophic wind model is appropriate for storms of large length scales, and is discussed below.

Geostrophic wind is a very useful approximation to the actual wind by means of which the direction and the width of the isobaric channels are converted into the wind velocity. The approximation implies the following simplifications-

1. The current is assumed to be nearly straight so that the centripetal (or turning) acceleration can be ignored. In other words, curvature of the isobars is neglected, which is a reasonable premise, since a Northeaster is characterized by isobars of very large radii of curvature. (In contrast, centripetal effects should be accounted for when dealing with hurricanes).

2. The motion is assumed to be free of friction.

3. It is assumed that the wind speed changes so slowly that acceleration along the path may be neglected.

The three assumptions imply that the horizontal pressure force $\frac{1}{\rho}(\partial p/\partial n)$ is balanced by the Coriolis force $V 2\Omega \text{Sin}\phi$, so that

$$V = \frac{1}{\rho 2\Omega \text{Sin}\phi} \frac{\partial P}{\partial n} \qquad (4)$$

or $V = \frac{1}{\rho f} \frac{\partial P}{\partial n}$

where

V = Geostrophic wind speed

$\partial P/\partial n$ = Pressure gradient normal to isobars

ρ = Air density = 0.00231 slugs/ft^3

$f = 2\Omega \text{Sin}\phi$ = Coriolis Parameter

Ω = Angular velocity of the rotation of the earth

ϕ = Latitude.

The angle of incurvature of the wind in a Northeast storm can range from 10 to 45 degrees. For this geostrophic model, the value of 22.5 degrees reported by Peterson, et al. (1964) is employed.

The balance between the Coriolis force and the horizontal pressure force given by equation 4 is nearly realized in the large scale currents above the layer directly influenced by surface friction (Petterssen, 1958). Surface friction sharply reduces the wind speed near land and sea surfaces. Surface winds can be estimated by reducing the geostrophic wind by a fixed percent; alternatively, some curve from Figure 7 could be chosen. The different curves in Figure 7 are from different sources, and details may be found in Peterson and Goodyear (1964). Curve V, based on 180 ship and Coast Guard observations of surface winds and computations of geostrophic winds, has been used in this simulation.

Application of the Wind Model

A set of the strongest Northeast storms to pass through the region has been prepared by Stone and Webster Engineering Corporation (1978b). From this, a subset of 22 storms which had a pressure difference ($P_p - P_o$) of 35 mb or greater was chosen for wave hindcasting.

The entire area under consideration is covered by a mesh. The input consists of storm central pressure, maximum peripheral radius and coordinates of the storm center at each instant of time. These can be obtained from synoptic pressure charts. For each instant of time, the program jumps from point to point in the grid and employs the geostrophic approach to calculate wind speed and direction at each point.

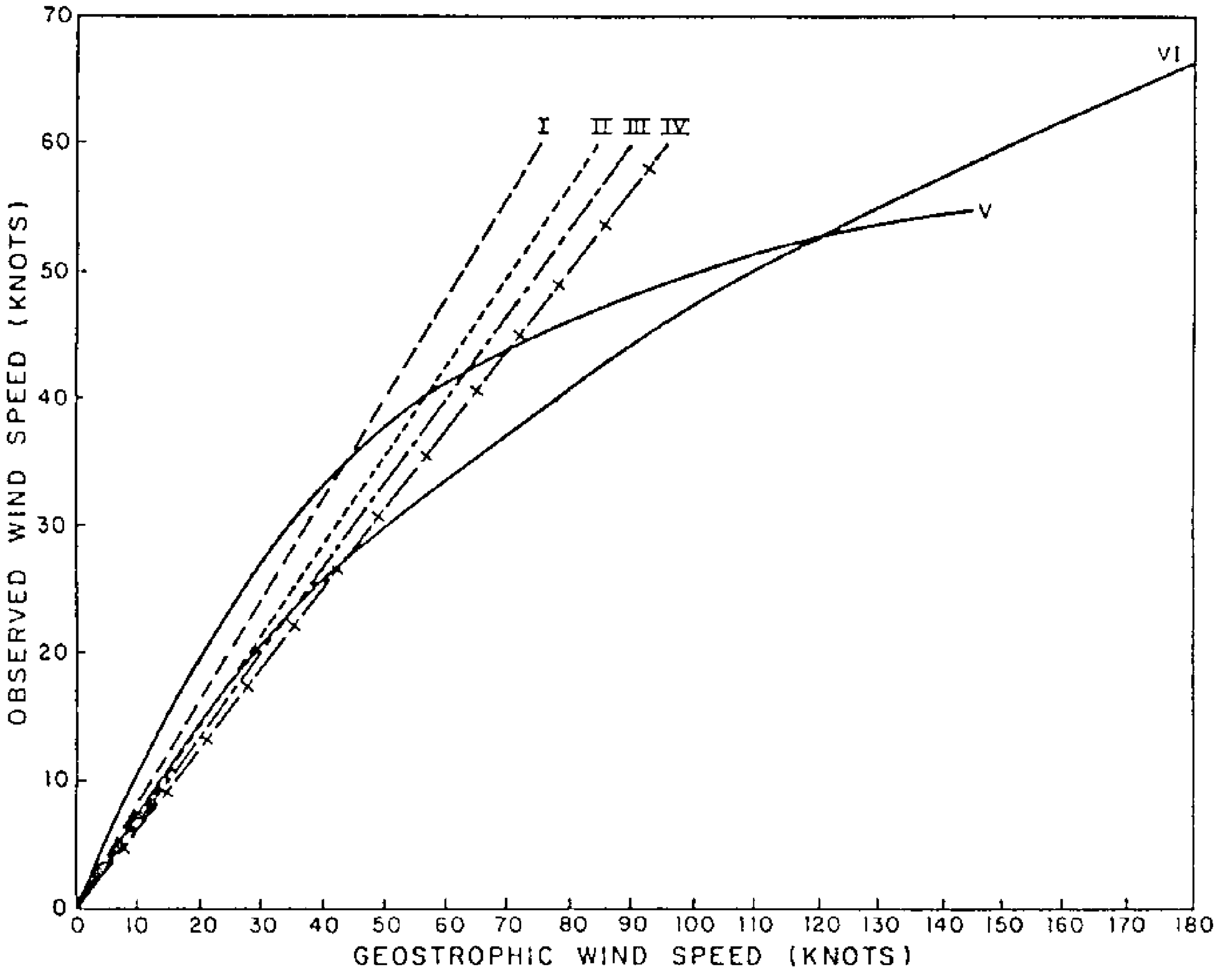


Figure 7. Relationship Between Surface and Geostrophic Winds.
(From Stone & Webster Engineering Corp., 1978a)

CHAPTER III

THE WAVE MODEL

The current approach to the study of sea waves identifies the energy spectrum as the ordering and governing principle in the apparent confusion of the ocean. The energy spectrum of a time record made at a fixed point is simply the Fourier cosine transform of the autocorrelation function of the record (Kinsman, 1965). Most modern wave prediction schemes are based on the energy balance equation for the wave spectrum:

$$\frac{\partial F}{\partial t} + V_i \frac{\partial F}{\partial x_i} = S \qquad (5)$$

Here $F(f,\theta;\bar{x},t)$ is the two-dimensional wave spectrum energy density, i.e. energy per unit frequency (f) per unit direction (θ) and V_1, V_2 are the components of the wave group velocity for frequency f and direction -. ($|V| = g/4\pi f$ in deep water). S is the energy source function governing all wave generating and dissipating processes. The resolution of equation 5 requires a knowledge of the source function. This has been a topic of intense theoretical and experimental research, and, following Hasselmann, et al. (1973), it can be resolved as

$$S = S_{in} + S_{ds} + S_{nl} \qquad (6)$$

where

S_{in} = Energy input from the atmosphere

S_{ds} = Energy dissipated due to various causes

S_{nl} = Energy transfer among the various frequency bands due to conserving non-linear wave-wave interactions.

Usually, S_{in} is explained on the basis of the theories of Phillips (1957) and Miles (1957). Phillips' mechanism leads to linear growth of wave energy in the early stages of wave development. The theoretical estimates of Miles' exponential growth are much smaller than those actually observed, and the exponential growth coefficients are taken from experiments. In a growing wind-sea, energy is dissipated mainly by wave-breaking in the high frequency parts of the spectrum. Other possible dissipative processes include attenuation of swell through interaction with short waves (Phillips, 1963), scattering by turbulence (Phillips, 1959) or topological effects (Long, 1973).

Spectral Approach

In the spectral approach to wave prediction, the spectrum is represented by some finite though large number of frequency-direction components at each point in a numerical model. The most commonly used source term representation which treats the spectral components that travel within ± 90° of the local wind is of the form

$$S = [A + B.E(f,\theta)] \; . [1 - (\frac{E(f,\theta)}{E_{\infty}(f,\theta,U})^2] \qquad (7)$$

where

A represents the linear growth rate

B represents the exponential growth rate

E_{∞} is the Pierson-Moskowitz spectrum (Figure 8), whose functional form was discovered by Pierson and Moskowitz (1964) as

$$E_{\infty}(f) = \frac{8.10x10^{-3} g^2}{(2\pi)^4 \; f^4} \; \exp[-0.74 \; (\frac{g}{2\pi U_{19.5} f})^4] \quad \text{in } m^2.\text{sec.}$$

In practice, the adopted magnitudes of A and B are empirically calibrated against field measurements of net observed wave growth. As the terms A and B adopted in any application are fitted constants, they reflect, in addition to Phillips' and Miles' mechanisms, other present mechanisms. The non-linear contribution S_{nl}, in this approach, enters in the balance equation 7 only indirectly through A and B, and a direct account of this component of S is simply ignored.

The equation 5, with 7, is now completely determinate and is numerically integrated for the energy spectrum over various frequency and direction bands. Thus, for instance, if the directional spectrum is divided into 24 direction bands and the frequency spectrum into 24 frequency bands, then 576 unknowns, each satisfying equation 5, will have to be solved for, leading thereby to an economically prohibitive and error-prone operation.

Physically, each of the above frequency-direction bands propagates with its own group velocity without any interaction from the other bands. This is an advective process which is quite appropriate to the propagation of swell where the advective term dominates the transport equation. In wind-sea, on the other hand, it has been established by Hasselmann, et al. (1973, 1976) that the non-linear transfer rates are an order of magnitude larger than the transfer rates associated with the advection. Accordingly, the spectrum responds much more rapidly to the non-linear interaction than to any other mechanism present and, therefore, it should not be ignored. The absence of a direct contribution from non-linear wave-wave work, required to solve the spectrum, leads us to a new generation of wave models, vis., the parametric models.

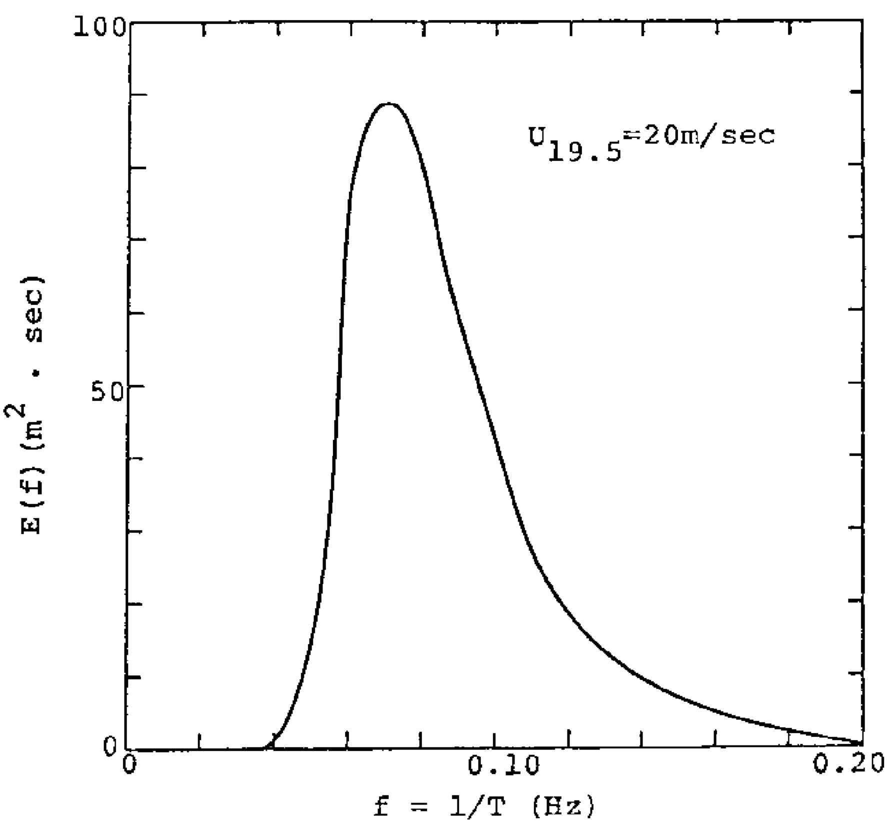


Figure 8. Pierson-Moskowitz Spectrum of a Fully Arisen Sea.
(From Horikawa, 1978)

Parametric Approach

The parametric models, introduced by Hasselmann, et al. (1973, 1976), assume that the spectrum can be given an analytical shape defined by five parameters. These models originated in the analysis of measurements of wave growth with fetch in the Joint North Sea Wave Project (JONSWAP) experiments. These models highlight the role played by the non-linear wave-wave interaction in the distribution of energy across the spectrum. Most of the observed characteristics of the development of the spectrum with fetch can be explained by non-linear energy transfer.

While the discrepancy in Miles' exponential growth of wave energy between the theoretical estimate and observations can be attributed to non-linear energy transfer, results from the JONSWAP study (Hasselmann, et al., 1973) have also shown that these non-linear wave-wave interactions in conjunction with the energy input from the atmosphere have a shape stabilizing effect on the wind-sea part of the spectrum giving rise to a mean spectral shape (Figure 9). If the spectrum becomes too broad, the wave-wave interactions act, as in Figure 10, to transfer energy to the peak from the central frequencies, causing the peak to grow. If the spectrum becomes too peaked, the wave-wave interactions move energy from immediately under the peak to the sides thus reducing and broadening the peak (Figure 11). In general when the spectrum has the mean shape, a dynamic balance exists where the wave-wave interactions act to transfer the energy input from the atmosphere at the peak and higher frequencies of the spectrum to the forward face of the spectrum (Figure 12), causing the peak to grow and migrate to lower frequencies, while maintaining the near spectral shape.

These findings are significant not only from the viewpoint of proper identification of the source function in equation 5, but also for the choice of the appropriate numerical procedure. Once the non-linear interaction is accepted as one of the effective agents in the process of spectral evolution, some form of parameterization of the spectrum becomes necessary, because the exact evaluation of the non-linear interaction is enormously complicated and extensively time-consuming on the computer. The following findings of the JONSWAP study indicate the process of such a parameterization:

1. The direction of the wave spectrum will follow the wind direction (Hasselmann, et al., 1976) as a consequence of the wave-wave interactions. Therefore, it is assumed that, for a growing wind-sea, the two-dimensional spectrum has a universal

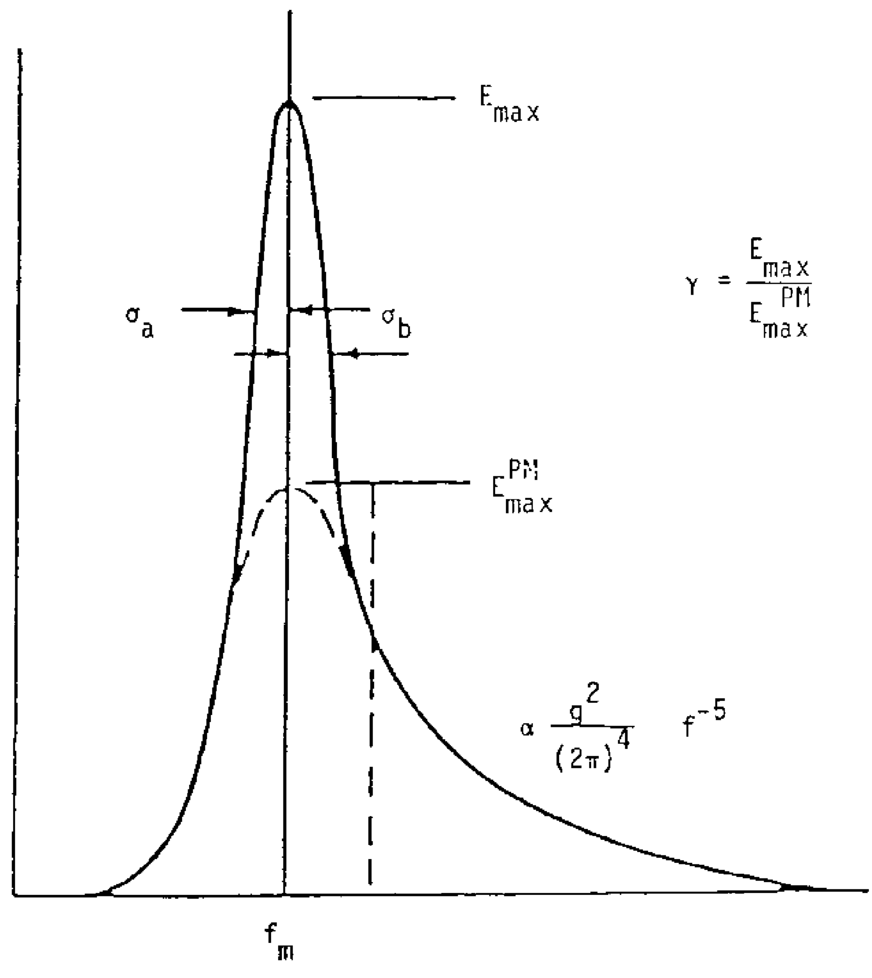


Figure 9. Mean Jonswap Spectrum. (From Hasselmann, et al, 1973)

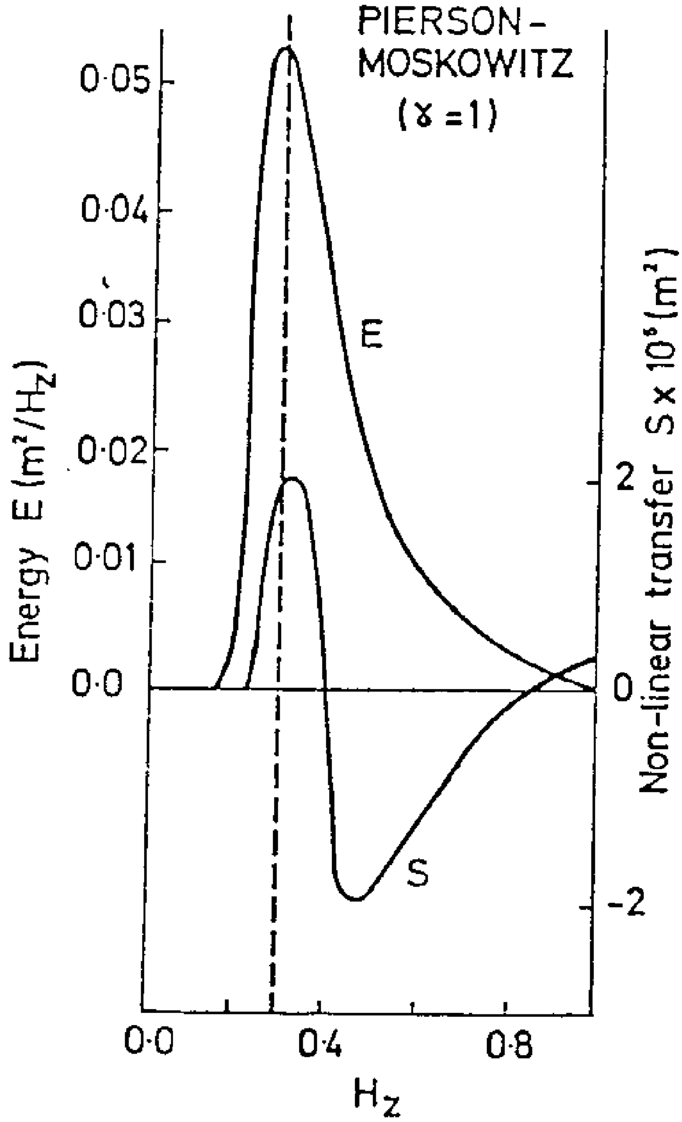


Figure 10. Energy Transfer for a Broad Spectrum. (From Hydraulics Research Station, 1977)

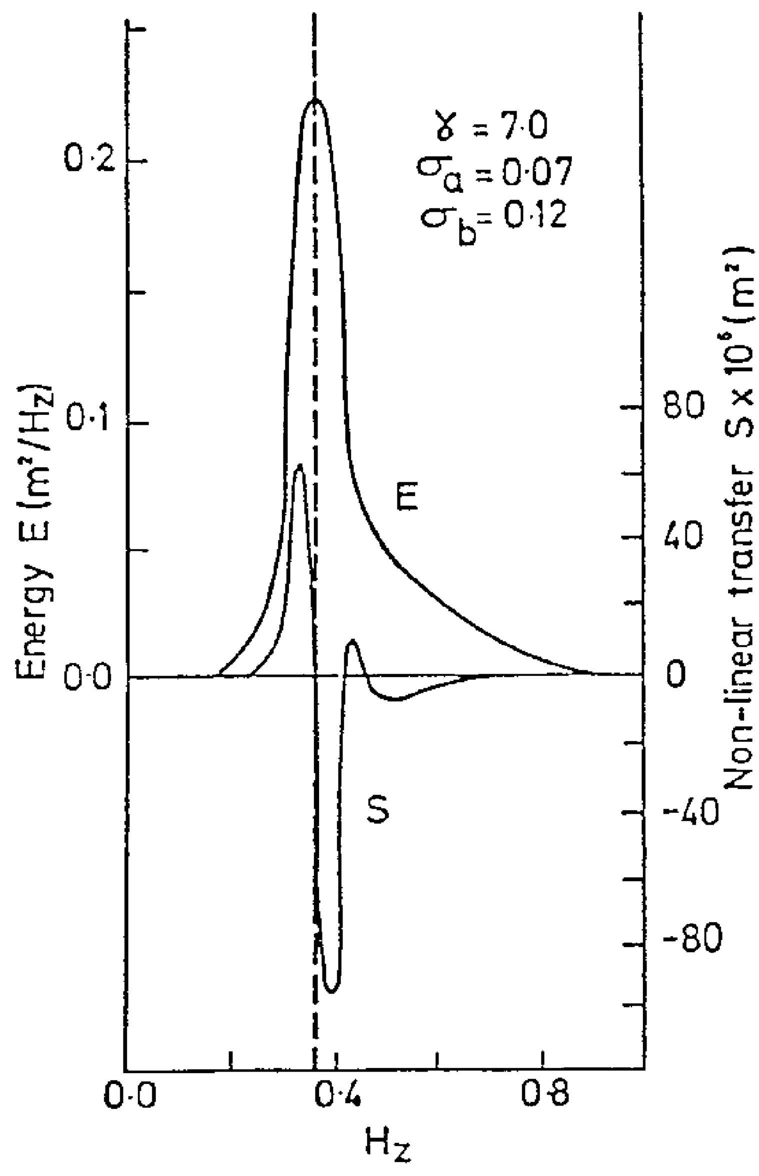


Figure 11. Energy Transfer for a Sharp Spectrum. (From Hydraulics Research Station, 1977)

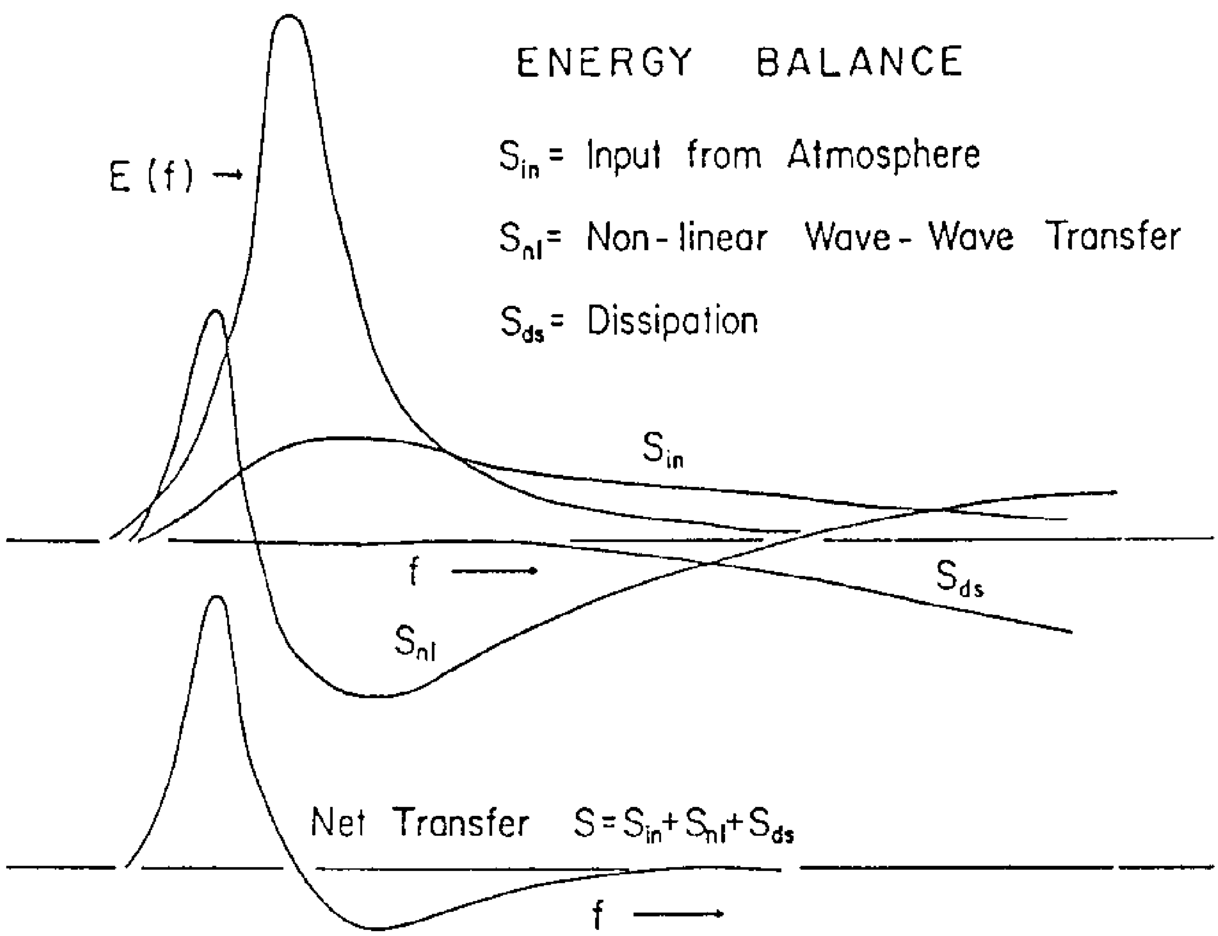


Figure 12. Energy Balance Schematic. (From Hydraulics Research Station, 1977)

(frequency-independent) angular distribution D, centered on the local wind direction -- i.e.

$$F(f,\theta;\overline{x},t) = D(\theta-\phi)\ E(f;\overline{x},t)$$

(It is also assumed that D is a $\cos^2$ distribution, in accordance with the JONSWAP study).

If equation 5 is integrated over direction, we have

$$\frac{\partial E}{\partial t} + \overline{V}_i \frac{\partial E}{\partial x_i} = S' \qquad (8)$$

where $E(f;x,t) = \int F(f,\theta;x,t)d\theta$ is the one-dimensional spectrum, $S' = \int Sd\theta$, and $\overline{V}$ is the directionally averaged group velocity in the direction of the wind, i.e.,

$\overline{V}_i = \frac{\int V_i F d\theta}{E}$, which can be shown to equal $\frac{8}{3\pi} \cdot \frac{g}{4\pi f} \cdot \frac{u_i}{|\overline{u}|}$,

where $\overline{u}$ is the wind velocity.

2. The one-dimensional energy spectrum can be approximated by the expression suggested by Hasselmann (JONSWAP, 1973), i.e.,

$$E(f)=\alpha g^2(2\pi)^{-4}\ f^{-5}\{\exp - \frac{5}{4}(\frac{f}{f_m})^{-4} + \ln\gamma.\exp[-\frac{(f-f_m)^2}{2\sigma^2 f_m^2}]\} \qquad (9)$$

where

α = Phillips' "constant"

f_m = frequency of the spectral peak

γ = peak enhancement factor (over a Pierson-Moskowitz spectrum having the same α, f_m; see Figure 8)

σ = a peak width parameter = σ_a if $f>f_m$

σ_b if $f<f_m$

The one-dimensional spectrum (equation 9) provides 5 parameters, α, f_m, γ, σ_a σ_b. The energy balance equation 5 can be transformed to a parametrical form

$$\frac{\partial a_i}{\partial t} + D_{ijk}\frac{\partial a_j}{\partial x_k} = S_i'; \qquad k = 1,\ 2 \quad i,j = 1,2,3,4,5 \qquad (10)$$

where $a_1 = f_m$, $a_2 =$, $a_3 =$, $a_{4,5} =$ a,b and the propagation velocities D_{ijk}^m and the source terms S_i, are now functions of these 5 parameters. The details of the procedure for the projection of the energy balance equation onto the JONSWAP parameter space are given in Hasselmann, et al. (1976).

The five equations in 10 are numerically intergrated by using the Leap-Frog difference scheme. At and near the model boundaries, the finite difference equations have to be modified because the variables at some grid-points are not available. For incoming boundaries, where wind is directed into the model, the NONSWAP fetch limited equations for the evolution of the wave spectra (Hasselmann, et al., 1973) are used with a fixed fetch length of one-half of the grid spacing. The outgoing boundaries are modeled to reflect no energy. Moreover, it is not generally necessary to solve for all five parameters; some may be held fixed, or directly related to others. Ewing, et al. (1979) reported negligible increase in accuracy and enormous increase in computation when using all five parameters instead of fewer; in this project, consequently, -- and -- were held fixed.

A complete description of the programs is given in a Hydraulic Research Station Report (1977), and a revised version (Puri and Pearce, 1981) was used. The values of the parameters a_i solved at a point are substituted into equation 9 to obtain the spectrum at that point.

The above parametrical model can be applied only to the wind-sea region of the spectrum, characterized by the wind-sea parameter $\nu = \frac{f_m U}{g} \geq$ 0.13. For wind speed $U \frac{0.13}{f_m} g$, the phase velocity of the wave exceeds the wind speed and then is assumed that no energy is absorbed from the atmosphere. The atmospheric input needed to support the spectrum at a level at which the wave-wave interactions become effective is thus lacking, and the wave field propagates freely as swell. The problem therefore reduces to solving the transport equation 5 without a source function input (i.e., S = 0). The transport equation can be modeled by using a finite difference scheme. This has the disadvantage of requiring the use of the high-order finite different schemes (Gunther, et al., 1979) which, moreover, are not amenable to the inclusion of refraction effects.

The alternative is to represent the swell field on a set of characteristics (rays). For each swell frequency f required, the model is covered with a mesh of rays at appropriate angles and spacing. Swell is represented as discrete energy packets at points spaced along each ray at intervals of $L = \Delta t.v$ where Δt is the model time step and v is the group velocity at frequency $f (v=g/4\pi f$ in deep water). The characteristics (shown in Figure 13 for the final grid) are thus divided into bins, and these bins resemble the way an energy parcel propagates during each time step. As the swell field is to interact with the Cartesian grid, it is necessary to transform values from the characteristics grids to the Cartesian. The method used can be summarized as follows: Energy density changes defined on the Cartesian wind-sea grid are transferred to the first ray point of each segment by linear interpolation from the surrounding four Cartesian grid points. The energy density changes at the remaining points are obtained by linear interpolation along each ray. For the inverse process the energy density at a Cartesian grid point is simply taken to be the value at the nearest ray point.

The overall model described here is thus a hybrid model in which the wind-sea component of the energy spectrum is treated by a parametrical approach and swell by a characteristic ray method. Energy needs to be transferred between the two fields as either swell becomes absorbed by a growing wind-sea in a rising wind, or energy from the wind-sea is lost to swell when the wind falls. For instance, if an additional energy peak in the wind-sea frequency domain is caused by incoming swell from a distant source, it will be attenuated by wave-wave interactions, and the energy and momentum of the swell will be partly distributed over a larger frequency-direction space or dissipated due to turbulence. Dynamical criteria for such redistribution of energy between the two domains of the wave spectrum are formulated on intuitive grounds. First, it is assumed that the total energy (wind-sea and swell) is conserved in any exchange. Second, the non-linear interaction between swell and wind-sea is very weak unless the swell and wind-sea frequency domains overlap, and when they do, the interactions is such that coupling or decoupling of the two is rapid (i.e., within one model timestep). The following criteria have been adopted for wind-sea to swell transfer:

a. When the peak frequency f_m falls below the peak frequency of a fully developed sea f_{PM} i.e. when $f_m < \frac{0.13g}{U} = f_{PM}$.

   o f_m is set equal to f_{PM}

   o is adjusted to a new value α' such that the spectral energy for frequencies above f_{PM} is conserved in the wind-sea, i.e.

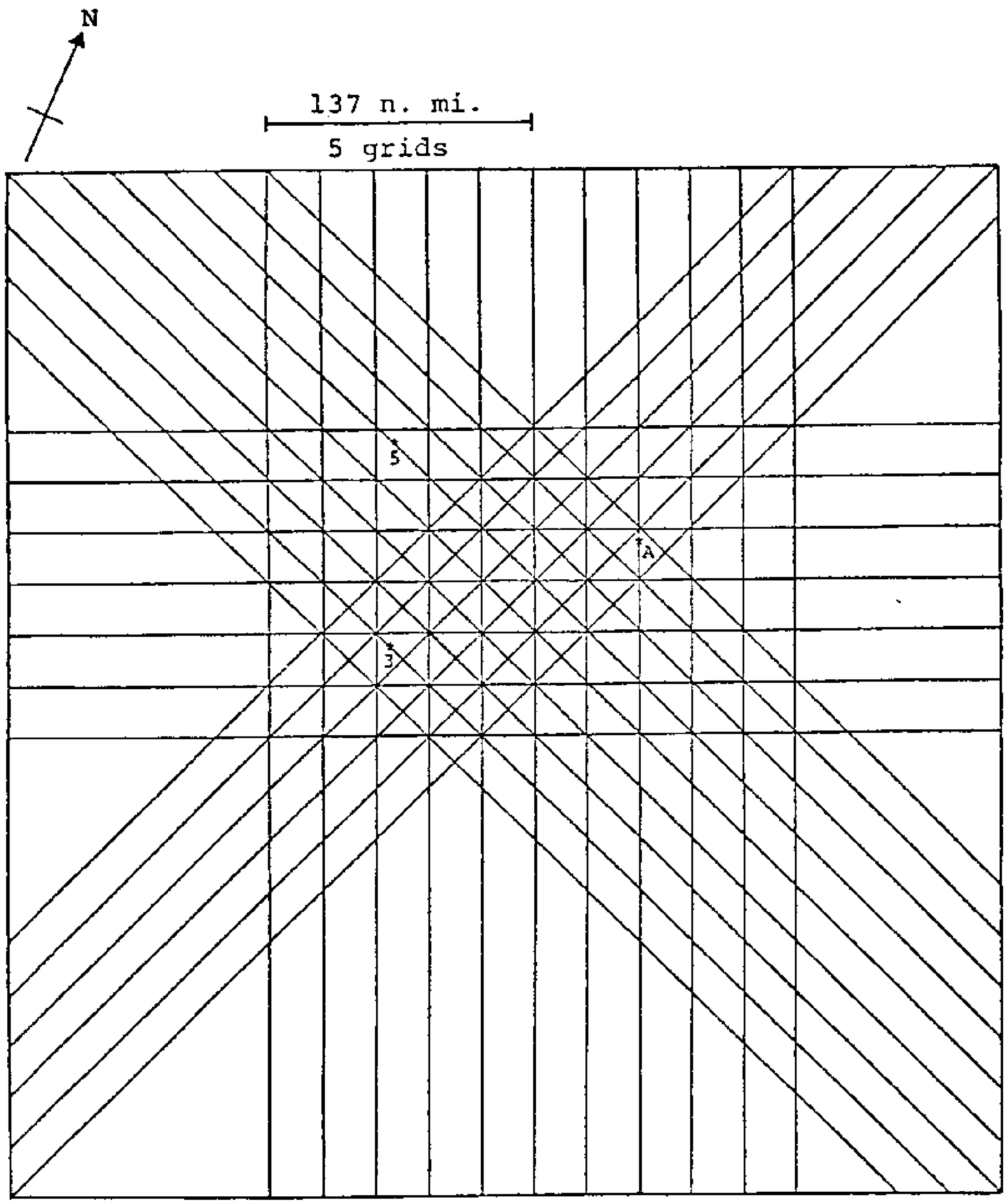


Figure 13. Rays Used to Represent Swell Field. Point A is Discussed in Chapter V, and Points 3 and 5 Show Locations of NOAA Data Buoys 44003 and 44005

$$\int_{f_{PM}}^{\infty} E_{ws} \; (\alpha, f_m, \gamma)\, df = \int_{f_{PM}}^{\infty} E_{ws} \quad (\alpha', f_{PM}, 1)\, df$$

E_{ws} denotes the parametrical wind-sea spectrum.

o Overall energy conservation is achieved by transferring to swell the wind-sea spectrum for $f<f_c$ where the cutoff frequency f_e defined so that the energy in the remaining wind-sea spectrum is equal to the energy in the original wind-sea spectrum for $f>f_c$,

$$\int_{f_c}^{\infty} E_{ws} \quad (\alpha, f_m, \gamma)\, df = \int_{0}^{\infty} E_{ws} \quad (\alpha', f_{PM}, \gamma)\, df$$

α' and f_c in the above integrals are computed numerically.

b. The reverse exchange of swell to wind-sea is called into play whenever swell energy is found at a frequency greater than $.9f_m$. The swell within this frequency range is assumed to be instantaneously absorbed (i.e. within one model time interval) into the wind-sea irrespective of the direction. The energy is conserved by adjusting the frequency of the wind-sea peak for f_m to f'_m keeping α, γ fixed, i.e.

$$E_{swell} + \int_{0}^{\infty} E_{ws} \; (\alpha, f_m, \gamma)\, df = \int_{0}^{\infty} E_{ws} \; (\alpha, f'_m, \gamma)\, df$$

where E_{swell} is the total swell energy transferred. The details of the process are given in Gunther, et al. (1979). The transfer process is illustrated in Figures 14 and 15.

