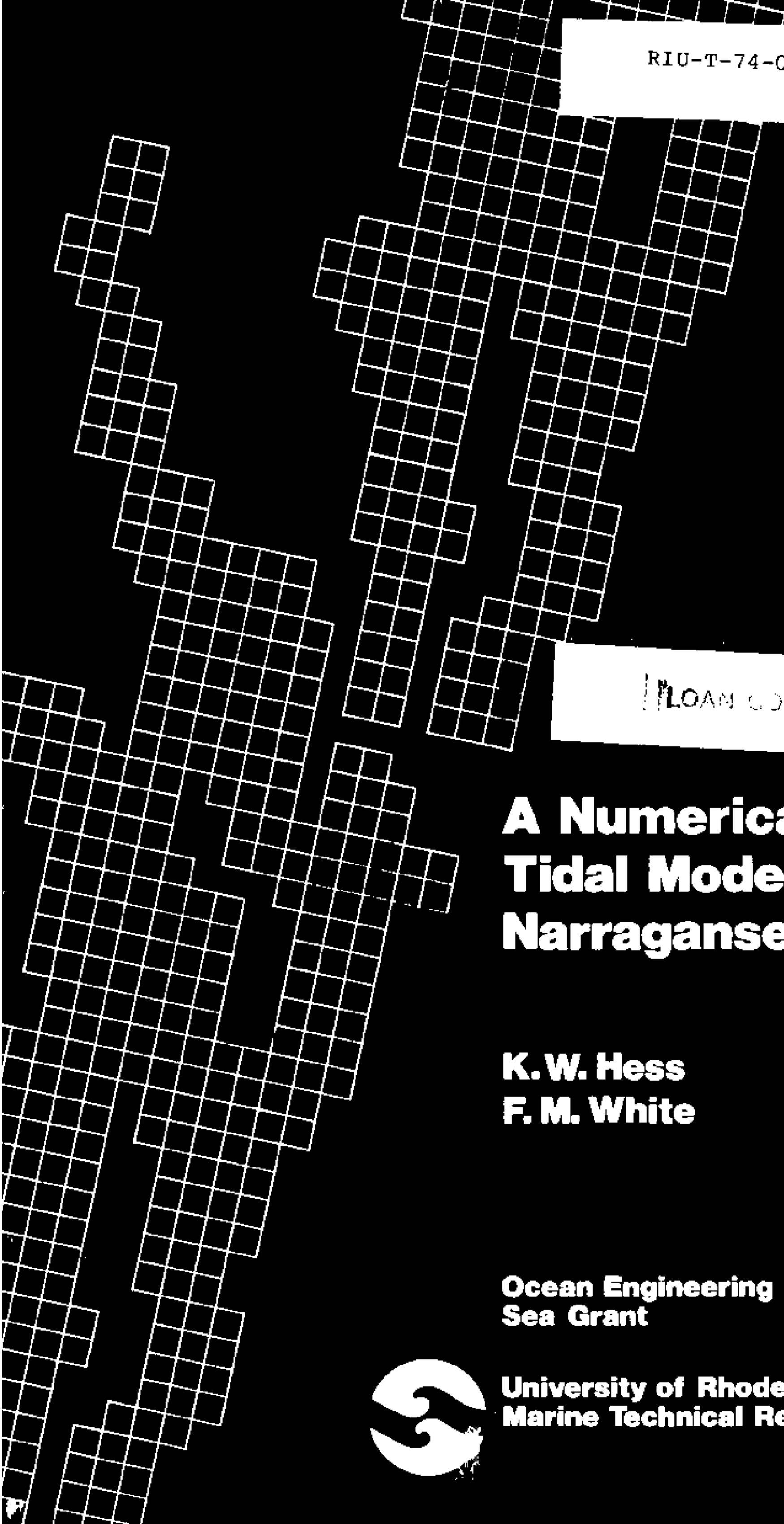


A Numerical Tidal Model of Narragansett Bay

K. W. Hess
F. M. White

Ocean Engineering
Sea Grant

University of Rhode Island
Marine Technical Report No. 20

A Numerical Tidal Model of Narragansett Bay

K. W. Hess
F. M. White

Ocean Engineering
Sea Grant

University of Rhode Island
Marine Technical Report No. 20
Kingston 1974

Additional copies of Marine Technical Report Number 20 are available from the Marine Advisory Service, University of Rhode Island, Narragansett Bay Campus, Narragansett, Rhode Island 02882.

CONTENTS

FIGURES

I. MATHEMATICS OF SOLUTION

1.0 INTRODUCTION

Over the past few years, several numerical tidal hydrodynamic models have been proposed for the study of estuarine behavior. Examples are the two-dimensional long-wave propagation models of Leendertse (1), Reid and Bodine (2), Masch and Brandes (3), and Mungall and Matthews (4). Pritchard (5) has given an excellent summary of the mathematical development of the vertically-averaged equations, and Sobey (6) has investigated the characteristics of several schemes.

The basic approach of Leendertse (1) was chosen for the development of the numerical model of Narragansett Bay (Fig. 1), which is a wide, shallow estuarine system dominated by tidal effects. The model had been successfully applied to a small harbor by Grimsrud (7) and has now been adapted to the Bay with several modifications. The following is an explanation of the mathematics of the solution as used in the Bay model, along with certain necessary modifications of the original approach.

The model was developed to provide information concerning the tidal dynamics of the Bay and the accompanying currents and flowrates. Verification studies have been carried out and reported here, and the model will be used as the basis of a concentration-transport model for the study of salinity, temperature, and biochemical parameters in the Bay.

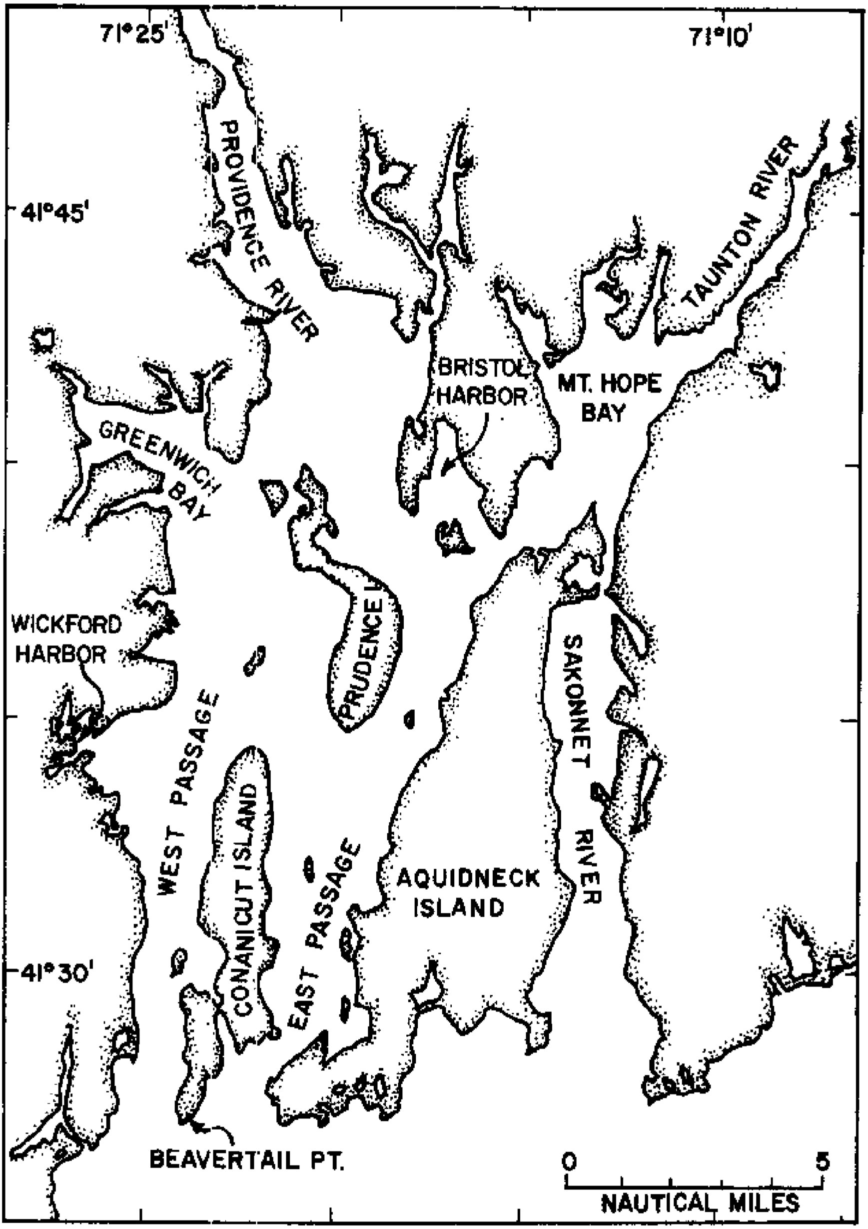


Fig. 1. Reference map of Narragansett Bay.

1.1 THE EQUATIONS OF MOTION

The basic equations for this model are the Navier-Stokes momentum equations, plus conservation of mass, In Eulerian form, with a right-handed coordinate system with the z-axis directed upward (Fig. 2a), the three momentum equations are

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + v\,\frac{\partial u}{\partial y} + w\,\frac{\partial u}{\partial z} = -\,\frac{1}{\rho}\,\frac{\partial p}{\partial x} + f\,v + \qquad 1.1.1$$

$$+\,\frac{1}{\rho}\,\left(\frac{\partial \tau}{\partial x}xx + \frac{\partial \tau}{\partial y}xy + \frac{\partial \tau}{\partial z}xz\right)$$

$$\frac{\partial v}{\partial t} + u\,\frac{\partial v}{\partial x} + v\,\frac{\partial v}{\partial y} + w\,\frac{\partial v}{\partial z} = -\,\frac{1}{\rho}\,\frac{\partial p}{\partial y} - f\,u + \qquad 1.1.2$$

$$+\,\frac{1}{\rho}\,\left(\frac{\partial \tau}{\partial x}yx + \frac{\partial \tau}{\partial y}yy + \frac{\partial \tau}{\partial z}yz\right)$$

$$\frac{\partial w}{\partial t} + u\,\frac{\partial w}{\partial x} + v\,\frac{\partial w}{\partial y} + w\,\frac{\partial w}{\partial z} = -\,\frac{1}{\rho}\,\frac{\partial p}{\partial z} - g + \qquad 1.1.3$$

$$+\,\frac{1}{\rho}\,\left(\frac{\partial \tau}{\partial x}zx + \frac{\partial \tau}{\partial y}zy + \frac{\partial \tau}{\partial z}zz\right)$$