Once E (f) has been calculated, this information is used to compute the significant wave height H_S and the zero crossing period T_z. Writing $m_n = \int_0^\infty E(f)\, f^n\, df$, the above quantities are defined as,

$$H_S = 4\, m_o^{1/2} \tag{11}$$

$$T_z = (m_o / m_2)^{1/2} \tag{12}$$

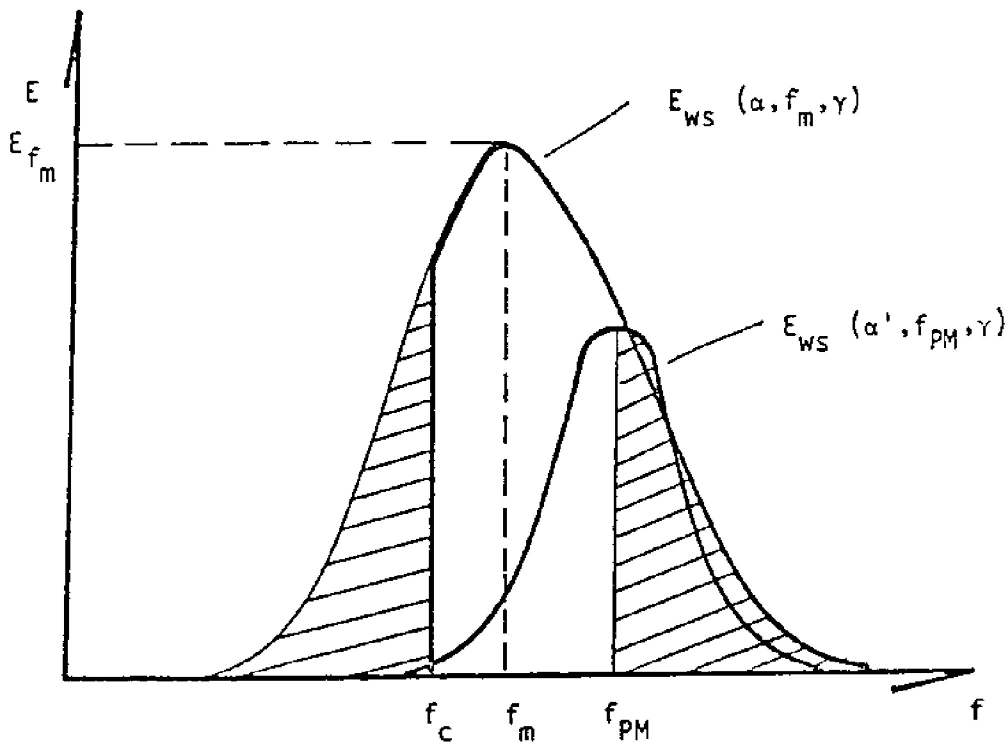


Figure 14. Transfer of Wind-Sea to Swell. (From Puri and Pearce, 1981)

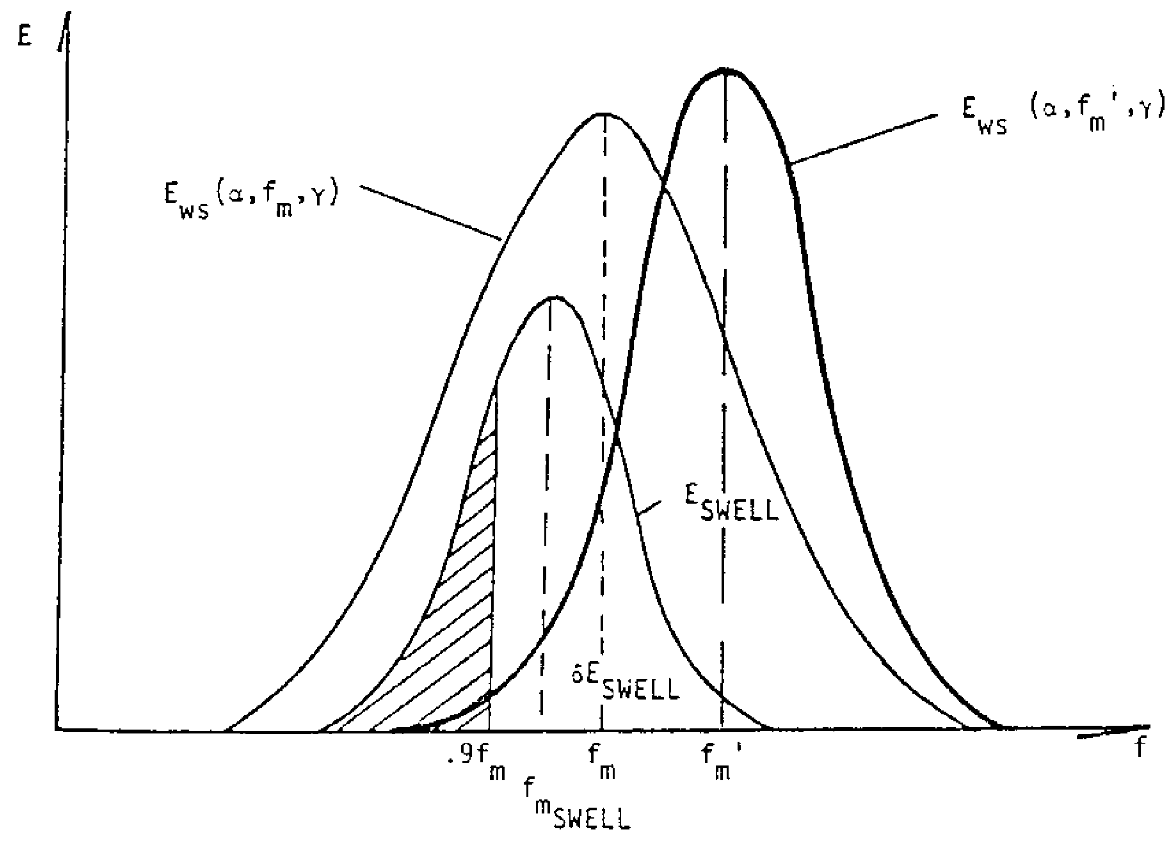


Figure 15. Transfer of Swell to Wind-Sea (From Puri and Pearce, 1981)

The JONSWAP spectrum is not analytically integrable for $\gamma \neq 1$ and so the quantities m_o, m_2 are evaluated numerically. To facilitate this, first an integration by parts is used to obtain

$$m_o = g^2 \quad (2\ )^{-4} \quad 1/5 \ (I+1) \tag{13a}$$

where $I = \int_{-1/\sigma}^{\infty} \exp[-\frac{5}{4}(1+y\sigma)^{-4}]y \quad \exp[-\frac{y^2}{2} + \ln\gamma \ e^{-y^2/2}] dy$

with

$$y = \frac{f-f_m}{\sigma f_m}$$

I is regarded as a function of α and γ only and a two-way table can be formed for I in terms of γ and σ, and H_S then calculated using the equation (11).

Similarly, m_2 can be written as

$$m_2 = \alpha g^2 \ (2\pi)^{-4} \ f_m^{-2} \ J \tag{13b}$$

where

$$J = \int_0^{\infty} x^{-3} \exp\{-\frac{5}{4} + \ln\gamma \quad \exp[\frac{-(x-1)^2}{2\sigma^2}]\} \ dx \ ; \ x = \frac{f}{f_m}$$

is again a function of α and γ.

Again a two-way table for J can be set up to calculate m_2 from the equation (13b). An application of the equation (12) then yields T_z.

The above approach produces H_S and T_z from the wind-sea. Swell energy is incorporated by first summing over the directions to produce swell frequency bins and then using the formulae for m_o and m_2. These are then added to the m_o and m_2 produced for the wind-sea and the final H_S and T_z for the total energy then follows.

CHAPTER IV

EXTRAPOLATION OF HINDCAST DATA FOR ESTIMATING DESIGN WAVE HEIGHTS

Hindcast wave height information for the design wave studies usually covers a period of historical record that is shorter than the return period selected for acceptable engineering risk. Return periods commonly used for selection of design waves are 50 years, 100 years or more, but good meteorological data, on which the calculated wave heights are based, can rarely be obtained for periods covering more than 50-60 years. As a consequence, extrapolations to longer return periods are necessary. The method suggested by Petruaskas and Aagard (1970) for making such extrapolation is employed here.

Most methods for making the extrapolation to longer return periods employ probabilistic models through the use of special probability graph papers on which a family of distribution functions plot as straight lines. The wave heights are plotted against their "plotting position" return period, and a straight line fitted to the plotted data is extended beyond the data to estimate extreme wave heights for return periods of interest (Gumbel, 1958). The drawbacks of these methods are: (1) the straight line drawn through the data is in most cases visually fitted to the data, and is thus subject to error; and (2) no information is available on the uncertainty of the resulting extrapolation. In contrast, the method used here offers (1) greater flexibility in the choice of distribution through computerized procedures, (2) guidelines for picking the "best" distribution from several implemented in the method, and (3) procedures for estimating the uncertainty of extrapolated wave heights.

In the following, those facets of the Weibull distribution that were used for actual working are described. Other details may be found in Petruaskas and Aagard (1970). The primary function of the present extrapolation method is to find a conditional distribution for wave heights fitting input hindcast wave height data. The following restrictions apply to the input data:

(1) The data must have a common definition and be derived by consistent methods, i.e., all "expected maximum wave heights" $E(H_{max})$ must be obtained from the same wave height calculation model. In this project, all wave heights are computed using the hybrid parametric model based on Hydraulics Research Station (1977).

(2) The $E(H_{max})$ value for each storm must be the largest $E(H_{max})$ at the site of interest.

(3) The data set must include all storms occurring in the hindcast period that exceed the threshold of the conditional distribution. A common practice is to specify a hindcast of the N largest storms that effect the site of interest. The threshold of the conditional distribution is then the smallest $F(H_{max})$ value greater than the threshold value.

(4) All storms must be from the same population. For example, hurricane generated waves should not be mixed with waves generated by frontal systems, etc. In this project, for instance, waves generated only by Northeast storms were considered.

Incidentally, it may be noted that the method is not applicable where the wave height population is truncated by natural phenomena such as breaking-height limitations in shallow water.

The Weibull distribution functions used here to describe the conditional probability that the $E(H_{max})$ is less than or equal to x meters, given that such an event occurred, is defined by

$$\text{Cond Pr } [E(H_{max}) \leq x] = 1 - e^{-\frac{(x-B)^k}{A}} \qquad (14)$$

where

Cond Pr [ ] = conditional probability statement

A, B, & k are parameters to be determined.

It may be noted that if k = 1, an exponential distribution results, and if k = 2 and B = 0, a Rayleigh distribution results. Equation 14 leads to the plotting-position formula

$$\text{Cond Pr } [E(H_{max}) \leq x_{m,N}] = 1 - \frac{m-\alpha}{N+\beta} \qquad (15)$$

where

N = sampling size

m = rank. The wave heights are ranked in descending order, so that m=1 refers to the highest $E(H_{max})$.

$x_{m,N}$ = height of the m^{th} highest out of N sample values of $E(H_{max})$.

α, β = constants that are functions of the distribution for which the parameters are being determined.

The parameters α and β are functions of k and N - 20.

$$\alpha (k) = 0.30 + 0.18/k$$
$$\beta (k) = 0.21 + 0.32/k \quad (16)$$

A value of k is chosen and and are calculated with equation 16. The conditional probability is then calculated for each m (m = 1, ----, N). The reduced variate, r_v, is related to the Cond Pr $[E(H_{max}) \leq x]$ for the Weibull distribution by

$$r_v = [- \ln \{1 - \text{Cond Pr } [E(H_{max}) \leq x]\}]^{1}/k \quad (17)$$

and the return period R_p is given by

$$R_p = \frac{1}{\lambda \text{ (Cond Pr } [E(H_{max}) > x])} \quad (18)$$

where

$$\lambda = \frac{N}{\text{time in years}}$$

and Cond Pr $[E(H_{max}) > x]$ = 1-Cond Pr $[E(H_{max}) \leq x]$

The value of k is changed until the reduced variates can be plotted against the wave heights as a straight line (Figure 16). This presentation of the distribution function as a straight line is advantageous in that it indicates how good the distribution is as a probabilistic model of the data and extrapolation to higher wave heights can be done simply by extending the straight line beyond the data.

The computer program that was developed to accomplish this reads the largest wave height at each grid point for each of the 22 storms from a disk file (on which these had been previously written by the program "HSTZ"), and for each point, ranked the 22 value of $E(H_{max})$ in descending order. An initial value of 0.5 was chosen for k and the reduced varieties were calculated using equation 17. It was impractical to plot the wave heights against the reduced variates for each trail of k, and therefore the correlation coefficient was calculated instead. k was then gradually increased until the correlation coefficient reached a maximum. A typical value for the maximum value of the correlation coefficient is 0.9768. (Theoretically, a coefficient of 1 indicates an exactly linear correlation between the wave heights and the reduced variates.)

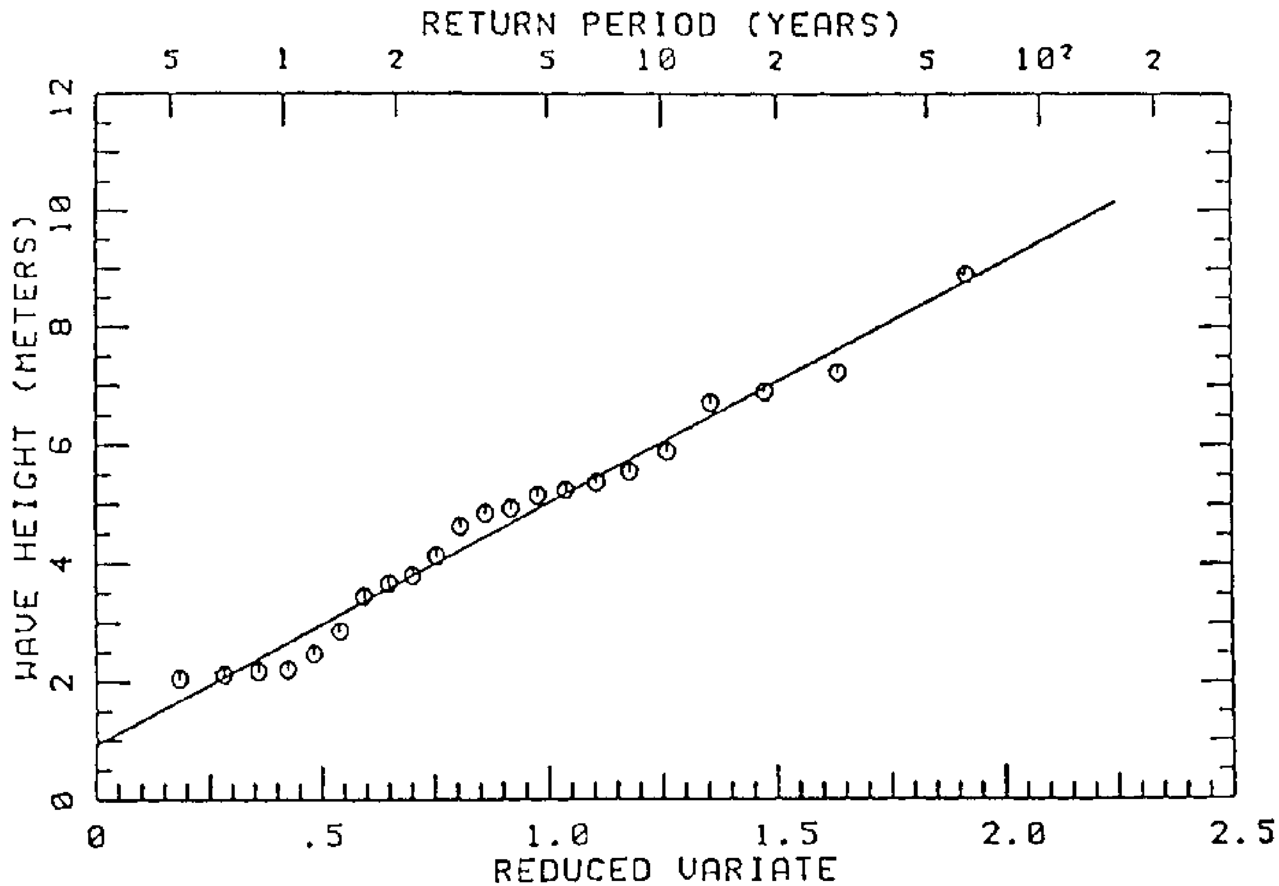


Figure 16. Typical Wave Height Distribution

CHAPTER V

ANALYSIS AND RESULTS

It was mentioned in Chapter II that the Northeaster program solved the geostrophic wind velocity equations for each time-step at each point of the grid system. The wind model and the wave model discretized the corresponding equations using the same grid. The extent of the grid was chosen to approximately correspond with the average length scale of a Northeast storm, i.e., 500-1000 nautical miles. The finite difference scheme discussed in Chapter II inputs wind speed and direction at each point and computes the significant wave height, H_S, for that point.

The boundary conditions play an important role in the choice of the grid. Wave heights calculated near the model boundary are not appropriate because of the limitations on the fetch imposed by the presence of the boundary. When wind is directed into the model at a boundary, the boundary is assigned near-zero energy a priori, and a fixed fetch equal in length to one-half of the grid spacing is used to generate waves in the first grid. Consequently, acceptable estimates of the wave heights are obtained only some distance away from the boundaries. A sensitivity study was therefore done by progressively expanding the grid until it was so large that the area of interest was free from the fetch-limiting effects of the model boundaries.

The original grid is shown in Figure 17, and the entire region encompassed by it is shown in Figure 18. Each square is 32.9 nautical miles on a side, and the mesh contains 11 columns. The 100-year return period wave heights computed with this grid are shown for the western points of this grid in Figure 19. Values at all points of the grid are not shown because the proximity of the model boundary to the Gulf reduces the computed wave heights there by limiting the fetch. It was felt that the points where the values are shown in Figure 19 were far enough to the west of the boundary to have no such fetch limitations, and that the computations were correct at these western points. These were subsequently found to be in error; even the most western points on this grid were effected by the effects of the boundary, as will be seen in the following paragraph. Nevertheless, this run did serve to illustrate this aspect of the working of the wave model (compare Figure 19 and Figure 25) and to eventually free the Gulf of Maine from the boundary effects.

A second attempt to compute the wave heights was made by expanding the grid to the east. This grid contained squares 27.4 nautical miles on a side, and it had 21 rows. The eastern boundary of the mesh is indicated by A-A in Figure 20. The wave heights calculated were now different from the values obtained in the first attempt. The

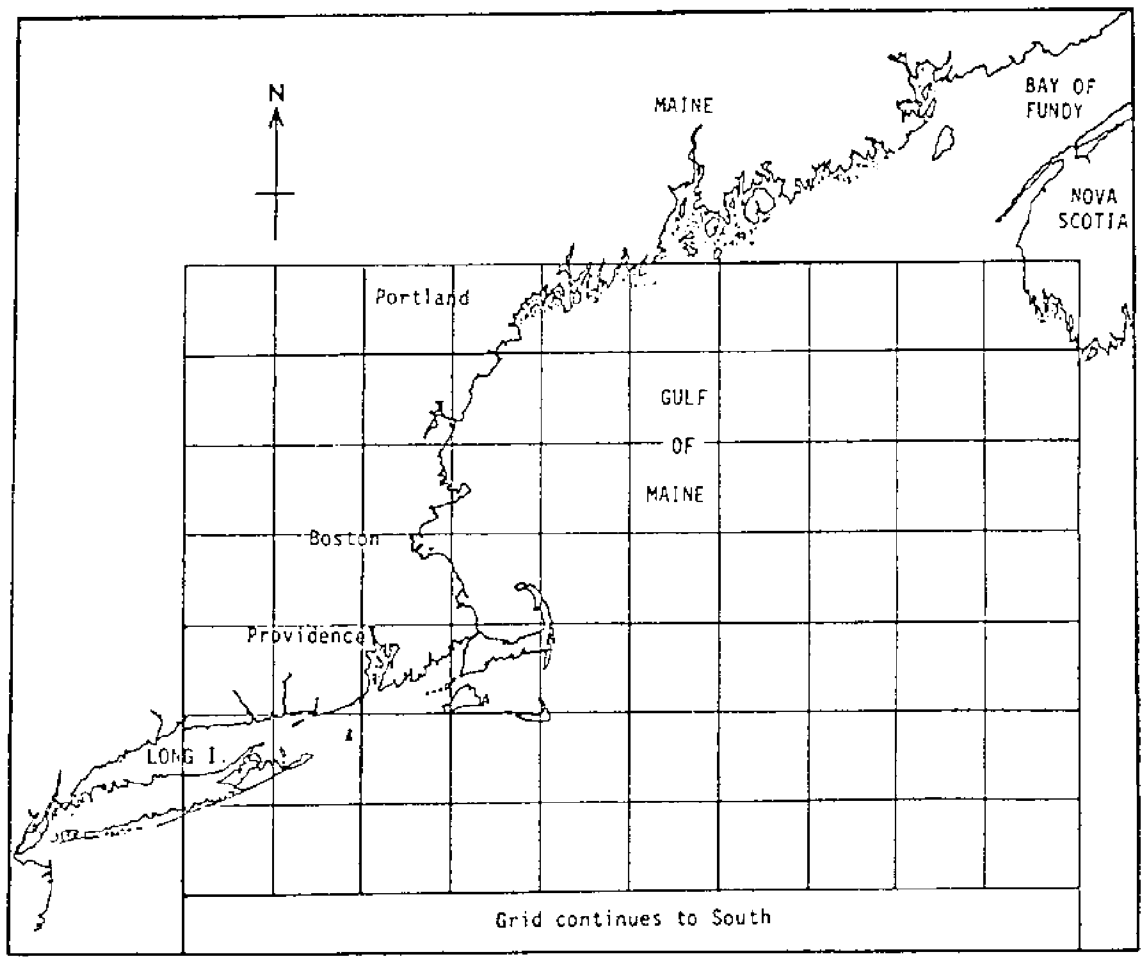


Figure 17. Initial Grid. (Refer to Figure 18 for entire grid.)

computations gave larger wave heights due to the increased fetch. This second grid was then extended further to the east to 26 columns (B-B in Figure 20). A comparison of wave heights in meters along a row (generated by the storm of February 2, 1976 for the seventh and thirteenth hours of the simulation) computed using grids A-A and B-B is presented (Figure 21 and 22). It can be seen that, while the wave heights in the eastern columns are different, there is little or no change in about the first 12 columns which cover the Gulf of Maine, indicating that an extension in the east-west dimension of the grid from 20 to 26 columns does not alter the computed wave heights of the Gulf of Maine. It can therefore be inferred that the area of interest is free of the fetch limiting effects of the boundaries when the second grid system was eventually chosen to calculate the extreme wave conditions. An enlarged view of this grid in the Gulf of Maine is shown in Figure 23, and the computed 50-year and 100-year return period wave heights at the grid points within the Gulf of Maine are presented in Figures 24 and 25.

A brief analysis of the uncertainty of the extrapolated wave heights was done. Neglecting possible inaccuracies in the hindcast data, uncertainty in the wave height, corresponding to a given return period, exists because the storms used in the extrapolation procedure represent only a small sample of the total storm population. The small sample size causes two problems in developing an adequate probabilistic model for estimating design wave heights: (1) we cannot reliably pick a "true" distribution function defining the storm population, and (2) the parameters calculated are only sample estimates. It is therefore desirable to obtain confidence limits for the extrapolated values. Point 308 of the model (where the 100-year significant wave height is 10.72 meters in Figure 25) was arbitrarily chosen for this uncertainty analysis, and the 90% confidence limits were constructed (Figure 26) by the method illustrated in Viessman, et al. (1977) and attributed to Beard (1962). It was found that the 90% reliability band for the 100-year significant wave height at point 308 is between 9.81 meters and 12.21 meters.

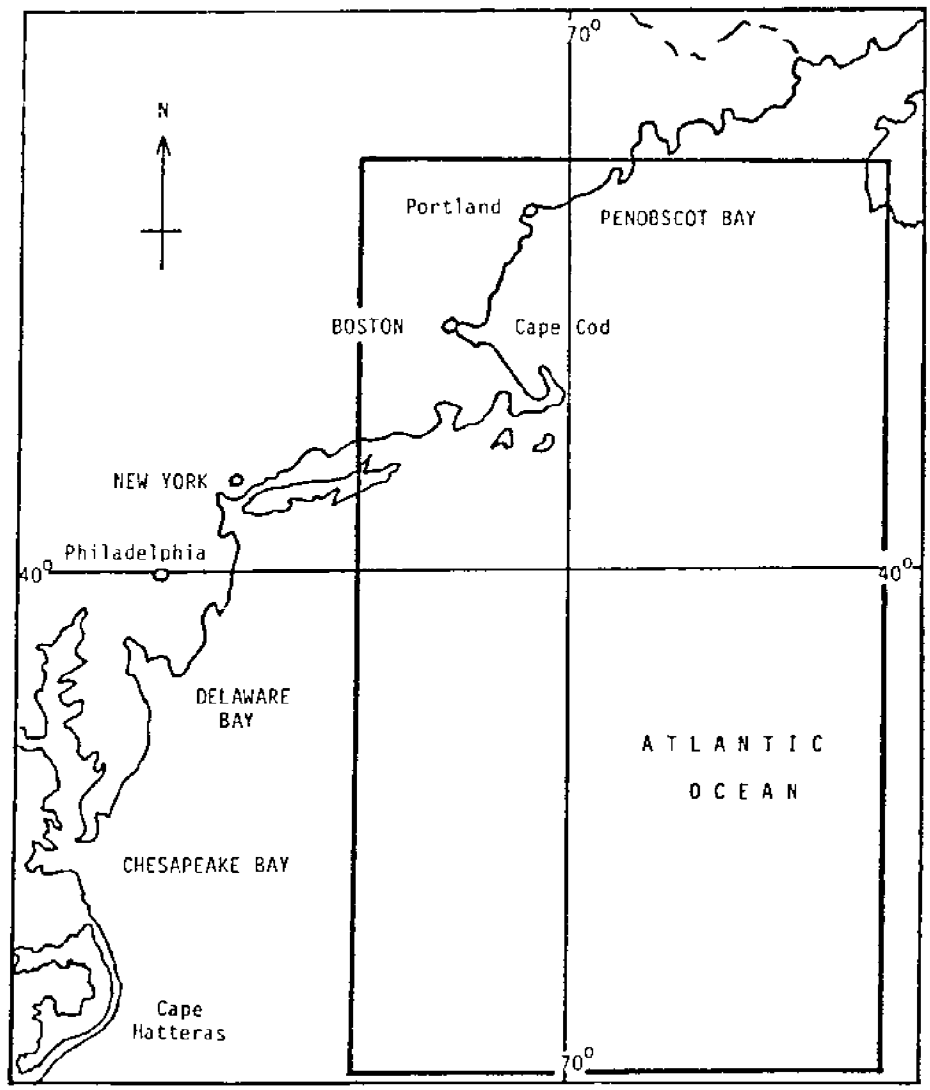


Figure 18. Entire Region of Initial Grid.

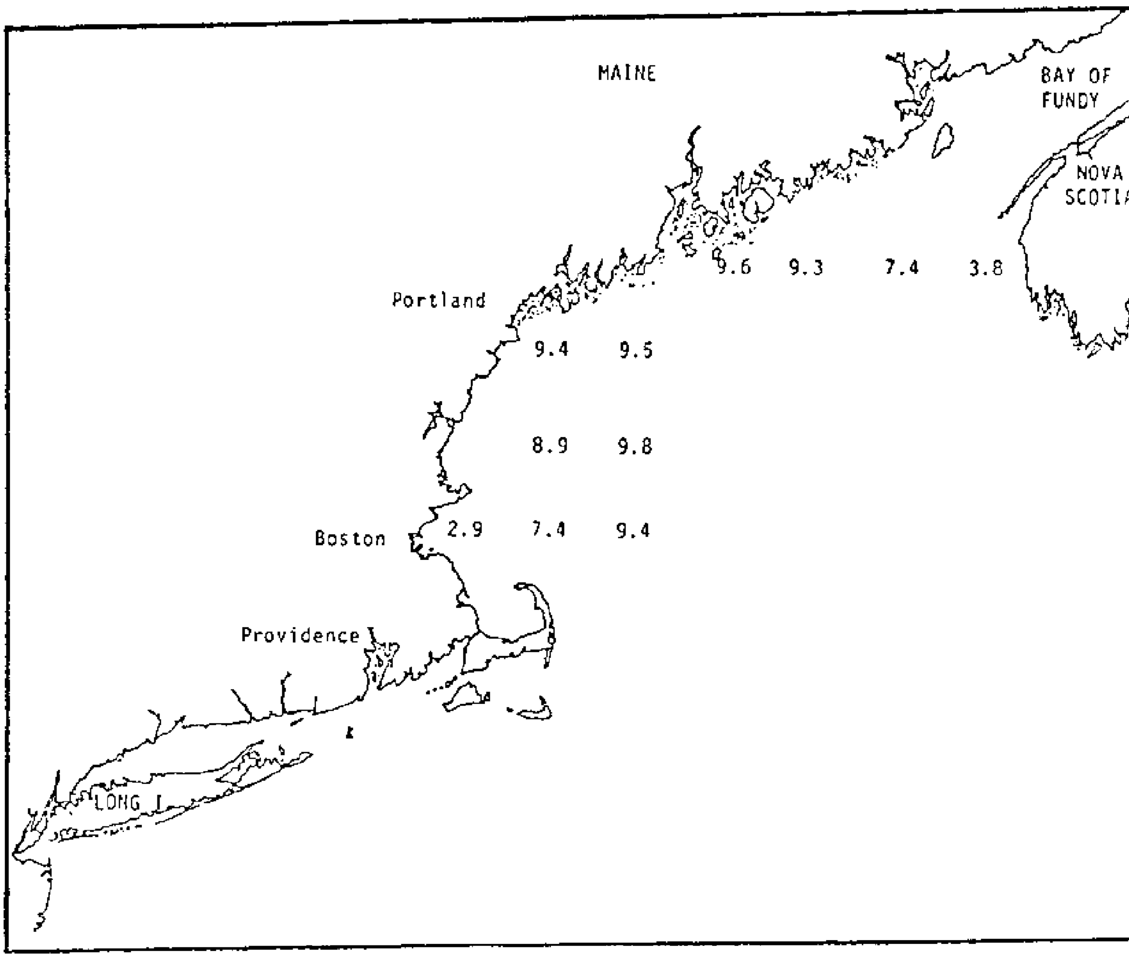


Figure 19. 100-year Significant Wave Heights (in meters) Computed Using Initial Grid

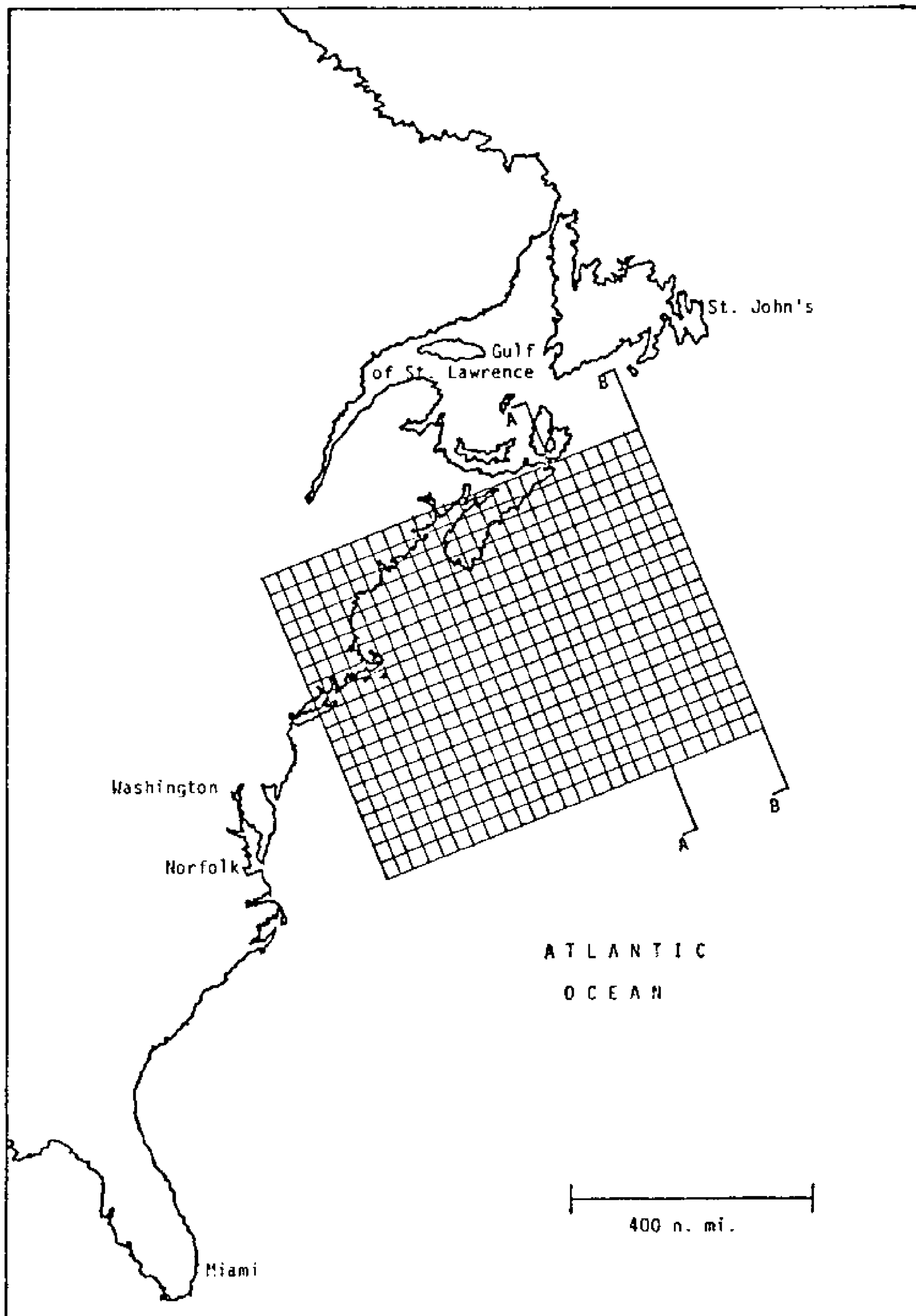


Figure 20. Second (A-A) and Third (B-B) Grid Systems

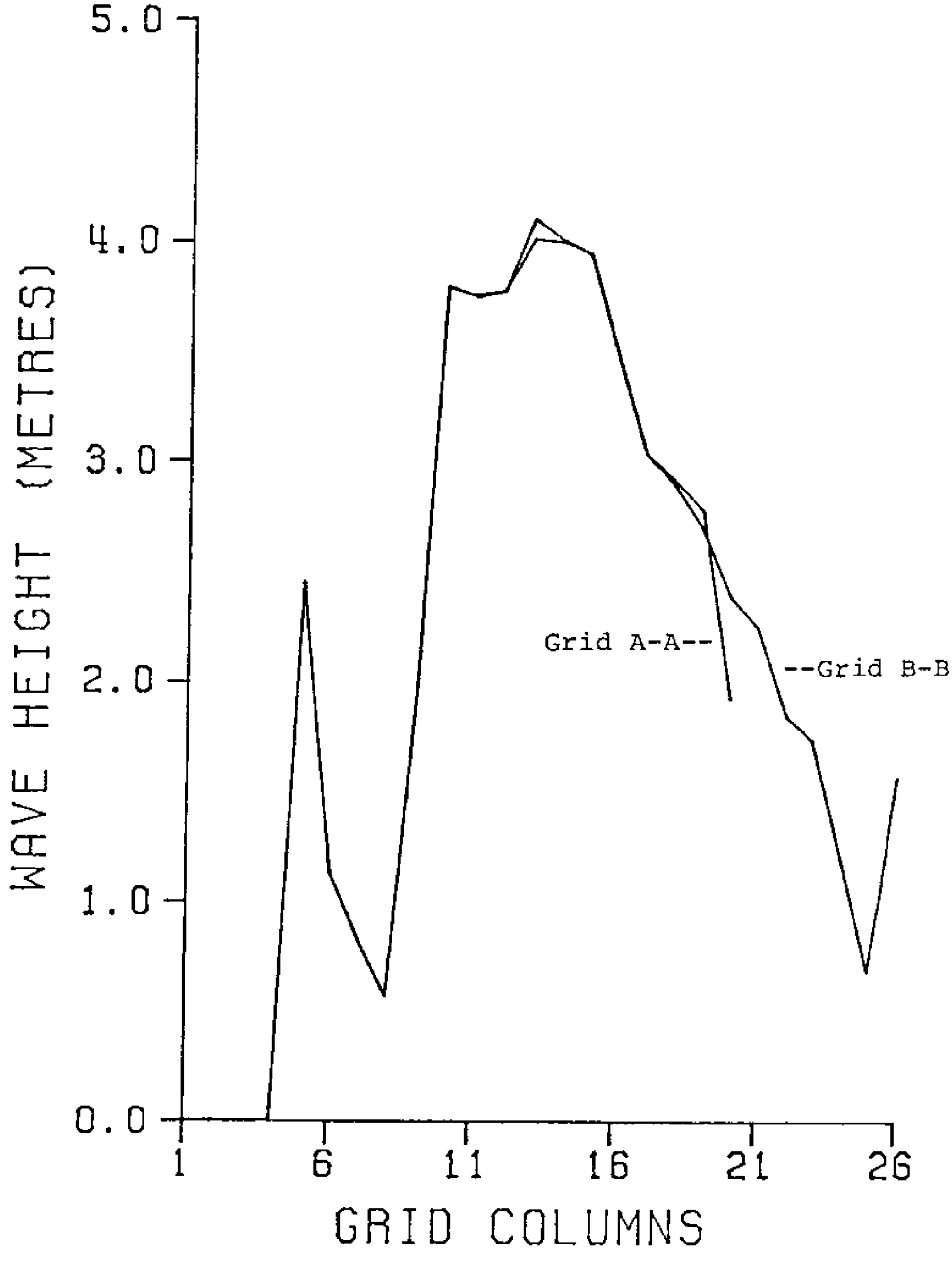


Figure 21. Wave Height Comparisons of Grid A-A and B-B. (Row 16, Hour 7)

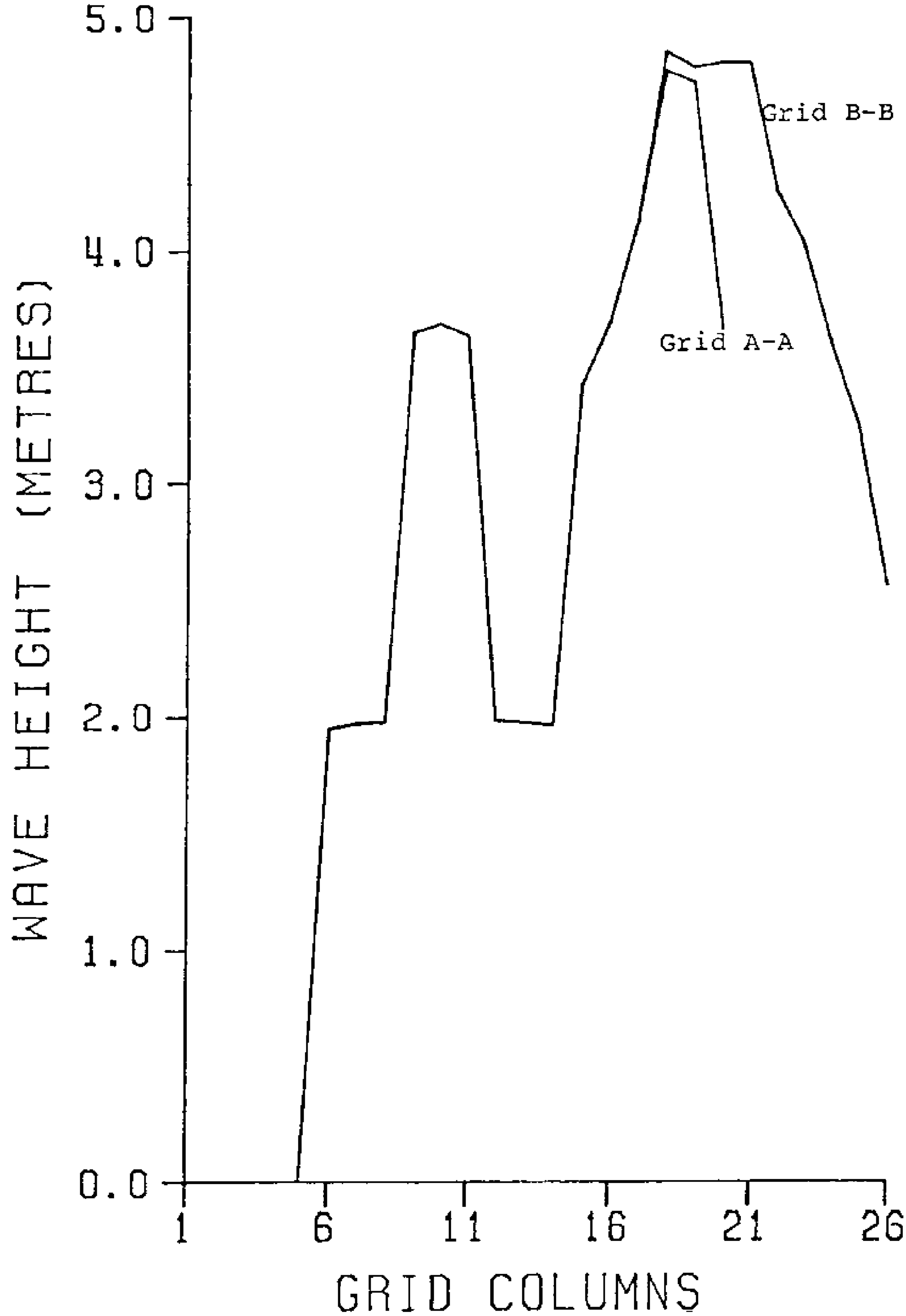


Figure 22. Wave Height Comparison of Grids A-A and B-B. (Row 17, Hour 13)

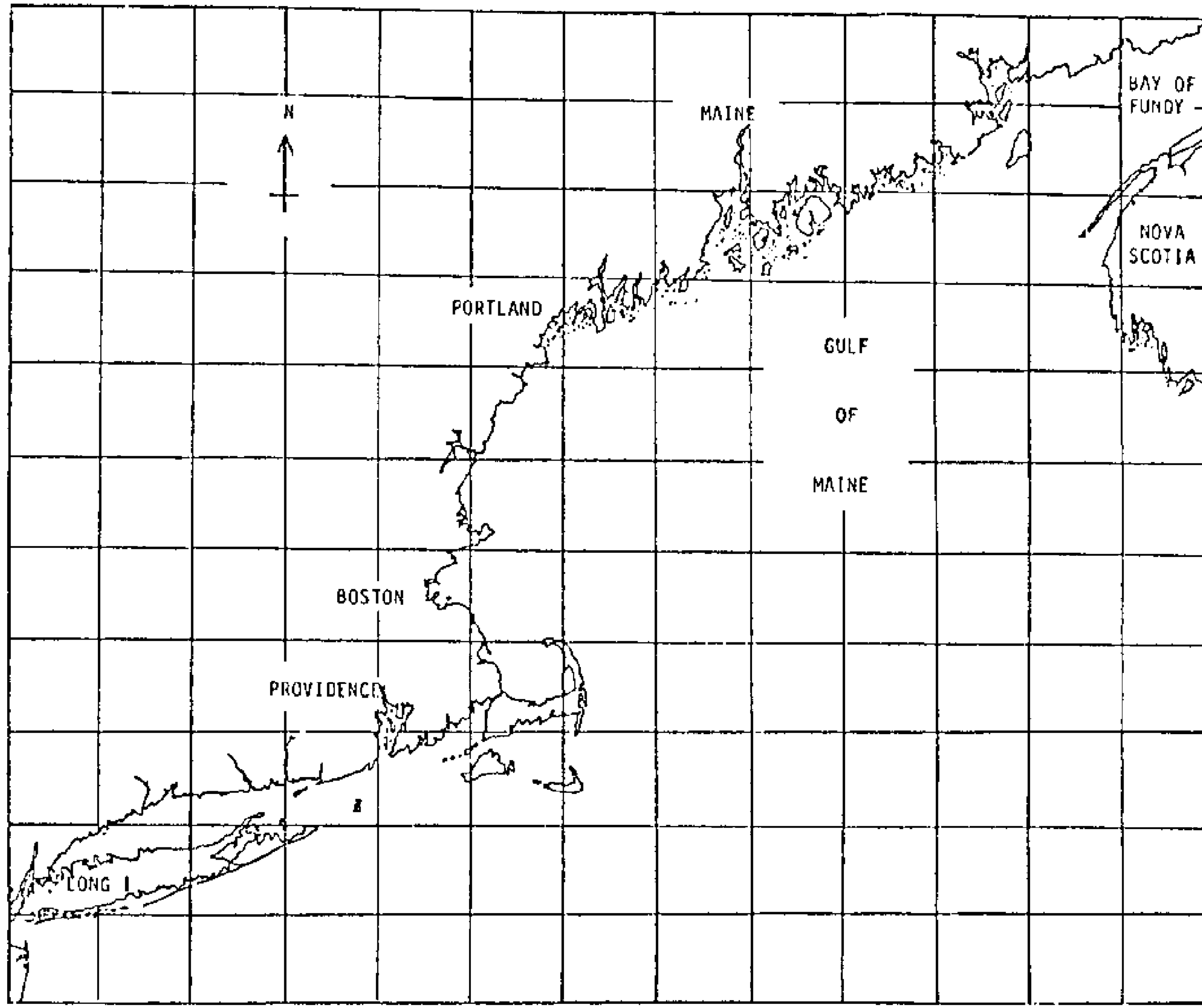


Figure 23. Enlarged View of Grid A-A Within the Gulf of Maine

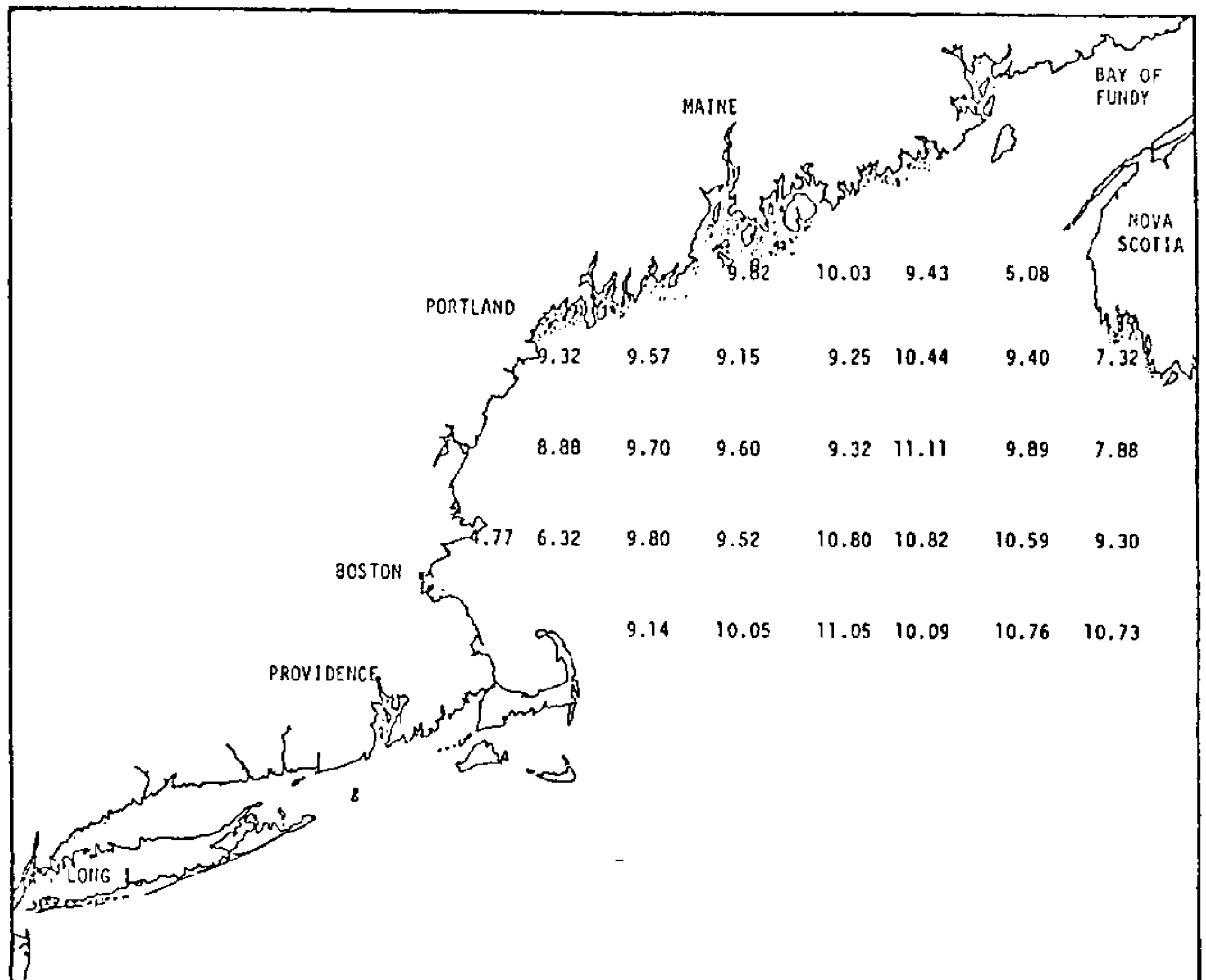


Figure 24. 50-Year Significant Wave Heights (in meters) in the Gulf of Maine

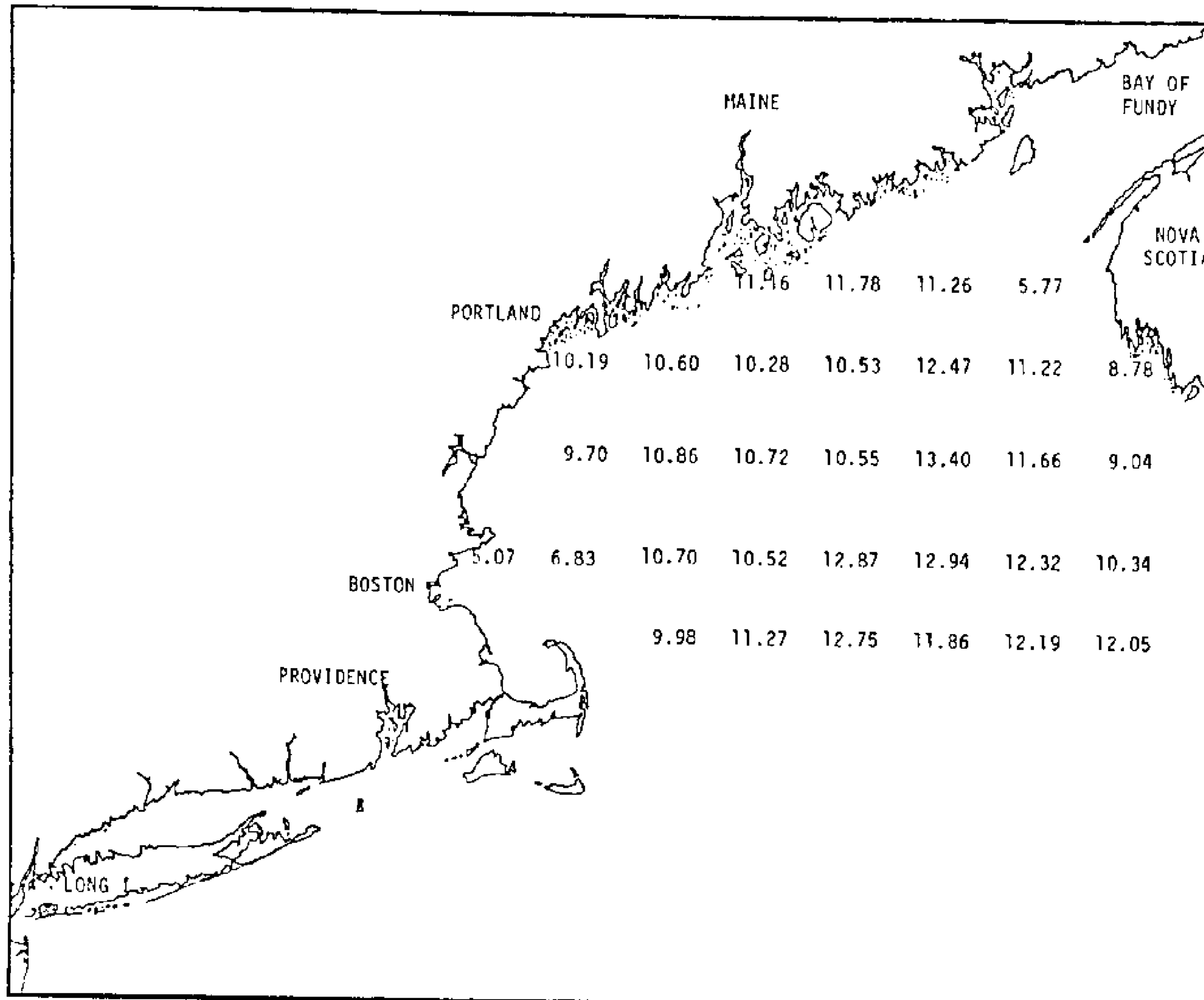


Figure 25. 100-Year Significant Wave Heights (in meters) in the Gulf of Maine

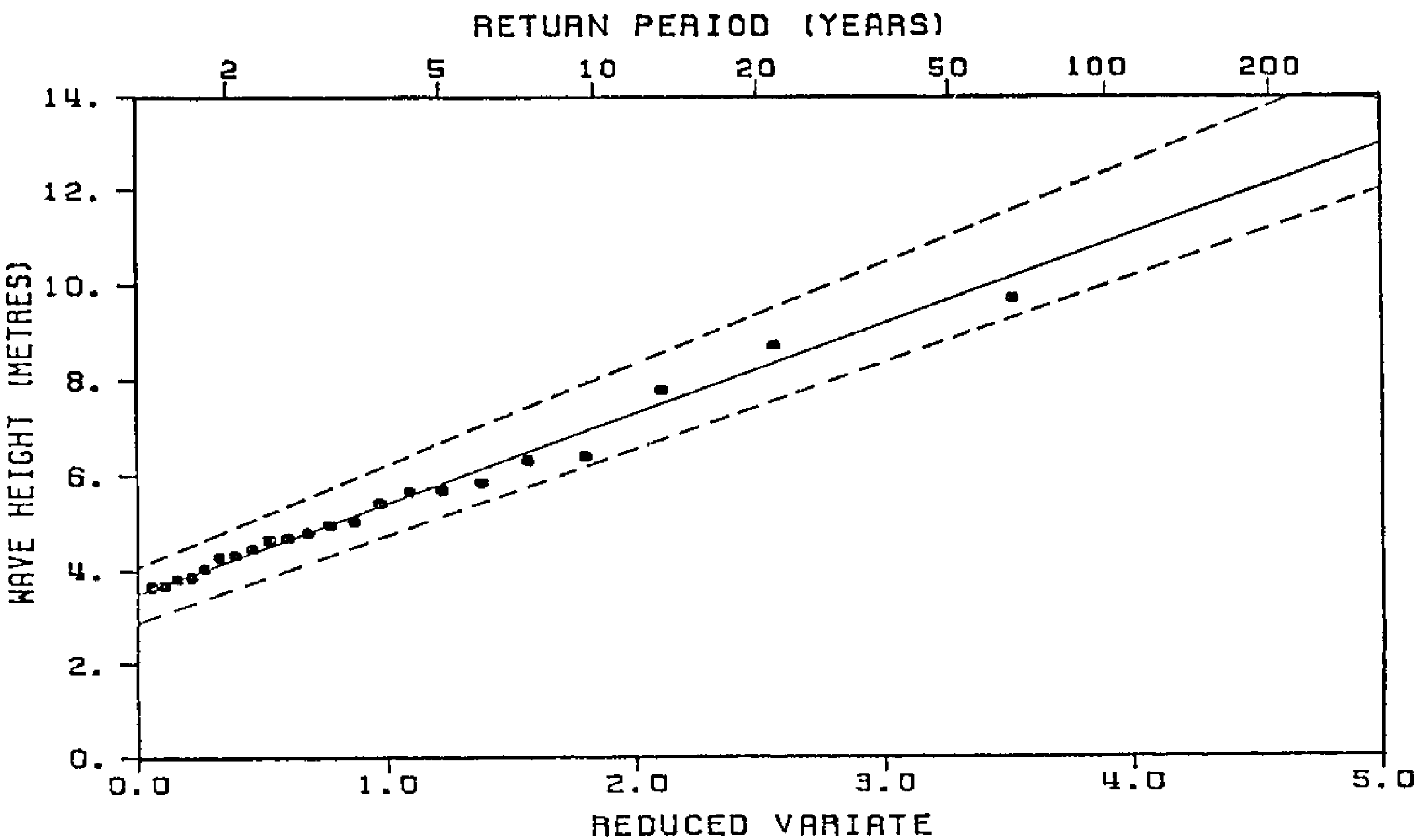


Figure 26. Wave Height Distribution for Point 308. The Dashed Lines Represent the 90% Reliability Band

CHAPTER VI

SUMMARY AND DISCUSSION

This project was initiated in order to generate some information about the wave climate in the Gulf of Maine. Such knowledge is needed because little quantitative wave data exists that can be used for engineering purposes in the Gulf of Maine. The availability of past meteorological information, however, enables the hindcasting of waves through the application of a suitable wave model. The hindcast wave information is then used to compute extreme wave statistics like 50-and 100-year return period waves.

Meteorological information used in this project consisted of synoptic surface pressures for Northeast storms. Twenty-two strongest Northeasters in the period between 1944 and 1976 were chosen for the computation of the design wave heights. The "Northeaster" model (Stone and Webster Engineering Corporation, 1978a) was used to simulate the complex pressure pattern of a Northeast storm by fitting a system of equations to the isobars. Wind velocities were calculated using the geostrophic approach, and these wind velocities were then used as input to a hybrid, parametric type wave model. This model was chosen in preference to a spectral model because it involves fewer computations and recognizes the redistribution of energy among different frequencies due to non-linear interactions as an important part of wave dynamics. It has also been verified with reasonably good results in the Gulf of Mexico (Puri and Pearce, 1981) and near the British Isles (Ewing, et al., 1979).

A sensitivity analysis was done by progressively enlarging the model grid before an adequate grid could be chosen. Thereafter, the wind and wave models were run repeatedly for the 22 strongest Northeast storms in the period between 1944 and 1976. For each run, the largest wave height at each grid point was obtained. Finally, for each point, a Weibull distribution was fitted to the set of the largest wave heights and the 50- and 100-year wave heights were calculated.

The results of this project were presented in Figures 24 and 25. It might have been interesting to compare the 100-year wave height at a point in this grid system computed by the U.S. Army Engineer Waterways Experiment Station (Corson, et al., 1981) with the results of the project. Regrettably, such a comparison was rendered impossible by the fact that point A (Figure 27), where the computation was made by the Army Engineers, was outside the Gulf of Maine, and large wave heights in that area can be generated by storms other than those chosen for this simulation. Large waves in the Gulf of Maine are primarily caused by Northeast storms, and a computer-generated plot of the tracks of the storms analyzed in this project is shown in Figure 27. While the

50-year and 100-year wave heights at point A are dependent on these Northeasters, they are also dependent on other storms within whose area of influence this point lies, and as these have not been considered in this project, a computation of the design wave heights at A would be improper. Nevertheless, it is interesting that the Army Engineers' computation of the 100-year significant wave height at A yielded 14.1 meters, which is slightly higher than the calculated wave heights in the Gulf of Maine (Figure 24). This difference can be attributed to the previous argument, and at this point, it would be reasonable to expect the results of the two models to be compatible.

The next logical step in the project would therefore be the inclusion of other types of storms in the construction of wave statistics. It would involve the use of other storm models, such as HURR (Puri and Pearce, 1981) for hurricanes, to obtain windfields. The 50- and 100-year wave heights at other points would be calculated, enabling a comparison of the wave heights at point A (Figure 27) as computed by this model and by the U.S. Army Engineer Waterways Experiment Station (Corson, et al., 1981).

As the area of interest progressively encompasses more easterly points in the grid, it will be necessary to expand the grid even beyond B-B (Figure 20) to the east. This would aggravate another problem that has been neglected so far--the curvature of the earth. In the present study, a Cartesian grid was drawn on a Mercator projection of the East Coast of the United States and the Atlantic Ocean. As the size of the model is increased, the curvature of the earth will cause discrepancies in the distances and positions of grid points in relation to latitudes. This problem must somehow be addressed, perhaps by transforming the coordinate system to a polar or spherical system.

A comparison of the results of this hybrid, parametric approach with those of the spectral approach would also be interesting. The spectral model of Resio (1981) makes a provision for the role of non-linear wave-wave interactions and could be used to compute the wave heights in the Gulf of Maine.

One question that arises is in regard to the propriety of using a deep water wave model, such as the one used here, to compute the wave heights in the shallow Georges Bank region. Georges Bank, well-known as a productive fishing ground, covers an extensive area to the immediate south of the Gulf of Maine. There is a sharp decrease in water depth into this shallow water body, and in some places, the depths are as small as 3.5 fathoms (6.4 meters). The results at some grid points in the Georges Bank area were therefore investigated. The computed wave periods were used to calculate the wave lengths L by linear theory, and the check revealed that d/L 0.5, where d is the depth of the water. This implies that the properties of the modeled waves are not those of deep water waves. The Georges Bank area must be treated differently, and a shallow water wave model (e.g. Shemdin, 1980) may possibly have to be used to compute the wave heights there. It is also not known if the distorted wave height computation in Georges Bank has resulted in a

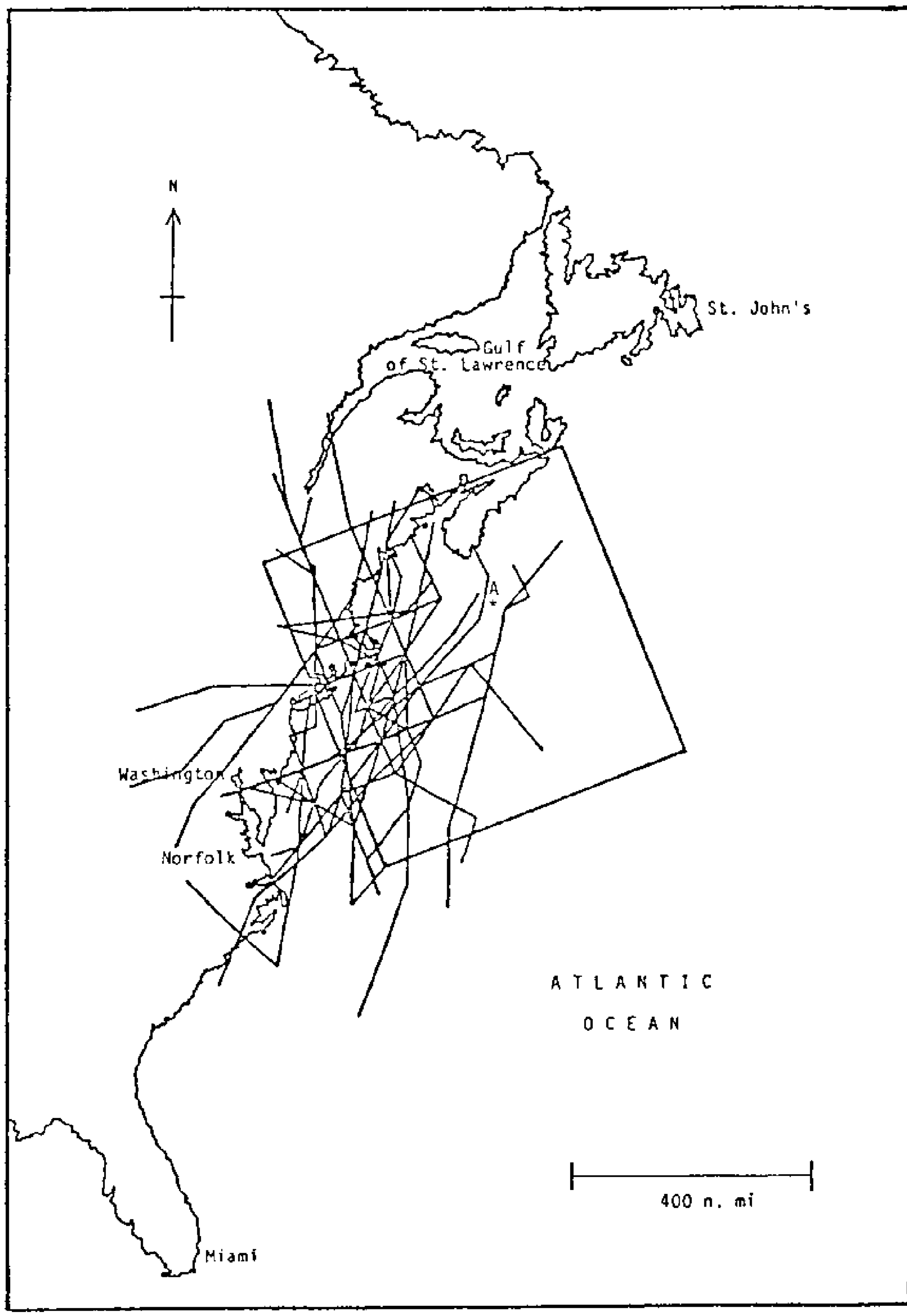


Figure 27. Storm Track Composite

slight over-estimation of the waves in the Gulf of Maine. In that case, the results shown in Figures 24 and 25 represent conservative estimates of the wave heights in the Gulf of Maine.

In conclusion, the results of Puri and Pearce (1981) and Ewing, et al. (1979) give an indication of the reliability of this modeling technique. In both studies, the maximum difference between the model results and the measured values of the significant wave heights was in the range of 3 meters. Investigation of the factors mentioned above will obviously lead to a refinement of the results presented in Figures 24 and 25. Owing to a paucity of data with which to compare the model results, because of an imperfect understanding of the various mechanisms, these numbers are perhaps as good as the state of the art will allow. It is a reasonable premise, however, that any engineering effort in the Gulf of Maine will have to reckon maximum wave heights in the range of 90 feet (corresponding to the maximum H_S = 13.4 meters in Figure 25) for a 1% change of occurrence.

REFERENCES

Beard, L.R. (1962). "Statistical Methods in Hydrology." Civil Works Investigations, U.S. Army Corps of Engineers, Sacramento District, USA.

Corson, W.D., et al. (1982). "Atlantic Coast Hindcast, Deepwater, Significant Wave Information." Report of US Army Engineers Waterways Experiment Station, Vicksburg, Mississippi.

Ewing, J.A., et al. (1979). "A Hindcast Study of Extreme Wave Conditions in the North Sea." J. Geophys. Res., 84: 5739-5747.

Gumbel, E.J. (1958). Statistics of Extremes. Columbia University Press, New York.

Gunther, H., et al. (1979). "A Hybrid Parametrical Wave Model." J. Geophys. Res., 84: 5727-5738.

Hasselmann, K., et al. (1973). Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP). Deutsche Hydro graphische Zeitschrift, Supplement A8, 12.

Hasselmann, K., et al. (1976). "A Parametric Wave Prediction Model." J. Phys. Ocean., 6: 200-228.

Horikawa, K. (1978). Coastal Engineering. John Wiley & Sons, New York.

Hydraulics Research Station. (1977). "Numerical Wave Climate Study for the North Sea." Technical Report 775 (see also 776), Wallingford, England.

Kinsman, B. (1965). Wind Waves. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Long, R. (1973). "Scattering of surface waves by bottom irregularities." J. Geophys. Res., 78: 7861-7870.

Miles, J.W. (1957). "On the generation of surface waves by shear flow." J. Fluid Mech., 3: 185-204.

Peterson, K.R., and Goodyear, H.V. (1964). "Criteria for a Standard Project Northeaster for New England North of Cape Cod." National Hurricane Research Project, Report No. 68, Hydrometeorological Section, Hydrologic Services Division, Weather Bureau, Washington, D.C.

Petruaskas, C. and Aagard, P.M. (1970). "Extrapolation of Historical Storm Data for Estimating Design Wave Heights." Offshore Technology Conference, Houston, Texas.

Petterssen, S. (1958). Introduction to Meteorology. McGraw-Hill, pp. 152.

Phillips, O.M. (1957). "On the generation of waves by turbulent wind." J. Fluid Mech., 2: 417-445.

Phillips, O.M. (1959). "The scattering of gravity waves by turbulence." J. Fluid Mech., 5: 177-192.

Phillips, O.M. (1963). "On the attenuation of long gravity waves by short breaking waves." J. Fluid Mech., 16: 321-332.

Pierson, W.J., and Moskowitz, L. (1964). "A Proposed Spectral Form for Fully Developed Wind Seas Based on the Similarity Theory of S.A. Kitaigorodskii." J. Geophys. Res., 69: 5181-5190.

Puri, K. and Pearce, B.R. (1981). "Numerical Hindcasting of Storm Generated Waves-Users Manual." Technical Report, Departments of Mathematics and Civil Engineering, University of Maine, Orono, Maine.

Resio, D.T. (1981). "The Estimation of Wind-Wave Generation in a Discrete Spectral Model." J. Phys. Ocean., 11: 510-525.

Shemdin, O. (1981). "Shallow Water Models." Transcript of Workshop on Wind-Wave Hindcasting and Forecasting Models, June 3-5, 1981; Gaithersburg, Maryland. National Ocean and Atmospheric Administration.

Stone & Webster Engineering Corporation. (1978). "Determination of Coastal Storm Tide Levels." Technical Report, Boston, Massachusetts.

Viessman, W. (1977). Introduction of Hydrology. Harper & Row, Publishers, New York.

APPENDIX A

Chapters II, III, and IV have dealt with the wind, wave, and statistical models, respectively. A synthesis of these models is obtained through an ordered sequence of steps, which are implemented in the computer as subroutines NOREASTR, TRACK, WAVE, INTEG, HSTZ and WEIBULL, resulting in the required program for hindcasting and design wave height estimation. The programs TRACK, WAVE, INTEG and HSTZ constitute the wave model, NOREASTR is the wind model and WEIBULL calculates the extreme wave statistics. The actual sequence, together with the device numbers of the input/output files containing the various pieces of information at different stages of the computations, is shown in Figures A-1. A description of each of the above subroutines is given in the following pages, along with programming details such as inputting, setting up disc file arrays, dimensioning of variables, etc. The final grid chosen in this project (20 columns x 21 rows) is used to illustrate these details.

A-I.

All programs require the use of direct access file systems. Such a system has an advantage over the sequential file system in that the former permits a programmer to read and write records randomly from any location within a data set and then go directly to any other point without having to process all the records in between.

The 'Read' and 'Write' statements cause transfer of data into or out of internal storage. These statements allow the user to specify the location within a data set from which the data is to be read or into which the data is to be written. The actual specification is made by assigning a reference number to each record. This is done, uniquely, through the 'Define File' statement. Other parameters which must be given in it include the number, size, and type of the records.

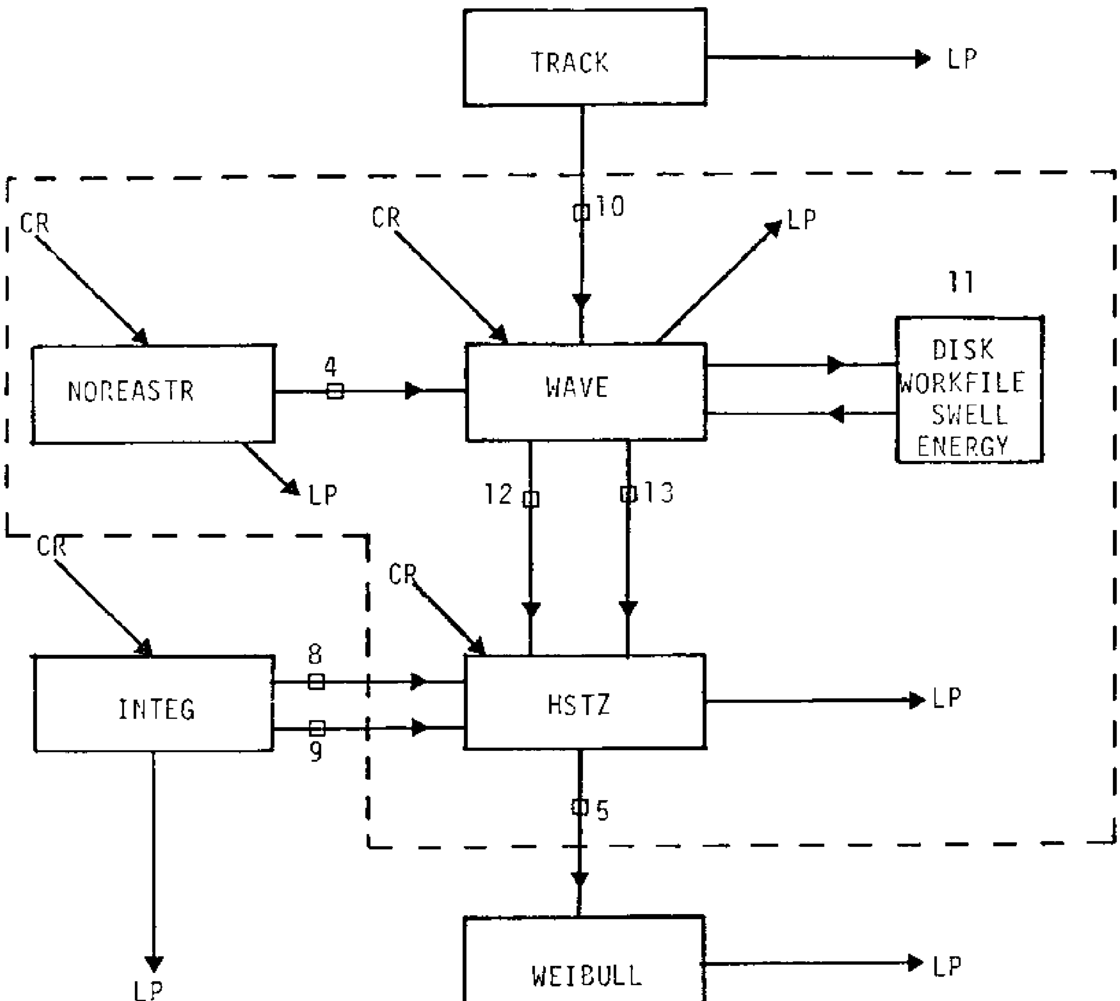
The complete form of such a statement is

Define File a(m,r,f,v)

where:

a is the unsigned integer constant that is the data reference number (File Number);

m is an integer constant that specifies the number of records in the data set associated with a;



DIRECT ACCESS DISK FILES

4 WINDA - Wind Field
5 BIGWA - Largest Waves
8 INTEGI
9 INTEGJ - Integral Table

10 SWELLR - Swell Ray Tracks
11 WORKF - Temporary Disk
12 SWELLE
13 WAVEP - Swell + Windsea

* Programs within dashed lines must be executed for each case.
Others need be run only once for each grid.

* CR - Card Reader. LP - Line Printer.

Figure A-1. Programming File Structure r is an integer constant that specifies the maximum size of each record associated with a;

f specifies that the data set is to be read or written either with or without format control. In the above program, all input/output, statements stored on the direct access files are unformatted. This is indicated by specifying f as U.

v is an integer variable, called an associated variable. At the conclusion of each read or write operation, v is set equal to a value that points to the record that immediately follows the last record transmitted.