The conservation of mass equation, assuming incompressible flow, is

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \qquad 1.1.4$$

where

$u(x,y,z,t)$ = velocity in the x-direction

$v(x,y.z,t)$ = velocity in the y-direction

$$U = \frac{1}{(h + \eta)} \int_{-h}^{\eta} u dz \qquad 1.1.9$$

$$V = \frac{1}{(h + \eta)} \int_{-h}^{\eta} v dz \qquad 1.1.10$$

so that the horizontal velocities may be expressed as

$$u = U \left[1 + \epsilon_u(z)\right] \qquad 1.1.11$$

$$v = V \left[1 + \epsilon_v(z)\right] \qquad 1.1.12$$

Assuming constant atmospheric pressure, small horizontal stress and that

$$\epsilon_u, \ \epsilon_v << 1$$

integration over z from the bottom, -h to the surface, η, of the horizontal momentum equations, along with 1.1.7 and 1.1.8, gives

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} = - g \frac{\partial \eta}{\partial x} + fV + \frac{1}{\rho(h + \eta)} (\tau_{sx} - \tau_{bx}) \qquad 1.1.13$$

$$\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} = - g \frac{\partial \eta}{\partial y} - fU + \frac{1}{\rho(h + \eta)} (\tau_{sy} - \tau_{by}) \qquad 1.1.14$$

In these equations, τ_{bi} and τ_{si} represent the bottom and surface stresses, respectively, and a number of terms arising from the application of the Leibnitz rule, shown to be small by Grimsrud (7), have been neglected. Similar integration of

$w(x,y,z,t)$	=	velocity in the z-direction
$p(x,y,z,t)$	=	pressure
$\rho(x,y,z)$	=	density of water
f	=	Coriolis parameter, $2\Omega \sin \varphi$
τ_{ij}	=	shear stress tensor
g	=	gravitational acceleration

If we make the Boussinesq assumption that vertical variations in pressure are predominantly the result of variations in depth, the z-momentum equation (1.1.3) reduces to the hydrostatic equation

$$\frac{\partial p}{\partial z} = - \rho g \qquad 1.1.5$$

Assuming uniform density and integrating from the water surface, $Z = \eta$, 1.1.5 becomes

$$p(x,y,t) = g \left[\eta(x,y,t) - z\right] + p_o \qquad 1.1.6$$

where p_o is the pressure at the surface. This expression may be used in the momentum equations, where the pressure gradients become

$$\frac{\partial p}{\partial x} = \rho g \frac{\partial \eta}{\partial x} + \frac{\partial p_o}{\partial x} \qquad 1.1.7$$

$$\frac{\partial p}{\partial y} = \rho g \frac{\partial \eta}{\partial y} + \frac{\partial p_o}{\partial y} \qquad 1.1.8$$

We now introduce the vertically-averaged velocities, U and V, where the conservation of mass equation yields

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x}\left[(h + \eta)\ U\right] + \frac{\partial}{\partial y}\left[(h + \eta)\ V\right] = 0 \qquad 1.1.15$$

where the vertical velocity at the surface has been replaced by

$$w(z = \eta) = \frac{\partial \eta}{\partial t} \qquad 1.1.16$$

The bottom stresses in the x and y directions may be approximated by the Chezy relationship

$$\tau_{b,x} = \frac{\rho g\ U\ (U^2 + V^2)^{1/2}}{C^2} \qquad 1.1.17$$

$$\tau_{b,y} = \frac{\rho g\ V\ (U^2 + V^2)^{1/2}}{C^2} \qquad 1.1.18$$

The Chezy coefficient, C, has the form

$$C = \frac{1.49}{N}\ (h + \eta)^{1/6} \quad , \quad \left[ft^{1/2}/sec\right] \qquad 1.1.19$$

where $(h + \eta)$ is in feet and the Manning friction factor, N, has units of $sec/ft^{1/3}$.

The surface stresses are due to wind and may be approximated by the quadratic law for turbulent flow:

$$\tau_{s,x} = k\ \rho_a |W_x| W_x \qquad 1.1.20$$

$$\tau_{s,y} = k\ \rho_a |W_y| W_y \qquad 1.1.21$$

where k is a dimensionless drag coefficient [herein taken as 0.0025], ρ_a is the air density, and W_x and W_y are the wind speed components in the x and y directions, respectively. Since these wind stresses are only applied to vertically-averaged momentum relations (1.1.13, 14), it follows that they do not truly model the wind-driven upper layers of an estuary. Rather, they cause bulk water movements (U and V) whose net values should approximate the total movement of water in the upper layers. The applications (Chapter 4) will illustrate this approximation.

The final differential equations are then

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} = - g \frac{\partial \eta}{\partial x} + fV + k \frac{\rho_a}{\rho} \frac{W_x |W_x|}{H} - \frac{g\ U\ (U^2 + V^2)^{1/2}}{C^2 H} \qquad 1.1.22$$

$$\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} = - g \frac{\partial \eta}{\partial y} - fU + k \frac{\rho_a}{\rho} \frac{W_y |W_y|}{H} - \frac{g\ V\ (U^2 + V^2)^{1/2}}{C^2 H} \qquad 1.1.23$$

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} (HU) + \frac{\partial}{\partial y} (HV) = 0 \qquad 1.1.24$$

where $$H = h + \eta \ . \qquad 1.1.25$$

With the development of these three equations describing the fluid motion, 1.1.22, 1.1.23, and 1.1.24, we may now proceed to obtain the necessary finite-difference approximations.

1.2 FINITE-DIFFERENCE EQUATIONS

For the solution of the equations 1.1.22 to 1.1.24, the approach of Leendertse (1) will be followed. In that scheme, the variables U and V are staggered in both space and time, and a semi-implicit method is used in the solution.

The space-staggered placement of the variables is shown in Figure 2.b. The velocities (U, V) are taken at points different in space from the point of the water surface (η). Consider the grid square, with side ΔL, denoted by the coordinate pair (m, n); then

$$x_c = (m - \tfrac{1}{2})\ \Delta L \qquad 1.2.1$$

$$y_c = (n - \tfrac{1}{2})\ \Delta L \qquad 1.2.2$$

are the coordinates of the center of the grid square. Each of the variables identified at (m, n) will have a different spatial position as follows:

$$\eta_{m,n} = \eta(x_c, y_c) \qquad 1.2.3$$

$$U_{m,n} = U(x_c + \frac{1}{2} \Delta L, y_c) \qquad 1.2.4$$

$$V_{m,n} = V(x_c, y_c + \frac{1}{2} \Delta L) \qquad 1.2.5$$

$$h_{m,n} = h(x_c + \frac{1}{2} \Delta L, y_c + \frac{1}{2} \Delta L) \qquad 1.2.6$$

$$C_{m,n} = C(x_c, y_c) \qquad 1.2.7$$

This scheme has the advantage that for the variable operated upon in time there is a centrally located spatial derivative for the linear term. For example, in the x-momentum equation (1.1.22), the time-derivative of U is associated with the spacially centered derivative of water level ($g \frac{\partial \eta}{\partial x}$).

In accordance with the semi-implicit method (see next section), the time step is split into two halves, and the time-derivative taken over the half time step. Thus, for the function F continuous in space and time, and with the notation

$$F(m \Delta L, n \Delta L, t \Delta T) = F^t_{m,n} , \qquad 1.2.8$$

the first forward time derivative is

$$\frac{\partial F}{\partial t} = \frac{2}{\Delta T} \delta_t F^t_{m,n} = \frac{2}{\Delta T} (F^{t+1/2}_{m,n} - F^t_{m,n}) \qquad 1.2.9$$

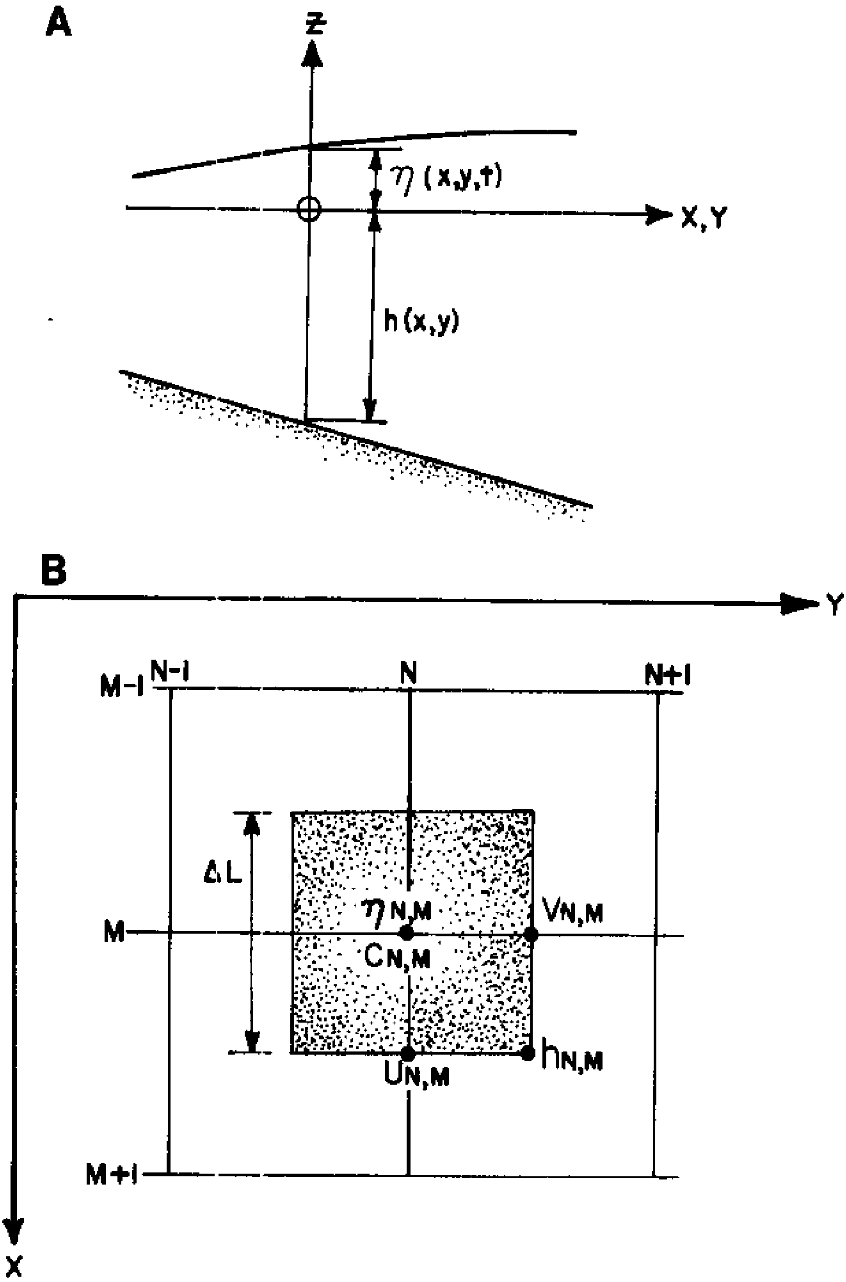


Fig. 2. Model coordinate system (A), and horizontal placement of variables in the space staggered scheme (B).

We adopt the following notation for various convenient functions of space and time:

$$\overline{F}^{x}_{m,n} = \frac{1}{2}\left(F_{m+1/2,n} + F_{m-1/2,n}\right) \qquad 1.2.10$$

$$\overline{F}^{y}_{m,n} = \frac{1}{2}\left(F_{m,n+1/2} + F_{m,n-1/2}\right) \qquad 1.2.11$$

$$\delta_x F_{m,n} = \left(F_{m+1/2,n} - F_{m-1/2,n}\right) \qquad 1.2.12$$

$$\delta_y F_{m,n} = \left(F_{m,n+1/2} - F_{m,n-1/2}\right) \qquad 1.2.13$$

$$\delta^*_x F_{m,n} = \frac{1}{2}\left(F_{m+1,n} - F_{m-1,n}\right) \qquad 1.2.14$$

$$\delta^*_y F_{m,n} = \frac{1}{2}\left(F_{m,n+1} - F_{m,n-1}\right) \qquad 1.2.15$$

$$F_{m,n} = \frac{1}{4}\left(F_{m+1/2,\ n+1/2} + F_{m-1/2,\ n+1/2} + F_{m+1/2,\ n-1/2} + F_{m-1/2,\ n-1/2}\right) \qquad 1.2.15$$

The momentum and conservation of mass equations may then be transformed to finite-difference equations (six equations result, three for each half of the time step) and solved for the new value in time. The equations are given in Appendix A. The solution method will be discussed in the next section.

1.3 METHOD OF SOLUTION

The solution of Equations A.1 to A.6 (Appendix A) is called by Leendertse (1) a "multi-operation" method, which is a modification of the "leap-frog" method. In the first half time step, values of U and η are computed implicitly along a grid row in the x-direction at the time $(t+1/2)\,\Delta T$. Then V is computed at the same time level explicitly. In the second half time step, V and η are computed implicitly at $(t+1)\,\Delta T$ along grid rows in the y-direction, after which U is calculated explicitly at $(t+1)\,\Delta T$.

In the first half of the time step, the time derivative of U in the x-momentum equation is approximated by a backward difference:

$$\frac{\partial}{\partial t}\left(U^{t+1/2}\right) = \frac{2}{\Delta T}\left(U^{t+1/2} - U^{t}\right) = \mathrm{fcn}\left(\eta^{t+1/2}\right) \qquad 1.3.1$$

In the second half time step, a forward difference is used:

$$\frac{\partial}{\partial t}\left(U^{t+1}\right) = \frac{2}{\Delta T}\left(U^{t+1} - U^{t+1/2}\right) = \mathrm{fcn}\left(\eta^{t+1/2}\right) \qquad 1.3.2$$

Thus, over a full time step, the time derivative is a central difference with respect to the water level:

$$\frac{\partial U}{\partial t} = \frac{U^{t+1} - U^{t}}{T} = \mathrm{fcn}\left(\eta^{t+1/2}\right) \qquad 1.3.3$$

This composite relation defines the leap-frog method.

The set of difference equations for the implicit time step on U and η may be written as

$$\left[A\right] \left\{U^{t+1/2} \text{ or } \eta^{t+1/2}\right\} = \left\{b\right\} \qquad 1.3.4$$

where $\left[A\right]$ is a tridiagonal matrix. Equation 1.3.4 may then be solved by Gaussian elimination [see Mitchell (8) for example] for the new values of U and η at (t+1/2). A similar procedure is used for the second implicit operation involving v and η at time (t+1). The details are given in Appendix B.

1.4 STABILITY

An extensive analytical treatment of stability has been given by Leendertse (1), and the reader is referred to that original exposition for details. Only a brief outline of the approach will be presented here.

The form of investigation of the stability is that introduced by von Neumann (see 8), which assumes a Fourier expansion of a line of errors propagating over time. Consider the harmonic decomposition of error, E(x), as

$$E(x) = \sum_j A_j \; e^{i\beta_j x} \qquad 1.4.1$$

for all spatial frequencies, β_j, where β is real. For linear equations, only one frequency need be examined. At the end of the time step, the error may be expressed as

$$E(x) = A \; e^{i\beta x} \; e^{\alpha\Delta T} ,$$

where $\alpha = \alpha\ (\beta)$ is, in general, complex. The von Neumann criterion for stability is that

$$|e^{\alpha\Delta T}| \leq 1 \qquad 1.4.2$$

Leendertse (1) has determined analytically that the multioperation method is unconditionally stable for the linear simplifications of the momentum and mass equations, which are (for the x-direction)

$$\frac{\partial U}{\partial t} + g\,\frac{\partial \eta}{\partial x} = 0 \qquad 1.4.3$$

$$\frac{\partial \eta}{\partial t} + h\,\frac{\partial U}{\partial x} = 0 \qquad 1.4.4$$

Numerical experiments (1,6) are used to establish the stability of the full equations 1.1.22-24 where exact analysis is not available.

The consequence of these studies is that the present method is stable for any size time step, ΔT, for regions of uniform geometry. However, for modeled regions with the irregular geometry that often occurs in nature, stability is not guaranteed. The situation may be remedied by reducing the time step or grid size, or by smoothing the bottom contours to eliminate steep depth gradients.

It should also be noted that the total water depth must remain positive at all times, the most critical time being low water. This trouble frequently arises in shallow grids near land boundaries. and care should be taken in the selection of depths at these locations, even to the extent of introducing some distortion of the bathymetry.

1.5 BOUNDARY CONDITIONS

Two different types of external grid interfaces, or boundaries, are possible in the numerical model: a water-water or water-land interface. At the first, either the water level, η, or one velocity component (U or V) must be specified. At the second, the appropriate normal velocity component is zero.

A difficulty is encountered when the spatial derivatives δ_x^* and δ_y^* (1.2.14, 1.2.15) in the convective terms are applied in a grid with a land boundary. At least one velocity component will lie outside the field of computation. Leendertse overcomes the problem by dropping the convective term in the momentum equation. Although this procedure produces an inaccuracy in the numerical results, it preserves stability (1).

1.6 GRID IDENTIFICATION

The method of solution involves solving for the dependent variables along a grid row. Therefore, each row in the x and y directions is described by a row identification number. Three types of gird squares occur in the field: land, water, and water-boundary grids.

The identification number gives the m (or n) values of the end (first and last) water grids in the row, and indicates the type of the grid adjacent (and in the same row) to the end grids. If an end grid is adjacent to a land grid, the normal velocity there will be zero; if adjacent to a water-boundary grid, a water level or velocity boundary condition will be searched for. In no case will the adjacent grids be water grids.

II. APPLICATION TO NARRAGANSETT BAY

2.0 INTRODUCTION

Now that the fundamentals of the numerical solution method have been investigated, the model may be applied to the specific case of Narragansett Bay. This entails the selection of the grid net which describes the Bay geography, and selection of the time step. Depth and Chezy coefficient data must be introduced. The boundary conditions must be prescribed as continuous time functions. The following sections outline the procedures involved.

2.1 GRID NET SELECTION

Few, if any, guidelines exist for the selection of an optimum grid system for a water body, especially one with complicated geography like Narragansett Bay. The first step taken however, was the choice of the water boundaries. The area of the Bay to be modeled is bounded on the south by Rhode Island Sound, on the east by the entrance of Mt. Hope Bay, and the north at the narrowing of the Seekonk River. This area represents about two-thirds of the entire Bay. The portion excluded, Mt. Hope Bay and the Sakonnet River, comprises another estuarine system, and is geomorphically connected to the main part of the Bay by a narrow passage.

Secondly, the computation scheme imposes a lower limit of two grids per row in the field. Thus the narrowest channel must be at least two grids wide. These critical areas occur in the lower Bay, in the East and West Passages, and in the upper Bay in the Providence River (Fig. 1). Therefore, a grid length of one-half nautical mile was chosen. The resulting grid net (Fig. 3) consists of 324 water and water-boundary grids within the rectangular (19 by 48) field. The x-axis is 10.1 degrees to the right of the true north direction for more accurate representation of the coastline geometry.

2.2 TIME STEP SELECTION

One property of the implicit solution method is the unconditional numerical stability, regardless of time step. However, the size of the time step has an effect on the accuracy of the solution.

Leendertse (1) has shown that the solution has high accuracy when

$$\beta = \frac{\Delta T}{\Delta L} \sqrt{gh} \qquad 2.2.1$$

is of the order of five or less, where h is the maximum depth of water. Hence, the factor $\sqrt{gh}$ is the maximum long-wave celerity. For a maximum depth of 152 feet and a ΔL of 3038 feet, a ΔT of 220 seconds would give a

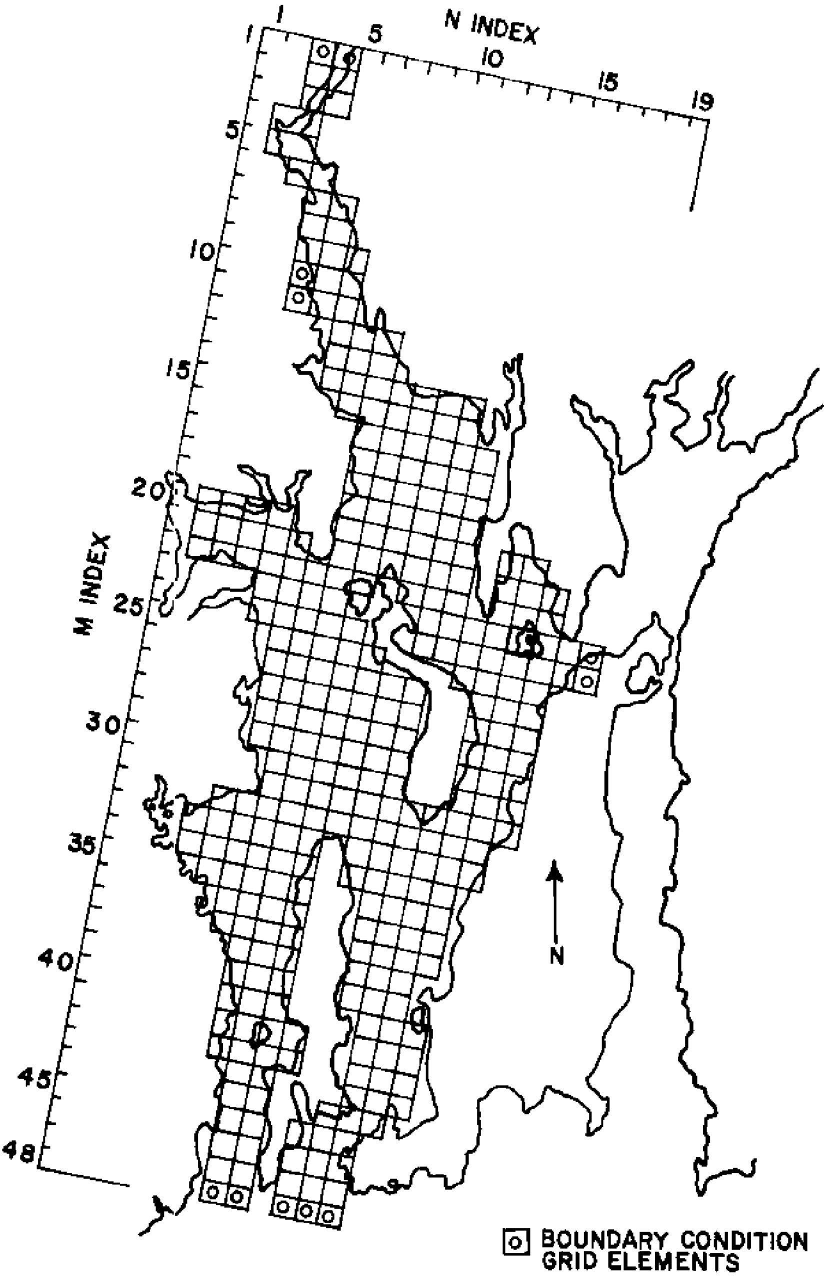


Fig. 3. Grid system for Narragansett Bay.