For example; the statement,

Define file 4(125,280,U, KOUT) (1)

defines a data set having reference number 4. It has 125 records each with a maximum length of 280 storage locations. The data are unformatted and KOUT is the associated variable that acts as a pointer to the next record.

The presence of a Define File statement in the body of a program has to be signalled through a statement in the job cards. This follows the form:

FI unit DISK fname DATA diskmode (LRECL lrecl Block block RECFM F DSORG DA XTENT # records [Perm]

where:

unit	is the reference number of the file
fname	is the title of the file;
'diskmode'	is A or G;
lrecl	is the record length = 4x Number of words = 4xr
block	= 4 x r
# records	= m; XTENT # records should be specified only if m 50.
Perm	is to be specified for files to be retained after the job is finished.

For example, associated with (1) will be the statement, FI 4 DISK WINDA DATA G (LRECL 1120 BLOCK 1120 RECFM f DSORG DA XTENT 125 PERM

As described later, eight files are used in the sequential running of these programs. Details of each file are given in the following discussion of each program.

A-II. Wind Model-Program Noreastr

This subroutine develops the wind model, i.e. computes the wind speed and direction, at each of the model points on the basis of the available physical and geographical data. It can formulate the wind field using either the geostrophic approximation described in Chapter II or the equilibrium wind equations. (As this program (1978a) was developed by Stone and Webster Engineering Corporation for the calculation of storm surges, it generates, besides a wind field, other information not germane to this project.) This program must be run once for each storm. Figure A-2 shows the coordinate system to be used.

Input:

TITLE - Any 80 alphanumeric characters to be printed at the top of the output to identify the job.

Namelist "CONST"

The following variables are input using a namelist for each in formatting.

ALAT - The latitude of the study areas in decimal degrees

PO - Storm central pressure in millibars

PP - Storm peripheral pressure in millibars

RMAX - Maximum radius to last closed isobar in nautical miles

DX - Individual grid size in size in feet - x direction

DY - Individual grid size in feet - y direction

NMAX - Number of grid points - y direction rows

MMAX - Number of grid points - x direction columns

MAXSTM - The number of sets of storm locations and track directions to be read

IPRNT - A print control switch

if: (IPRNT.NE.1) only the arrays of wind speed, incurvature angle, surface rise, wind direction, and pressure are printed. These are printed for any value of IPRNT at intervals specified by PTIME. If IPRNT.EQ.1 arrays of TSX, TSY, DPDX, and DPDY are also printed.

RHO - density of air (slugs/ft^3)

IPLOT - If (IPLOT.EQ.1), this will produce a data set for input to a Calcomp plotting program. Additional JCL, which is installation dependent, must be provided for this option. Otherwise, set (IPLOT.NE.1).

CHOICE - Controls which wind model is employed and controls several debugging options.

if: Choice = 1, debug output for the equilibrium wind model is produced. Be wary of using this as a lot of output is generated.

if: CHOICE is greater than zero, the equilibrium wind model is employed.

if: CHOICE is less than zero, the geostrophic wind model is used.

if: CHOICE - 7, xxxxx is set equal to zero. This is the concentric isobar assumption used in ref. 12.

VS - The storm translational speed in knots; necessary for equilibrium wind model only.

KN,KT - Frictional coefficients for the equilibrium wind model only.

IPO - If IPO is greater than zero, P_0 is read in at each storm position.

If IPO is less than zero, P_0 is assumed constant.

PTIME - Arrays of storm characteristics will be printed every PTIME hours. (If(MOD(TIME,PTIME)=0) then print)

After namelist CONST is read in, the program looks for MAXTSM sets of storm positions and track angles. Each set of storm positions occupies one record (or card image). The variables on each record are, in order, TIME, which is the hour that the storm is at that location, X1 and Y1, which are the distances of the storm center from the grid origin in grid units, track angle relative to the grid Y axis, and central pressure. The record format is 6F10.0.

A typical data set is shown in Figure A-3. The coordinates of the storm centers in this data set do not correspond to the final grid

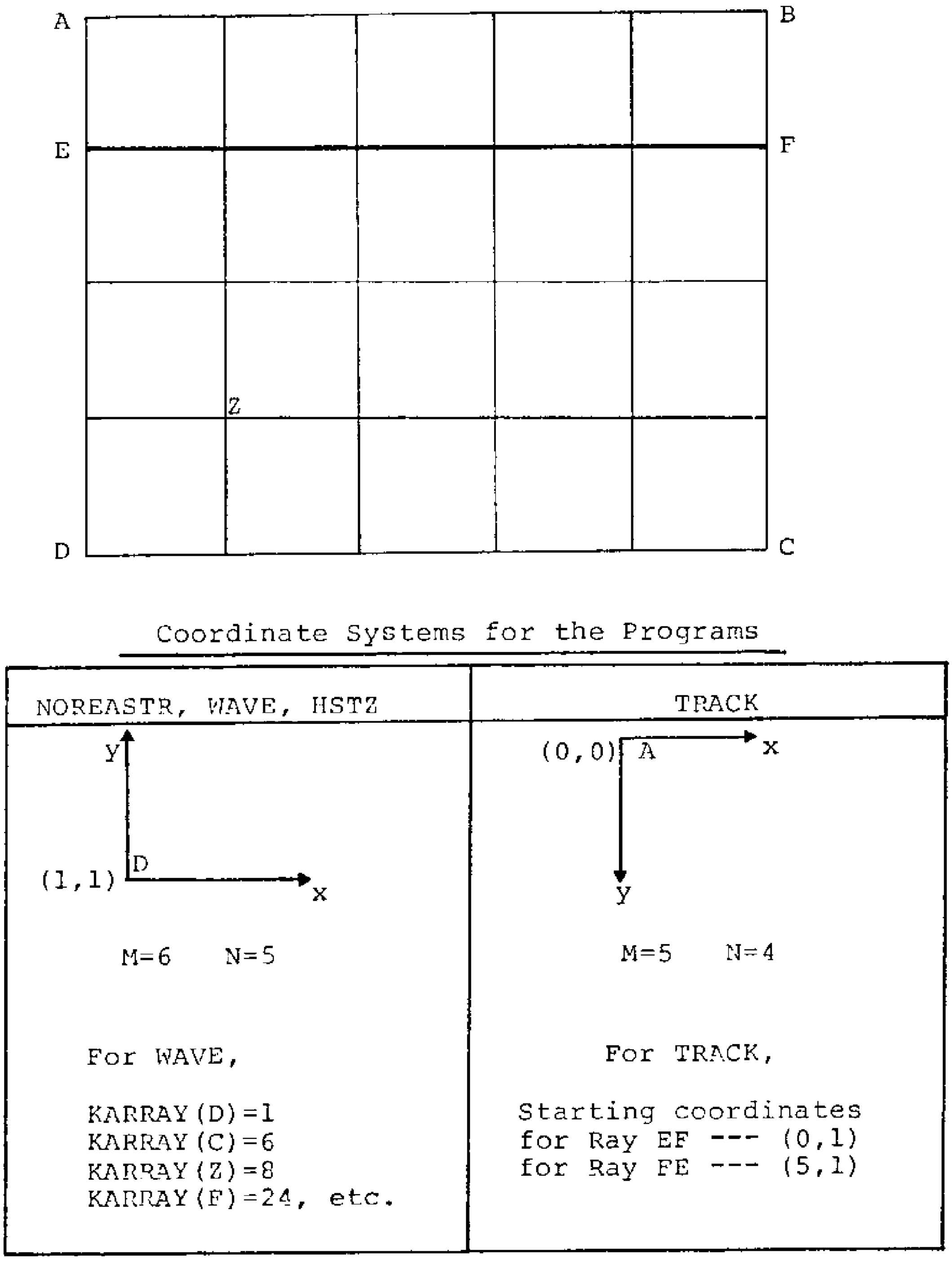


Figure A-2. Coordinate and Ray Illustrations chosen for modeling, and, therefore, are not X1 and Y1. The coordinates shown correspond to another grid whence this data set was obtained, and appropriate correction in the program is made when the coordinates are read in, and are marked by asterisks in the program listing. These lines must be omitted if the data set contains coordinates pertaining to the grid actually being used.

Output

a. Wind speeds at each grid point.

b. Wind velocities at each grid point.

c. Atmospheric pressure at each grid point.

d. The disc file (#4) holds the horizontal and vertical components U and V of the computed wind velocity in the forms of a complex number UV.

Dimensioning

The following statements must be attended to,

```
COMPLEX UV (M*N)

DIMENSION WSP (M,N), UWSP (M,N VWSP (M,N), PRS (M,N),
              ABETA (M,N), THETAD(M,N)

COMMON/WIND1/TSX(M,N), TSY(M,N), DPDX (M,N), DPDY(M,N),
              HO(M,N)

DIMENSION Z(N,M)
```

where

N - NMAX and M=MMAX

Disc File

The 'Define File' statement should read

```
DEFINE FILE 4(NT, KMAX2, U, KOUT)
```

where

NT = Number of time steps

KMAX2 = 2xNumber of grid points

```
    ****STORM  11-30-44 :M0129--RUN 23*******
&CONST  ALAT=39.6,PO=979.,PP=1015.,RMAX=755.,DX=166666.,
        DY=166666.,NMAX=21,MMAX=20,MAXSTM=19,IPRNT=1,
        RHO=0.00231,IPLOT=0,CHOICE=-2.,VS=48.,KN=0.004,
        KT=0.005,IPO=2,PTIME=1.,&END
    1.0      -105.65     -10.95       9.56   1001.00
    2.0       -95.02      -9.17       9.56    999.84
    3.0       -84.41      -7.33       9.56    998.68
    4.0       -73.82      -5.42       9.56    997.53
    5.0       -59.03      -0.79       9.56    996.37
    6.0       -44.28       3.94       9.56    995.21
    7.0       -29.58       8.77       9.56    994.05
    8.0       -21.14      14.14       9.56    992.89
    9.0       -12.71      19.51       9.56    991.74
   10.0        -4.27      24.88       9.56    990.58
   11.0         6.12      27.21       9.56    989.42
   12.0        16.47      29.60       9.56    988.26
   13.0        26.79      32.07       9.56    987.11
   14.00       35.17      37.53       9.56    985.95
   15.00       43.55      43.00       9.56    984.79
   16.0        51.92      48.46       9.56    983.63
   17.0        66.32      53.86       9.56    982.47
   18.0        80.67      59.35       9.56    981.32
   19.0        94.94      64.99       9.56    980.16
```

Figure A-3. Input Data for 'NOREASTR'

A-III. Ray Paths - Program Track

This program inputs the model cells in the form of flags (0 if land and 1 if water) and the starting coordinates of the rays, and tracks each ray in the modeled region until an inactive cell is reached. Information required for the transfer of the energy from a ray to the Cartesian grid is calculated and stored. This program is run only once.

TRACK works in terms of 'cells' so that for the grid with 20 columns and 21 rows, M-19 and N=20. The coordinate system (Figure A-2) is such that the origin is at the top left corner of the grid and the bottom right corner is (M,N).

Input

GRID - Grid spacing in metres

DT - Timestep in seconds

M,N - Number of cells in x and y directions, respectively.

FLAG - is an MxN array containing, for each element, a '0' if that cell is inactive (i.e. land) or a '1' if it is active.

NQUAD - Number of ray directions per quadrant.

NFREQ - Number of frequency bins

FINIT - Start of first (lowest) frequency bin in Hertz

FBIN - Frequency bin size in Hertz.

This is followed by a record (card image) for the number of rays and a record giving the starting coordinates of the rays, for each direction, starting with rays going in the positive x-direction. For example, in Figure 13, there are seven rays going in the positive x-direction with starting coordinates

(0,5), (0,6), (0,7), (0,8), (0,9), (0,10), (0,11)

which are written in format I2 as shown in Figure A-4. Then there are six rays making -45° with the positive x-axis, and so on. The whole set for all directions is repeated as many times as there are direction bins (NFREQ). The format of the starting coordinates may be changed if desired by duly amending the 'Format' Statement 1040 and the appropriate 'Read' statement in the program.

```
00010 **********  ENLARGED GRID : RUN 1  **********
00020  &GRIDD   GRID=50000.,DT=1800.,M=19,N=20,
00030           FLAG=0.0,0,0,0,0,0,0,0,0,0,1,1.0,0,0,0,0,1,
00040                0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,1,1,1,1,
00050                0,0,0,0,0,0,1,1,1,1,1,0,0,1,1,1,1,1,1,
00060                0,0,0,0,0,1,1,1,1,1,1.0,1,1,1,1,1,1,1,
00070                0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00080                0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00090                0,0,0,0,1,1,1,1,1,1,1,1,1,1,1.1,1,1,1,
00100                0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00110                0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1.1,
00120                0,1,1,1,1,1.1,1,1,1,1,1,1,1,1,1,1,1,1,
00130                1,1,1,1,1,1,1,1.1,1,1,1,1,1,1,1,1.1,1,
00140                1,1,1,1,1.1,1,1.1,1,1,1,1,1,1,1.1,1,1,
00150                1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00160                1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00170                1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00180                1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1.1,1,
00190                1,1.1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00200                1,1,1,1,1,1,1.1,1,1,1,1,1,1,1,1,1,1,1,
00210                1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
00220                1,1,1,1,1,1,1.1,1,1,1,1,1,1,1,1,1,1,1,&END
00230  &DIFR    NQUAD=2,NFREQ=5,FINIT=0.05,FBIN=0.10, &END
00240   7
00250 0005000600070008000900100011
00260   6
00270 001500160017001800190020
00280  11
00290 05200620072008200920102011201220132014201520
00300   9
00310 172018201920191919181917191619151914
00320   7
00330 1911191019091908190719061905
00340   6
00350 150016001700180019001901
00360  11
00370 05000600070008000900100011001200130014001500
00380   9
00390 000300020001000001000200030004000500
REPEAT LINES 00240 THROUGH 00390 4 MORE TIMES
```

Figure A-4. Input Data for 'TRACK'

Output

a. The line printer output includes model size and flags, and the direction and frequencies used for tracking rays.

b. For each frequency, details concerning the ray-points generated, the density distribution of the cells crossed by the rays and the nearest ray points are printed.

Dimensioning

```
INTEGER FLAG(MN), IDUM(10), COUNT(MN)

DIMENSION THETA (NTHETA), FREQ(NFREQ), KINFT(NFREQ* NTHETA)

DIMENSION XSTART (NRAYmax), YSTART(NRAYmax), XY(2), RDUM(4)

DIMENSION XX(KMAX), YY(KMAX), KK(KMAX), LL(KMAX)

DIMENSION LNEAR(MNE), DNEAR(MNE)
```

where:

MN = M*N and MNE = (M+1)*(N+1)

KMAX = # of active points

NTHETA = total # directions represented

NFREQ = total # frequencies represented

$NRAY_{max}$ = maximum # of rays in any direction for any frequency.

Disc File

```
DEFINE FILE 10(NFT3P2, MEG, V, KOUT)
```

where NFT3P2 = (NFREQ*NTHETA*3) + 2

MEG = MAX(MNE+2, ISEG*2)

in which ISEG is the maximum of the total number of cell crossings in any direction. It is generally best to assign MEG a value a little larger than MNE.

A-IV. Wave-Swell Model - Program Wave

This program uses a hybrid approach between a one-dimensional parametrical wind-sea and a characteristic representation for swell to calculate the wave spectra of an area, given the wind conditions.

The program produces, for each timestep, the calculated parameters for the wind-sea and the contents of the swell frequency-direction bins. (The numbers given in the parenthesis correspond to those used in the example.)

Input

a. Number of free parameters (f_m, α, γ, σ_a, σ_b) to be used in run (N=3).

b. Number of iterations (variable, ISCM=19 in Figure A-5).

c. Timestep of windfield to be used as first timestep in model (IISC=1).

d. Frequency of printing results on line printer (IPRINT=3).

e. Fetch length (in metres) at boundary, for use in determining the conditions at an incoming boundary (Xa=1/2 x grid size).

f. Initial frequency (in hertz) of wind-sea spectral peak (FM=0.20).

g. Initial Phillips' constant α (0.01). (γ, σ_a, σ_b assumed to be 3.3, 0.07, 0.09, respectively).

h. Factor of subdivision of timestep to ensure stability of finite difference schemes (NW=2).

i. Factor for subdivision of first timestep to ensure smooth start (NS=4).

j. Maximum frequency (in hertz) of spectral peak after dumping swell occurs (FMMAX=1).

k. Frequency of storage of swell information (ISSTOR=1).

A description of all the input variables is given at the beginning of the program. The namelist CONTR contains KMAX, the total number of grid points and KARRAY, a list of active grid-point numbers. The bottom left point is the first point and the top right point with coordinates (M,N) is the MxN)th point (See Figure A-2). Those points which are on land are assigned a KARRAY value of zero.

```
RETYPED INPUT DATA NORIN23   :GULF OF MAINE --ENLARGED GRID 1
 &CONTR  N=3,ISCH=19,IISC=1,KPSC=19,IPRINT=3,XA=25000.,
         FH=0.20,ALPHA=0.010,NW=2,NS=4,&END
 &CONTS  FHMAX=1.0,ISSTOR=1,&END
 &CONTK  KMAX=420,KARRAY=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,
         17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,
         35,36,37,38.39,40,41,42,43,44,45,46,47,48;49,50,51,
         52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,
         69,70,71,72,73,74,75,76,77,78,79,80,
         81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,
         96,97,98,99,100,101,102,103,104,105,106,107,108,109.
         110,111,112,113,114,115,116,117,118,119,120,121,122,
         123,124,125,126,127,128,129,130,131,132,
         133,134,135,136,137,138,139,140,141,142,
         143,144,145,146,147,148,149,150,
         151,152,153,154,155,156,157,158,159,160,
         161,162,163,164,165,166,167,168,169,170,171,
         172,173,174,175,176,177,178,179,180,181,182,
         183.184,185,186,187,188,189,190,191,192,193,
         194,195.196,197,198,199,200,201.202,203,204,205,206,
         207,208,209,210.211,212,213,214,215,216,217.218,219,220,
         0,0,0,224,225,226,227,228,229,230,231,232,233,234,235,
         236,237,238,239,240,0,0,0,0,0,246,247,248,249,250,
         251,252,253,254,255,256,257,258,259,260,0,0,0,0,0,0,
         267,268,269,270,271,272,273,274,275,276,277,278,279,280,
         0,0,0,0,285,286,287,288,289,290,291,292.293,294,295,296,
         297,298,299,300,0,0,0,0,305,306,307,308,309,310,311,312,
         313,314,315,316,317,318,319,320,0,0,0,0,0,326,327,328,
         329,330,331,332,333,334,335,336,337,338,339,340,0,0,0,0,
         0,0,0,348.349,350,351,0,0,354,355,356,357,358,359,360,
         0,0,0,0,0,0,0,0,0,370,371,0,0,0,0,376,377,378,379,380,0,
         0,0,0,0,0,0,0,0,0,391,392,393,0,0,0,0,398,399,400,0,0,0,
         0,0,0,0,0,0,0,0,0,413,0,0,0,0,0,419,420,&END
```

Figure A-5. Input Data for 'WAVE'

Input/Output

Number 11: Disc file WORKF. This is used as a workfile to hold the swell energy at each ray point for all frequencies and directions. Information for each frequency-direction is then read in as required and written back displaced one element to model advection when energy exchanges have taken place between wind-sea and swell. The information in this file is generally not kept at the end of the run.

Output

Line printer. General listing of data used by program, followed by, at the specified frequency:

a. the calculated wave parameters

b. the frequency bin number corresponding to the cutoff frequency of points involved in wind-sea to swell transfer (frequency bin 1 is the lowest).

c. The direction bin number corresponding to wind direction for each point (bin number 1 equals 90° or Eastwards, then increasing as direction changes anticlockwise).

All are printed in geographical format. At the end of every sub-timestep, a count of the number of points involved, in wind-sea to swell transfer, for that sub-timestep is printed.

d. Number 13: Disc File WAVEP. This holds grid information followed by the wave parameters for each active point for each timestep, in order (f_{m_1}, α_1, γ_1, σ_{a_1}, σ_{b_1}, f_{m_2}, α_2,...etc.) where f_{m_1} etc. correspond to active cell 1.

Number 12: Disc file(s) SWELLE. Swell energy (as transformed to the Cartesian grid) is stored on this file according to the frequency specified. The storage layout is as follows:

1. iteration number of following information (initially first iteration number that wind-sea to swell energy occurs - excluded first iteration as the swell routines are not used until iteration number 2 because of the non-physical starting conditions).

2. frequency bin number (initially highest).

3. direction bin number (initially 1).

4. energy for specified frequency-direction bin at each active cell.

Parts a-d would then be repeated for all direction bins within each frequency for each iteration.

A great amount of swell information is thus generated and the program therefore keeps a count of how much information has been written to the file; when this exceeds 2400,000 words, the program halts with the code '66'. A fresh file must now be assigned to the disc before the program can continue.

Dimensioning

In the following:

N = Number of parameters used.

KMAX = Number of grid points,

KMAX2 = 2 x KMAX

KMAX5 = 5 x KMAX

LRAY1M = Maximum of total number of ray points in any direction, usually a large number.

NAMED COMMONS

/CMSP/ A(KMAX5), C(KMAX5), WI (KMAX2)

/GRID/ KARRAY (KMAX), IFLAG(KMAX)

/INTER/ IFLAGT (KMAX)

/PREV/ WP(KMAX)

/BOUND/ IBDY(KMAX)

/PREBDY/IPBDY(KMAX)

/SWLFLG/ ISWELL (KMAX),.... ISFC(KMAX)

/WSTSWL/ DEF, DEFD, IDBIN, FW, BETA, FC (KMAX)

/ANGDIS/FRACT (10), AVECOS (10)

/SWLFLD/ EC (LRAY1M), KINFT (NFREQ*NTHETA), MXY, LNEAR(KMAX)

/SWLTWS/ DE (KMAX)

/ORDER/ JVAL(3), E(KMAX)

/CWIND/ UP (2)....

```
/COPPAR/ACOP(KMAX2*N^2)

/PM/AFPM(KMAX)
```

MAIN PROGRAM

```
DIMENSION IVAL(4), WN (KMAX2)

DIMENSION RDUM(4)
```

SUB FLAGST

```
DIMENSION WI (KMAX2), WP(KMAX2)
```

SUB PRINT

```
DIMENSION A(M), AP(20), MESS(10)
```

where M is specified in the calling argument as KMAX5

SUBPRINTI

```
DIMENSION IA(M), IAP (30), MESS(2)
```

where M is specified in the calling argument as KMAX5

SUB STEUER

```
DIMENSION H(5)
```

SUB LPRFOG

```
DIMENSION PARE95), PARW(5), PARN(5), PARS(5), PAR(5)

DIMENSION ADVECT(5), HS(5)
```

SUB PRDICT

```
DIMENSION PARE(5), PARW(5), PARN(5), PARS(5), PAR(5)

DIMENSION ADVECT(5), HS(5)

DIMENSION UN(2), US(2), UE(2), UW(2)
```

SUB BDY

```
DIMENSION(5)
```

SUB SWELL

DIMENSION D(KMAX), LL(KMAX), KK(KMAX), XX(KMAX, YY(KMAX)

DIMENSION LRAY(2)

FUNCTIONS D110, D11, D12, D13, D21, D22, D23, D31, D32, D33, D41, D42, D43, D44, D51, D42, D53, D55, S2 and subroutines MHOL and MBESE:

DIMENSION PAR(5)

The 'BLOCK DATA' should be appropriately modified according to the above dimensions.

Disc Files

File 4 - As in 'NOREASTR'

File 10 - As in 'TRACK'

DEFINE FILE 11 (NFT, LRAY1M, U, KSWIN)

where

NFT = NFREQ*NTHETA

DEFINE FILE 12 (NFT*NT, KMAX+3, U, KSOUT)

where

NT = Number of time steps

DEFINE FILE 13 (NT+1, KMAX5, U, KWOUT)

A-V. Tables for I and J - Program INTEG

This program sets up the two-way tables I and J for use in equations 13a and 13b. The numerical integration is based on a simple trapezoidal summation rule.

Input

A typical data set is shown in Figure A-6. The variables used are:

a. number of values of sigma for which the integrals are to be calculated (II=3).

b. number of values of gamma for which the integrals are to be calculated (JJ=31).

c. upper limit to be used in integration for I (UL1=10).

d. upper limit for J (UL2=25).

e. integration step for I (DY=0.5).

f. integration step for J (DX=0.10)

g. starting and finishing values of gamma (generally GAMS=1 and GAMF=4).

h. starting and finishing values of sigma (SIGS=0.07 and SIGF=0.09).

Output

The line printer lists the input data and tables for I and J. The Disc file #8 holds the gammas and sigmas used plus the two-way tables for the Integral I. The Disc file #9 holds the two-way table for the Integral J.

Dimensioning

DIMENSION SIG(II), GAM(JJ), CINTI(II,JJ), CINT2(II,JJ)

REAL NATLOG (JJ)

where:

II = Number of Sigma values

JJ = Number of Gamma Values, as in the input

Disc Files

DEFINE FILE 8(II+2, JJ, U, KEOUT)

DEFINE FILE 9(II+2, JJ, U, KNOUT)

A-VI. Significant Wave Height and the Zero Crossing Period - Program HSTZ

This calculates the significant wave height H_S and the period T_Z at each grid point. It also finds the maximum H_S that occurred during the course of the storm at each grid point.

```
             *********: GULF OF MAINE  *********
&IINT    II=3,JJ=31,UL1=10.0,UL2=25.0,DY=0.05,DX=0.10, &END
&GS      GAMS=1.0,GAMF=4.0,SIGS=0.07,SIGF=0.09, &END
```

Figure A-6. Input Data for 'INTEG'

```
************-GULF OF MAINE:STORM NORIN22 ----ENLARGED GRID 3*****
 &HTPAR     NGOES=19,NSKIP=0,IDO=-1,NDIR=8,NFREQ=5,ISTNO=22,&END
```

Figure A-7. Input Data for 'HSTZ'

Input

Figure A-7 shows a typical data set.

a. number of iterations (NGOES).

b. number iterations to be skipped before printing (generally zero, but useful if only a few iterations in the middle of a run are to be analyzed).

c. flag (IDO) signifying if swell energy is to be included (positive if not).

d. number of frequency bins used for storing the swell energy (NFREQ), immaterial if flag positive.

e. number of directional bins (NDIR), immaterial if flag positive.

f. number of the storm (ISTNO=22, indicating NORIN23 was the twenty-second storm)

g. Disc file #8, INTEGI

h. Disc file #9, INTEGJ

i. Disc file #13, WAVEP

j. Disc file #13, SWELLE

Output

The output contains in geographical format, H_S and T_Z for each timestep. The maximum significant wave height at each grid point and the corresponding period for a particular storm are written on disc file #5.

As described in the wave program, the swell information may well fill up more than one file. This program therefore halts with the code '66' when a swell file has been exhausted, ready for the next swell file to be assigned.

Dimensioning

NAMED COMMON

/GRID/ KARRAY (MN), MM, NN

MAIN PROGRAM

```
DIMENSION GAM (II), SIG (II), C1NT1(II,JJ), CINT2)
               (II, JJ)

DIMENSION ARRAY (KMAX5)

DIMENSION EINT (KMAX2)

DIMENSION EC (KMAX), IVAL (3), ES(KMAX), EINTS
               (KMAX2), SIGMA(KMAX), SUMDIR(KMAX)
```

SUB PRINT

```
DIMENSION A(M), AP (20)
```

where M is defined in the calling argument as KMAX2

```
DIMENSION MESS (8)
```

Also values of MX and MY in Subroutine PRINT must indicate the correct size of the grid.

Disc Files

File 8 - As in 'INTEG'
File 9 - As in 'INTEG'
File 12 - As in 'WAVE'
File 13 - As in 'WAVE'

```
DEFINE FILE 5 (MSTM, KMAX2, U, KAT)
```

where MSTM = Total number of Storms Simulated.
KMAX2 = 2*Total number of grid points.

A-VII, Extrapolation to 50 - 100 Years-Program Weibull

The program WEIBULL calculates the 50- and 100-year significant wave heights, working through each point of the grid. This program is run only once after the wind and wave models are run for all the storms.

Input

The 12th and 13th lines of the program

```
TIME = 32

NPTS = 22
```

indicate that 22 storms spanning a period of 32 years were chosen for the statistical analysis. The 15th line

```
KMAX = 420
```

inputs the number of points in the grid. The 50- and 100-year wave heights are calculated for each of these points. Appropriate data should be inputted in these statements.

Dimensioning

NAMED COMMONS

```
/B/ YR (NPTS)
/C/ EHMAX (NPTS), X(NPTS)
```

MAIN PROGRAM

```
DIMENSION M(NPTS), A(NPTS)

DIMENSION BWH (NPTS, KMAX), CTP (NPTS, KMAX)

DIMENSION DIFF(20), (RR920), RRR(40), II(40)

REAL K(40), K1, LAMBDA, KA(40)
```

SUB RANK

```
DIMENSION A(NPTS), Y(NPTS)
```

SUB CORR

```
DIMENSION X(NPTS), Y(NPTS), XCAP(NPTS), YCAP(NPTS)
```

Disc File

File 5 - As in 'HSTZ'.

Appendix B

A description of all the programs is given in Appendix A. This section contains the appropriate Job Control Language (JCL) for the programs and the program listings. Typical outputs may be found in Puri and Pearce (1981).

'NOREASTR' JCL

```
/JOB RCE37204 RCE372 BOX31
/SET PRINT 100000
CP SPOOL CON TO RCE37204
CP SPOOL PUN TO RCE37204
CP SPOOL PRT TO RCE37204
CP LINK RCE372 192 122 WR WRCE372
ACCESS 122 H
FI 4 DISK WINDA DATA H1(LRECL 4800 BLOCK 4800 RECFM F DSORG DA XTENT 40 PERM
FORTRAN
```

'TRACK' JCL

```
/JOB RCE37204 RCE372 BOX31
/SET PRINT 10000
CP SPOOL CON TO RCE37204
CP SPOOL PRT TO RCE37204
CP SPOOL PUN TO RCE37204
CP LINK RCE372 192 122 WR WRCE372
ACCESS 122 G
FI 10 DISK SWELLR1 DATA G1(LRECL 2400 BLOCK 2400 RECFM F DSORG DA XTENT 125 PERM
FORTRAN
```

'WAVE' JCL

```
/JOB RCE37204 RCE372 BOX31
/SET PRINT 15000
CP SPOOL CON TO RCE37204
CP SPOOL PRT TO RCE37204
CP SPOOL PUN TO RCE37204
CP LINK RCE372 192 122 WR WRCE372
ACCESS 122 H
 FI 4 DISK WINDA DATA H1(LRECL 4800 BLOCK 4800 RECFM F DSORG DA XTENT 40 PERM
FI 10 DISK SWELLR1 DATA H1(LRECL 2400 BLOCK 2400 RECFM F DSORG DA XTENT 125 PERM
FI 11 DISK WORKF DATA A1(LRECL 32000 BLOCK 32000 RECFM F DSORG DA PERM
FI 12 DISK SWELLE DATA H1(LRECL 2640 BLOCK 2640 RECFM F DSORG DA XTENT 1600 PERM
FI 13 DISK WAVEP DATA H1(LRECL 14000 BLOCK 14000 RECFM F DSORG DA XTENT 40 PERM
FI 3 PRINTER
FORTRAN
```

'INTEG' JCL

```
/JOB RCE37204 RCE372 BOX31
/SET PRINT 5000
CP SPOOL CON TO RCE37204
CP SPOOL PRT TO RCE37204
CP SPOOL PUN TO RCE37204
CP LINK RCE372 192 122 WR WRCE372
ACCESS 127 G
FI 3 PRINTER
FI 8 DISK INTEGI DATA G1(LRECL 140 BLOCK 140 RECFM F DSORG DA PERM
FI 9 DISK INTEGJ DATA G1(LRECL 140 BLOCK 140 RECFM F DSORG DA PERM
FORTRAN
```

'HSTZ' JCL

```
/JOB RCE37204 RCE372 BOX31
/SET PRINT 10000
CP SPOOL CON TO RCE37204
CP SPOOL PRT TO RCE37204
CP SPOOL PUN TO RCE37204
CP LINK RCE372 192 122 WR WRCE372
ACCESS 122 G
FI 8 DISK INTEGI DATA G1(LRECL 140 BLOCK 140 RECFM F DSORG DA PERM
FI 9 DISK INTEGJ DATA G1(LRECL 140 BLOCK 140 RECFM F DSORG DA PERM
FI 12 DISK SWELLE DATA G1(LRECL 2640 BLOCK 2640 RECFM F DSORG DA XTENT 1600 PERM
FI 13 DISK WAVEP DATA G1(LRECL 14000 BLOCK 14000 RECFM F DSORG DA XTENT 40 PERM
FI 5 DISK BIGWA DATA G1(LRECL 5200 BLOCK 5200 RECFM F DSORG DA XTENT 25 PERM
FORTRAN PRINT
```

'WEIBULL' JCL

```
/JOB RCE37204 RCE372 BOX31
/SET PRINT 100000
CP SPOOL CON TO RCE37204
CP SPOOL PUN TO RCE37204
CP SPOOL PRT TO RCE37204
CP LINK RCE372 192 122 WR WRCE372
ACCESS 122 G
FI 5 DISK BIGWA DATA G1(LRECL 5200 BLOCK 5200 RECFM F DSORG DA XTENT 25 PERM
FORTRAN
```