β value of 4.91. Therefore, a time step of this size or less insures good accuracy, especially since the average depth of the Bay is only 30 feet.

2.3 SPECIFICATION OF DEPTHS

Bathometric variations are accounted for in the depth specification at each grid square. In accordance with the placement of variables within the grid (Fig. 2b) the depth in the corner of the grid at $(x_c + \frac{1}{2}\Delta L, y_c + \frac{1}{2}\Delta L)$ is entered as data for all grids in the computation field. The number entered is the actual depth at mean sea level at that point on the grid, and not the average depth over the grid square. Depths may also be entered at grid squares outside the computation field, such as those adjacent to water grids.

General information on the bathymetry was obtained from the U.S. Coast and Geodetic Survey Chart No. 353, which gives depths at mean low water. It should be noted that while such charts are useful, certain small-scale features may not be evident from them. For certain critical locations, therefore, depth surveys would be quite useful. These were carried out in the West Passage at the Jamestown Bridge and at the Mt. Hope Bridge.

2.4 CHEZY COEFFICIENTS

The effects of bottom friction are introduced through the Chezy coefficient.

$$C = \frac{1.49}{N} (h + \eta)^{1/6} \qquad 1.1.19$$

The dependence on η makes C a time-varying function. However, since the water level, η, is usually much smaller than the depth, h, at mean sea level, its influence is small. This was borne out by a model study of tidal flow in which the Chezy coefficient was computed each half-hour; in no case was the maximum variation more than ten percent. Values of C are computed at the start of each run (for $\eta = 0$), and are not changed afterward.

The selection of the Manning factor (N) poses a somewhat more difficult problem, due to the lack of extensive studies of rivers and bays. Masch and Brandes (3), for example, use values between 0.018 and 0.054, which corresponds to "rubble set in cement" and "natural river channels: winding, with pools and shoals," respectively, in a table given by Henderson (9). The essential concept is bottom roughness, which varies considerably in an area as large as Narragansett Bay. For approximation, then, the Manning factor was taken as a linear function of m, the model grid row number:

$$N(m) = N_{avg} \left[1.3 - 0.6m/max\right] \qquad 2.4.1$$

which varies from 1.3 N_{avg} in the Providence River to 0.7 N_{avg} at the mouth of the Bay. The average value, N_{avg}, was determined from comparisons of predicted and observed velocities, and was taken as .020. It is expected that further model testing and bottom surveys may change this representation.

2.5 THE RHODE ISLAND SOUND BOUNDARY

The primary driving force at the mouth of Narrangansett Bay is the astronomical tide, and thus is entered as a water level boundary condition at the location, grids m = 48, n = 8, 9, 11, 12, 13 (Fig. 3; Fig. 4). Other types of boundary conditions are used in the model, and these will also be discussed.

The Coast and Geodetic Survey regularly collects and analyzes tidal elevations at several locations around the Bay. The primary stations are at Newport, Bristol, and Providence, and the data obtained from them is the amplitude and phase angle of the twenty or so largest tidal constituents (10). A number of secondary stations have been occupied, and the times of high and low water relative to Newport are given for them in Reference 11.

The tidal forcing function may be represented by the sum of several sinusoidally varying terms, each with a unique amplitude, angular speed, and phase angle (12).

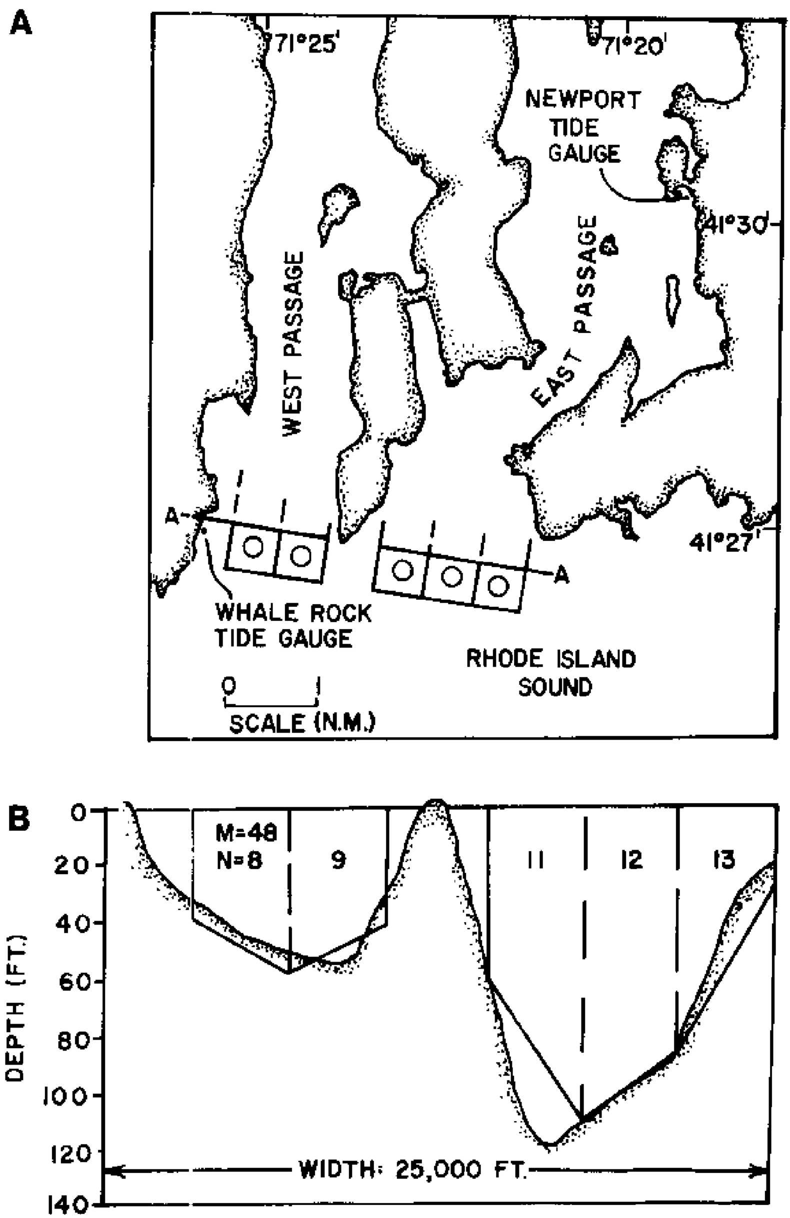


Fig. 4. Geography of Narragansett Bay at Rhode Island Sound boundary (A), and bathymetry of section A-A (B).

The phase angle is taken relative to Greenwhich, England; the amplitude is modified by a function of lunar position, (f_n). The equation for the water level, η, is

$$\eta(t) = \sum_n f_n (t) H_n \cos \left[\omega_n t + (V_o + u)_n - k_n\right] \qquad 2.5.1$$

where for each constituent, n,

$f_n(t)$ = amplitude factor depending on the position of the moon's line of nodes

H_n = amplitude of the constituent

ω_n = angular speed (degrees per hour) of the constituent

$V_o + u$ = value of the equilibrium argument when $t = 0$

k_n = epoch (angular phase difference from Greenwich)

t = time (hours) from reference time

The values of H_o, H_n, and k_n are calculated for each tide station. The angular speed (ω), lunar node function (f_n) and equilibrium argument ($V_o + u$) can be calculated from knowledge of astronomical motions, and are tabulated in Reference 12. A more detailed description is given in the description of the subroutine KURIH.

The tide at the lower boundary is calculated at each of the end grids (m = 48, m = 8, 13) by an equation of the form 2.5.1. The tide at the intermediate grids is obtained by linear interpolation. The amplitude and epoch of each constituent was originally obtained from analysis of their values at the three other stations. The values are now being modified by data obtained from the Ocean Engineering Department Whale Rock tide gauge.

Several other types of boundary conditions are included in the model, and are used in various tests and experiments. At present six different conditions are possible:

1. tidal input — The astronomical tidal function as described above, is used for water level.
2. zero tide — No tide variations of water level occurs.
3. extrapolated water level — The boundary water level is extrapolated from the interior field.
4. extrapolated velocity — The velocity is extrapolated from the flow in the interior field.
5. surge — The water level corresponding to hurricane surge at the mouth of the bay is entered.
6. surge plus tide — The sum of 1 and 5 is used.

These six conditions provide quite a measure of flexibility to the model application. Their usage is described in the next two chapters.

2.6 PROVIDENCE RIVER BOUNDARIES

The boundaries in the northern part of the Bay represent river entrances, and velocity boundary conditions are used to model them. Providence Harbor is the confluence of several rivers; and further down the Bay the Pawtuxet River joins the Providence River (Fig. 5). Several smaller rivers also flow into the Bay, but their discharges are relatively small and have been neglected.

The total volumetric flowrate from the Blackstone-Seekonk, Moshassuck, and Woonasquatucket Rivers is entered at boundary grid m = 1, n = 3, 4 to simplify the model grid in that region. The mean annual flow rate, about 890 c.f.s. including discharge from the City of Providence, is fairly small compared to tidal flowrate, but local tidal velocities computed by the model are significant in the adjacent area.

The daily average flowrate may either be obtained from surface water records (13) or estimated from the ratio of monthly to yearly mean discharges (Fig. 6).

The Pawtuxet River boundary (m = 10, 11; n = 4) is handled in the same manner as the Providence Harbor boundary.

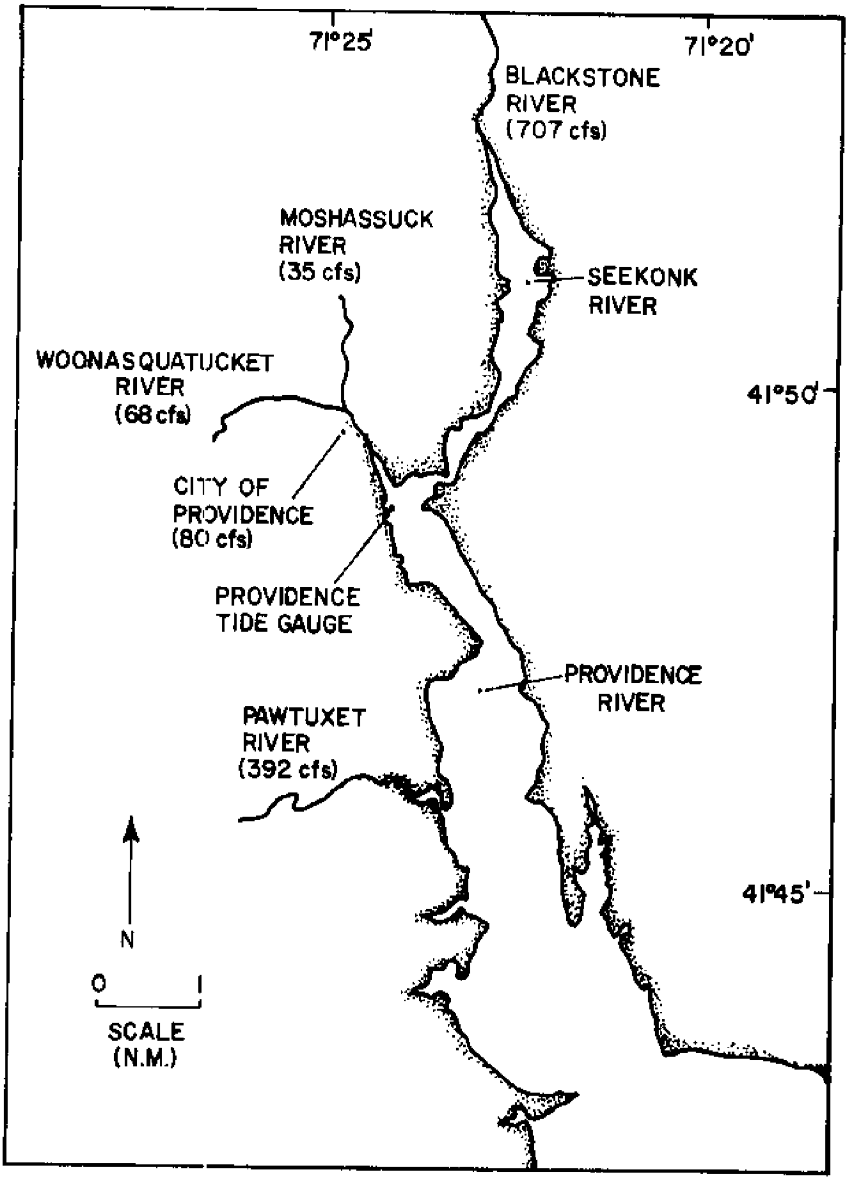


Fig. 5. Geography of Narragansett Bay near the Providence River.

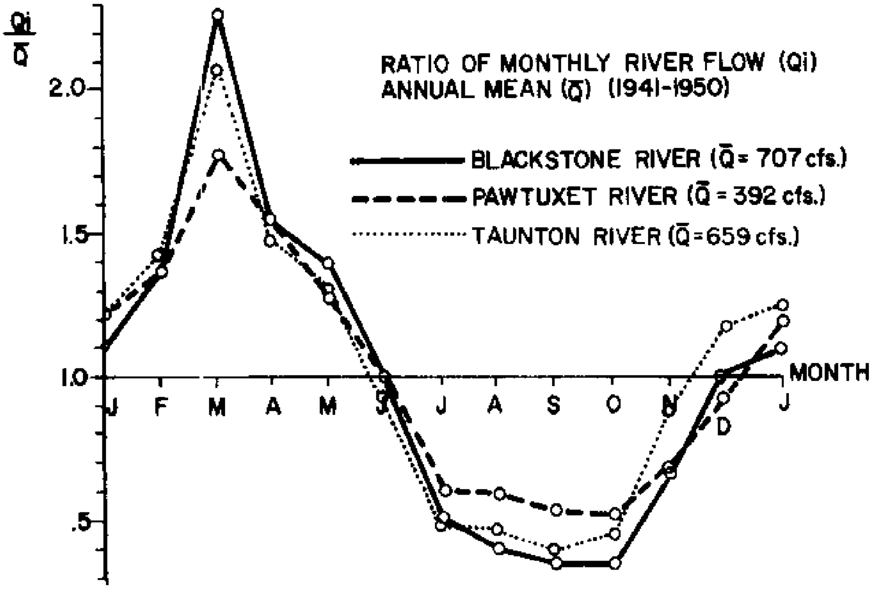


Fig. 6. Monthly variations in discharge for local rivers.

2.7 THE MT. HOPE BOUNDARY

The boundary at the entrance to Mt. Hope Bay probably is the most difficult to model accurately. The local geography (Fig. 7) does not permit the use of the Bristol Harbor tide as a water level boundary condition, so the tidal velocity, based upon the volumetric flowrate, is used.

The total flow under the Mt. Hope Bridge is determined by tidal differences, river discharge, and wind effects. The tidal flow results from water level variations between the Bay proper and Mt. Hope Bay, which itself is connected to Rhode Island Sound through the Sakonnet River. Also, a certain fraction of the fresh water discharge into the Mt. Hope Bay, primarily from the Taunton River (mean annual flowrate: 660 c.f.s.), passes under the bridge. Local winds may contribute to daily variations in the flow, but they are neglected since no data on wind currents are available.

The earliest available measurements of the flow under the bridge are reported by Haight (14), which made use of a 7-foot pole and three current meters on August 7 and 8, 1930. Recent measurements (August 5 and 18, 1971) were taken by using several poles spaced across the section under the bridge. The general approach of analyzing the data used by Haight was applied to the newer observations.

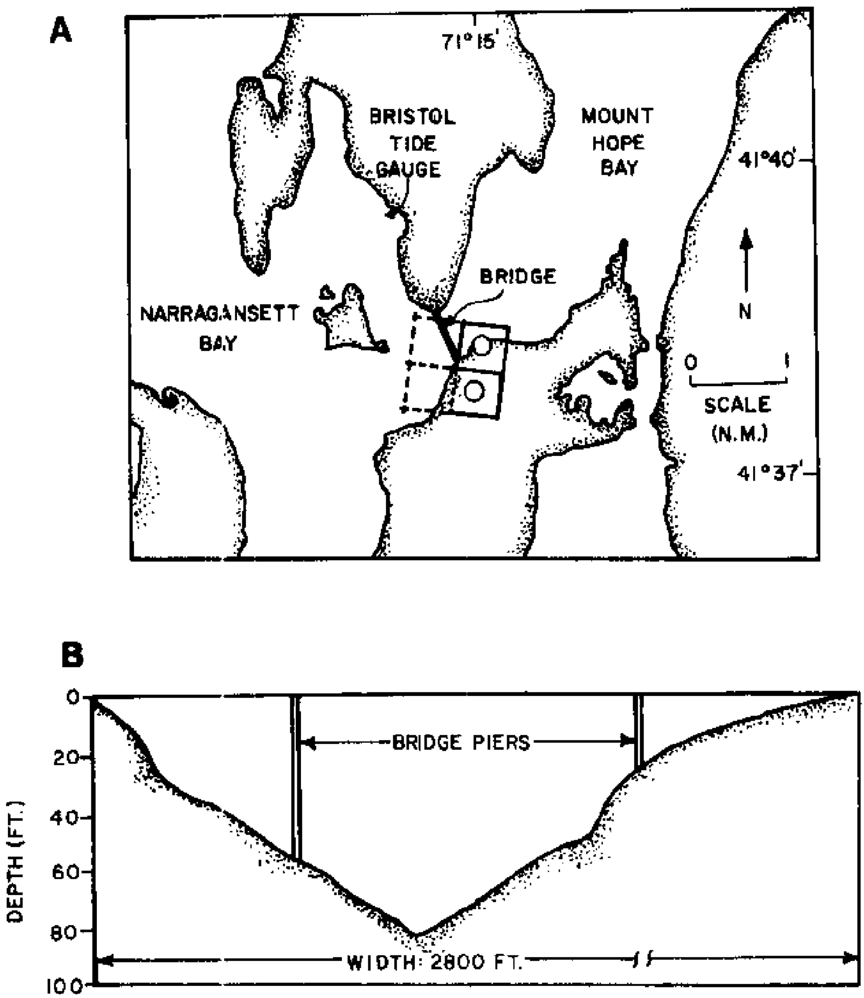


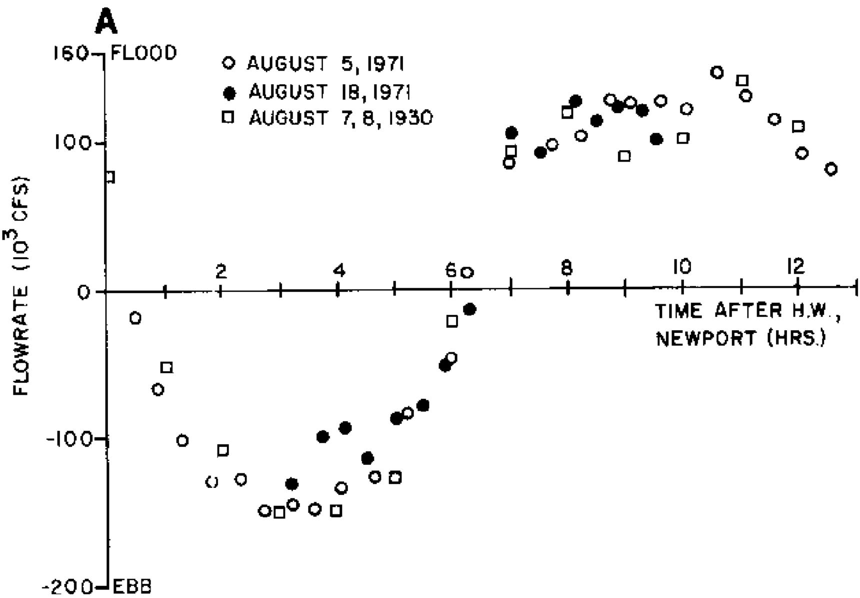
Fig. 7. Geography of Narragansett Bay near the Mount Hope boundary (A), and bathymetry at the Mount Hope Bridge (B).

Due to the nature of the Bay geometry, Haight (14) showed that the currents due to the lunar (M_2, M_4, and M_6) constituents of the tide accounted for most of the observed current. The flowrate can then be approximated by

$$q = \sum_{k=1}^{3} q_k \cos \left[\frac{2 \pi k}{12.42} (t - \tau_k) \right] \qquad 2.7.1$$

where q is the flowrate, and τ the time to first flood after high water. The flowrate was deduced from the 1930 data by integrating the velocity over the depth, and multiplying by a weighted area under the bridge (90,600 ft^2). The flowrates for the other observations were calculated by summing the products of the pole velocity and the incremental area; the resultant values were adjusted for the tidal range and smoothed. A weighted average was then analyzed by least squares, using an equation similar to 2.7.1. The results are shown in Table 2.7.1, and in Figure 8. The tidal velocity is obtained by dividing the flowrate, q, by the area at the boundary.

The portion of the Taunton River discharge passing under the bridge is obtained from Hicks, (15), who estimated the river outflow from the ebb flowrates through each Bay passage. The value used here is 72 percent of the annual mean flow or 475 c.f.s.