'NOREASTR' LISTING

```
C*************************************************************************
C
C         SYNTHETIC NORTHEASTER WIND AND PRESSURE FIELD PROGRAM
C                   VERSION 0 ,LEVEL 0
C                   AUGUST 30,1978
C
C         THIS PROGRAM IS USED TO GENERATE THE WIND SHEAR STRESS AND
C         ATMOSPHERIC PRESSURE FIELDS OF A POSTULATED NORTHEASTER
C         COASTAL STORM FOR INPUT TO A COASTAL STORM SURGE PROGRAM(2-D)
C         (REF: DEVELOPMENT AND VERIFICATION OF A SYNTHETIC NORTHEASTER
C         MODEL,SWECO,1978)
C
C*************************************************************************
C      THIS PROGRAM WAS DEVELOPED BY STONE AND WEBSTER ENGINEERING
C      CORPORATION IN THE IMPACT ANALYSIS SECTION OF THE ENVIRONMENTAL
C      ENGINEERING DIVISION. IT WAS DEVELOPED EXPRESSLY FOR FIA.(HUD)
C
C      IF YOU HAVE ANY QUESTIONS ABOUT THE USE OR OPERATION OF THIS
C      COMPUTER PROGRAM, PLEASE FEEL FREE TO CALL DR. Y.J. TSAI AT
C      (617) 973-2771, DR. F.K. CHOU AT (617) 973-2703, OR DOUG
C      CROCKER AT (617) 973-0639. WE WILL BE DELIGHTED TO ASSIST YOU.
C      ALSO, IF YOU SHOULD DISCOVER A "BUG" WE WOULD APPRECIATE BEING
C      INFORMED.                                  DMC 8/30/78
C*************************************************************************
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RHO
     $,KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPC,PTIME
      COMMON/CURVES/CCC(19,2,3)
      COMMON/B/KOUT
      DEFINE FILE 4 (40,1200,U,KOUT)
      COMPLEX UV(650)
      DIMENSION TITLE(20)
      REAL KN,KT
      NAMELIST/CONST/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,
     3RHO,IPLOT,CHOICE,VS,KN,KT,IPO,PTIME
      READ(1,10)TITLE
   10 FORMAT(20A4)
      WRITE(3,20)TITLE
   20 FORMAT(/////,T36,20A4,////)
      READ(1,CONST)
      ALAT=ALAT*3.1415/180.
C    ***********RMAX SHOULD BE IN NAUTICAL MILES; PO AND PP IN MB
      WRITE(3,30)PO,PP,RMAX
   30 FORMAT(////,T30,'CENTRAL PRESSURE(MB)=',F6.1,'      PERIPHERAL'
     #,' PRESSURE(MB)='
     #,F6.1,//,T30,'THE MAXIMUM STORM RADIUS(RMAX)=',F6.1,
     #' NAUTICAL MILES',///)
      CALL ERRSET(207,256,-1,2,1,209)
C    MASK IS AN IBM SUBROUTINE THAT MASKS UNDERFLOWS. IT REPLACES THE
C    UNDERFLOW VALUE WITH ZERO AND CONTINUES EXECUTION. THIS IS
C    NECCESSARY IN REGIONS NEAR THE STORM CENTER WHERE PRESSURE
C    GRADIENTS CAN BE QUITE SMALL. SINCE THERE IS LITTLE OR NO
C    WIND IN THIS AREA THE SUBSTITUTION IS VALID.
      CALL STORM
      STOP
      END
      SUBROUTINE STORM
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RHO
     $,KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPC,PTIME
      COMMON/CURVES/CCC(19,2,3)
      COMMON/B/KOUT
C   CURVE 1 IS A AS A FUNCTION OF ANGLE FROM RMAX
C   CURVE 2 IS THE CURVE OF THE SURFACE WIND AS A
C   FUNCTION OF THE GEOSTROPHIC WIND AS CALCULATED
C   FROM THE PRESSURE GRADIENT.(REF: STANDARD PROJECT
C   NORTHEASTER REPORT)
C   CURVE 3 IS DRTDT AS A FUNCTION OF ANGLE FROM RMAX
      COMMON/WIND1/ TSX(32,32),TSY(32,32),DPDX(32,32)
     #,DPDY(32,32),HO(32,32)
C   INITIALIZE ALL ARRAYS TO ZERO
      DO 10 I=1,NMAX
      DO 10 J=1,MMAX
      TSX(I,J)=0.0
      TSY(I,J)=0.0
      DPDX(I,J)=0.0
      HO(I,J)=0.0
   10 DPDY(I,J)=0.0
         KOUT=1
      DO 20 L=1,MAXSTM
C     READ STORM PARAMETERS AT EACH STORM POSITION
      IF(IPO.LT.0)READ(1,30)TIME,X11,Y11,PSI
      IF(IPO.GT.0)READ(1,30)TIME,X11,Y11,PSI,PO
31      FORMAT(//' ',8F10.2)
C *************THE NEXT 5 STATEMENTS TRANSFORM THE INPUT STORM
C*************COORDINATES TO MATCH WITH X1 & Y1 WHICH CORRESPOND
C*************TO THE MODEL GRID.IN THIS PROJECT.INPUT X1 & Y1
C*************CORRESPONDED TO A DIFFERENT GRID,HENCE A TRANSFORMATION
C*************WAS NECESSARY.THE NEXT 5 STATEMENTS SHOULD BE
C*************OMITTED IF X1 & Y1 ARE READ IN.
      ANGL=57.*3.14159/180.
      XC=X11/6.85
      YC=Y11/6.85
      X1=XC*COS(ANGL)-YC*SIN(ANGL)+5.3
      Y1=YC*SIN(ANGL)+YC*COS(ANGL)+13.4
      WRITE(3,31)TIME,X11,Y11,PSI,PO,X1,Y1
      WRITE (3,30)TIME,X11,Y11,PSI,PO
   20 CALL WIND(X1,Y1,PSI,TIME,NMAX,MMAX)
   30 FORMAT(6F10.2)
      RETURN
      END
      SUBROUTINE WIND(X1,Y1,PSI,TIME,NMAX,MMAX)
C     THIS ROUTINE DETERMINES THE PRESSURE AND WIND IN EACH GRID POINT
      COMMON /CURPVES/CCC(19,2,3)
      COMMON /B/KOUT
      COMPLEX UV(650)
      COMMON /C/ALAT,PO,PP,RMAX,DX,CY,NDUM,MDUM,MAXSTM,IPRNT,RHO
     $,KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPC,PTIME
      REAL KAPPA
      DIMENSION WSP(32,32),UWSP(32,32),VWSP(32,32),PRS(32,32),
      ABETA(32,32),THETAD(32,32)
```

```
      COMMON/WIND1/ TSX(32,32),TSY(32,32),DPOX(32,32),
     *DPOY(32,32),HO(32,32)
C     LOOP THROUGH EACH POINT IN THE STORM SURGE GRID
C       KOUNT=0
        DO 40 I=1,NMAX
        DO 30 J=1,MMAX
        DELTX=FLOAT(J)-X1-1
        DELTY=FLOAT(I)-Y1-1
C    6076 FEET PER NAUTICAL MILE
C    CALCULATE RADIUS FROM STORM CENTER TO GRID POINT
        DELTR=SQRT((DELTX*DX)**2+(DELTY*DY)**2)/6076.
        R=DELTR
        IF(DELTX.EQ.0.0.AND.DELTY.EQ.0.0)GO TO 10
CALCULATE THE ANGLE OF THE GRID POINT RELATIVE TO THE GRID AXIS
        PHI=ATAN2(DELTX*DX,DELTY*DY)*180./3.1415+360.
        PHI=AMOD(PHI,360.)
        KAPPA=720.-135.+PHI-PSI
        KAPPA=AMOD(KAPPA,360.)*3.1415/180.
   13   BETA=0.0
C     WGEO*****************************************************
C     THIS ROUTINE ACCEPTS THE LOCATION OF THE GRID POINT
C     IN POLAR COORDINATES RELATIVE TO THE STORM CENTER
C     AND RETURNS THE GEOSTROPHIC WIND SPEED AS CALCULATED
C     FROM THE LOCAL PRESSURE GRADIENT
C     *********************************************************
        IF(CHOICE.LT.0.0)BETA=22.5
        IF(CHOICE.LT.0.0)CALL WGEO(KAPPA,DELTR,PR,WGS)
        IF(CHOICE.GT.0.0)CALL WINDEQ(KAPPA,DELTR,PR,WGS,BETA)
C    WINDEQ******************************************
C    THIS ROUTINE CALCULATES THE EQUILIBRIUM WIND AS DESCRIBED IN
C    THE DOCUMENTATION AND IN THE STANDARD PROJECT NORTHEASTER REPORT
C    ************************************************
        THETA=180.-PHI+360.+BETA
        THETAD(I,J)=AMOD(THETA,360.)
CALCULATE THE WIND DIRECTION AT EACH GRID POINT
        THETA=AMOD(THETA,360.)*3.1415/180.
        WGS1=WGS*6076./3600.
        IF(WGS.LT.1.0)THETAD(I,J)=0.0
CALCULATE THE WIND SHEAR STRESS
        TS=WST(WGS1,1.93)
        PRS(I,J)=PR
CALCULATE THE WIND SHEAR COMPONENTS ALONG THE GRID AXIS
        TSX(I,J)=TS*COS(THETA)
        TSY(I,J)=TS*SIN(THETA)
        WSP(I,J)=WGS
        ABETA(I,J)=BETA
C       KOUNT=KOUNT+1
  303 FORMAT (10X,2F15.3)
        UWSP(I,J)=(WGS*COS(THETA))/2.
        VWSP(I,J)=(WGS*SIN(THETA))/2.
C       IF (TIME.EQ.1.00) WRITE (3,901) UWSP(I,J),VWSP(I,J),
C      &WSP(I,J),THETAD(I,J)
  901 FORMAT (10X, 4F15.3)
C     DIVISION BY 2 CONVERTS KNOTS TO MET/SEC REQD. ON DISC
C       UV(KOUNT)=CMPLX(UWSP(I,J),VWSP(I,J))
   14 CONTINUE
      GO TO 20
   10 WSP(I,J)=0.0
      PRS(I,J)=PO
CALCULATE THE INVERTED BAROMETER EFFECT
   20 HO(I,J)=(PP-PRS(I,J))*1.105*29.53/1000.
   30 CONTINUE
   40 CONTINUE
C       WRITE(4'XOUT) (UV(K),K=1,KOUNT)
C       IF ((TIME.EQ.1.00).AND.(I.EQ.1))WRITE(3,304)
C      &WSP(I,J),WGS
  304 FORMAT (5X,'WSP(I,J)=',F10.3,'WGS=',F10.3)
   16 CONTINUE
      N=NMAX-1
      M=MMAX-1
CALCULATE THE X AND Y PRESSURE GRADIENTS
      DO 50 I=2,N
      DO 50 J=1,MMAX
   50 DPOY(I,J)=(PRS(I+1,J)-PRS(I-1,J))/(2.*DY/6076.)
      DO 60 J=2,M
      DO 60 I=1,NMAX
   60 DPDX(I,J)=(PRS(I,J+1)-PRS(I,J-1))/(2.*DX/6076.)
      DO 70 J=1,MMAX
      DPOY(NMAX,J)=(PRS(NMAX,J)-PRS(NMAX-1,J))/(DY/6076.)
   70 DPOY(1,J)=(PRS(2,J)-PRS(1,J))/(DY/6076.)
      DO 80 I=1,NMAX
      DPDX(I,1)=(PRS(I,2)-PRS(I,1))/(DX/6076.)
   80 DPDX(I,MMAX)=(PRS(I,MMAX-1)-PRS(I,MMAX))/(DX/6076.)
      DO 90 I=1,NMAX
      DO 90 J=1,MMAX
      DPOX(I,J)=DPDX(I,J)*(14.5038/1000.*144./6076.)
   90 DPOY(I,J)=DPOY(I,J)*(14.5038/1000.*144./6076.)
C  THIS IS THE FIRST OF TWO PRINT CONTROL "IF STATEMENTS"
C      IF(TIME.EQ.1.)GO TO 100
C       IF((AMOD(TIME,PTIME)).NE.0.0)GO TO 200
  100 WRITE(3,110)TIME
  110 FORMAT(1H1,4X,'TIME=',F4.1,//8X,'WIND SPEED IN KNOTS')
      KOUNT=0
      DO 41 I=1,NMAX
      DO 41 J=1,MMAX
      KOUNT=KOUNT+1
      UV(KOUNT)=CMPLX(UWSP(I,J),VWSP(I,J))
      IF (TIME.EQ.1.0) WRITE (3,901) UWSP(I,J),VWSP(I,J),
     &WSP(I,J),THETAD(I,J)
   41 CONTINUE
      WRITE(4'KOUT)(UV(K),K=1,KOUNT)
      CALL OUTPUT(NMAX,MMAX,WSP)
      CALL OUTPUT (NMAX,MMAX,UWSP)
      CALL OUTPUT (NMAX,MMAX,VWSP)
      WRITE(3,120)
  120 FORMAT(1H1,8X,'WIND DIR. IN DEG. W.R.T. X-AXIS')
      CALL OUTPUT(NMAX,MMAX,THETAD)
      WRITE(3,130)
  130 FORMAT(1H1,8X,'ATMOSPHERIC PRESSURE AT EACH GRID POINT'
     *,'IN MILLIBARS')
```

```
      CALL OUTPUT(NMAX,MMAX,PRS)
      GO TO (1)
  213 WRITE(3,140)
  140 FORMAT(1H1,8X,'ANGLE OF WIND INCURVATURE AT EACH GRID POINT')
C     CALL OUTPUT(NMAX,MMAX,ABETA)
C THIS IS THE SECOND PRINT CONTROL STATEMENT
C     IF(IPRNT.NE.1)GO TO 200
      WRITE(3,150)
  150 FORMAT(1H1,8X,'INITIAL WATER SURFACE RISE DUE TO "INVERTED'
     *,'BAROMETER" EFFECT(FT)')
C     CALL OUTPUT(NMAX,MMAX,HO)
      WRITE(3,160)
  160 FORMAT(1H1,8X,'TSX(I,J)')
C     CALL OUTPUT(NMAX,MMAX,TSX)
      WRITE(3,170)
  170 FORMAT(1H1,8X,'TSY(I,J)')
C     CALL OUTPUT(NMAX,MMAX,TSY)
      WRITE(3,180)
  180 FORMAT(1H1,8X,'DPCX(I,J)')
C     CALL OUTPUT(NMAX,MMAX,DPDX)
      WRITE(3,190)
  190 FORMAT(1H1,8X,'DPCY(I,J)')
C     CALL OUTPUT(NMAX,MMAX,DPDY)
C  200 WRITE(3)TIME,
C    &         ((TSX(I,J),I=1,NMAX),J=1,MMAX),
C    &         ((TSY(I,J),I=1,NMAX),J=1,MMAX),
C    &         ((DPDX(I,J),I=1,NMAX),J=1,MMAX),
C    &         ((DPDY(I,J),I=1,NMAX),J=1,MMAX),
C    &          ((HO(I,J),I=1,NMAX),J=1,MMAX))
C 201    FORMAT(6E13.7)
CALCOMP PLOTTER OPTIONAL OUTPUT STATEMENT: SUPPLY YOUR OWN INSTALLATION
C      SPECIFIC JCL
C       IF(IPLOT.EQ.1)WRITE(44)(((WSP(I,J),UWSP(I,J),VWSP(I,J)),I=1,NMAX)
C    1   ,J=1,MMAX)
  113 CONTINUE
      RETURN
      END
      SUBROUTINE WGEO(KAPPA,RR,PR,WEQ)
      COMMON /CURVES/CCC(19,2,3)
      COMMON/C/ALAT,PO,PP,RMAX
      REAL KAPPA
      R=RR
      A=CURVB(KAPPA,1)
      DELTP=PP-PO
      RT=RMAX*(1.-.3143*KAPPA+5.003*.01*KAPPA**2)
   10 FORMAT(1X,F10.2,1X,F10.8,1X,5(1X,F10.2))
      PR=(DELTP*EXP(-A*RT/R+A)+PO)
      DPDR=-DELTP*EXP(-A*RT/R+A)*(-A*RT/R**2)
      WGS1=DPDR*607.0/SIN(ALAT)
      IF (WGS1.GT.180.0) WEQ=60.
      IF (WGS1.LE.180.) WEQ=CURVB(WGS1,2)
      IF(R.GT.RT)WEQ=0.0
      IF(R.GT.RT)PR=PP
      RETURN
      END
      FUNCTION U(B)
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RRHO
     $,KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPO,PTIME
      REAL KN,KT,KAPPA
      RHO=RRHO*6076.**3
      AA=CURVB(KAPPA,1)
      UU=((1./RHO)*DPDR(AA)*SIN(B)+(1./(RHO*R))*DPDT(AA)*
     1COS(B))/KT
      IF(CHOICE.EQ.1.0)CALL DEBG(2,UU,B,AA)
      IF(UU.LE.0.) UU=0.
      U=UU**.5
      RETURN
      END
      FUNCTION DPDR(A)
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RHO,
     $KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPO,PTIME
      IF(CHOICE.EQ.1)CALL DEBG(1,A,99.,99)
      DPDR=(PP-PO)*EXP(-A*RT/R+A)*A*RT/R**2
      DPDR=DPDR*2.084*2132.98*6076.**2
      RETURN
      END
      FUNCTION DPDT(A)
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RHO,
     $KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPC,PTIME
      REAL KAPPA
      DPDT=(PP-PO)*EXP(-A*RT/R+A)*(-DADT(KAPPA+2.36)*(RT/R-1.)
     1-(A/R)*CURVB(KAPPA,3)*RMAX)
      DPDT=DPDT*2.084*2132.98*6076.**2
      IF(CHOICE.EQ.7.0)DPDT=0.0
      RETURN
      END
      FUNCTION DADT(KAPPA)
      REAL KAPPA
      ANG=AMOD(KAPPA,6.283)
      DADT=3.806*.01-0.2668*(ANG)+
     *(1.586*.01*(ANG)**2-1.1676*.01*(ANG)**3
      RETURN
      END
      FUNCTION FCT(B)
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RRHO,
     $KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPC,PTIME
      REAL KN,KT,KAPPA
      RHO=RRHO*6076.**3
      F=0.5235*SIN(ALAT)
      AA=CURVB(KAPPA,1)
      C=COS(B)
      A1=(1./RHO)*DPDR(AA)*C
      A2=(1./(RHO*R))*SIN(B)*DPDT(AA)
      UU=U(B)
      A3=F*UU
      A4=UU**2*C/R
      A5=KN*UU**2
      A6=UU*VS*SIN(KAPPA+2.35)/R
      FCT=A1-A2-A3-A4-A5+A6
      IF(CHOICE.EQ.1.0)CALL DEBG(3,UU,B,FCT)
```

```
      RETURN
      END
      SUBROUTINE WINDEQ(KAPPA,DELTR,PR,WGS,BETA)
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RHO,
     $KN,KT,VS,THETA,R,RT,II,IDUMMY,CHOICE,IPO,PTIME
      EXTERNAL FCT
      REAL KAPPA,KT,KN
      XST=18.5*3.1415/180.
      EPS=0.02
      RT=RMAX*(1.-0.3143*KAPPA+5.003*.01*KAPPA**2)
      IF(R.GT.RT)GO TO 20
      CALL BISECT(BETA,FCT,XST,EPS,IER)
      IF(CHOICE.EQ.1.0)CALL DEBG(4,KAPPA,BETA,FCT)
C     IF(IER.NE.0)PRINT 100,IER,KAPPA,DELTR
   10 FORMAT(1X,'IER=',I2,1X,'AT KAPPA=',F8.2,1X,'AND DELTR=',F10.2/)
      WGS=U(BETA)
      BETA=BETA*180./3.1415
      A=CURVB(KAPPA,1)
      DELTP=PP-PO
      PR=DELTP*EXP(-A*RT/R+A)+PO
      RETURN
   20 PR=PP
      WGS=0.0
      BETA=0.0
      RETURN
      END
      SUBROUTINE DEBG(N,X,Y,Z)
C    THIS IS AN AUXILARY PROGRAM USED TO TRACE THE FLOW OF THE
C    PROGRAM IN THE EVENT OF A PROBLEM GETTING IT TO RUN. IT
C    GENERATES A LOT OF OUTPUT AND SHOULD BE USED DISCRETELY.
C    IT IS CALLED BY SETTING CHOICE = 1. THIS MEANS THAT IT ONLY
C    WORKS FOR THE EQUILIBRIUM WIND MODEL. THE GEOSTROPHIC MODEL
C    IS TOO SIMPLE TO REQUIRE THIS FEATURE.
      COMMON/C/ALAT,PO,PP,RMAX,DX,DY,NMAX,MMAX,MAXSTM,IPRNT,RHO,
     $KN,KT,VS,THETA,R,RT,II,KAPPA,CHOICE,IPC,PTIME
      DIMENSION MO(10)
      DATA MO/'DPDR','U   ','FCT ','WEQ ','DADT','DPDT','SPAR'
     &,'SPAR','SPAR','SPAR'/
      REAL KN,KT,KAPPA
      WRITE(3,10)MO(N),RHO,KN
   10 FORMAT(//,20X,A4,5X,2F10.8/)
C     PRINT *,X,Y,Z,KAPPA,R,RT
C     PRINT *,NMAX,MMAX,MAXSTM,IPRNT,RHO
C     PRINT *,KN,KT,VS,THETA,R,RT,II
C     PRINT *,KAPPA,CHOICE
      WRITE(3,20)
   20 FORMAT(1X)
      RETURN
      END
        FUNCTION CURVB (X,N)
      COMMON /CURVES/CCC(19,2,3)
   10 FORMAT(    F30.2,35H IS LESS THAN LOWEST VALUE ON CURVE,I5,28H CUR
     1E FUNCTION EXTRAPOLATES)
   20 FORMAT(    F9.2,39H IS GREATER THAN HIGHEST VALUE ON CURVE,I5,28H
     1CURVE FUNCTION EXTRAPOLATES)
   30 FORMAT(10X,'SOMETHING IS WRONG WITH ABSCISSA OF CURVE',I2)
C
   40 K=19
      IF(N.EQ.1)K=9
   50 IF (CCC(K,1,N)) 80,60,80
   60 IF (CCC(K,2,N)) 80,70,80
   70 K=K-1
      GO TO 50
   80 DO 100 I=2,K
      IF (CCC(I,1,N)-CCC(I-1,1,N)) 90,90,100
   90 WRITE (3,30) N
      STOP
  100 CONTINUE
C                    MAKING LINEAR INTERPOLATION
      IF (X-CCC(1,1,N)) 120,110,130
  110 CURVB = CCC(1,2,N)
      RETURN
  120 CURVB = CCC(1,2,N) - (CCC(1,1,N)-X)*(CCC(2,2,N)-CCC(1,2,N))/
     1        (CCC(2,1,N) - CCC(1,1,N))
      WRITE(3,10)X,N
      STOP
  130 IF (X-CCC(K,1,N)) 160,140,150
  140 CURVB = CCC(K,2,N)
      RETURN
  150 CONTINUE
      CURVB = CCC(K,2,N) + (X-CCC(K,1,N))*(CCC(K,2,N)-CCC(K-1,2,N))/
     1        (CCC(K,1,N) - CCC(K-1,1,N))
      WRITE(3,20)X,N
      STOP
  160 DO 170 I=1,K
      IF (X-CCC(I,1,N)) 190,180,170
  170 CONTINUE
  180 CURVB = CCC(I,2,N)
      RETURN
  190 II = I
      CURVB = CCC(II-1,2,N) + (X-CCC(II-1,1,N))*(CCC(II,2,N)-CCC(II-1,2
     1        ,N))/(CCC(II,1,N)-CCC(II-1,1,N))
      RETURN
      END
      SUBROUTINE OUTPUT(NMAX,MMAX,Z)
C     THIS IS THE OUTPUT SUBROUTINE(OBVIOUSLY) THAT WILL PRINT ANY
C     ARRAY PASSED IN THE CALLING ARGUMENTS
      DIMENSION Z(32,32)
      JOUT=(MMAX-1)/10+1
      DO 20 MM=1,JOUT
      JAA=10*(MM-1)+1
      JZZ=10*MM
      IF(MM.EQ.JOUT)JZZ=MMAX
      IF(MM.GT.1) WRITE(3,60)
      WRITE(3,30) (J,J=JAA,JZZ)
      WRITE(3,40)
      DO 10 NN=1,NMAX
      N=NMAX+1-NN
      WRITE(3,50) N,(Z(N,J),J=JAA,JZZ)
   10 CONTINUE
```

```
   20 CONTINUE
   30 FORMAT(/54X,'COLUMN'/'  ROW',5X,10(I2,10X))
   40 FORMAT('  ---',5X,10('--',10X))
   50 FORMAT(2X,I3,'.',1X,10(G11.4,1X))
   60 FORMAT(/10X,'CONTINUED FROM PREVIOUS PAGE')
      RETURN
      END
      FUNCTION WST(V,RHO)
C     THIS ROUTINE CALCULATES THE WIND SHEAR STRESS GIVEN THE WIND SPEE
C     IN FEET/SECOND. VALUES ARE BASED ON REFERENCE #28 IN THE REPORT
C     DESCRIBING THIS MODEL.
      IF(V.LT.16.9)TS=RHO*(1.25*.000001*V**2)
      IF(V.GE.16.9.AND.V.LT.50.6)TS=RHO*(1.25*.000001+1.75*.000001)*
     1  SIN((V-16.9)*3.1415/67.6)*V**2
      IF(V.GE.50.6.AND.V.LT.121.5)TS=RHO*3.*.000001*V**2
      IF(V.GE.121.5)TS=RHO*(1.21*.000001+2.75*.000001*((1.-23.47/V)**2)
     1  *V**2)
      WST=TS
      RETURN
      END
      BLOCK DATA
      COMMON/CURVES/CCC(19,2,3)
C     DATA CCC/0.,.785,1.57,2.356,3.142,3.927,4.712,
C    #5.498,6.2832,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,.42,.44,.52,.53,.64,
C    #.7,.73,.61,.42,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,10.,20.,
C    #,30.,40.,50.,60.,70.,80.,90.,100.,110.,120.,130.,140.,150.,
C    #160.,170.,180.,0.,11.,19.,27.,33.,38.,42.,43.5,45.,47.,50.,52.5,
C    #53.5,54.,55.,56.,57.,58.,59.,0.0,0.349,0.698,1.04,1.4,1.75,
C    #2.09,2.44,2.79,3.14,3.49,3.84,4.19,4.54,
C    #4.88,5.23,5.58,5.93,6.283,0.0,-.194,-.17,-.07,0.0,.004,
C    #-.033,-.07,-.08,-.054,0.11,.08,.098,0.089,.104,.153,.164,.068,
C    #   0.0/
      DATA CCC/0.,.785,1.57,2.356,3.142,3.927,4.712,
     #5.498,6.2832,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,.49,.44,.46,.54,.65,
     #.72,.69,.59,.49,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,10.,20.,
     #,30.,40.,50.,60.,70.,80.,90.,100.,110.,120.,130.,140.,150.,
     #160.,170.,180.,0.,11.,19.,27.,33.,38.,42.,43.5,45.,47.,50.,52.5,
     #53.5,54.,55.,56.,57.,58.,59.,0.0,0.349,0.698,1.04,1.4,1.75,
     #2.09,2.44,2.79,3.14,3.49,3.84,4.19,4.54,
     # 4.88,5.23,5.58,5.93,6.283,-.314,-.279,.244,-.2095,
     #-.175,-.1397,-.1047,-.0698,-.0347,-0.,.035,.0699,.1048,
     #.1397,.1747,.2096,.2445,.2794,.3144/
      END
      SUBROUTINE BISECT(B3,FCT,XST,ERR,ISWT)
C   THIS ROUTINE EMPLOYS THE BISECTION METHOD TO SOLVE THE EQUILIBRIUM
C   WIND EQUATIONS FOR THE WIND VELOCITY AND ANGLE OF INCURVATURE.
      ISWT=0
      MC=0
      Z=1.
      LCOUNT=0
   10 MCOUNT=0
      ISIGN=0
      B1=XST+.3*Z
      FB1=FCT(B1)
      IF(FB1.EQ.0.) GO TO 110
   20 B2=XST-.3*Z
      IF(MCOUNT.NE.0) GO TO 60
      B3=B2
      GO TO 70
   30 LCOUNT=LCOUNT+1
      IF(LCOUNT.EQ.10) STOP233
   40 IF(MC.GE.2) GO TO 130
   50 Z=Z+1.
      MC=MC+1
      GO TO 10
   60 B3=(B1+B2)/2.
   70 MCOUNT=MCOUNT+1
      IF (MCOUNT.GT.20) GO TO 30
      FB3=FCT(B3)
      IF(FB1*FB3) 80,120,90
   80 B2=B3
      ISIGN=1
      GO TO 100
   90 B1=B3
  100 IF(ABS(B2-B1).LE.ERR) GO TO 140
      GO TO 60
  110 B3=B1
  120 RETURN
  130 ISWT=1
      RETURN
  140 IF(ISIGN.EQ.0) GO TO 40
      RETURN
      END
```