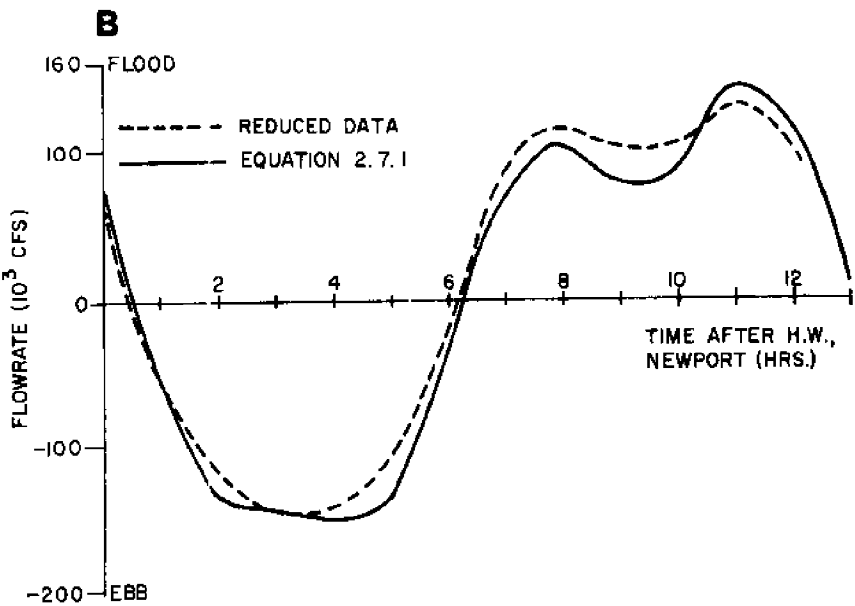


Fig. 8. Flowrate data at the Mount Hope Bridge (A), and comparison of reduced data and the analytic expression used in the model (B).

TABLE 2.7.1 Lunar Constituent Analysis of Flow Under Mt. Hope Bridge

k	Lunar Constit-uent	Period (hr.)T	Time to first flood τ(hrs.)	Current (kts)	q_k (10^3 cfs)
1	M_2	12.42	9.87	1.12	150.5
2	M_4	6.21	6.29	0.29	33.2
3	M_6	4.14	3.32	0.15	35.4

III. MODEL DYNAMIC RESPONSE CHARACTERISTICS

3.0 INTRODUCTION

It is helpful to know how the model will respond under a variety of input conditions, so that the relative influence of parameters may be assessed. Since the two-dimensional equations are non-linear, an analytic sensitivity analysis is quite difficult. Leendertse (1), for example, discussed only the simplified momemtum equation with linear damping and the continuity equation with constant depth

$$\frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} + ku = 0 \qquad 3.0.1$$

$$\frac{\partial \eta}{\partial t} + h \frac{\partial u}{\partial y} = 0 \qquad 3.0.2$$

Therefore, a series of numerical experiments were carried out, using the computer model of Narragansett Bay for the hydraulic system. Two parameters were the subject of investigation: the time step and the Chezy friction factor. These were varied, along with several types of boundary conditions. As a result, insight into the computed solution was gained, and its dependence upon the input explored.

No experiments involving variable grid size or bathymetry were undertaken. The grid net was considered acceptable on the basis of computer-imposed limitations. The geography of the Bay is essentially constant, that is, the bottom is not subject to variations during a tidal cycle, and no important shoreline changes occur.

3.1 PROPERTIES OF THE NUMERICAL SOLUTION

The computed solution may be examined in a manner similar to that used in the error analysis. Following Sobey (6), we consider the following set of linear equations:

$$\frac{\partial u}{\partial t} + g\,\frac{\partial \eta}{\partial x} = 0 \qquad 3.1.1$$

$$\frac{\partial v}{\partial t} + g\,\frac{\partial \eta}{\partial y} = 0 \qquad 3.1.2$$

$$\frac{\partial \eta}{\partial t} + h\,\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = 0 \qquad 3.1.3$$

The Fourier series representation of the solution is

$$\overline{F} = \sum_m F_m^* \quad e^i\ (\beta_m t + \sigma_{m_1} x + \sigma_{m_2} y) \qquad 3.1.4$$

where the vector $\overline{F}$ is

$$\overline{F} = \begin{Bmatrix} u \\ v \\ \eta \end{Bmatrix} \qquad 3.1.5$$

and β_m and σ_m are the real wave time frequency and wave number of the m^{th} component, respectively. The substitution of 3.1.4 into 3.1.1 to 3.1.3 leads to

$$A\ (\beta,\sigma_1,\sigma_2)\overline{F} = 0 \qquad 3.1.6$$

where A is the amplification matrix. Since 3.1.1 to 3.1.3 are linear, only one component need be examined. For the difference equation equivalents of 3.1.1 to 3.1.3, the solution is

$$F^1 = F^{1^*}\exp\left[i(\beta' n\ \Delta T + \sigma_1 j\ \Delta x + \sigma_2 k\ \Delta y)\right] \qquad 3.1.7$$

which yields the computed wave amplifications matrix from which β' is solved. The computed wave number, β', is such that $Re(\beta')$ is the computed wave frequency, and $Im(\beta')$ is a measure of the computed wave deformation.

Sobey shows that essentially zero deformation results in Leendertse's scheme for

$$\frac{\Delta T}{\Delta L}\sqrt{gh} \leqq 5$$

and negligible frequency distortion for a tidal wavelength to grid length ratio above 100. Leendertse (1) shows that, when linear damping is added (as in 3.0.1), the computed velocity and tidal amplitudes approach unity from above for decreasing time step. His corresponding frequency results are similar to those of Sobey. However, the amplitude distortion is a function of tidal wavelength, so one can expect different amounts of distortion for different tidal constituents. This effect may be important in Narragansett Bay, where it was shown (section

2.7) that the M_2, M_4, and M_6 lunar constituents of currents are large.

The tidal wavelength may be estimated by standing wave relationship (17)

$$\frac{\eta(\text{head})}{\eta(\text{mouth})} = \sec(k_t \ell) \qquad 3.1.8$$

where k_t is the tidal wave number, and ℓ the length of the Bay. Data from Narragansett Bay (10) indicate that the tidal wavelength is of the order of 200 n.m. The half nautical mile grid length should therefore give adequate spatial resolution.

3.2 FREE OSCILLATION EXPERIMENTS

In this series of experiments, a linear tide was imposed upon the Bay (zero tide at the mouth and two to three feet at Providence Harbor), and then allowed to oscillate freely with a zero tide at the mouth. Experiments involving changes in the time step and the Chezy coefficient were conducted.

The time step was varied from 1.5 to 12.0 minutes, and the current and water level at a specific grid (M = 17, N = 9) were examined. The Chezy coefficient was held constant throughout the Bay. As expected, the water level and velocity appeared as a damped oscillation (Fig. 9). It was found that for the first 300 time steps, the

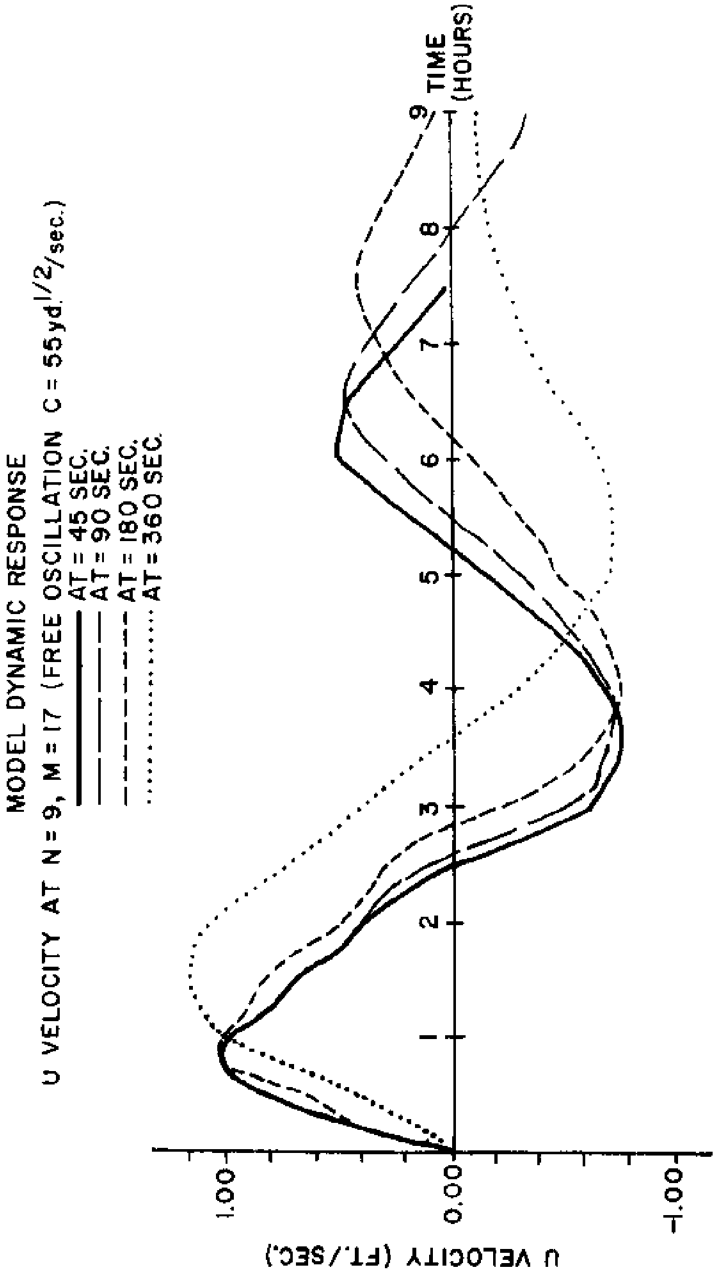


Fig. 9. Variation of computed, free-oscillation velocity with time step size.

amplitude decrease would be approximated by

$$\frac{\eta(t)}{\eta(0)} = e^{-\mu\ t^{1/2}} \qquad 3.2.1$$

where t is the time in minutes and μ is a damping factor. The damping factor was found to be a function of time step size, ΔT. The values obtained are shown in Figure 10a and indicate that distortion increases greatly for ΔT above four minutes.

The phase distortion (Fig. 8) became extreme for ΔT larger than six minutes, although there appears to be only small amplitude distortion. The results for the velocity solution are similar: the amplitude of the first peak decreased by two percent when the time step was increased from 1.5 to 6.0 minutes.