'TRACK' LISTING

```
C         MASTER TRACK
C
C                          PROGRAM TO CALCULATE CHARACTERISTIC RAYS FOR SWELL
C
      INTEGER FLAG(650),IDUM(10),COUNT(650)
      DIMENSION THETA(18),FREQ(10),KINET(60)
      DIMENSION XSTART(20),YSTART(20),XY(2),RDUM(4)
      DIMENSION IXSTAR(20),IYSTAR(20)
      DIMENSION XX(650),YY(650),KK(650),LL(650)
      DIMENSION LNEAR(650),DNEAR(650),TITLE(20)
      NAMELIST /GRIDD/ GRID,DT,M,N,FLAG
      NAMELIST /DIFR/ NQUAD,NFREQ,FINIT,FBIN
      DATA PI/3.14159265361/,G/9.81/
      DEFINE FILE 10 (125,600,U,KOUT)
         DEGRAD = PI/180.0
C
C                                        READ IN GRID INFO AND FLAGS AND WRITE TO LP
C
      READ (1,2000) TITLE
      READ(1,GRIDD)
      WRITE(3,4000) TITLE,M,N
      DO 10 J=1,N
         KST = (J-1)*M+1
         KFIN = KST+M-1
   10 WRITE(3,4210)(FLAG(K),K=KST,KFIN)
      MN = M*N
C
C                                        WRITE OUT GRID SPACING AND TIME STEP
C
      WRITE(3,4020)GRID,DT
C
C                          READ DIRECTION AND FREQUENCY INFO.
C
      READ(1,DIFR)
      NTHETA = 4*NQUAD
      DBIN = 90.0/NQUAD
C
C                          SET UP DIRECTIONS
C
      DIR = 90.0+DBIN
      DO 20 I=1,NTHETA
      DIR = DIR - DBIN
 20   THETA(I) = AMOD(DIR,360.0)
      WRITE(3,4030) NQUAD,NTHETA
      WRITE(3,4040) (THETA(I),I=1,NTHETA)
C
C                          SET UP FREQUENCIES
C
      FREQ(1) = FINIT+FBIN*0.5+(NFREQ-1)*FBIN
      DO 30 I=2,NFREQ
 30   FREQ(I) = FREQ(I-1)-FBIN
      WRITE(3,4050) NFREQ,FINIT,FBIN
      WRITE(3,4060) (FREQ(I),I=1,NFREQ)
C
C                          PUT HEADER ON ED FILE
C
         KOUT = 1
         IDUM(1) = NTHETA
         IDUM(2) = NFREQ
         IDUM(3) = M+1
         IDUM(4) = N+1
      RDUM(1) = FINIT
      RDUM(2) = FBIN
      RDUM(3) = GRID
      RDUM(4) = DT
      WRITE (10'KOUT) (IDUM(KI),KI=1,4),(RDUM(KJ),KJ=1,4)
         MNE = (M+1)*(N+1)
C
C                                        SET POINTER FOR ED OUTPUT
C
      KOUT = 3
      DO 500 IF = 1,NFREQ
C
C                                        CALC GROUP VELOCITY * D IN UNITS OF GRID
C
         DS = G*DT/(4.0*PI*FREQ(IF))
         DS = DS/GRID
C
C                                         TRACK RAYS, ONE DIRECTION AT A TIME
C
      DO 400 IT = 1,NTHETA
C
C                                        READ IN STARTING COORDINATES
C
      READ(1,1000) NRAY
      READ(1,1040)(IXSTAR(IR),IYSTAR(IR),IR=1,NRAY)
      DO 78 IRS=1,NRAY
      XSTART(IRS)=FLOAT(IXSTAR(IRS))
      YSTART(IRS)=FLOAT(IYSTAR(IRS))
   78 CONTINUE
      WRITE(3,4080) FREQ(IF),THETA(IT),NRAY
C
C                                        SET COUNT TO ZERO
C
      DO 100 K=1,MN
  100    COUNT(K) = 0
C
C                                        INITIALISE LNEAR AND DNEAR
C
      DO 120 K=1,MNE
         LNEAR(K) = 0
  120    DNEAR(K) = 2.0
         L = 0
         ISEG = 0
C
C                                        CALC RAY STEP AND SET ED POINTER
C
         DX = DS*SIN(THETA(IT)*DEGRAD)
         DY = DS*COS(THETA(IT)*DEGRAD)
         IFT = (IF-1)*NTHETA+IT
```

```
      DO 300 IR=1,NRAY
          X = XSTART(IR)-DX * 0.99999
          Y = YSTART(IR)+DY * 0.99999
          I = X+1
          J = Y+1
          KST = 0
          IS = ISEG + 1
  150     L = L+1
          X = X+DX
          Y = Y-DY
          I = X+1
          J = Y+1
          K = (J-1)*(M+1)+I
      IF(KST-K)190,155,190
C
C                                    CHECK FOR NEAR NEIGHBOUR
C
  155     XC = X-I+1.0
          YC = Y-J+1.0
          DC = XC*XC+YC*YC
          KC = K
      IF(DC-DNEAR(KC))161,161,160
  161     DNEAR(KC) = DC
          LNEAR(KC) = L
  160 CONTINUE
          XC = 1.0-XC
          DC = XC*XC+YC*YC
          KC = K+1
      IF(DC-DNEAR(KC))171,171,170
  171     DNEAR(KC) = DC
          LNEAR(KC) = L
  170 CONTINUE
          YC = 1.0-YC
          DC = XC*XC+YC*YC
          KC = K+M+2
      IF(DC-DNEAR(KC))181,181,180
  181     DNEAR(KC) = DC
          LNEAR(KC) = L
  180 CONTINUE
          XC = 1.0-XC
          DC = XC*XC+YC*YC
          KC = K+M+1
      IF(DC-DNEAR(KC))151,151,150
  151     DNEAR(KC) = DC
          LNEAR(KC) = L
      GO TO 150
  190 CONTINUE
          ISEG = ISEG+1
C
C                                    RAY HAS CROSSED A CELL BOUNDARY-TEST FLAG
C
      IF(J.GT.N.OR.J.LT.1) GO TO 200
      IF(I.GT.M.OR.I.LT.1) GO TO 200
          KF = (J-1)*M+I
      IF(FLAG(KF))200,200,201
C
C                                    CELL IS ACTIVE
C
  201     COUNT(KF) = COUNT(KF)+1
          LL(ISEG) = L
          KK(ISEG) = K
          XX(ISEG) = X - AINT(X)
          YY(ISEG) = Y - AINT(Y)
          KST = K
C     WRITE(3,18)ISEG,LL(ISEG),KK(ISEG),XX(ISEG),YY(ISEG),X,Y
  18    FORMAT(' ',I3,I5,I5,4(F7.3,4X,'CHEK'))
      GOTO 155
  200 CONTINUE
C
C                                    RAY HAS CROSSED MODEL BOUNDARY
C                                    OUTPUT END OF RAY MARKER
C
          LL(ISEG) = L
          KK(ISEG) = 0
      IF(IR.EQ.NRAY) KK(ISEG) = -1
          X = X-DX
          Y = Y+DY
          XX(ISEG) = X - AINT(X)
          YY(ISEG) = Y - AINT(Y)
C
C                                    PRINTOUT DETAILS OF RAY ON LP
C
      NCC=ISEG-IS+1
      WRITE(3,4100) IR,XSTART(IR),YSTART(IR),X,Y,NCC
      WRITE(3,4110) (KK(ISS), ISS = IS,ISEG)
      WRITE(3,4110) (LL(ISS), ISS = IS,ISEG)
  300 CONTINUE
C
C                                    PRINTOUT DENSITY DISTRIBUTION OF CELLS CROSSED
C
      WRITE(3,4200) FREQ(IF),THETA(IT),NRAY,ISEG,L
      DO 350 J=1,N
          KST = (J-1)*M+1
          KFIN = KST+M-1
  350 WRITE(3,4210)(COUNT(K),K=KST,KFIN)
C
C                                    PRINTOUT NEAREST RAYPOINTS
C
      WRITE(3,4220)
      NPAGE=M/24+1
      DO 360 IP=1,NPAGE
      WRITE (3,4240) IP
      IST=(IP-1)*24+1
      IFIN=IST+23
      IF (IP.EQ.NPAGE) IFIN=M+1
      NNN = N+1
      DO 360 J=1,NNN
      KST=(J-1)*(M+1)+IST
      KFIN=(J-1)*(M+1)+IFIN
  360 WRITE(3,4230) (LNEAR(K),K=KST,KFIN)
```

```
C
C                                    STORE RAY LENGTH
C
C
C                                    OUTPUT THE SEGMENT COORDS ETC TO EO FILE
C
         KINFT(IFT) = KOUT
      WRITE (10'KOUT) L,ISEG,(LNEAR(KI),KI=1,MNE)
      WRITE (10'KOUT) (LL(KJ),KJ=1,ISEG),(KK(KL),KL=1,ISFG)
      WRITE (10'KOUT) (XX(KM),KM=1,ISEG),(YY(KN),KN=1,ISEG)
C      WRITE(3,1) L,ISEG,(LNEAR(KI),KI=1,MNE)
C      WRITE(3,2)(LL(KJ),KJ=1,ISEG),(KK(KL),KL=1,ISEG)
C      WRITE(3,11)(XX(KM),KM=1,ISFG),(YY(KN),KN=1,ISEG)
 1    FORMAT(' ',I3,I3,20(I4,2X))
 2    FORMAT(' ',20(I4,2X))
 400  CONTINUE
 11   FORMAT(' ',20(F6.2))
  500 CONTINUE
C
C                                    OUTPUT RAY POSITIONS TO DISC FILE
C
         KOUT = 2
         NFT = NFREQ*NTHETA
      WRITE (10'KOUT) (KINFT(KI),KI=1,NFT)
      STOP
 1000 FORMAT(2I3)
 1010 FORMAT(80I1)
 1020 FORMAT(2F10.1)
 1030 FORMAT(10F6.1)
 1040 FORMAT(30I2)
 1050 FORMAT(I3,2F6.3)
 2000 FORMAT (20A4)
 4000 FORMAT ('1'///30X,'NORSWAM  PROGRAM  TRACK'///26X,20A4///30X,
     1        'FLAG GRID HAS DIMENSIONS',I4,'  BY',I4//'FLAG = 1 IS',
     2        ' ACTIVE - FLAG = 0 IS OUTSIDE MODEL'//)
 4010 FORMAT(1H0,20X,10CI1)
 4020 FORMAT(1H0//30X,14HGRID SPACING =,F20.1,3H MS,10X,10HTIMESTEP =,
     *        F20.1,6H  SECS//)
 4030 FORMAT(1H0,30X,39HNUMBER OF RAY DIRECTIONS PER QUADRANT =,I6//
     1        1H ,54X,9HOVERALL =,I6/1H ,40X,12HIN DEGREES -//)
 4040 FORMAT(1H0,30X,20F6.1)
 4050 FORMAT(1H0//30X,27HNUMBER OF FREQUENCY BINS = ,I6/1H ,30X
     1        ,13HSTARTING FROM,F7.4/1H ,30X,20HBIN SIZE =,F7.4/1H ,40X,
     2       22HCENTRED ON (IN SEC-1)'//)
 4060 FORMAT(1H0,30X,20F6.3)
 4080 FORMAT(1H1,20X,11HFREQUENCY =,F20.3,10X,12HDIRECTION = ,F10.1//1H0
     *,30X,I6,12H  RAYS INPUT//)
 4100 FORMAT(1H ,3HRAY,I4,5X,7HSTART (,F6.3,1H,,F6.3,8H)  END (,F6.3,1H,
     *,F6.3,10H) CROSSING,I5,5HCELLS)
 4110 FORMAT(1H ,20I6)
 4200 FORMAT(1H1,20X,20HSUMMARY OF FREQUENCY,F20.3,5X,9HDIRECTION,F10.1/
     */30X,I6,11H RAYS INPUT//30X,I6,14H CELLS CROSSED//30X,I6,21H RAY P
     *OINTS GENERATED//30X,31HDISTRIBUTION OF CELL CROSSING -///)
 4210 FORMAT(1H ,40I3)
 4220 FORMAT(1H0//20X,29HINDICES OF NEAREST RAY POINTS///)
 4230 FORMAT(1H ,24I5)
 4240 FORMAT(1H0,20X,4HPAGE,I3)
      END
```

```
C        MASTER   WAVE      (VERSION 1  REVISION 2)
C
C         REVISION 2 PROVIDES FOR OUTPUT TO BE PRINTED SO THAT GRID IS
C         VISUALLY CORRECT,  THAT IS, WITH THE ORIGIN IN THE LOWER LEFT.
C
C
C         ************************************************************
C
C              LISTING OF VARIABLES USED IN COMMON AREAS AND MASTER SEGMENT
C              ------------------------------------------------------------
C
C
C
C         VARIABLE        TYPE       USE
C         --------        ----       ---
C           XA            REAL       FETCH LENGTH AT INCOMING BOUNDARY
C           MX            INT        NO. OF CELLS IN E-W DIRN OF GRID
C           MY            INT               "          N-S     "
C           DELX          REAL       GRID SIZE IN METRES (ALTERED IN LW SCHEME)
C           DELT          REAL       TIMESTEP (OF WIND DATA) IN SECS
C           N             INT        NO. OF PARAMETERS USED IN MODEL
C                                      (GENERALLY 3: FM ALPHA GAMMA)
C           T             REAL       TOTAL TIME IN SECS
C           NN            INT        N*N
C           KMAX          INT        NO. OF ACTIVE GRID POINTS
C           ISC           INT        ITERATION COUNTER
C           A             ()REAL     WIND-SEA PARAMS AT MAIN GRID POINTS
C           B             ()REAL           "        AT INTERMEDIATE GRID POINTS
C           C             ()REAL           "        AT MAIN GRID POINTS (AT T + DELT)
C           WI            ()REAL     INTERPOLATED WIND DATA (U,V,U,V,U.......)
C           KARRAY        ()INT      HOLDS POSITION OF DATA IN CONDENSED ARRAYS
C           IFLAG         ()INT      SIGNIFIES PRESENCE OR ABSENCE OF WIND
C           PI            REAL       3.14159.....
C           G             REAL       GRAVITY
C           ECONST        REAL       G*G/(80*PI**4). CONSTANT IN JONSWAP SPECTRUM
C           EH            REAL       EXP(0.5)
C           FA            REAL       F ALPHA 1.8 (USED IN COUPLING COEFFICIENTS)
C           FA4           REAL       FA**(-4)
C           RROOT2        REAL       RECIPROCAL OF ROOT 2
C           EPS           REAL       0.0001
C           SLIM          REAL       0.1 (10%) MAXIMUM CHANGE IN PARAMETERS
C                                      DUE TO SOURCE FUNCTION AT EACH SUB-TIMESTEP
C           IFLAGT        ()INT      SIGNIFIES WHETHER INTERMEDIATE POINT EXISTS
C           WP            ()REAL     WIND DATA FROM PREVIOUS TIMESTEP
C           IBDY          ()INT      SIGNIFIES WHETHER GRID POINT IS AN
C                                      INCOMING BOUNDARY
C           IPBDY         ()INT      SIGNIFIES WHETHER GRID POINT WAS AN
C                                      INCOMING BOUNDARY LAST TIMESTEP
C           IVAL          ()INT      HOLDS HEADER INFORMATION FROM WIND FILE
C                                      (MX, MY ETC.) USED TO PRODUCE HEADER
C                                      ON SWELL FILE
C           RDUM          ()REAL     HOLDS HEADER INFO FROM RAY FILE
C           WN            ()REAL     WIND DATA FOR NEW TIMESTEP
C           ISWELL        ()INT        -VE IF SW -> WS ENERGY EXCHANGE
C           KSWELL        INT        NO. OF POINTS WS -> SW OCCURS
C           NFSWL         INT        NO. OF SWELL FREQUENCY BINS
C           FBIN          REAL       SWELL BIN SIZE (HERTZ)
C           FMMAX         REAL       UPPER LIMIT ON FM
C           NQUAD         INT        NO. OF SWELL RAY DIRNS PER QUADRANT
C           DBIN          REAL       SWELL DIRN INTERVAL IN RADIANS
C           FINIT         REAL       FREQUENCY OF LOWEST SWELL BIN
C           ISFC          ()INT      SWELL FREQ BIN NO. FOR WS -> SW TRANSFER
C           DEF           ()REAL     ENERGY TO BE DUMPED INTO FREQ BIN
C           DEFD          ()REAL           "              INTO FREQ-DIRN BIN
C           IDBIN         ()INT      DIRN BIN TO BE USED FOR WS -> SW
C           FW            ()REAL     G/(2*PI*U) FREQ PHASE SPEED OF WAVES = WIND
C           BETA          ()REAL     SWELL GROWTH PARAMETER
C           FC            ()REAL     CUT-OFF FREQ FOR WS -> SW
C           FRACT         ()REAL     PROPORTION OF ENERGY TO BE DISTRIBUTED
C                                      INTO DIRN BINS
C           AVECOS        ()REAL     AVERAGE COS OF ANGLE BETWEEN DIRN BIN
C                                      AND WIND
C           EC            ()REAL     SWELL ENERGY ALONG CHARACTERISTIC RAY
C          KINFT         ()INT      POINTERS TO SWELL RAY INFORMATION
C           MXY           INT        MX * MY:  TOTAL NO. OF POINTS IN MODEL
C           LNEAR         ()INT      NEAREST RAY POINT TO THE GRID POINT
C           DE            ()REAL     ENERGY IN SWELL BIN TO BE ABSORBED
C           JVAL          ()INT      USED TO OUTPUT INFORMATION TO SWELL FILE
C           E             ()REAL     SWELL ENERGY AT GRID POINT
C           LRAY          ()INT.     LENGTH OF SWELL CHARACTERISTIC RAY
C           KIN           INT        INPUT FILE POINTER
C           KSOUT         INT        SWELL OUTPUT FILE POSITION POINTER
C           KMAX5         INT        KMAX * 5
C           KMAX2         INT        KMAX + KMAX
C           KMAXP3        INT        KMAX + 3
C           ISCM          INT        NO. OF ITERATIONS FOR RUN
C           IPRINT        ()INT      FREQUENCY OF PRINTING TO LP AT UP TO
C                                      5 DIFFERENT INTERVALS
C           KPSC          ()INT      ENDING POINTS FOR EACH IPRINT INTERVAL
C           IISC          INT        INITIAL TIMESTEP (OF WIND)
C           FM            REAL       INITIAL VALUE OF FM
C           ALPHA         REAL             "          ALPHA
C           NW            INT        TIMESTEP DIVISION FOR STABILITY
C           NS            INT        TIMESTEP DIVISION FOR FIRST ITERATION FOR
C                                      ENSURING SMOOTH START
C           ISSTOR        INT        FREQUENCY OF STORAGE OF SWELL INFORMATION
C                                      (GENERALLY EVERY TIMESTEP)
C           NDSWL         INT        NO. OF SWELL DIRECTION BINS
C           NFDBIN        INT        NO. OF SWELL FREQ-DIRN BINS
C           KOUT          INT        OUTPUT FILE POINTER
C           JPRINT        INT        NO. OF ITERATIONS SINCE LAST OUTPUT TO LP
C           JSSTOR        INT        NO. OF ITERATIONS SINCE LAST STORAGE OF SWELL
C           DELTS         REAL       TIMESTEP (UNALTERED)
C                                      (DELT IS ALTERED TO EQUAL SUB-TIMESTEP)
C           NT            INT        ACTUAL TIMESTEP SUBDIVISION
C                                      (INITIALLY NS, THEN NW)
C           R             REAL       DETERMINES PROPORTION OF WN,WP IN FORMING WI
C           KSWIN         INT        POINTER TO SWELL ENERGY ON DISC
C           NOSWLL        INT        FLAG TO INITIATE SWELL ROUTINES
C                                      (ROUTINES ARE NOT STARTED UNTIL WS -> SW)
C           IFR           INT        FREQ-BIN POINTER REVERSED
```

```
C           IF             INT        FREQ-BIN POINTER
C           IST            INT        SWELL ITERATION COUNTER
C           ID             INT        DIRN-BIN POINTER
C           I              INT        DO LOOP VARIABLE
C           J              INT             "
C           K              INT             "
C                                       (USUALLY RUNS OVER MODEL GRID)
C           KK             INT        POSITION IN CONDENSED ARRAY OF CELL
C                                       (I.E.  KK=KARRAY(K)  )
C           ACOP           ()REAL     HOLDS COUPLING PARAMETERS
C           AFPM           ()REAL     PIERSON-MOSKOWITZ FREQUENCY
C           UP             ()REAL     WIND U,V COMPONENTS
C           IUX            INT        SIGN OF U COMPONENT OF WIND
C           IUY            INT          "     V       "
C           U              REAL       WIND SPEED
C           USEFILE        SUBR.      ATTACH DISC FILE
C           FREEFILE       SUBR.      RELEASE DISC FILE
C           RLEASE         SUBR.      RELEASE BASIC PERIPHERAL
C           GETVAR         SUBR.      READ FROM BACKING STORE
C           PUTVAR         SUBR.      WRITE TO BACKING STORE
C           FLAGST         SUBR.      SET MAIN POINT FLAGS ACCORDING TO WIND
C           FLGTST         SUBR.      SET INTER GRID FLAGS ACCORDING TO SURROUNDINGS
C           PRINT          SUBR.      WRITE REAL ARRAY IN GEOGRAPHICAL FORMAT
C           PRINTI         SUBR.        "   INTEGER         "
C           STEUER         SUBR.      CONTROLS UPDATING OF WIND-SEA
C           LPFROG         SUBR.      UPDATES NEW WIND-SEA ON MAIN POINTS
C           PRDICT         SUBR.      CALCULATES WIND-SEA ON INTERMEDIATE POINTS
C           BOY-           SUBR.      CALCULATES VALUES AT INCOMING BOUNDARY
C           SZ             FUNC.      CALCULATES SOURCE TERMS
C           WSTOSW         SUBR.      DETERMINES IF WS -> SW TRANSFER
C           FREBIN         SUBR.      CALCULATES WHERE AND HOW MUCH ENERGY TO TRANSFER
C           DIRBIN         SUBR.      PROPORTIONS ENERGY BETWEEN DIRNS
C                                       AND CALCULATES SWELL GROWTH
C           SHELL          SUBR.      ADVECTS SWELL, APPLIES WS <-> SW TRANSFER
C           SWTOWS         SUBR.      ABSORBS SWELL INTO WIND-SEA
C           COPZ           FUNC.      CALLS APPROPRIATE FUNC. TO CALC. COUPLING COEFFS
C           D11C           FUNC.      CORRECTION TERM FOR COEFF D11
C           D11      )
C           D12      )
C                    )
C            .       )
C            .       )     FUNC.      CALCULATE COUPLING COEFFS
C            .       )
C           D53      )
C           D55      )
C           ISIG           FUNC.      SIGN OF PARAMETER
C           MHOL           SUBR.      GETS DATA FROM STORAGE ARRAYS
C           MBESE          SUBR.      PUTS DATA INTO STORAGE ARRAYS
C           ************************************************************
C
C                       WIND-SEA COMMON AREAS
C
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CMSP/A(3500),B(3500),C(3500),W((1300)
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON/CONST/ PI,G,EH,EA,EA4,RROOT2,EPS,SLIM,ECONST
      COMMON /INTER/ IFLAGT(650)
      COMMON/PREV/ WP(1300)
      COMMON /BOUND/ IBDY(650)
      COMMON /PREBDY/ IPBDY(650)
      DIMENSION IVAL(2),WN(1300)
C
C                       SWELL COMMON AREAS
C
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FBIN,FMMAX,NQUAD,DBIN
     C              ,FINIT,ISFC(650)
      COMMON/WSTSWL/DEF(650),DEFD(650),IDBIN(650),FW(650),BETA(650)
     C              ,FC(650)
      COMMON/ANGDIS/FRACT(10),AVECOS(10)
      COMMON/SWLFLD/EC(4000),KINFT(50),MXY,LNEAR(650)
      COMMON/SWLTWS/DE(650)
      COMMON/ORDER/ JVAL(3),E(650)
C
      DIMENSION ROUM(4),TITLE(20)
      COMMON/KKRP/ KPSC(5),IPRINT(5)
C
C                       SET UP DISC FILES
C
      DEFINE FILE 4 (40,1200,U,KHIN)
      DEFINE FILE 10 (125,600,U,KIN)
      DEFINE FILE 11 (100,8000,U,KSWIN)
      DEFINE FILE 12 (1600,660,U,KSOUT)
      DEFINE FILE 13 (40,3500,U,KWOUT)
      NAMELIST /CONTR/ N,ISCM,IISC,KPSC,IPRINT,XA,FM,ALPHA,NW,NS
      NAMELIST /CONTS/ FMMAX,ISSTOR
      NAMELIST /CONTK/ KMAX,KARRAY
C
C           CALL SYSTEM SUBR. TO AVOID EXPONENT UNDERFLOW.
      CALL ERRSET(208,256,0,0,1,208)
C
C
C                 READ OR GET CONTROL AND GRID INFORMATION.
C
      READ (1,105) TITLE
  105 FORMAT (20A4)
      WRITE(3,222)TITLE
 222  FORMAT(20A4)
      READ (1,CONTR)
      READ (1,CONTS)
      READ (1,CONTK)
      KIN = 1
 1001 KWOUT = 1
      KHIN = 1
      READ (10'KIN) (JVAL(KI),KI=1,2),(IVAL(KM),KM=1,2),
     C     (ROUM(KJ),KJ=1,4)
      DELX = ROUM(3)
      DELT = ROUM(4)*NW
C
C                    SET UP GRID INFORMATION.
```

```
      MX=IVAL(1)
      MY=IVAL(2)
      MXY = MX*MY
C
C                   OUTPUT GRID INFORMATION TO WAVE PARAM FILE
C
      WRITE (13'KWOUT) KMAX,MX,MY,MXY,(KARRAY(K),K=1,MXY)
      KSOUT=1
C
C                   AND TO SWELL FILE
      WRITE(12'KSOUT) (IVAL(I),I=1,2),MXY,KMAX
      WRITE(12'KSOUT) (KARRAY(K),K=1,MXY)
C
C
C                   WRITE INFO TO LP
C
      WRITE (3,106) TITLE,ISCH,N,MY,IPRINT(1),FM,MX,ITSC,FMMAX,KMAX,
     &              DELT,ALPHA,XA,NW,DELX
  106 FORMAT ('1'//43X,'NORSWAM  WAVE  MODEL  (VERSION I  REVISION 2)',
     &        ////26X,20A4,/////////14X,'NO. OF ITERATIONS     = ',I8,
     &        10X,'NO. OF PARAM. = ',I6,10X,'NO. OF ROWS',8X,'= ',I10,
     &        //14X,'INIT. PRINTOUT FREQ. = ',I8,10X,'INITIAL FM     = ',
     &        F6.4,10X,'NO. OF COLUMNS      = ',I10,//14X,'STARTING ',
     &        'TIMESTEP      = ',I8,10X,'MAXIMUM FM     = ',F6.4,10X,
     &        'NO. OF ACT. ELEM.   = ',I10,//14X,'TIMESTEP (SECONDS)    ',
     &        '= ',F8.1,10X,'INITIAL ALPHA = ',F6.4,10X,'FETCH (METER',
     &        'S)       = ',F10.1,//14X,'TIMESTEP SUBDIVISION = ',I8,42X,
     &        'GRID SIZE (METERS)  = ',F10.1)
C
C                   SET UP CONSTANTS
C
      T = 0.
C
C                   SLIM GIVES MAXIMUM RELATIVE CHANGE IN PARAMETERS
C                   DUE TO SOURCE FUNCTION
C
      SLIM=0.1
      RROOT2=1./SQRT(2.)
      ECONST=G*G/(80.0*PI*PI*PI*PI)
      FA4=(.9**(-4.)
      EH=EXP(0.5)
      NN=N*N
      KMAX5=KMAX*5
      KMAX2=KMAX+KMAX
      KMAXP3=KMAX+3
C
C                   ALTER GRID SIZE FOR LW SCHEME
C
      DELX = DELX/RROOT2
C
C                   SET UP INITIAL CONDITIONS ON GRID
C
      DO 20 K=1,KMAX5,5
      A(K) = FM
      B(K) = FM
      C(K) = FM
      A(K+1) = ALPHA
      B(K+1) = ALPHA
      C(K+1) = ALPHA
      A(K+2) = 3.3
      B(K+2) = 3.3
      C(K+2) = 3.3
      A(K+3) = 0.07
      B(K+3) = 0.07
      C(K+3) = 0.07
      A(K+4) = 0.09
      B(K+4) = 0.09
      C(K+4) = 0.09
   20 CONTINUE
C
C                             GET CONTROL PARAMETERS FOR SWELL
C
      NDSWL = JVAL(1)
      NFSWL = JVAL(2)
      NQUAD = NDSWL/4
      FINIT = RDUM(1)
      FBIN = RDUM(2)
         DBIN = PI*0.5/NQUAD
C
C                   WRITE TO LP
C
      WRITE (3,306) NFSWL,FINIT,FBIN,NQUAD,ISSTOR
  306 FORMAT (' ',///43X,'NO. OF SWELL FREQUENCY BINS',11X,'= ',I6,
     &        //43X,'MINIMUM SWELL FREQUENCY (HERTZ)',7X,'= ',F6.3,
     &        //43X,'SWELL FREQUENCY BIN SIZE (HERTZ)',6X,'= ',F6.3,
     &        //43X,'NO. OF SWELL DIRECTIONS PER QUADRANT  = ',I6,
     &        //43X,'ITERATION FREQ. OF SWELL DATA STORAGE = ',I6,//)
      NFDBIN = NDSWL*NFSWL
C
C                   READ POINTERS TO RAYTRACKS
C
      READ (10'KIN) (KINFT(KJ),KJ=1,NFDBIN)
C
C                   INITIALISE RAY ENERGY
C
      KSWIN = 1
      DO 120 IFT=1,NFDBIN
      KIN = KINFT(IFT)
      READ (10'KIN) LR1
      KCT = LR1/2401+1
      KFIN = 0
      IF (KCT.EQ.1) GO TO 135
      DO 130 KIF=2,KCT
      KST = KFIN+1
      KFIN = KST+2399
  130 WRITE (11'KSWIN) (EC(KE),KE=KST,KFIN)
  135 KST = KFIN+1
      KFIN = LR1
  120 WRITE (11'KSWIN) (EC(KE),KE=KST,KFIN)
```

```
C                              SET UP ANGULAR DISTRIBUTION
C
      NQDP1 = NQUAD+1
      DO 300 I=1,NQDP1
      AVECOS(I)=(SIN((I-0.5)*DBIN)-SIN((I-1.5)*DBIN))/DBIN
  300    FRACT(I) = (DBIN+SIN(DBIN)*COS(2*(I-1)*DBIN))/PI
         FRACT(NQUAD+1) = 0.5*FRACT(NQUAD+1)
C
C                    SET POINTER TO START OF WIND DATA
C
      KHIN = IISC
      KPR = 1
      JPRINT=0
      JSSTOR=1
      IFLAGT(KMAX+1)=0
      DELTS=DELT
      ISC = -1
C
C                    ------------------------------
C                    START OF MAIN ITERATION LOOP.
C                    ------------------------------
C
  900 ISC = ISC +1
      JPRINT=JPRINT-1
C
C                      TIME TO PRINT OUT RESULTS?
C
      IF (JPRINT) 911,911,910
 911  JPRINT=IPRINT(KPR)
      CALL WPRINT(NW)
      CALL PRINT(A,KMAX5,5,N,NW)
  910 IF(ISC.GE.ISCM) GOTO 999
C
C                    GET WIND DATA
C
      READ (4'KHIN) (WN(KW),KW=1,KMAX2)
      WRITE(3,103) KHIN
 103  FORMAT(' ','KHIN=',I5)
      WRITE(3,104) (WN(K),K=1,KMAX2)
 104  FORMAT(' ',13(F7.3,3X))
      IF (ISC.GE.KPSC(KPR)) KPR=KPR+1
C
C                     SET IFLAG ACCORDING TO PRESENCE OF WIND
C
      CALL FLAGST(WN,WP)
C
C                    SET UNKNOWN WIND TO ZERO
C
      DO 920 K=1,KMAX2,2
      IF (WN(K).NE.-1.0.OR.WN(K+1).NE.-1.0) GO TO 920
      WN(K) = 0.0
      WN(K+1) = 0.0
  920 CONTINUE
      NT=NW
      IF (ISC) 931,931,930
 931  NT=NS
C
C                     SUBDIVIDE TIMESTEP
C
      DELT=DELTS/NS
 930  RNT=NT
C
C                  ITERATE OVER SUBDIVIDED TIMESTEPS - 1 WHOLE TIMESTEP IN TOTAL
C
      DO 950 I=1,NT
      R=I
C
C                     LINEARLY INTERPOLATE WIND IN TIME
C
      DO 940 K=1,KMAX2
C     WRITE (3,111)NT,RNT,NW,K,R,WN(K),WP(K)
  111 FORMAT (5X,'NT=',I4,4X,'RNT=',F6.3,4X,'NW=',I6,4X,
     C'K=',I3,'R=',F6.2,4X,'WN(K)=',F8.3,4X,'WP(K)=',F8.3)
      WI(K)=((RNT-R)*WP(K)+R*WN(K))/RNT
C     WRITE (3,113)WI(K)
  113 FORMAT (10X,'WI(K)=',F10.5)
  940 CONTINUE
C
C                     SET IFLAGT FOR INTERMEDIATE POINTS ACCORDING TO
C                     CONDITIONS
C
      CALL FLGTST
          KSWELL = 0
          KSWIN = 1
          NOSWLL = 0
C
C
C                     ----------------------
C                     START OF SWELL SECTION
C                     ----------------------
C
C
C
C                   IGNORE SWELL ROUTINES ON FIRST TIMESTEP
C
      IF (ISC) 250,250,251
C
C                              SET ISWELL FLAG. ADJUST FM TO LOSS OF W-S
C
 251  CALL WSTOSW
      IF(KSWELL.GT.0) NOSWLL = 1
C
C                     IF NOSWLL NOT SET, WS -> SW DUMPING HAS NOT OCCURRED AND
C                     SO SKIP SWELL ROUTINES
C
      IF(NOSWLL)250,250,252
 252  IF (I.EQ.NT) JSSTOR=JSSTOR-1
      DO 200 IFR=1,NFSWL
C
C                      REVERSE FREQ-BIN POINTER SO AS TO SCAN FROM HIGHEST TO
C                      LOWEST BIN
C
```

```
      IF=NFSWL-IFR+1
C
C
C                             SET W-S ENERGY IN IF FREQ BIN : SET ISWELL FOR
C                             SWELL TO W-S TRANSFER
      CALL FREBIN(IF)
      DC 200 ID=1,NDSWL
C
C
C                             SET W-S ENERGY IN IF ID FREQ - DIR BIN
C                             AND SWELL GROWTH
      CALL DIRBIN(IF,ID)
C
C
C
C                             PROPAGATE SWELL AND INTERCHANGE W-S SWELL ENERGY
      IFI=(IFR-1)*NDSWL+ID
      CALL SWELL(IFI,DELT,KSWIN)
      IF(JSSTOR) 201,201,200
  201 CONTINUE
C
C                   SET SWELL FIELD UP ON CARTESIAN GRID
C
      DC 400 K=1,MXY
      KK=KARRAY(K)
      IF(KK) 400,400,401
  401 L=LNEAR(K)
      E(KK)=0.0
      IF(L.GT.0) E(KK)=EC(L)
  400 CCNTINUE
C
C                      SET UP HEADER INFORMATION OF FREQ AND DIRN NUMBERS
C
      IST=ISC+1
      JVAL(1)=IST
      JVAL(2)=IF
      JVAL(3)=ID
      WRITE (12'KSOUT) (JVAL(KL),KL=1,3),(E(KM),KM=1,KMAX)
  200 CONTINUE
      IF (JSSTOR.EQ.0) JSSTOR=ISSTOR
C
C                            ADJUST FM TO ABSORB SWELL
C
      CALL SWTCHS
      TIM = ISC*NW*DELT/60.
   55 WRITE(3,6)KSWELL,TIM
    6 FORMAT(1H ,10X,45HNUMBER OF POINTS AT WHICH WS TO SWELL OCCURS=,
     &        I8,10X,'TIME =',F8.2,' MINUTES')
  250 CONTINUE
C
C                      -------------------------------
C                       UPDATE PARAMETRICAL WIND-SEA
C                      -------------------------------
C
C
      CALL STEUER
C
C                            COPY PARAMETERS C TO A AND B
C
      DO 190 K=1,KMAX5
      A(K) = C(K)
  190 B(K) = C(K)
C
C                   UPDATE BOUNDARY POINTERS
C
      DO 945 K=1,KMAX
  945 IPBDY(K)=IBDY(K)
  950 CONTINUE
C
C                   PRINTOUT ISFC FLAG AND IOBIN FIELDS
C
      IF(JPRINT-1) 955,956,955
  956 WRITE (3,957)
      CALL PRINTI(ISFC,KMAX,NW,1)
      WRITE(3,7)
    7 FORMAT(1H0,///,50X,7H*IDBIN*)
      CALL PRINTI(IOBIN,KMAX,NW,2)
C
C                      UPDATE PREVIOUS WIND DATA
C
  955 DO 960 K=1,KMAX2
  960 WP(K)=WN(K)
C
C
C                      WRITE WAVE PARAMETERS TO OUTPUT FILE
      WRITE(13'KWOUT) (A(I),I=1,KMAX5)
C
C                      SET TIMESTEP
C
      DELT=DELTS/NW
C
C
C                      GO TO DO NEXT ITERATION
      GOTC 900
  957 FORMAT('1',///,52X,'ISFC')
C
C                     ------------
C                      END OF RUN
C                     ------------
C
C
C                      TEST TO SEE IF FINAL ITERATION HAS BEEN PRINTED
  999 IF (JPRINT.EQ.IPRINT(KPR)) STOP
      CALL PRINT (A,KMAX5,5,N,NW)
      ISC=ISC-1
      WRITE (3,957)
      CALL PRINTI(ISFC,KMAX,NW,1)
      WRITE(3,7)
      CALL PRINTI(IOBIN,KMAX,NW,2)
```

```
      STOP
      DEBUG UNIT(3),SUBCHK,TRACE
C     AT 100
C     TRACE ON
C     END
C    *******************************************************
C    *******************************************************
C
      BLOCK DATA
C
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CMSP/ A(3500),B(3500),C(3500),W(1300)
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON/CONST/ PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      COMMON /INTER/ IFLAGT(650)
      COMMON/PREV/ WP(1300)
      COMMON /BOUND/ IBDY(650)
      COMMON /PREBDY/ IPBDY(650)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FBIN,FMMAX,NQUAD,DBIN
     &               ,FINIT,ISFC(650)
      COMMON/WSTSWL/DEF(650),DEFD(650),IDBIN(650),FW(650),BETA(650)
     &               ,FC(650)
      COMMON/ANGDIS/FRACT(10),AVFCOS(10)
      COMMON/SWLFLD/FC(4000),KINFT(50),MXY,LNEAR(650)
      COMMON/SWLTHS/DE(650)
      COMMON/ORDER/ JVAL(3),E(650)
      COMMON/CWIND/ UP(2),IUX,IUY,U
      COMMON /COPPAR/ ACCP(11700)
      COMMON /PM/ AFPM(650)
      COMMON/KKPR/ KPSC(5),IPRINT(5)
C
      DATA XA,DELX,DELT,T,EH,FA4,RROOT2,SLIM,EPS/8*0.0,0.0001/
      DATA ECONST,FBIN,FMMAX,DBIN,FINIT,U,UP/8*0.0/
      DATA FRACT,AVECOS,PI,G,FA/20*0.0,3.14159,9.806,1.8/
      DATA MX,MY,N,NN,KMAX,ISC,KSWELL,NFSWL,NQUAD/9*0/
      DATA IUX,IUY,MXY,JVAL/6*0/,KINFT,LNEAR/700*0/
      DATA KARRAY,IFLAG,IFLAGT,IBDY,IPBDY,ISWELL/3900*0/
      DATA ISFC,IDBIN/1300*0/,DEF,DEFD,FW,BETA/2600*0.0/
      DATA FC,DE,E,AFPM/2600*0.0/,WI,WP/2600*0.0/
      DATA A,B,C/10500*0.0/,ACOP,EC/11700*0.0,4000*0.0/
      DATA KPSC,IPRINT/10*0/
      END
C
C    *******************************************************
C    *******************************************************
C
      SUBROUTINE FLAGST(WI,WP)
C
C                    SET IFLAG ACCORDING TO PRESENCE OF WIND
C
      DIMENSION WI(1300),WP(1300)
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      DO 100 I=1,MY
      KST=(I-1)*MX
      DO 100 J=1,MX
      K=KST+J
      KK=KARRAY(K)
      IF(KK) 100,100,101
 101  IFLAG(KK)=1
      KWIND=KK+KK-1
      IF(WI(KWIND).EQ.-1.0.AND.WI(KWIND+1).EQ.-1.0.AND.
     &   WP(KWIND).EQ.0.0.AND.WP(KWIND+1).EQ.0.0) IFLAG(KK)=0
 100  CONTINUE
      RETURN
      END
C    *******************************************************
C    *******************************************************
C
      SUBROUTINE FLGTST
C
C                    SUBROUTINE SETS IFLAGT ACCORDING TO SURROUNDING
C                    CONDITIONS.
C
C                      IFLAGT ZERO => ONE OR MORE SURROUNDING POINTS DO
C                                     NOT EXIST OR HAVE NO WIND
C                             +VE  => ALL OK
C                             -VE  => A SURROUNDING POINT IS EITHER AN
C                                     INCOMING BOUNDARY POINT OR A
C                                     PIERSON-MOSKOWITZ POINT
C
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON /INTER/ IFLAGT(650)
      COMMON /BOUND/ IBDY(650)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FBIN,FMMAX,NQUAD,DBIN
     &               ,FINIT,ISFC(650)
C
C                    SCAN THROUGH GRID
C
      DO 200 I=1,MY
      KST=(I-1)*MX
      DO 200 J=1,MX
      K=KST+J
C
C                  SET COMPASS
C
      KK=KARRAY(K)
      KW=KK
      IF(KK) 200,200,203
 203  IFLAGT(KK)=0
      IF(I.EQ.1.OR.J.EQ.MX) GOTO 200
      KN=KARRAY(K-MX)
      KS=KARRAY(K+1)
      KE=KARRAY(K-MX+1)
C
C                    TEST IF ALL SURROUNDING POINTS EXIST
C
      IF(KN.EQ.0.OR.KS.EQ.0.OR.KE.EQ.0) GOTO 200
C
```

```
C                      TEST IF WIND DATA EXISTS
C
      IF(IFLAG(KN).EQ.0.OR.IFLAG(KS).EQ.0.OR.IFLAG(KW).EQ.0.OR.
     &   IFLAG(KE).EQ.0) GOTO 200
C
C                      SET FLAG +VE, UNLESS......
C
      IFLAGT(KK)=1
C
C                      ANY SURROUNDING POINTS ARE INCOMING BOUNDARIES
C
      IF(IBDY(KN).LT.0.OR.IBDY(KS).LT.0.OR.IBDY(KW).LT.0.OR.
     &   IBDY(KE).LT.0) IFLAGT(KK)=-1
C
C                      OR ARE PIERSON-MOSKOWITZ POINTS
C
      IF(ISFC(KN).GT.0.OR.ISFC(KS).GT.0.OR.ISFC(KW).GT.0.OR.
     &   ISFC(KE).GT.0) IFLAGT(KK)=-1
C      WRITE (3,350) KK,KN,KS,KW,KE
 200  CONTINUE
C      WRITE(3,102)(IFLAG(KK),IFLAGT(KK),KK=1,KMAX)
C     WRITE (3,300) (IBDY(K),ISFC(K),K=1,KMAX)
  300 FORMAT (/,' ',(21I5))
  350 FORMAT (/,' ',7I7)
      RETURN
      END
C   ******************************************************
C   ******************************************************
C
      SUBROUTINE WPRINT(NW)
C
C            PRINTS OUT WINDFIELD BY X AND Y COORDINATES.
C
      DIMENSION AP(20),MES1(2),MES4(3)
      COMMON /CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON/PREV/ WP(1300)
      DATA MES1/'** W','IND:'/,MES2/'  X '/,MES3/'  Y '/
      DATA MES4/'DIRE','CTIO','N **'/
      TIME = NW*ISC*DELT/60.
      MES=MES2
C
C        SET NO. OF VALUES TO BE PRINTED ACROSS PAGE
C
      NU = 15
C
C        CALCULATE NO. OF PAGE WIDTHS REQUIRED
C
      NPAGE = (MY-1)/NU+1
      WRITE (3,100)
      KW = 1
   10 WRITE (3,400) MES1,MES,MES4
      DO 60 L=1,NPAGE
      WRITE (3,200) L,TIME,ISC
C
C                     CALCULATE NO. OF VALUES TO BE PRINTED ACROSS PAGE
C
      LL=NU
      IF(NU*L.GT.MX) LL=MX-(L-1)*NU
      LF=LL+(L-1)*NU
      LS=LF-LL+1
      WRITE (3,250)
      WRITE (3,270) (LI,LI=LS,LF)
      WRITE (3,280)
      DO 50 IJ=1,MY
      I=MY-IJ+1
      KST=(I-1)*MX+(L-1)*NU
      DO 70 J=1,LL
      K=KST+J
      AP(J)=0.0
      KK=KARRAY(K)
C                     IF CELL NOT ACTIVE, PRINT ZERO
C
      IF (KK.NE.0) AP(J) = WP((KK-1)*2+KW)
   70 CONTINUE
      WRITE (3,300) I,(AP(J),J=1,LL)
   50 CONTINUE
 60   CONTINUE
      IF (KW.EQ.2) GO TO 20
      KW = 2
      MES=MES3
      GO TO 10
   20 WRITE (3,500)
  100 FORMAT ('1',//)
  200 FORMAT (' ',//5X,4HPAGE,I3,34X,7HTIME = ,F8.2,8H MINUTES,33X,
     &     13HITERATION NO.,I4//)
  250 FORMAT (' ',59X,'X DIRECTION')
  270 FORMAT (' ','     Y ',5X,15(I2,6X))
  280 FORMAT (' ','     - ',5X,15('--',6X))
  300 FORMAT (' ',2X,I3,'. ',15F8.2)
  400 FORMAT ('0',///44X,6A4//)
  500 FORMAT (///)
      RETURN
      END
C   ******************************************************
C   ******************************************************
C
      SUBROUTINE PRINT (A,M,NP,NS,NW)
C
C                    SUBROUTINE PRINTS REAL ARRAY A IN GEOGRAPHICAL FORM
C                    ARRAY HAS NP VALUES PER CELL (I.E. FM,ALPHA ETC)
C                    REQUIRED TO PRINT FIRST NS (USUALLY 3: FM, ALPHA, GAMMA)
C
      DIMENSION A(M),AP(20),MESS(10)
      COMMON /CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      DATA MESS /'** F','M **','*ALP','HA* ','*GAM','MA *',
     &      '*SIG','MA A','*SIG','MA B'/
      TIME=NW*ISC*DELT/60.
C
```

```
C                   SET NO. OF VALUES TO BE PRINTED ACROSS PAGE
C
      NU=15
C
C                   CALCULATE NO. OF PAGE WIDTHS REQUIRED
C
      NPAGE=(MX-1)/NU+1
      WRITE (3,600)
      DO 100 II=1,NS
C
C                   WRITE APPROPRIATE TITLE
C
      IJ = 2*(II-1)+1
      WRITE (3,500) MESS(IJ),MESS(IJ+1)
      DO 60 L=1,NPAGE
      WRITE (3,200) L,TIME,ISC
C
C                   CALCULATE NO. OF VALUES TO BE PRINTED ACROSS PAGE
C
      LL=NU
      IF(NU*L.GT.MX) LL=MX-(L-1)*NU
      LF=LL+(L-1)*NU
      LS=LF-LL+1
      WRITE (3,250)
      WRITE (3,270) (LI,LI=LS,LF)
      WRITE (3,280)
      DO 50 JI=1,MY
      I=MY-JI+1
      KST=(I-1)*MX+(L-1)*NU
      DO 70 J=1,LL
      K=KST+J
      AP(J)=0.0
      KK=KARRAY(K)
C                   IF CELL NOT ACTIVE, PRINT ZERO
C
      IF (KK.NE.0) AP(J)=A((KK-1)*NP+II)
   70 CONTINUE
      WRITE (3,300) I,(AP(J),J=1,LL)
   50 CONTINUE
 60   CONTINUE
      WRITE (3,400)
  100 CONTINUE
  200 FORMAT (' ',//10X,4HPAGE,I3,34X,7HTIME = ,F8.2,8H MINUTES,33X,
     C       13HITERATION NO.,I4//)
  250 FORMAT (' ',59X,'X DIRECTION')
  270 FORMAT (' ',' Y ',5X,15(I2,6X))
  280 FORMAT (' ',' - ',5X,15('--',6X))
  300 FORMAT (' ',2X,I3,'. ',15F8.4)
  400 FORMAT(///)
 500  FORMAT ('0',///50X,2A4)
  600 FORMAT ('1',//)
      RETURN
C      DEBUG UNIT(3),SUBCHK
      END
C   ****************************************************
C   ****************************************************
C
       SUBROUTINE PRINTI(IA,M,NW,IM)
C
C                             PRINTS INTEGER ARRAY IA IN GEOGRAPHICAL FORMAT
C
      DIMENSION IA(M),IAP(30)
      COMMON/CINT/XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
          NU = 30
          NPAGE = (MX-1)/NU+1
          IST=ISC+1
      TIME = NW*IST*DELT/60.
      DO 60 L=1,NPAGE
      WRITE(3,200)L,TIME,IST
          LL = NU
      IF(NU*L.GT.MX) LL=MX-(L-1)*NU
      LF=LL+(L-1)*NU
      LS=LF-LL+1
      WRITE (3,250)
      WRITE (3,270) (LI,LI=LS,LF)
      WRITE (3,280)
      DO 50 JI=1,MY
      I=MY-JI+1
          KST = (I-1)*MX+(L-1)*NU
      DO 70 J=1,LL
          K = KST+J
          IAP(J) = 0
          KK = KARRAY(K)
      IF(KK.NE.0) IAP(J) = IA(KK)
   70 CONTINUE
      WRITE(3,300) I,(IAP(J),J=1,LL)
   50 CONTINUE
   60 CONTINUE
  200 FORMAT (' ',//10X,4HPAGE,I3,34X,7HTIME = ,F8.2,8H MINUTES,33X,
     C       13HITERATION NO.,I4//)
  250 FORMAT (' ',59X,'X DIRECTION')
  270 FORMAT (' ',' Y ',3X,30(I2,2X))
  280 FORMAT (' ',' - ',3X,30('--',2X))
  300 FORMAT (' ',2X,I3,'.',30I4)
      RETURN
      END
C   ****************************************************
C   ****************************************************
C
      SUBROUTINE STEUER
C
C                             SUBROUTINE CONTROLS UPDATING OF WIND-SEA
C                             I.E. DECIDES IF POINT IS BOUNDARY POINT OR
C                             ADVECTION POINT.
C                                  ALSO CALCULATES COUPLING COEFFICIENTS FOR
C                             ALL ACTIVE POINTS
C                                  AS FINITE DIFFERENCE SCHEME USED (LAX-WENDROFF)
C                             IS TWO-LEVEL METHOD, POINTS ARE UPDATED BY
C                             FIRST PREDICTING INTERMEDIATE POINTS AND THEN
```

```
C                                   LEAPFROGING THROUGH
C                                       POINTS ARE BOUNDARY POINTS IF LAND LIES
C                                    IN UPWIND QUADRANT
C
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON /INTER/ IFLAGT(650)
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CWIND/ UP(2),IUX,IUY,U
      COMMON /COPPAR/ ACCP(11700)
      COMMON /CONST/PI,G,EH,EA,EA4,RROOT2,EPS,SLIM,ECONST
      COMMON /BOUND/ IBDY(650)
      DIMENSION H(5)
      WMIN=0.5
      IV = 1
      IVH = 2
      JJ=0
C
C                                     SET COUPLING COEFFICIENTS AT EACH ACTIVE POINT
C
  130 DO 160 K=1,KMAX
      IF(IV-1) 11,11,10
   11 IF(IFLAG(K)) 15,15,135
   10 IF(IFLAGT(K)) 135,15,135
   15 JJ=JJ+NN
      GOTO 160
C
C                       GET PARAMS FOR CELL K
C
  135 CALL WHOL(K,H,IV)
      DO 150 IG=1,N
      DO 150 L=1,N
      JJ=JJ+1
C
C                       CALCULATE COEFFICIENTS
C
  150 ACCP(JJ)=COP2(H,IG,L)
  160 CONTINUE
C
C                             SCAN THROUGH GRID
C
      DO 300 I=1,MY
      KST=(I-1)*MX
      DO 300 J=1,MX
      K=KST+J
      KK=KARRAY(K)
C
C                             CELL ACTIVE?
C
      IF (KK) 300,300,301
C
C                             FIRST STAGE? IF SO, PREDICT ACTIVE INTERMEDIATE
C                             POINTS
C
  301 IF (IV-1) 21,21,20
   21 IF (IFLAGT(KK)) 302,300,302
  302 CALL PREDICT (K)
      GOTO 300
   20 CONTINUE
C
C                             SECOND STAGE - I.E. CALL LPFROG OR BDY
C
C                             DOES WIND DATA EXIST?
C
      IF(IFLAG(KK)) 300,300,303
  303 IBDY(KK)=1
C
C                       GET WIND
C
      CALL WHOL (KK,UP,4)
      IUX=ISIG(UP(1))
      IUY=ISIG(UP(2))
C
C                             FIND QUADRANT WIND IS BLOWING FROM
C                              NE = 1, SE = 2, SW = 3, NW = 4
C
      IQ=(5-IUX*IUY+IUX+IUX)/2
      IF (IUX.EQ.0.AND.IUY.EQ.0) GOTO 300
      U=SQRT(UP(1)*UP(1)+UP(2)*UP(2))
C
C                       ALTER WIND SPEED TO MINIMUM IF LESS BUT NOT ZERO
C
      IF(U-WMIN) 31,31,30
   31 UP(1)=UP(1)*WMIN/U
      UP(2)=UP(2)*WMIN/U
      U=WMIN
   30 CONTINUE
C
C                       TURN WIND THROUGH 45 DEGREES
C                         - TO CORRESPOND WITH LAX-WENDROFF GRID
C
      Y=(UP(2)-UP(1))*RROOT2
      UP(1)=(UP(1)+UP(2))*RROOT2
      UP(2)=Y
C
C                         SET COMPASS
C
      KN=0
      IF(J.NE.1) KN=KARRAY(K-1)
      IF (KN.EQ.0) KN=KMAX+1
      KE=KK
      KS=0
      IF(I.NE.MY) KS=KARRAY(K+MX)
      IF (KS.EQ.0) KS=KMAX+1
      KW=0
      IF(I.NE.MY.AND.J.NE.1) KW=KARRAY(K+MX-1)
      IF (KW.EQ.0) KW=KMAX+1
C
C                      CHECK FOR WIND BLOWING PARALLEL TO WIND GRID
C
      IF (IUX) 402,400,402
```

```
  402  IF (IUY) 501,500,501
C
C                       CHECK IF LAND OR NO WIND IN UPWIND QUADRANT
C
 501   GOTO (1,2,3,4),IQ
    1  IF(IFLAGT(KE)) 116,200,116
    2  IF(IFLAGT(KS)) 117,200,117
    3  IF(IFLAGT(KW)) 118,200,118
    4  IF(IFLAGT(KN)) 119,200,119
C
C                       IF DATA EXISTS, LPFROG
C
  116 CALL LPFROG(K,KN,KS,KE,KW,K)
      GOTO 300
  117 CALL LPFROG(K,KN,KS,KE,KW,K+MX)
      GOTO 300
  118 CALL LPFROG(K,KN,KS,KE,KW,K+MX-1)
      GOTO 300
  119 CALL LPFROG(K,KN,KS,KE,KW,K-1)
      GOTO 300
C
C                       INCOMING BOUNDARY - CALCULATE USING JONSWAP FORMULAE
C
  200 CALL BOY(KK,IVH)
C      WRITE (3,250) KK,KN,KS,KW,KE,I,J,K
  250 FORMAT (/,' ',8I7)
C
C                       SET MARKER, TO INDICATE NOT TO USE THIS POINT AS
C                       DOWNWIND SIDE OF A DERIVATIVE
C
      IBOY(KK)=-1
      GOTO 300
C
C                       WIND IN Y-DIRECTION ONLY (ON WIND GRID)
C                       EITHER QUADRANT WILL SUFFICE AS UPWIND
C
  400 IF(IUY) 414,300,423
  414 IF(IFLAGT(KE).NE.0) GOTO 116
      IF(IFLAGT(KN).NE.0) GOTO 119
      GOTO 200
  423 IF(IFLAGT(KS).NE.0) GOTO 117
      IF(IFLAGT(KW).NE.0) GOTO 118
      GOTO 200
C
C                       WIND IN X-DIRECTION ONLY
C
  500 IF(IUX) 512,300,534
  512 IF(IFLAGT(KE).NE.0) GOTO 116
      IF(IFLAGT(KS).NE.0) GOTO 117
      GOTO 200
  534 IF(IFLAGT(KN).NE.0) GOTO 119
      IF(IFLAGT(KW).NE.0) GOTO 118
      GOTO 200
  300 CONTINUE
C
C                       CHANGE TO SECOND STAGE OR FINISH
C
      IF (IV.EQ.2) GOTO 140
      IV = 2
      IVH = 1
      GOTO 130
  140 CONTINUE
C
C                       UPDATE TIMESTEP
C
      T = T+DELT
      RETURN
      END
C   ******************************************************
C   ******************************************************
C
      SUBROUTINE LPFROG (K,KN,KS,KE,KW,KU)
C
C                       SUBROUTINE TO LPFROG THROUGH INTERMEDIATE POINTS
C                       DEFINED AT TIMESTEP T + 0.5 DELT.
C                         ONLY OUTGOING BOUNDARIES NEED BE CONSIDERED AS
C                       SUBROUTINE STEUER WOULD HAVE CALLED BOY IF
C                       INCOMING.
C                         BOUNDARIES ARE MODELLED TO REFLECT NO ENERGY BY
C                       USING APPROPRIATE ONE-SIDED DERIVATIVES.  THESE
C                       ARE CENTRED ON AN INTERMEDIATE POINT UPWIND IF
C                       POSSIBLE (I.E. THE MAIN GRID POINTS ARE USED).
C                         IF THIS WOULD ENTAIL USING EITHER AN INCOMING
C                       BOUNDARY POINT OR A PIERSON-MOSKOWITZ POINT(ONE
C                       THAT IS FULLY DEVELOPED) AS ONE OF THE DOWNWIND
C                       POINTS, SIMPLE UPWIND DERIVATIVES ARE USED IF DATA
C                       EXISTS ON THEM. (PROBLEMS DEVELOP IF A DOWNWIND
C                       POINT IS HELD ARTIFICIALLY HIGH IN THE FREE
C                       ADVECTION OF ENERGY)
C                         WHEN UPDATING THE PARAMETERS, IF THE SOURCE TERM
C                       IS FOUND TO PRODUCE A CHANGE LARGER THAN ALLOWED
C                       (SLIM), THE TIMESTEP IS SUB-DIVIDED AND, ASSUMING
C                       THE ADVECTION TERMS TO BE UNALTERED, THE SOURCE
C                       TERM IS CALCULATED AND APPLIED PROGRESSIVELY.
C                         THE TIMESTEP, IF IT IS TO BE DIVIDED, IS ALTERED
C                       BY AT MOST HALF.
C                          TO STOP A LARGE ADVECTION TERM CAUSING AN
C                       OVERSHOOT, A SEMI-IMPLICIT FORM IS TAKEN IF
C                       NECESSARY.
C
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON /CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON /CWIND/ UP(2),IUX,IUY,U
      COMMON /COPPAR/ ACOP(11700)
      COMMON /INTER/ IFLAGT(650)
      COMMON/CONST/ PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      COMMON /PREBOY/ IPBOY(650)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FMIN,FMMAX,NQUAD,DBIN
     C             ,FINIT,ISFC(650)
      DIMENSION PARE(5),PARW(5),PARN(5),PARS(5),PAR(5)
```

```
      DIMENSION ADVECT(5),HS(5)
      UX=UP(1)
      UY=UP(2)
      KK=KARRAY(K)
C
C                 STORE COMPASS
C
      KKW=KW
      KKE=KE
      KKN=KN
      KKS=KS
      CALL MHOL (KK,PAR,1)
C
C                SET COMPASS TO POINTS TO BE USED FOR X-DERIV
C                IVX=2 == INTERMEDIATE TIME LEVEL
C                IVX=1 == OLD TIME LEVEL
C                IVX=0 == NO DERIV CAN BE TAKEN
C
      IVX=2
      IF(IFLAGT(KE)) 61,95,60
   61 IF(UX) 60,60,95
   60 IF(IFLAGT(KW)) 71,95,70
   71 IF(UX) 95,70,70
   70 CONTINUE
C
C                 SET COMPASS FOR Y-DERIV
C
      IVY=2
      IF(IFLAGT(KN)) 81,95,80
   81 IF(UY) 80,80,95
   80 IF(IFLAGT(KS)) 91,95,90
   91 IF(UY) 95,90,90
C
C                   TRY CENTRED DERIV UPWIND
C
   95 IVX=1
      IVY=1
      KW=KARRAY(KU)
      IF (KW.EQ.0) KW=KMAX+1
      KKU = KW
      KN=KARRAY(KU-MX)
      IF (KN.EQ.0) KN=KMAX+1
      KE=KARRAY(KU-MX+1)
      IF (KE.EQ.0) KE=KMAX+1
      KS=KARRAY(KU+1)
      IF (KS.EQ.0) KS=KMAX+1
C
C                   CHECK IF ALL POINTS OK
C
      IF(IFLAGT(KKU)) 92,92,90
C
C                   SOME POINTS ARE BDY OR PM VALUES
C                   MAKE SURE NOT DOWNWIND IN X OR Y DERIV
C                   UNLESS DOWNWIND POINT IS UPDATE POINT
C
   92 IF((IPBDY(KW).LT.0.OR.ISFC(KW).GT.0).AND.UX.LT.0.0.AND.KK.NE.KW)
     & GOTO 110
      IF((IPBDY(KE).LT.0.OR.ISFC(KE).GT.0).AND.UX.GT.0.0.AND.KK.NE.KE)
     & GOTO 120
      IF((IPBDY(KN).LT.0.OR.ISFC(KN).GT.0).AND.UY.GT.0.0.AND.KK.NE.KN)
     & GOTO 130
      IF((IPBDY(KS).LT.0.OR.ISFC(KS).GT.0).AND.UY.LT.0.0.AND.KK.NE.KS)
     & GOTO 140
C
C                   POINTS OK - SKIP
C
      GOTO 90
C
C                   A DOWNWIND POINT IS A BDY OR PM POINT
C                   TRY TO TAKE DERIV IN NORMAL UPWIND DIRN
C                   - FIRST CHECKING IF POINT EXISTS
C
  110 IVX=1
      IF(IFLAGT(KKE)) 151,150,151
  151 KW=KK
      KE=KARRAY(K-MX+1)
      GOTO 90
  120 IVX=1
      IF(IFLAGT(KKW)) 152,150,152
  152 KW=KARRAY(K+MX-1)
      KE=KK
      GOTO 90
  130 IVY=1
      IF(IFLAGT(KKS)) 171,170,171
  171 KN=KK
      KS=KARRAY(K+MX+1)
      GOTO 90
  140 IVY=1
      IF(IFLAGT(KKN)) 172,170,172
  172 KN=KARRAY(K-MX-1)
      KS=KK
      GOTO 90
C
C                   IF CANNOT GET ONE DERIV NORMALLY CHECK
C                   IF POSSIBLE TO GET IT BY ALLOWING DOWNWIND
C                   PM VALUE IF UPWIND POINT IS PM ALSO
C
  150 IF(ISFC(KE).GT.0.AND.ISFC(KW).GT.0) GOTO 90
      IVX=0
      GOTO 90
  170 IF(ISFC(KN).GT.0.AND.ISFC(KS).GT.0) GOTO 90
      IVY=0
   90 CONTINUE
  100 U1=UX/U
      U2=UY/U
C
C                   GET REQUIRED PARAMETERS
C
      IF (IVX) 250,250,253
  253 CALL MHOL (KE,PARE,IVX)
```

```
      CALL MHOL (KW,PARW,IVX)
  250 IF (IVY) 300,300,304
  304 CALL MHOL (KN,PARN,IVY)
      CALL MHOL (KS,PARS,IVY)
  300 KNP=(KN-1+(IVY-1)*KMAX)*NN
      KSP=(KS-1+(IVY-1)*KMAX)*NN
      KEP=(KE-1+(IVX-1)*KMAX)*NN
      KWP=(KW-1+(IVX-1)*KMAX)*NN
      REM=DELT
C
C                    CALCULATE ADVECTION ASSUMING CONSTANT OVER DELT
C
      DO 40 I=1,N
      HDX1=0.
      HDX2=0.
      IF (IVX) 30,30,32
   32 DO 20 L=1,N
      KEP=KEP+1
      KWP=KWP+1
   20 HDX1=HDX1+(ACOP(KEP)+ACOP(KWP))*(PARE(L)-PARW(L))
   30 IF (IVY) 40,40,41
   41 DO 35 L=1,N
      KNP=KNP+1
      KSP=KSP+1
   35 HDX2=HDX2+(ACOP(KNP)+ACOP(KSP))*(PARN(L)-PARS(L))
   40 ADVECT(I)=0.425*(HDX1*U1+HDX2*U2)/DELX
  305  SMAX=0.
C
C                    CALC SOURCE TERM AND FIND THE LARGEST RELATIVE
C                    INCREASE OF PAR
C
      DO 310 IG=1,N
      S = S2(PAR,IG)
      HS(IG) = S
      RSIZE=ABS(S*REM/PAR(IG))
      IF(RSIZE-SMAX) 310,310,311
  311 SMAX=RSIZE
  310 CONTINUE
      DT=REM
      REMH=REM*0.5
C
C                    TEST IF LARGEST CHANGE IS GREATER THAN LIMIT
C
      IF(SMAX-SLIM) 320,320,321
C
C                    CHANGE DT ACCORDINGLY - BUT MAKE CHANGE
C                     AT MOST HALF
C
  321 DT=AMIN1(REMH,SLIM/SMAX*REM)
  320 REM=REM-DT
      DO 10 IG=1,N
      ADV=ADVECT(IG)*DT
      S=HS(IG)*DT
      IF (ADV) 4,4,5
    4 PAR(IG)=PAR(IG)-ADV+S
      GOTO 50
C
C                    USE SEMI-IMPLICIT FORM FOR ADV TO ENSURE NO
C                    OVERSHOOT
C
    5 PAR(IG)=(PAR(IG)+S)/(1.0+ADV/PAR(IG))
   50 IF(IG-3) 200,204,200
C
C                    GAMMA HAS MINIMUM OF ONE
C
  204 IF(PAR(3).LT.1.0) PAR(3)=1.0
  200 IF(PAR(IG)) 9,9,10
    9 PAR(IG)=0.0
   10 CONTINUE
C
C                    TEST IF ANY TIMESTEP LEFT IF SUBDIVISION HAS BEEN USED
C
      IF(REM-EPS) 306,306,305
  306 CALL MBESE (KK,PAR,L)
      RETURN
      END
C   ******************************************************
C   ******************************************************
C
      SUBROUTINE PROICT (K)
C
C                    SUBROUTINE TO PROICT THE VALUE OF THE INTERMEDIATE
C                    POINTS AT TIMESTEP T + 0.5 DELT.
C                      NO PROBLEMS ARISE AT BOUNDARIES BECAUSE THE
C                    INTERMEDIATE POINTS ARE NOT DEFINED UNLESS COMPLETELY
C                    SURROUNDED BY MAIN GRID POINTS
C
C                     THE TREATMENT OF THE SOURCE TERM IS THE SAME AS THAT
C                     OUTLINED FOR SUBROUTINE LPFROG
C
      DIMENSION PARN(5),PARS(5),PARE(5),PARW(5),PAR(5)
      DIMENSION ADVECT(5),HS(5)
      DIMENSION UN(2),US(2),UE(2),UW(2)
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON /CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON /CWIND/ UP(2),IUX,IUY,U
      COMMON /COPPAR/ ACOP(1700)
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      WMIN=0.5
C
C                    SET COMPASS
C
      KW=KARRAY(K)
      KN=KARRAY(K-MX)
      KS=KW+1
      KE=KN+1
C
C                    GET NECESSARY PARAMETERS AND WIND DATA
C
      CALL MHOL(KE,PARE,1)
```

```
      CALL MHOL(KN,PARN,1)
      CALL MHOL(KS,PARS,1)
      CALL MHOL(KW,PARW,1)
      CALL MHOL(KE,UE,4)
      CALL MHOL(KN,UN,4)
      CALL MHOL(KS,US,4)
      CALL MHOL(KW,UW,4)
C
C                  CALC AVERAGE PAR AND WIND AT CENTRE
C
      DO 2 I=1,5
 2    PAR(I)=(PARE(I)+PARW(I)+PARN(I)+PARS(I))*0.25
      DO 6 I=1,2
 6    UP(I)=(UE(I)+UW(I)+UN(I)+US(I))*0.25
      U=SQRT(UP(1)*UP(1)+UP(2)*UP(2))
      IF (U) 100,100,102
 102  IF(U-WMIN) 8,8,7
   8  UP(1)=UP(1)*WMIN/U
      UP(2)=UP(2)*WMIN/U
      U=WMIN
 7    CONTINUE
C
C                  TURN WIND THROUGH 45 DEGREES
C
      Y=(UP(2)-UP(1))*RROOT2
      UP(1)=(UP(1)+UP(2))*RROOT2
      UP(2)=Y
      U1=UP(1)/U
      U2=UP(2)/U
      KEP=(KE-1)*NN
      KWP=(KW-1)*NN
      KNP=(KN-1)*NN
      KSP=(KS-1)*NN
      REM=DELT*0.5
C
C                  CALC ADVECTION ASSUMING CONSTANT OVER HALF DELT
C
      DO 25 I=1,N
      HDX1=0.
      HDX2=0.
      DO 20 L=1,N
      KEP=KEP+1
      KWP=KWP+1
      KNP=KNP+1
      KSP=KSP+1
      HDX1=HDX1+(ACOP(KWP)+ACOP(KEP))*(PARE(L)-PARW(L))
      HDX2=HDX2+(ACOP(KNP)+ACOP(KSP))*(PARN(L)-PARS(L))
 20   CONTINUE
 25   ADVECT(I)=0.425*(HDX1*U1+HDX2*U2)/DELX
 305  SMAX=0.
C
C                  CALC SOURCE TERM AND FIND THE LARGEST RELATIVE
C                  INCREASE OF PAR
C
      DO 310 IG=1,N
      S=S2(PAR,IG)
      HS(IG) = S
      RSIZE=ABS(S*REM/PAR(IG))
      IF(RSIZE-SMAX) 310,310,312
 312  SMAX=RSIZE
 310  CONTINUE
      DT=REM
      REMH=REM*0.5
C
C                  TEST IF LARGEST CHANGE IS GREATER THAN LIMIT
C
      IF(SMAX-SLIM) 320,320,322
C
C                  CHANGE DT ACCORDINGLY - BUT MAKE CHANGE
C                   AT MOST HALF
C
 322  DT=AMIN1(REMH,SLIM/SMAX*REM)
 320  REM=REM-DT
      DO 10 IG=1,N
      ADV=ADVECT(IG)*DT
      S=HS(IG)*DT
      IF (ADV) 3,3,5
   3  PAR(IG)=PAR(IG)-ADV+S
      GOTO 50
C
C                  USE SEMI-IMPLICIT FORM FOR ADV TO ENSURE NO
C                  OVERSHOOT
C
 5    PAR(IG)=(PAR(IG)+S)/(1.0+ADV/PAR(IG))
 50   IF(IG-3) 200,205,200
C
C                  GAMMA HAS MINIMUM OF ONE
C
 205  IF(PAR(3).LT.1.0) PAR(3)=1.0
 200  IF(PAR(IG)) 1,1,10
   1  PAR(IG)=0.0
 10   CONTINUE
C
C                  TEST IF ANY TIMESTEP LEFT IF SUBDIVISION HAS BEEN USED
C
      IF(REM-EPS) 100,100,305
C
C                  STORE UPDATED PARAMETERS
C
 100  CALL MRESE(KW,PAR,2)
      RETURN
      END
C   *****************************************************
C   *****************************************************
C
      SUBROUTINE BDY(KK,IVH)
C
C                  SUBROUTINE SETS PARAMETERS AT INCOMING BOUNDARIES
C                  ACCORDING TO JONSWAP-TYPE EQUATIONS DERIVED FROM
C                  ALPHA - NU RELATION
```

```
C
      COMMON/CONST/ PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      COMMON/CWIND/ UP(2),IUX,IUY,U
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      DIMENSION PAR(5)
    4 FETCH=XA*G/(U*U)
      PAR(1)=2.898*FETCH**(-0.3718)*G/U
      PAR(2)=0.0644*FETCH**(-0.1982)
      PAR(3)=3.3
      PAR(4)=0.07
      PAR(5)=0.09
C
C                     SET UPPER LIMIT ON FM AND LOWER LIMIT ON ALPHA
C
      IF(PAR(1).GT.0.5) PAR(1)=0.5
      IF(PAR(2).LT.0.003) PAR(2)=0.003
      CALL MBESE (KK,PAR,IVH)
      RETURN
      END
C   ******************************************************************
C   ******************************************************************
C
      FUNCTION S2 (PAR,I)
C
C                     FUNCTION CALCULATES THE VALUE OF THE SOURCE TERM
C                     FOR THE I TH PARAMETER (FM, ALPHA, ETC.)
C
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CWIND/ UP(2),IUX,IUY,U
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      GOTO (1,2,3,4,5),I
    1 H = 0.
C
C                     FM SOURCE TERM ZERO IF GAMMA = 1
C
      IF ( PAR(3).GT.1.) H = -0.58613      *(PAR(1)*PAR(2))**2*(PAR(3)
     &-1.1/2.3
      S2 = H
      GOTO 100
    2 H = PAR(1)*U/G
      HF = H
      IF (H.LE.0.14) HF = 0.14
C
C                     ALPHA - NU RELATION MODELLED HERE
C                       SEE REPORT FOR DETAILS
C
      S2= (0.00512     *HF **(1.314)-5.*PAR(2)**2)*PAR(1)
     &*PAR(2)
      GOTO 100
    3 H = 3.2787
      HF = PAR(1) *U/G
C
C                     GAMMA -> 1 BETWEEN 0.16 AND 0.14
C
C
      IF( HF.LT.0.16) H = -14.951     +113.94       *HF
      IF( HF.LT.0.14) H = 1.
      S2  = -16.*PAR(1)*PAR(2)**2*(PAR(3)-H)
      GOTO 100
    4 H =(4./(PAR(3)+0.7))**2
      S2  = -0.5*PAR(1)*PAR(2)**2*(25.*(PAR(4)+PAR(5)-0.16373   *H)
     &+26.*(PAR(4)-PAR(5)+0.019870 *H))
      GOTO 100
    5 H =(4./(PAR(3)+0.7))**2
      S2  = -0.5*PAR(1)*PAR(2)**2*(25.*(PAR(4)+PAR(5)-0.16373   *H)
     &-26.*(PAR(4)-PAR(5)+0.019870 *H))
  100 CONTINUE
      RETURN
      END
C   ******************************************************************
C   ******************************************************************
C
      SUBROUTINE WSTOSW
C
C                                 SETS FLAG ISFC = FREQ BIN NUMBER FOR TRANSFER
C                                 W.S TO SWELL
C                                 ADJUSTS FM PARAMETER TO CONSERVE ENERGY
C                                 DETERMINES THE DIRECTION BIN(IDBIN)OF W.S DIR
C
      COMMON/CMSP/A(3500),B(3500),C(3500),WI(1300)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FMIN,FMMAX,NQUAD,DBIN
     &                ,FINIT,ISFC(650)
      COMMON/CINT/XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON/CONST/ PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      COMMON/WSTSWL/DEF(650),DEFD(650),IDBIN(650),FW(650),BETA(650)
     &                ,FC(650)
      COMMON/PREV/ WP(1300)
      COMMON /PM/ AFPM(650)
          CPM = G*0.1396
          CW=G*0.5/PI
          E54=EXP(-1.25)
      DBIN2=-DBIN*0.5
C
      DO 100 K=1,KMAX
      IF(IFLAG(K))100,100,101
C
C                                 ACTIVE POINT CALCULATE PM FREQ AND FW
C
  101 ISFC(K)=0
          ISWELL(K) = 0
          U = WI(K+K-1)
          V = WI(K+K)
          W = SQRT(U*U+V*V)
      FW(K)=0
      IF(W.LE.0.00001) W = 0.00001
          FPM = CPM/W
      AFPM(K)=FPM
          FWIND=CW/W
```

```
         FW(K)=FWIND
         KK = (K-1)*5+1
         FM = B(KK)
      ALPHA=B(KK+1)
C
C                         ** CRITERION FOR W.S TO SWELL **
C
      IF(FPM.GT.FMMAX) FPM=FMMAX
      IF(FM.GE.FPM) GO TO 25
C
C                   CALC CUT-OFF FREC
C
      X1=FM/FPM
      X1=X1*X1*X1*X1
      X3=-1.25/ALOG(1.0-X1)
      X3=X3**0.25
      FC(K)=FM*X3
C
C                   TRANSFORM TO BIN NUMBER
C
      IFC=(FC(K)-FINIT)/FBIN+1
      IF(IFC.LE.0) IFC=1
C
C                         ADJUST FM
C
      A(KK)=FPM
      C(KK)=FPM
C
C                   STORE BIN NUMBER
C
         ISFC(K) = IFC
C
C                   UPDATE COUNT OF TRANSFERS
C
         KSWELL = KSWELL+1
C
C                   ACCORDING TO PREVIOUS WIND
C                         DETERMINE THE DIRECTION BIN
C
 25   U=WP(K+K-1)
      V=WP(K+K)
      IF (U.NE.0.0.OR.V.NE.0.0) GOTO 50
C
C                         PREVIOUS WIND ZERO - ARBITRARY
C
      U=1.0
 50      THETA = ATAN2(V,U)
      IF(THETA.LT.DBIN2) THETA = THETA+PI+PI
         IDBIN(K) = (THETA+1.5*DBIN)/DBIN
 100  CONTINUE
      RETURN
      END
C  ********************************************************
C  ********************************************************
C
      SUBROUTINE FREBIN(IBIN)
C
C                         CALCULATES THE W-S ENERGY IN FREQ. BIN IBIN(DEF)
C                   IF ENERGY IS TO BE DUMPED, ELSE
C                         TESTS FOR SWELL TO W-S TRANSFER
C
      COMMON/CMSP/A(3500),B(3500),C(3500),W1(1300)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FBIN,FMMAX,NQUAD,DBIN
     C                  ,FINIT,ISFC(650)
      COMMON/CONST/ PI,G,FH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      COMMON/WSTSWL/DEF(650),DEFD(650),IDBIN(650),FW(650),BETA(650)
     C                  ,FC(650)
      COMMON/CINT/XA,WX,WY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/GRID/ KARRAY(650),IFLAG(650)
C
C                   SET POINTERS TO TOP AND BOTTOM OF FREQ BIN
C
         F2 = IBIN*FBIN+FINIT
         F1 = F2-FBIN
      IF(IBIN.EQ.1) F1=0.
      DO 100 K=1,KMAX
      IF(IFLAG(K)) 100,100,101
 101     DEF(K) = 0.0
         KK = (K-1)*5+1
         FM = B(KK)
C
C                   TEST IF WS -> SW OCCURING AT ALL
C
      IF(ISFC(K)) 50,50,51
C
C                   TEST IF OCCURRING AT THIS FREQ YET
C
C
C
C                   SET MARKER
C
 51   IF(IBIN.GT.ISFC(K)) GOTO 100
C
C                         ENERGY IS TO BE TRANSFERRED FROM W-S TO SWELL
C                         CALCULATE THE ENERGY IN FREQUENCY INTERVAL IBIN
C
 5    ISWELL(K)=1
         X = FM/F2
      IF(F2.GT.FC(K)) X=FM/FC(K)
      IF(IBIN.EQ.NFSWL)X=FM/FC(K)
         E2 = EXP(-1.25*X*X*X*X)
      IF(F1.LE.0.0) GO TO 10
         X = FM/F1
         E1 = EXP(-1.25*X*X*X*X)
C
C                   E2-E1 GIVES ENERGY IN BIN F1-*F2
C
      GO TO 20
 10      E1 = 0.0
 20      ALPHA = B(KK+1)
         DEF(K) = ALPHA*ECONST*(E2-E1)/(FM*FM*FM*FM)
```

```
C                    ADJUST FOR NON-UNIT GAMMA
C
      GAMMA = B(KK+2)
      DEF(K) = DEF(K)*(.774+.226*GAMMA)
      GO TO 100
   50 CONTINUE
      ISWELL(K)=0
C
C                                 IS ENERGY TO BE TRANSFERRED FROM SWELL TO WS?
C
C
C                      ADJUST FREQ BIN POINTER IF TOP BIN
C
      FM=A(KK)
      IF(IBIN.EQ.NFSWL) F2=(FMMAX+F2)*0.5
C
C                        SET MARKER ISWELL IF OCCURS
C
      IF((F2-0.5*FBIN).GT.(0.9*FM))ISWELL(K) = -1
  100 CONTINUE
      RETURN
      END
C  ************************************************************
C  ************************************************************
C
      SUBROUTINE DIRBIN(IF,ID)
C
C                                 CALCULATES THE W-S ENERGY IN DIR BIN IBIN
C                                 AND SETS SWELL SOURCE FUNCTION BETA
C
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON/ANGDIS/FRACT(10),AVECOS(10)
      COMMON/WSTSWL/DEF(650),DEFD(650),IDBIN(650),FW(650),BETA(650)
     C               ,FC(650)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FBIN,FMMAX,NQUAD,DBIN
     C               ,FINIT,ISFC(650)
      COMMON/CINT/XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      DATA C/0.000396/
         N2QUAD = NQUAD+NQUAD
         N4QUAD = N2QUAD+N2QUAD
C
C                        CALC FREQ AT CENTRE OF BIN
C
      F=IF*FBIN-0.5*FBIN+FINIT
      DO 100 K=1,KMAX
      IF(IFLAG(K)) 100,100,101
 101  BETA(K)=0
      DEFD(K)=0.0
C
C                                 IS IBIN WITHIN ANGDIS?
C
         IDIFF = IABS(IDBIN(K)-ID)
      IF(IDIFF.GT.N2QUAD) IDIFF = N4QUAD-IDIFF
      IF(IDIFF.GT.NQUAD) GOTO 100
      IF (ISWELL(K)) 100,103,50
C
C                      NO WS <-> SW EXCHANGE,SO
C                       CHECK FOR GROWING SWELL
C
 103  FWIND=FW(K)
      IF(FWIND) 100,100,102
 102  BETA(K)=C*F*(F*AVECOS(IDIFF+1)/FWIND-1.0)
      IF(BETA(K).LT.0.0) BETA(K)=0.0
      GOTO 100
C
C                                SET DEFD FOR WS TO SWELL
C
 50   DEFD(K)=FRACT(IDIFF+1)*DEF(K)
  100 CONTINUE
      RETURN
      END
C  ************************************************************
C  ************************************************************
C
      SUBROUTINE SWELL(IFT,DT,KSWIN)
C
C                                 PROPAGATES SWELL FIELD ONE TIMESTEP
C                                 TRANSFERS WS ENERGY DEFD TO SWELL
C                                 ABSORBS SWELL ENERGY INTO DE.
C
C                     RAY SCHEME FOR PROPAGATION OF SWELL
C
      COMMON/CMSP/A(3500),B(3500),C(3500),WI(1300)
      COMMON/SWLFLD/EC(4000),KINFT(50),MXY,LNEAR(650)
      COMMON/WSTSWL/DEF(650),DEFD(650),IDBIN(650),FW(650),BETA(650)
     C               ,FC(650)
      COMMON/SWLFLG/ISWELL(650),KSWELL,NFSWL,FBIN,FMMAX,NQUAD,DBIN
     C               ,FINIT,ISFC(650)
      COMMON/GRID/ KARRAY(650),IFLAG(650)
      COMMON/SWLTWS/DE(650)
      COMMON/CINT/XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CONST/ PI,G,FH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      COMMON /PM/ AFPM(650)
      DIMENSION D(650),LL(650),KK(650),XX(650),YY(650)
      DIMENSION LRAY(2)
C
C                     GET POINTER TO INFORMATION ABOUT RAY
C
      KRIN = KINFT(IFT)
      READ (10'KRIN) (LRAY(KI),KI=1,2),(LNEAR(KJ),KJ=1,MXY)
      LRAYL = LRAY(1)
      NSEG = LRAY(2)
      KRIN = KRIN+1
      READ (10'KRIN) (LL(KL),KL=1,NSEG),(KK(KM),KM=1,NSEG)
      KRIN = KRIN+1
      READ (10'KRIN) (XX(KN),KN=1,NSEG),(YY(KI),KI=1,NSEG)
C
C                     READ IN RAY ENERGY - PROPAGATED ONE SPACE
C
```

```
      KOUTE = KSWIN
      KOT = LRAY1/2401+1
      KFIN = 0
      IF (KOT.EQ.1) GO TO 135
      DO 130 KIF=2,KOT
      KST = KFIN+1
      KFIN = KST+2399
  130 READ (11'KSWIN) (EC(KE),KE=KST,KFIN)
  135 KST = KFIN+1
      KFIN = LRAY1
      READ (11'KSWIN) (EC(KE),KE=KST,KFIN)
      DE2 = 0.0
      EC(1) = 0.0
      IF (LRAY1.LT.2) GO TO 550
C
C                    EXCHANGE WIND - SEA AND SWELL
C
      DO 300 K=1,MXY
      K2 = KARRAY(K)
      IF(K2)300,300,301
C
C                    FIND NEAREST RAY POINT - SKIP IF THERE IS NOT
C
 301  D(K2)=0.0
      IF(IFLAG(K2)) 300,300,302
 302  L = LNEAR(K)
      IF(L)300,300,303
 303  IF(ISWELL(K2))251,260,250
C
C                    ABSORB SWELL
C
C
C                    CALC MAX E THAT CAN BE ABSORBED WITHOUT WS BECOMING
C                     OVERDEVELOPED
C
 251  KP=(K2-1)*5+1
      FM=A(KP)
      ALPHA=A(KP+1)
      GAMMA=A(KP+2)
      FM4=FM*FM*FM*FM
      FPM=AFPM(K2)
      FPM4=FPM*FPM*FPM*FPM
      EMAX=ALPHA*ECONST*(1.0/FPM4-1.0/FM4)*(.774+.226*GAMMA)
      IF(EMAX.LT.0.0) EMAX=0.0
      EA=EC(L)
      IF(EA.GT.EMAX) EA=EMAX
         DE(K2) = DE(K2)+EA
C
C                    ABSORB BY INJECTING NEGATIVE AMOUNT
C
      D(K2) = -EA
      GO TO 300
C
C                    INJECT W-S
C
  250 D(K2) = DEFD(K2)
      GO TO 300
C
C                    APPLY SWELL GROWTH
C
  260 D(K2) = BETA(K2)*DT*EC(L)
  300 CONTINUE
C
C                                  INJECT ENERGY AND INTERPOLATE
C
      ISEG = 1
   75 CONTINUE
      L1 = LL(ISEG)
      K1 = KK(ISEG)
      IF(K1)400,350,77
   77 X2 = XX(ISEG+1)
      Y2 = YY(ISEG+1)
      L2 = LL(ISEG+1)
      K2 = KK(ISEG+1)
      IF(K2)81,81,80
  81  L2 = L2-1
      K2 = K1
   80 CONTINUE
C
C                    INTERPOLATE DE AT ENDS OF SEGMENT
C
      IF(L2-L1) 110,110,111
 111  DE1 = DE2
      DE2 = (1.0-X2)*((1.0-Y2)*D(KARRAY(K2))+Y2*D(KARRAY(K2+MX)))
     C        +X2*((1.0-Y2)*D(KARRAY(K2+1))+Y2*D(KARRAY(K2+MX+1)))
C
C                    INTERPOLATE ALONG SEGMENT
C
      DEL = (DE2-DE1)/(L2-L1)
      DEL12 = DE1
      L1P1 = L1+1
      DO 100 L12 = L1P1,L2
      DEL12 = DEL12+DEL
      EC(L12) = EC(L12)+DEL12
 100  IF(EC(L12).LT.0.0) EC(L12) = 0.0
  110 ISEG = ISEG+1
      GO TO 75
  350 CONTINUE
C
C                    START OF RAY - SET E TO ZERO
C
      ISEG = ISEG+1
      L1 = LL(ISEG)
      DE2 = 0.0
      EC(L1) = 0.0
      GO TO 75
  400 CONTINUE
C
C                    END OF RAY - PROPAGATE AND OUTPUT TO EO
C
```

```
      MRAY = LRAY1-1
      DO 500 JLR = 1,MRAY
      JR=LRAY1-JLR+1
      EC(JR) = EC(JR-1)
  500 CONTINUE
  550 KFIN = 0
      IF (KOT.EQ.1) GO TO 650
      DO 600 KIF=2,KOT
      KST = KFIN+1
      KFIN = KST+2399
      WRITE (11*KOUTE) (EC(KF),KF=KST,KFIN)
C      WRITE(3,91)(EC(KF),KF=KST,KFIN)
C 91   FORMAT(' ','EC=',9F8.3)
  600 KOUTE = KOUTE+1
  650 KST = KFIN+1
      KFIN = LRAY1
      WRITE (11*KOUTE) (EC(KF),KF=KST,KFIN)
      RETURN
C      DEBUG UNIT(3),SUBCHK
      END
C   ***********************************************************
C   ***********************************************************
      SUBROUTINE SWTCHS
C
C                     SUBROUTINE TO ABSORB SWELL ENERGY DE INTO WS
C                     ADJUSTS FM TO COMPENSATE
C
      COMMON /GRID/ KARRAY(650),IFLAG(650)
      COMMON/CMSP/A(3500),B(3500),C(3500),WI(1300)
      COMMON/SWLTWS/DE(650)
      COMMON/CINT/XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CONST/ PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DO 100 K=1,KMAX
      IF(IFLAG(K)) 100,100,101
C
C                     CHECK IF ANY ENERGY TO ABSORB
C
  101 IF(DE(K))100,100,102
  102     KK = (K-1)*5+1
C
C                    ADJUST FOR NON-UNIT GAMMA
C
      GAMMA = A(KK+2)
      DE(K)=DE(K)/(.774 + .226*GAMMA)
          FM = A(KK)
      ALPHA = A(KK+1)
      X=FM*FM*FM*FM
      A(KK)=FM*(1.0+DE(K)*X/ECONST/ALPHA)**(-0.25)
      C(KK)=A(KK)
          DE(K) = 0.0
  100 CONTINUE
      RETURN
      END
      FUNCTION COP2 (PAR,I,K)
C
C                  CALLS RELEVANT FUNCTION TO CALC COUPLING COEFFS
C
      DIMENSION PAR(1)
      GOTO (1,2,3,4,5),I
    1 GOTO (11,12,13,60,60),K
    2 GOTO (21,22,23,60,60),K
    3 GOTO (31,32,33,60,60),K
    4 GOTO (41,42,43,44,60),K
    5 GOTO (51,52,53,60,55),K
   11 COP2=D11(PAR)
      GOTO 100
   12 COP2= D12(PAR)
      GOTO 100
   13 COP2= D13(PAR)
      GOTO 100
   21 COP2= D21 (PAR)
      GOTO 100
   22 COP2= D22(PAR)
      GOTO 100
   23 COP2= D23(PAR)
      GOTO 100
   31 COP2= D31(PAR)
      GOTO 100
   32 COP2= D32(PAR)
      GOTO 100
   33 COP2=D33(PAR)
      GOTO 100
   41 COP2= D41(PAR)
      GOTO 100
   42 COP2= D42(PAR)
      GOTO 100
   43 COP2= D43(PAR)
      GOTO 100
   44 COP2= D44(PAR)
      GOTO 100
   51 COP2=D51(PAR)
      GOTO 100
   52 COP2= D52(PAR)
      GOTO 100
   53 COP2= D53(PAR)
      GOTO 100
   55 COP2= D55(PAR)
      GOTO 100
   60 COP2 = 0.
  100 CONTINUE
      RETURN
      END
C   ***********************************************************
C   ***********************************************************
C
      FUNCTION D11C(PAR)
C
C                     FUNCTION COMPUTES THE CORRECTION TERM OF D11
C
      DIMENSION PAR(1)
```

```
      SIG2 = 0.25*(PAR(4)*PAR(5))**2
      D11C = -SIG2/(20.*SIG2+ALOG(PAR(3)))
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D11 (PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D11
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D11 = G/(4.*PI*PAR(1))*(1.0 - 5.0*D11C(PAR))
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D12(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D12
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D12 = G/(4.*PI*PAR(2))*D11C(PAR)
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D13(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D13
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D13 = G/(4.*PI*PAR(3))*D11C(PAR)
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D21(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D21
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D21 = -5.*PAR(2)*FA4*(G/(4*PI*PAR(1)*FA)-D11(PAR))/PAR(1)
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D22(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D22
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D22 = G/(4.*PI*FA*PAR(1))
     C      +D12(PAR)*5.0*PAR(2)*FA4/PAR(1)
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D23(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D23
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D23 = D13(PAR)*5.*PAR(2)*FA4/PAR(1)
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D31(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D31
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      H = 5*D11C(PAR)
      D31 = 5.*PAR(3)/PAR(1)**2*G/(4.*PI)*(FA4/FA-(1.-H)*FA4-H)
      RETURN
      END
C     ***********************************************************
C     ***********************************************************
C
      FUNCTION D32(PAR)
C
C                     FUNCTION COMPUTES THE COUPLING COEFFICIENT D32
C                     OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D32 = PAR(3)/PAR(2)*G/(4.*PI*PAR(1))*(1.+5.*D11C(PAR)
     &*(1.-FA4)-1./FA)
      RETURN
```

```
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D33(PAR)
C
C                       FUNCTION COMPUTES THE COUPLING COEFFICIENT D33
C                       OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D33 = G/(4.*PI*PAR(1))*(1.+5.*D11(PAR)*(1.-FA4))
      RETURN
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D41(PAR)
C
C                       FUNCTION COMPUTES THE COUPLING COEFFICIENT D41
C                       OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D41 = 0.
      IF (PAR(3).LT.1.5 ) GOTO 10
      HA = 1.-PAR(4)
      FAK = PAR(4)*EH/ALOG(PAR(3))
      D41 = HA/PAR(1)*D11(PAR)+FAK*(-G*(5.*HA**(-4)+HA/FAK)
     E       /(4.*PI*HA*PAR(1)**2)
     E       +5.*HA**(-4)/PAR(1)*D11(PAR)-D21(PAR)/PAR(2)
     E       -D31(PAR)/(EH*PAR(3)))
   10 CONTINUE
      RETURN
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D42(PAR)
C
C                       FUNCTION COMPUTES THE COUPLING COEFFICIENT D42
C                       OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D42 = 0.
      IF (PAR(3).LT.1.5 ) GOTO 10
      HA = 1.-PAR(4)
      D42 = HA/PAR(1)*D12(PAR)+PAR(4)*EH/ALOG(PAR(3))*
     E       (G/(4.*PI*HA*PAR(1)*PAR(2))+5.*HA**(-4)/PAR(1)*D12(PAR)
     E       -D22(PAR)/PAR(2)-D32(PAR)/(EH*PAR(3)))
   10 CONTINUE
      RETURN
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D43(PAR)
C
C                       FUNCTION COMPUTES THE COUPLING COEFFICIENT D43
C                       OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D43 = 0.
      IF (PAR(3).LT.1.5 ) GOTO 10
      HA = 1.-PAR(4)
      D43 = HA/PAR(1)*D13(PAR)+PAR(4)*EH/ALOG(PAR(3))*
     E       (G/(4.*PI*HA*PAR(1)*PAR(3)*EH)+5.*HA**(-4)/PAR(1)*D13(PAR)
     A-D23(PAR)/PAR(2)-D33(PAR)/(EH*PAR(3)))
   10 CONTINUE
      RETURN
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D44(PAR)
C
C                       FUNCTION COMPUTES THE COUPLING COEFFICIENT D44
C                       OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D44 = G/(4.*PI*(1.-PAR(4))*PAR(1))
      RETURN
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D51(PAR)
C
C                       FUNCTION COMPUTES THE COUPLING COEFFICIENT D51
C                       OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D51 = 0.
      IF (PAR(3).LT.1.5 ) GOTO 10
      HB=1.+PAR(5)
      FAK=PAR(5)*EH/ALOG(PAR(3))
      D51=-HB/PAR(1)*D11(PAR)+FAK*(-G*(5.*HB**(-4)-HB/FAK)/(4.*PI*HB*PAR
     E(1)**2)+5.*HB**(-4)/PAR(1)*D11(PAR)-D21(PAR)/PAR(2)-D31(PAR)
     E/(EH*PAR(3)))
   10 CONTINUE
      RETURN
      END
C     *****************************************************
C     *****************************************************
C
      FUNCTION D52(PAR)
```

```
C
C                        FUNCTION COMPUTES THE COUPLING COEFFICIENT D52
C                        OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D52 = 0.
      IF (PAR(3).LT.1.5 ) GOTO 10
      HB=1.+PAR(5)
      D52=-HB/PAR(1)*D12(PAR)+PAR(5)*EH/ALOG(PAR(3))*
     C(G/(4.*PI*HB*PAR(1)*PAR(2))+5.*HB**(-4)/PAR(1)*D12(PAR)
     C-D22(PAR)/PAR(2)-D32(PAR)/(EH*PAR(3)))
   10 CONTINUE
      RETURN
      END
C    **************************************************
C    **************************************************
C
      FUNCTION D53(PAR)
C
C                        FUNCTION COMPUTES THE COUPLING COEFFICIENT D53
C                        OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D53 = 0.
      IF (PAR(3).LT.1.5 ) GOTO 10
      HB = 1.+PAR(5)
      D53 = -HB/PAR(1)*D13(PAR)+PAR(5)*EH/ALOG(PAR(3))*
     C         (G/(4.*PI*HB*PAR(1)*PAR(3)*EH) +5.*HB**(-4)/PAR(1)*D13(PAR)
     A-D23(PAR)/PAR(2)-D33(PAR)/(EH*PAR(3)))
   10 CONTINUE
      RETURN
      END
C    **************************************************
C    **************************************************
C
      FUNCTION D55(PAR)
C
C                        FUNCTION COMPUTES THE COUPLING COEFFICIENT D55
C                        OF THE JONSWAP WAVE MODEL
C
      COMMON /CONST/PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      D55 = G/(4.*PI*(1.+PAR(5))*PAR(1))
      RETURN
      END
C    **************************************************
C    **************************************************
C
      FUNCTION ISIG(X)
C
C                        FUNCTION RETURNS SIGN OF PARAMETER
C
      ISIG = 0
      IF(X.LT.-0.00000001) ISIG =-1
      IF(X.GT.0.00000001) ISIG = 1
      RETURN
      END
C    **************************************************
C    **************************************************
C
      SUBROUTINE MHOL (K,PAR,IV)
C
C                        SUBROUTINE GETS DATA FROM STORAGE ARRAYS AND
C                        PUTS INTO PARAMETER ARRAY PAR
C
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      COMMON/CMSP/A(3500),B(3500),C(3500),WI(1300)
      COMMON /CONST/ PI,G,EH,FA,FA4,RROOT2,EPS,SLIM,ECONST
      DIMENSION PAR(1)
      I=5*(K-1)
      GOTO (1,2,100,4),IV
    1 DO 10 J = 1,5
   10 PAR(J) = A(I+J)
      GOTO 100
    2 DO 20 J = 1,5
   20 PAR(J) = B(I+J)
      GOTO 100
    4 PAR(1)=WI(K+K-1)
      PAR(2)=WI(K+K)
  100 CONTINUE
      RETURN
      END
C    **************************************************
C    **************************************************
C
      SUBROUTINE MBESE(K,PAR,IV)
C
C                        SUBROUTINE PUTS DATA INTO STORAGE ARRAYS FROM
C                        PARAMETER ARRAY PAR
C
      COMMON/CMSP/A(3500),B(3500),C(3500),WI(1300)
      COMMON/CINT/ XA,MX,MY,DELX,DELT,N,T,NN,KMAX,ISC
      DIMENSION PAR(1)
      I=5*(K-1)
      GOTO(1,2),IV
    1 DO 11 J = 1,N
   11 C(I+J) = PAR(J)
      GOTO 100
    2 DO 12 J = 1,N
   12 B(I+J) = PAR(J)
  100 CONTINUE
      RETURN
      END
C    **************************************************
C    **************************************************
```