One interesting result is the length of the natural period of the oscillation. With decreasing time step, the period approached a value of about 4.8 hours (Fig. 10b). A value of 5.72 hours was computed by Haight (14), who used the rectangular estuary approximation (17)

$$T = 4\,\ell / \sqrt{gh} \qquad 3.2.2$$

for the fundamental period. He used a bay length (ℓ) of 24 n.m., and a depth (h) of 25 feet. More realistic values for length and depth can be taken; for example, ℓ = 22 n.m. (mouth to Providence Harbor) and h = 30 feet

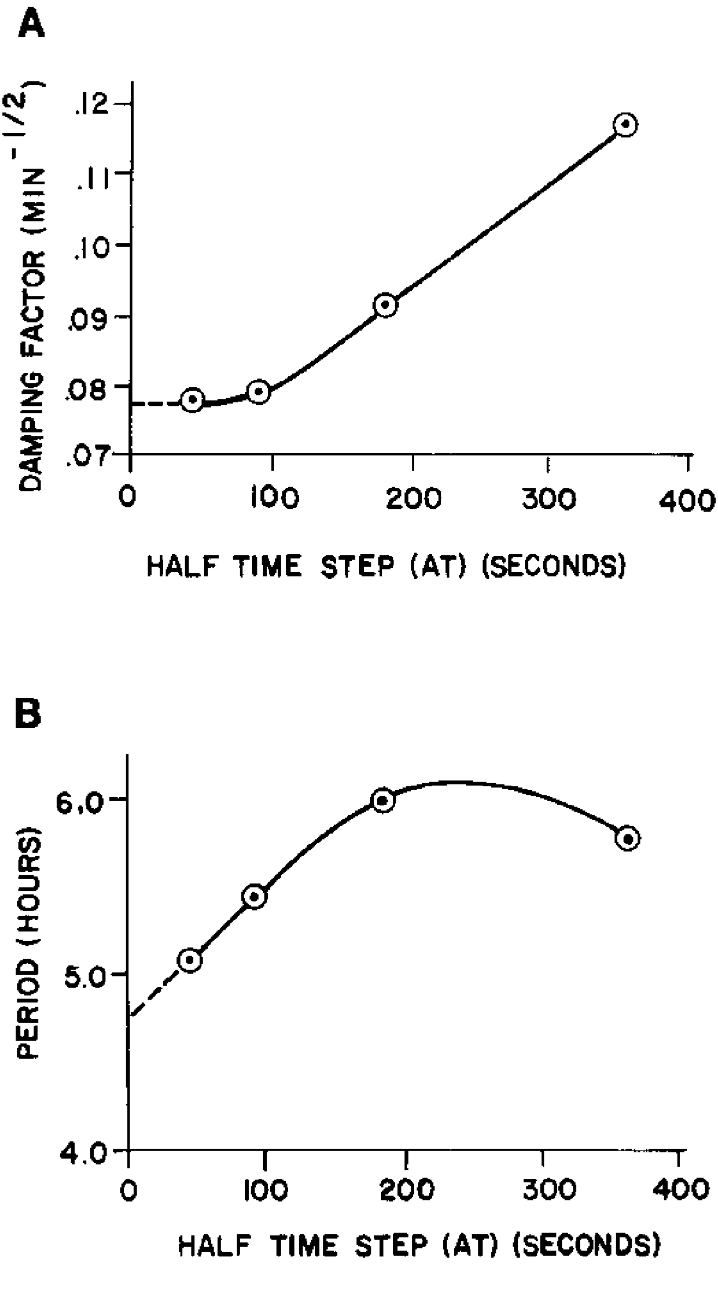


Fig. 10. Variation of free-oscillation damping factor (A), and natural period (B) with time step size.

fives a period of 4.73 hours. Thus the period obtained by the model is quite reasonable.

Another series of free oscillation experiments were conducted varying only the Chezy coefficient (through a variable Manning factor, N). The results are shown in Fig. 11. As expected, the velocity magnitude was a strong function of N. The amplitude of the first peak for N = 0.015 is about 60 percent greater than the amplitude for N = 0.040. Note that the phase shows very little variation over this range of N.

One application of these results is the calculation of the time required for transients to damp down to arbitrarily small values. Using the exponential damping representation (Eq. 3.2.1) with a damping factor of 0.073 $\min^{-1/2}$, an interval of about 42 hours is necessary for water level to damp to two percent of its initial value.

3.3 FORCED OSCILLATION EXPERIMENTS

Another series of experiments involved dynamic boundary conditions, usually a tidal variation. The primary object of these tests was the determination of the optimum initial conditions of water level and velocity to be used in predictive model studies. The previously found running time for the elimination of transients of 42 hours (three and one-half tidal cycles) is only an

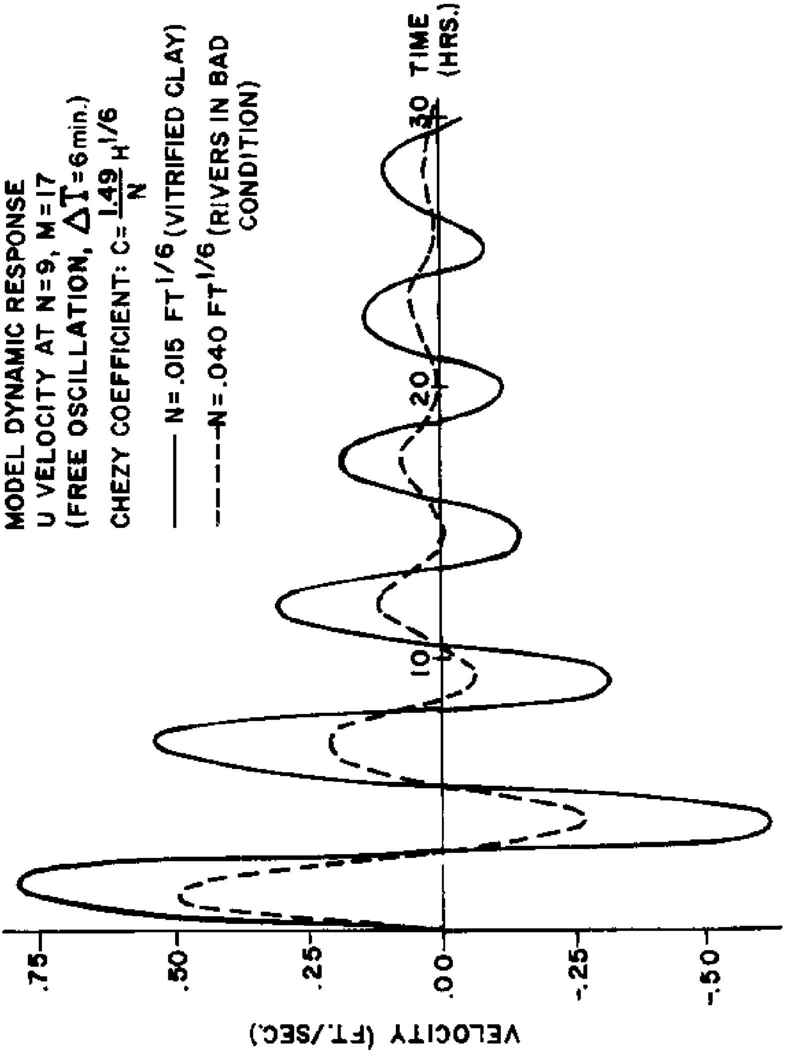


Fig. 11. Variation of computed free-oscillation velocity with Manning factor N.

approximation, since 3.2.1 may not apply for large times.

The first case of interest was the application of the tidal forcing function to the Bay completely at rest ($U = V = \eta = 0$). The results (Fig. 12) bear out our intuition that this is a poor initial condition. Transients in the Newport tide persisted for at least two full tidal cycles. Another case, using a linear tide, also gave similar results.

A more interesting initial condition was a (U, V, η) field obtained from a long-time computation with the values stored on punched cards after transients were eliminated. The Bay is essentially in a fully dynamic state for this condition. When the model was started at a random time, the phase mismatch produced small-scale oscillations (Fig. 13a) similar to those found in the static case (Fig. 12). However, when the model was started in phase with the (U, V, η) data field, the improvement was remarkable. Transients were not evident after one full tidal cycle. Thus it is recommended that such an in-phase dynamic starting condition be used for predictive model runs. (Fig. 13b)

3.4 FLOWRATE EXPERIMENTS

The above-mentioned experiments and analysis of the computed solution give insight into the nature and effects

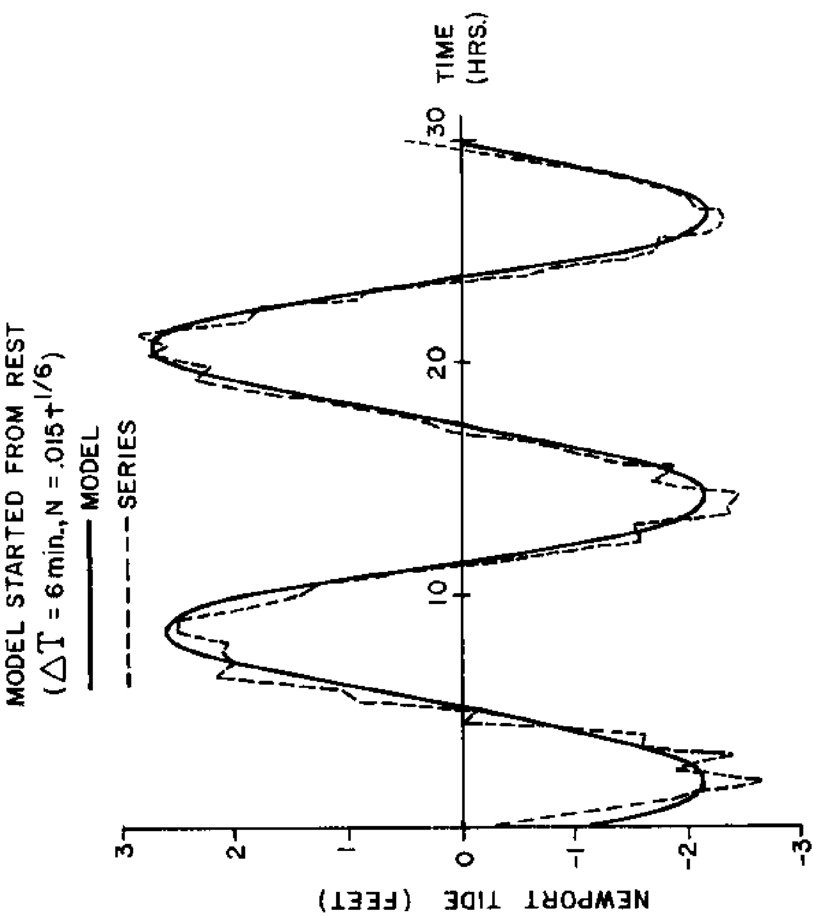


Fig. 12. Comparison of predicted (historical) tide with computed tide, starting the model at rest conditions.