'INTEG' LISTING

```
C       MASTER INTEGRATE      (VERSION 1  REVISION 2)
C
C       REVISION 2 PROVIDES AN IMPROVED HEADING.
C
C
C
C                 PROGRAM CALCULATES 2-WAY TABLES OF INTEGRALS I AND J
C                 FOR USE IN CALCULATING HS AND TZ FOR SELECTED VALUES
C                 OF GAMMA AND SIGMA
C                 (HS USES INTEGRAL I AND TZ USES BOTH INTEGRAL I AND J)
C                 INTEGRAL I IS OUTPUT ON CH 8, INTEGRAL J ON CH 9
C
      DIMENSION SIG(10),GAM(35),CINT1(10,35),CINT2(10,35)
      DIMENSION TITLE(20)
      REAL NATLOG(35)
      NAMELIST /IINT/ II,JJ,UL1,UL2,DY,DX
      NAMELIST /GS/ GAMS,GAMF,SIGS,SIGF
      DEFINE FILE 8 (12,35,U,KEOUT)
      DEFINE FILE 9 (12,35,U,KNOUT)
      READ (1,3000) TITLE
 3000 FORMAT (20A4)
      READ (1,IINT)
      READ (1,GS)
      GAMINT = (GAMF-GAMS)/(JJ-1)
      SIGINT = (SIGF-SIGS)/(II-1)
C
C                 II = NO. OF ROWS(SIGMAS),  JJ= NO. OF COLS(GAMMAS)
C                 UL1 = UPPER LIMIT FOR NUMERICAL INTEGRATION OF I
C                 UL2 = UPPER LIMIT FOR J
C                 DY = INTERVAL FOR CALCULATING INTEGRAL I
C                 DX = FOR J
C
      SUM1 = 0.0
      SUM2 = 0.0
      SIG(1) = SIGS
      GAM(1) = GAMS
      NATLOG(1) = ALOG(GAM(1))
C
C                 SET UP INTERMEDIATE GAMMA, SIGMAS
C
      DO 1 J=2,JJ
      GAM(J)=GAM(J-1)+GAMINT
    1 NATLOG(J) = ALOG(GAM(J))
      DO 11 I=2,II
   11 SIG(I)=SIG(I-1)+SIGINT
C
C                 WRITE OUT GAMMAS, SIGMAS USED TO FILES
C
      KECUT = 1
      KNCUT = 1
      WRITE (8'KEOUT) II,JJ,UL1,DY
      WRITE (9'KNOUT) II,JJ,UL2,DX
      WRITE (8'KEOUT) (SIG(I),I=1,II)
      WRITE (9'KNOUT) (GAM(J),J=1,JJ)
C                 CALCULATE INTEGRAL I FOR EACH GAMMA, SIGMA
C
      DO 2 J=1,II
      YY = (-1.0/SIG(J))+0.01
      YP = YY
      DO 3 K=1,JJ
    4 IF(YY.GT.UL1) GO TO 5
      C = 1.0+YY*SIG(J)
      A = EXP(-1.25*1.0/(C*C*C*C)-(YY*YY*0.5))
      B = EXP(NATLOG(K)*EXP(-0.5*YY*YY))
      SUM1 = SUM1+(A*YY*B)
      YY = YY+DY
      GO TO 4
    5 CINT1(J,K) = SUM1*NATLOG(K)*DY
      YY = YP
      SUM1 = 0.0
    3 CONTINUE
C
C                 WRITE COMPLETED ROW TO STORE
C
      WRITE(8'KEOUT) (CINT1(J,K),K=1,JJ)
    2 CONTINUE
C
C                 CALCULATE INTEGRAL J FOR EACH GAMMA, SIGMA
C
      DO 7 J=1,II
      XP = 0.1
      XX = XP
      DO 8 K=1,JJ
    9 IF(XX.GT.UL2) GO TO 10
      C2 = NATLOG(K)*EXP((-(XX-1.0)*(XX-1.0))/(2.0*SIG(J)*SIG(J)))
      X3 = 1/(XX*XX*XX)
      X4 = X3/XX
      B2 = X3*EXP(-1.25*X4+C2)
      SUM2 = SUM2+B2
      XX = XX+DX
      GO TO 9
   10 CINT2(J,K) = SUM2*DX
      XX = XP
      SUM2 = 0.0
    8 CONTINUE
C
C                 STORE ROW
C
      WRITE(9'KNOUT) (CINT2(J,K),K=1,JJ)
    7 CONTINUE
C
C                 WRITE VALUES USED TO LP
C
      WRITE (3,2009) TITLE
      WRITE(3,2007)
      WRITE(3,2000) (GAM(M),M=1,JJ)
      WRITE(3,2008)
      WRITE(3,2000) (SIG(M),M=1,II)
      WRITE(3,2005) II,JJ,UL1,UL2,DY,DX
```

```
C
C                    WRITE INTEGRAL I TO LP ROW BY ROW
C
      WRITE(3,2002)
      DO 6 L=1,II
      WRITE(3,2000) (CINT1(L,N),N=1,JJ)
    6 CONTINUE
C
C                    WRITE INTEGRAL J
C
      WRITE(3,2006)
      DO 16 L=1,II
      WRITE(3,2000) (CINT2(L,N),N=1,JJ)
 16   CONTINUE
C1010 FORMAT(4F6.3)
 2000 FORMAT(1H0,21F5.3)
 2002 FORMAT(1H1,10HINTEGRAL I)
 2005 FORMAT(1H0,I5,12H SIGMAS USED,I5,7H GAMMAS/1H ,
     1          28HUPPER LIMIT FOR INTEGRAL I =,F6.3/1H ,16X,
     2          12HINTEGRAL J =,F6.3/1H0,
     3          16HINTERVAL FOR I =,F6.3,5X,7HFOR J =,F6.3)
 2006 FORMAT(1H1,10HINTEGRAL J)
 2007 FORMAT(1H0,5HGAMMA)
 2008 FORMAT(1H0,5HSIGMA)
 2009 FORMAT('1',///52X,'NORSWAM  PROGRAM  INTEGRATE'///26X,20A4///)
      STOP
      END
```

'HSTZ' LISTING

```
C        MASTER HSTZ     (VERSION 1  REVISION 3)
C
C                  PROGRAM CALCULATES HS AND TZ(WIND-SEA ONLY), AND IF
C                  REQUIRED HS AND TZ (INCLUDING SWELL)
C
C            REVISION 2 PROVIDES FOR OUTPUT TO BE PRINTED SO THAT GRID IS
C            VISUALLY CORRECT,  THAT IS, WITH THE ORIGIN IN THE LOWER LEFT.
C            REVISION 3 CAUSES INITIALIZATION OF INTEGRAL TABLES AND
C            ENERGY ARRAYS TO 0.0 IN A BLOCK DATA.  DIMENSION
C            STATEMENTS WERE CHANGED TO COMMONS TO DO THIS.
C
C                  INPUT REQUIRED (ON UNIT 1):
C
C                          FIRST RECORD  :   TITLE  RECORD  (<81 CHARACTERS)
C                         NAMELIST HTPAR  :
C                         NGOES        NUMBER OF ITERATIONS
C                         NSKIP        NUM. OF INITIAL ITERS. SKIPPED
C                         IDO          SWELL ANALYSIS INDICATOR
C                                      (- IDO => SWL INCLUDED)
C                         NDIR         TOTAL NUM. OF SWL DIRECTIONS
C                         NFREQ        NUM. OF FREQ. PER DIRECTION
C
C
      COMMON /ITAB/ GAM(35),SIG(10),CINT1(10,35),CINT2(10,35)
      COMMON /ARR/ ARRAY(3250),EINT(1300)
      DIMENSION TITLE(20),IVAL(3)
      DIMENSION WH(50,650),TP(50,650),BWH(650),CTP(650)
      COMMON /EARS/ EC(650),ES(650),EINTS(1300),SIGMA(650),
     &          SUMDIR(650)
      COMMON/TIME/KTIME
      COMMON/GRID/KARRAY(650),MM,NN
      NAMELIST /HTPAR/ NGOES,NSKIP,IDO,NDIR,NFREQ,ISTNO
      DEFINE FILE 5 (25,1300,U,KAT)
      DEFINE FILE 8 (12,35,U,KEIN)
      DEFINE FILE 9 (12,35,U,KNIN)
      DEFINE FILE 12 (1600,660,U,KSIN)
      DEFINE FILE 13 (40,3500,U,KWIN)
      READ (1,89) TITLE
   89 FORMAT (20A4)
      READ (1,HTPAR)
      NDC=NDIR/4
      WRITE(3,99) TITLE,NGOES,NSKIP,NDO,NFREQ
   99 FORMAT ('1'//42X,'NORSWAM  PROGRAM  HSTZ  (VERSION 1  ',
     &        'REVISION 3)',////26X,20A4///51X,I4,' ITERATIONS ',
     &        'SKIPPING',I4//51X,I4,' DIRECTIONS PER QUADRANT'//51X,
     &        I4,' FREQUENCIES PER DIRECTION'/)
      IF(IDC)33,33,30
C
C                  IDO -VE ==HS AND TZ(+SWL) TO BE CALCULATED ALSO
C
   33 WRITE(3,93)
   93 FORMAT (' '/54X,'SWELL ANALYSIS REQUIRED'/)
 30   CONTINUE
C
C                  READ INTEGRAL TABLE INFORMATION(SET UP BY PROGRAM INTEGRATE)
C
      KEIN = 1
      KNIN = 1
      READ (8'KEIN) II,JJ,UL1,DY
      READ (9'KNIN) II,JJ,UL2,DX
      READ (8'KEIN) (SIG(I),I=1,II)
      READ (9'KNIN) (GAM(J),J=1,JJ)
      GAMS=GAM(1)
      GAMF=GAM(JJ)
      SIGS=SIG(I)
      SIGF=SIG(II)
      GAMINT = (GAMF-GAMS)/(JJ-1)
      SIGINT = (SIGF-SIGS)/(II-1)
      WRITE(3,2002) GAMS,GAMF,SIGS,SIGF
 2002 FORMAT (' '/39X,'STARTING GAMMA ',F6.3,10X,'FINISHING GAMMA ',
     &        F6.3//39X,'STARTING SIGMA ',F6.3,10X,'FINISHING SIGMA ',
     &        F6.3//)
C
C                  READ INTEGRAL TABLES
C
      DO 3 K=1,II
      READ (8'KEIN) (CINT1(K,L),L=1,JJ)
      READ (9'KNIN) (CINT2(K,L),L=1,JJ)
    3 CONTINUE
C
C                  READ GRID INFORMATION
C
      KWIN = 1
      READ(13'KWIN)KMAX,MM,NN,MN,(KARRAY(K),K=1,MN)
C
C                  AND WRITE SAME TO RESULTS FILES
C
C       IF(IDO)44,44,43
C   44 WRITE(7,1040)KMAX,MM,NN,MN,(KARRAY(K),K=1,MN)
 43   CONTINUE
      KMAX5 = KMAX*5
      KMAX2 = KMAX+KMAX
      IF(NSKIP)17,17,18
C
C                  SKIP SOME ITERATIONS IF REQUIRED
C
   18 KWIN = KWIN+NSKIP
   17 IF(IDO)32,32,31
   32 KSIN = 3
      READ (12'KSIN) (IVAL(KL),KL=1,3)
      KSST=IVAL(1)
C
C                  KSST GIVES ITERATION NUMBER CORRESPONDING TO FIRST SWELL
C                  INFORMATION
C
      KSIN = 3
 31   CONTINUE
      DO 20 KK=1,NGOES
      KTIME = NSKIP+KK
```

```
C                    READ WIND-SEA INFORMATION (FM,ALPHA,GAMMA,SIGA,SIGB)
C
      READ(13'KW(N)) (ARRAY(N),N=1,KMAX5)
      L = -5
      DO 10 I=1,KMAX2,2
      L = L+5
      SIGA = ARRAY(L+4)
      SIGB = ARRAY(L+5)
      SIGX = (SIGA+SIGB)/2.0
      IF(SIGX.GT.SIGF) SIGX=SIGF
      FM = ARRAY(L+1)
      ALPHA = ARRAY(L+2)
      GAMX = ARRAY(L+3)
C
C                    FIND ROW, COL CORRESPONDING TO INPUT GAMMA, SIGMA
C
      IF(GAMX.GT.GAMF) GAMX=GAMF
      IG = (GAMX-GAMS)/GAMINT
      IF(IG.GT.(JJ-1)) IG=JJ-1
      GAM0 = GAMS+IG*GAMINT
      GAM1 = GAM0+GAMINT
C
C                    GAM0, GAM1 ARE TABULATED GAMMAS EITHER SIDE OF INPUT GAMMA
C
      IS = (SIGX-SIGS)/SIGINT
      IF(IS.GT.(II-1)) IS=II-1
      SIG0 = SIGS+IS*SIGINT
      SIG1 = SIG0+SIGINT
C
C                    SIMILARLY FOR SIGMA
C
      GN = (GAMX-GAM0)/GAMINT
      SN = (SIGX-SIG0)/SIGINT
      J = IS+1
      K = IG+1
C
C                    INPUT GAMMA,SIGMA LIE IN CELL (J,K) IN INTEGRAL TABLE
C
C
C                     LINEARLY INTERPOLATE USING 4 CORNERS OF CELL TO GET
C                    VALUES OF INTEGRALS FOR INPUT GAMMA, SIGMA
C
      CINT10 = CINT1(J,K)+GN*(CINT1(J,K+1)-CINT1(J,K))
      CINT11 = CINT1(J+1,K)+GN*(CINT1(J+1,K+1)-CINT1(J+1,K))
      CINT12 = CINT10+SN*(CINT11-CINT10)
      CINT20 = CINT2(J,K)+GN*(CINT2(J,K+1)-CINT2(J,K))
      CINT21 = CINT2(J+1,K)+GN*(CINT2(J+1,K+1)-CINT2(J+1,K))
      CINT22 = CINT20+SN*(CINT21-CINT20)
C
C                    CALCULATE HS
C
      ENT = (1.0+CINT12)/5.0
      F = FM*FM*FM*FM
      E = ALPHA*(1.0/F)*0.01233941*(1.0+CINT12)
      EINT(I) = 4.0*SQRT(E)
C                    CALCULATE TZ
C
      EINT(I+1) = SQRT(ENT/CINT22)/FM
   10 CONTINUE
C
C                    TEST TO SEE IF SWELL ENERGY TO BE INCLUDED
C
      IF(IDC)64,64,76
C
C                    ANY SWELL ENERGY FOR THIS ITERATION?
C
   64 IF(KTIME-KSST)77,78,78
C
C                    TOP SWELL BIN FIRST AT 0.145 HZ, BIN SIZE =0.01 HZ
C
   78 F = 0.145
      DELF=0.01
      DO 80 K=1,KMAX
      ES(K)=0.0
 80   SIGMA(K)=0.0
C
C
C                    CALCULATE TOTAL SWELL ENERGY AND SECOND MOMENT
C
C
      DO 75 J=1,NFREQ
      DO 81 K=1,KMAX
 81   SUMDIR(K)=0.0
      DO 70 J2=1,NDIR
      READ (12*KSTN) (IVAL(KL),KL=1,3),(EC(KM),KM=1,KMAX)
      DO 82 K=1,KMAX
 82   SUMDIR(K)=SUMDIR(K)+EC(K)
   70 CONTINUE
      DO 83 K=1,KMAX
      SIGMA(K)=SIGMA(K)+F*F*SUMDIR(K)
 83   ES(K)=ES(K)+SUMDIR(K)
C
C                    SIGMA = SECOND MOMENT.        ES = TOTAL ENERGY
C
C
C                    ALTER FREQUENCY TO THAT OF NEXT BIN
C
      F=F-DELF
   75 CONTINUE
 77   CONTINUE
C
C                    CALCULATE HS AND TZ INCLUSIVE OF SWELL ENERGY
C
      DO 84 K=1,KMAX
      I=K+K-1
      HS=EINT(I)
      TZ=EINT(I+1)
      WH(KK,K)=HS
      TP(KK,K)=TZ
```

```
      EW=HS*HS/16.0
      EINTS(I)=4.0*SQRT(EW+ES(K))
84    EINTS(I+1)=SQRT((EW+ES(K))/ (EW/(TZ*TZ)+SIGMA(K)))
      CALL PRINT(EINTS,KMAX2,2,2,16)
76    CONTINUE
C
C                   WRITE TO LP AND FILES
C
      CALL PRINT(EINT,KMAX2,2,0,14)
   20 CONTINUE
      DO 851 K=1,KMAX
      IF (KARRAY(K).NE.0) GO TO 22
      BWH(K)=0.0
      CTP(K)=0.0
      KTIME=0.0
      GO TO 132
   22 BWH(K)=WH(1,K)
      CTP(K)=TP(1,K)
      KTIME=1
      DO 861 KK=2,NGOES
      IF (WH(KK,K).LE.BWH(K)) GO TO 861
      BWH(K)=WH(KK,K)
      CTP(K)=TP(KK,K)
      KTIME=KK
  861 CONTINUE
  132 WRITE(3,909) K,BWH(K),CTP(K),KTIME
  909 FORMAT (6X,'K=',I5,4X,'BWH(K)=',F8.3,4X,'CTP(K)=',
     $F8.3,4X,'TIME=',I4)
  851 CONTINUE
      KAT=ISTNO
      WRITE (5'KAT) (BWH(K),K=1,KMAX),(CTP(K),K=1,KMAX)
      STOP
      END
C   ****************************************************
C   ****************************************************
      BLOCK DATA
      COMMON /ITAB/ GAM(35),SIG(10),CINT1(10,35),CINT2(10,35)
      COMMON /ARR/ ARRAY(3250),EINT(1300)
      COMMON /EARS/ EC(650),ES(650),EINTS(1300),SIGMA(650),
     C              SUMDIR(650)
      DATA GAM,SIG/45*0.0/,CINT1,CINT2/700*0.0/
      DATA ARRAY/3250*0.0/,EINT,EINTS/2600*0.0/
      DATA EC,ES,SIGMA,SUMDIR/2600*0.0/
      END
C   ****************************************************
C   ****************************************************
      SUBROUTINE PRINT(A,M,NP,JI,IC)
      DIMENSION A(M),AP(20)
      DIMENSION MESS(8)
      COMMON/TIME/KTIME
      COMMON /GRID/ KARRAY(650),MX,MY
      DATA MESS /'  H1','/3  ','   T','Z   ',' HS+','SWL ',
     C    ' TZ+','SWL '/
      NU = 10
      MX=20

      MY=21
      NPAGE = (MX-1)/NU+1
      DO 100 II=1,2
      WRITE(3,600)KTIME
      JII = 2*(II+JI)-1
      DO 60 L=1,NPAGE
      WRITE(3,200) L,MESS(JII),MESS(JII+1)
      LL=NU
      IF(NU*L.GT.MX) LL=MX-(L-1)*NU
      LF=LL+(L-1)*NU
      LS=LF-LL+1
      WRITE (3,250)
      WRITE (3,270) (LI,LI=LS,LF)
      WRITE (3,280)
      DO 50 IJ=1,MY
      I=MY-IJ+1
      KST=(I-1)*MX+(L-1)*NU
      DO 70 J=1,LL
      K = KST+J
      AP(J) = 0.0
      KK = KARRAY(K)
      IF(KK.NE.0) AP(J) = A((KK-1)*NP+II)
   70 CONTINUE
      WRITE(3,300) I,(AP(J),J=1,LL)
   50 CONTINUE
   60 CONTINUE
      WRITE(3,400)
  100 CONTINUE
C     WRITE (IC,700) A
  200 FORMAT(//1H0,5X,4HPAGE,I3,38X,2A4)
  250 FORMAT (' ',59X,'X DIRECTION')
  270 FORMAT (' ','     Y ',8X,10(I2,9X))
  280 FORMAT (' ','     - ',8X,10('--',9X))
  300 FORMAT(' ',2X,I3,'.    ',10G11.4)
  400 FORMAT(///)
  600 FORMAT(1H0,10H TIMESTEP ,I4)
      RETURN
      END
```

'WEIBULL' LISTING

```
      DEFINE FILE 5 (25,1300,U,KAT)
      DIMENSION H(50),A(50)
      DIMENSION BWH(25,650),CTP(25,650)
      COMMON/FD/ACAP,BCAP
      COMMON/C/EHMAX(50),X(50)
      COMMON/B/YR(50)
      DIMENSION DIFF(20),RR(20),RRR(40),DI(40)
      COMMON/AP/NPTS
      COMMON/D/RV
      REAL K(40),K1,LAMBDA,KA(40)
      CALL ERRSET (209,256,0,0,1,209)
      TIME=32.
      NPTS=22
      LAMBDA=NPTS/TIME
      KMAX=420
      KAT=1
   99 KS=KAT
      READ (5*KAT)(BWH(KS,N),N=1,KMAX),(CTP(KS,N),N=1,KMAX)
      WRITE (3,58)KS
   58 FORMAT (8X,'STORM NO.=',I4)
      WRITE (3,59) (BWH(KS,N),N=1,KMAX)
   59 FORMAT (' ',11(F7.3,3X))
      IF (KAT.EQ.(NPTS+1)) GO TO 971
      GO TO 99
  971 DO 456 NA=1,KMAX
      WRITE (3,934)
  934 FORMAT (4X,'________________________________________')
      WRITE (3,964) NA
  964 FORMAT (10X,'POINT NO.',I4)
      WRITE (3,934)
      DO 10 KS=1,NPTS
      H(KS)=BWH(KS,NA)
   10 CONTINUE
      IF (NA.EQ.1) WRITE (3,31) (H(KS),KS=1,NPTS)
      CALL RANK (H,YR)
      WRITE (3,31) (YR(M),M=1,NPTS)
   31 FORMAT (5X,5F8.3)
      R=0.
      L=0
   26 L=L+1
   35 K(L)=0.25+L*0.25
      ST=K(L)
      CALL FINDR(ST,R)
      RR(L)=R
      DIFF(L)=1.-RR(L)
      WRITE(3,18)L,K(L),R,DIFF(L)
   18 FORMAT (5X,I5,3X,3(F9.3,2X))
      IF (L.EQ.1) GO TO 26
      IF ((RR(L)-RR(L-1)).LE.0.001) GO TO 33
      GO TO 26
   33 WRITE (3,43)
   43 FORMAT (5X,'FINER SEARCH STARTS')
      M=0
      IF (L.EQ.2) SP=0.25
      IF (L.GT.2) SP=K(L-2)
   47 M=M+1
      KA(M)=SP+0.05
      SP=KA(M)
      CALL FINDR (SP,R)
      RRR(M)=R
      DI(M)=1.0-R
      WRITE (3,18) M,KA(M),R,DI(M)
      IF (M.EQ.1) GO TO 47
      IF ((RRR(M)-RRR(M-1)).GT.0.001) GO TO 47
      WRITE (3,414)
  414 FORMAT (5X,'STILL FINER SEARCH')
      IF (M.EQ.2) K1=KA(M-1)
      IF (M.GT.2) K1=KA(M-2)
      WRITE (3,321) K1
      R=0.
   34 K1=K1+0.01
      AD=R
      WRITE (3,321) K1
  321 FORMAT (6X,'K1=',F7.3)
      CALL FINDR (K1,R)
      WRITE (3,145) K1,R
  145 FORMAT (5X,'K1=',F7.4,4X,'R=',F7.4)
      IF ((R-AD).GT.0.001) GO TO 34
      WRITE (3,12) K1,R
   12 FORMAT (10X,'K1=',F7.2,3X,'R=',F7.2)
      WRITE (3,202)
  202 FORMAT (6X,'RANK',3X,'RED.VAR.',3X,'HEIGHT',4X,'EHMAX',6X,
     C'RP')
      DO 40 M=1,NPTS
      RP=1.0/(LAMBDA*(1.0-EHMAX(M)))
      WRITE (3,222) M,X(M),YR(M),EHMAX(M),RP
  222 FORMAT (5X,I3,3X,4(F9.3,2X))
   40 CONTINUE
      WRITE (3,93) NA
   93 FORMAT (5X,'POINT NO.=',I3)
      DO 56 KL=1,2
      T=KL*50.
      EH50=1./(T*LAMBDA)
      RV50=(-ALOG(EH50))**(1./K1)
      W50=ACAP+(BCAP*RV50)
      WRITE (3,98) T, W50
   98 FORMAT (4X,F9.2,2X,'YEAR WAVE=',F7.3)
   56 CONTINUE
  456 CONTINUE
   41 STOP
      END
C *****************************************************************
      SUBROUTINE FINDR (S,R)
      COMMON/C/ EHMAX(50),X(50)
      COMMON/D/RV
      COMMON/B/YR(50)
      COMMON/AP/NPTS
      DO 30 M=1,NPTS
      ALPHA=0.3+0.18/S
      BETA=0.21+0.32/S
```

```
      EMMAX(M)=1.0-(M-ALPHA)/(NPTS+BETA)
      RV=(-ALOG(1.0-EMMAX(M)))**(1.75)
      X(M)=RV
   30 CONTINUE
      CALL CORR (X,YR,R)
      RETURN
      END
C ******************************************************
      SUBROUTINE RANK (A,Y)
      COMMON/AP/NPTS
      DIMENSION A(50),Y(50)
      DO 22 M=1,NPTS
      BMAX=A(1)
      K=1.
      DO 21 I=2,NPTS
      IF (A(I).LT.BMAX) GO TO 21
      BMAX=A(I)
      K=I
   21 CONTINUE
      Y(M)=BMAX
      WRITE (3,32) M,Y(M)
   32  FORMAT (6X,'M(',I3,')=',F7.2)
      A(K)=0.
   22 CONTINUE
      RETURN
      END
C ******************************************************C
      SUBROUTINE CORR(X,Y,R)
      DIMENSION X(50),Y(50),XCAP(50),YCAP(50)
      COMMON/FD/ACAP,BCAP
      COMMON/B/YR(50)
      COMMON/AP/NPTS
      XSUM=0.
      YSUM=0.
      DO 115 I=1,NPTS
      XSUM=XSUM+X(I)
      YSUM=YSUM+Y(I)
  115  CONTINUE
      XBAR=XSUM/NPTS
      YBAR=YSUM/NPTS
      B1=0.
      B2=0.
      DO 116 I=1,NPTS
      B1=B1+(X(I)-XBAR)*(Y(I)-YBAR)
      B2=B2+(X(I)-XBAR)**2
  116 CONTINUE
      BCAP=B1/B2
      ACAP=YBAR-BCAP*XBAR
      DO 117 I=1,NPTS
      YCAP(I)=ACAP+BCAP*X(I)
  117 CONTINUE
      R1=0.
      R2=0.
      DO 118 I=1,NPTS
      R1=R1+(YBAR-YCAP(I))**2

      R2=R2+(YBAR-Y(I))**2
  118 CONTINUE
      R=R1/R2
      RETURN
      END
```