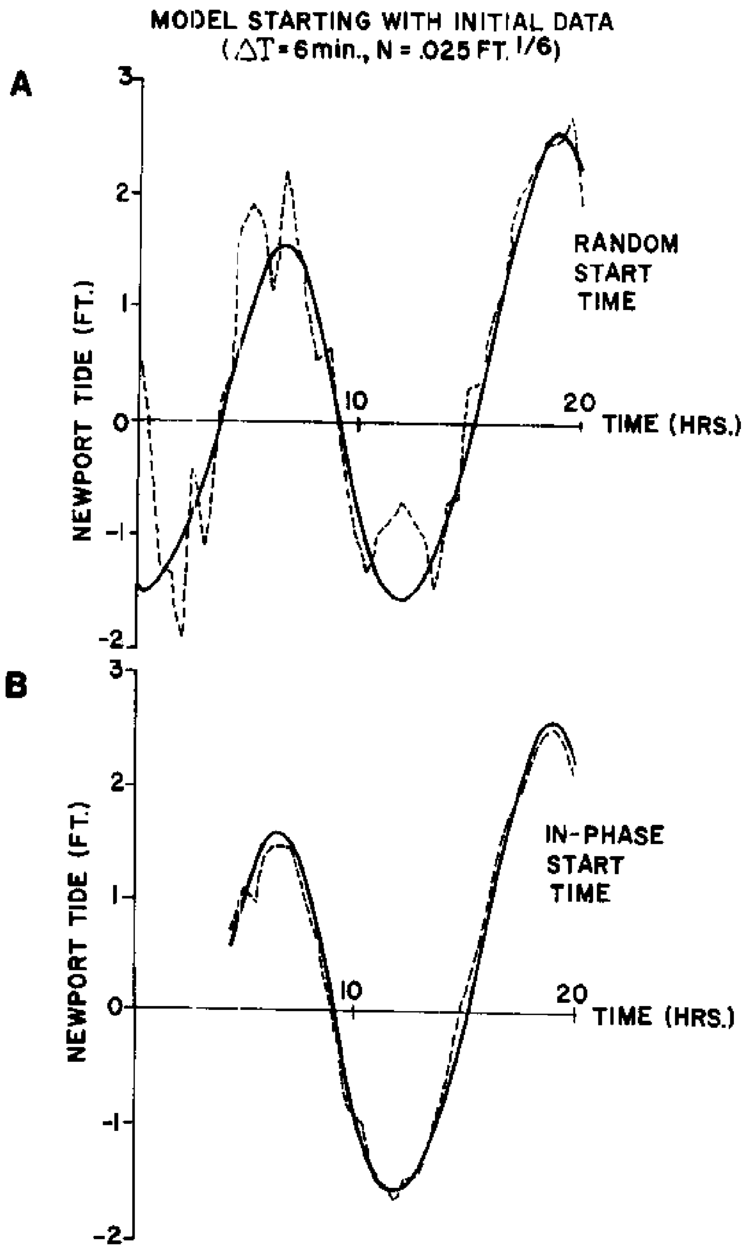


Fig. 13. Comparison of predicted (historical) tide with computed tide, starting the model with an initial velocity-water level field out of phase (A), and in-phase (B) with the tidal driving function.

of various parameters on the solution. Their influence was tested in the calculation of the flow past a vertical section of the Bay.

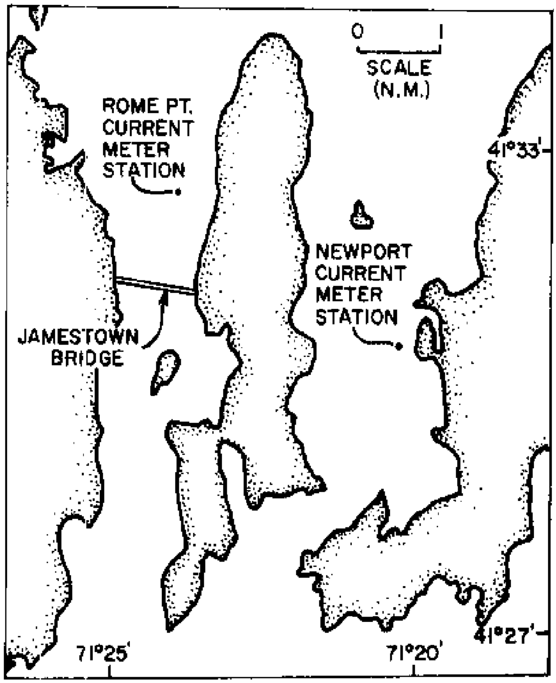
The flowrate is computed by summing the products of the area and the velocity for each grid across the section. At the Jamestown Bridge, the flowrate is computed by

$$Q = \frac{\Delta L}{4} \sum_{n=7}^{n=9} (h_{m,n} + h_{m,n-1} + \eta_{m,n} + \eta_{m+1,n}) u_{m,n} \qquad 3.5.1$$

for m = 38 (Fig. 14). Both the time step and Manning factor were varied.

The effect of the time step is seen in Figure 15a. The magnitude of the flowrate is a function of the time step and decreases with decreasing time step, as stated by Leendertse (1). However, the effect is not linear, that is, the curves also vary in shape. This can likely be attributed to the differential effect on each tidal constituent. The ebb peak decreases by five precent when T decreases from 3.0 to 1.5 minutes, which is a relatively small difference (a ΔT of 6.0 minutes has been judged too large because of its effect on the damping factor).

The influence of the Manning factor N is seen in Figure 15b. The magnitude and shape are both dependent upon N, which was expected. The change of N from 0.025 to 0.020 increases the ebb peak by about ten percent, A



B

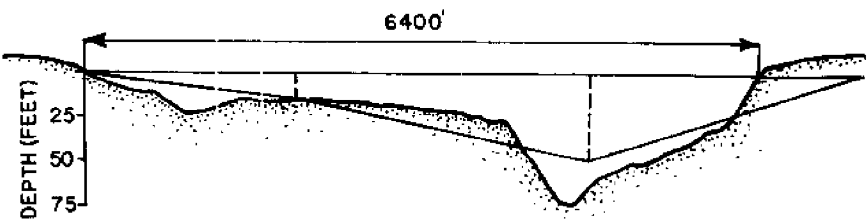


Fig. 14. Geography of lower Narragansett Bay showing the positions of the data stations used in the verification studies (A), and bathymetry at the Jamestown Bridge (B).

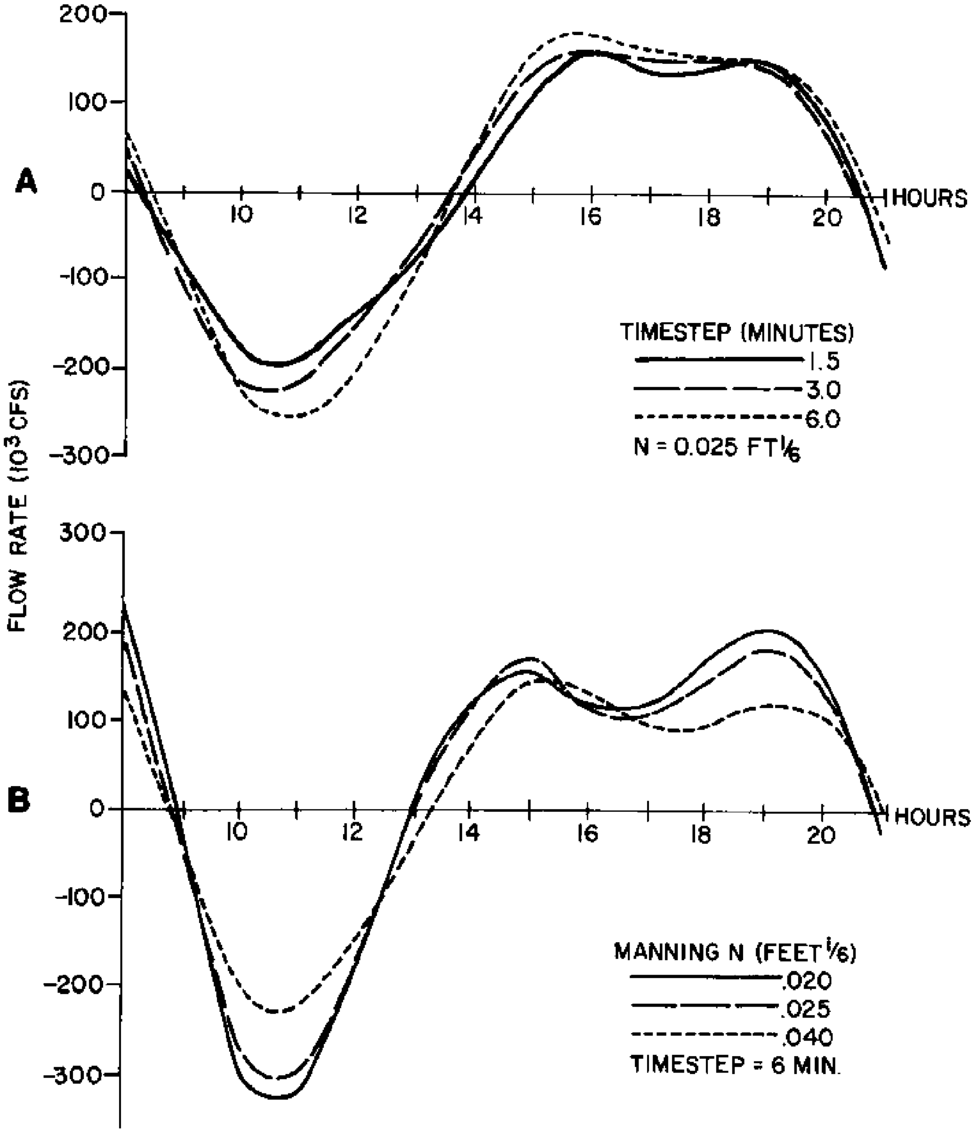


Fig. 15. Variation of computed flowrate with time step (A), and Manning factor N (B).

but has only a small effect on the shape ($N = 0.040$ is judged too large based on comparisons with observations - Chapter 4).

IV. MODEL VERIFICATION AND APPLICATIONS

4.0 INTRODUCTION

At this phase of model development, the physical grid has been selected, and insight into the dynamic response characteristics of the model has been gained. It now remains to compare hydraulic quantities computed by the model with those measured in the field. The primary quantities are water levels and current velocities. Secondary quantities such as flowrates and particle paths are also useful in the verification studies.

When the comparison is unfavorable, an attempt is made to isolate the factors contributing to the discrepancy. Modifications, if necessary, are introduced to the model. As the number of data sets used for comparison increases, a series of such modifications will eventually lead to realistic modeling, and greater understanding of the limits of the model. It is obvious that a large number of data sets representing a variety of conditions and parameters is necessary for this process.

Several applications of the model are included to indicate the potential usage of the numerical approach. It is believed that the model has great potential for many areas of investigation.

4.1 COMPUTED WATER LEVELS

The computed water level, η, in the East Passage is rather easily checked against the tide as measured by the U.S. Navy at the Newport tide gauge (Fig. 4). The data for several days in March, 1972, was obtained (18) and checked against the computed water level (Fig. 16). The Newport tide is plotted as the deviation from the mean of the two days modeled. The mean was 451 cm, or 69 cm above the datum (mean low water). This is about 19 cm larger than the historical mean; the difference is probably due to a number of rainstorms which occurred in that week. These storms may also account for the small variations between the computed and observed tides.

The tides for this period were also checked against the historical tides at Newport, Bristol, and Providence (see Section 2.5), which are generated by a series of the form of Eq. 2.5.1. The results (Fig. 16) show that the computed curves are very similar to the historical, especially at the Newport station. The computed tide at the Bristol and Providence stations is somewhat larger than the historical, although like the Newport curves, they are very close in phase. The differences are likely due to inadequate representation of friction, and in the fact that these two stations are located in areas of the Bay with locally complicated geometry.

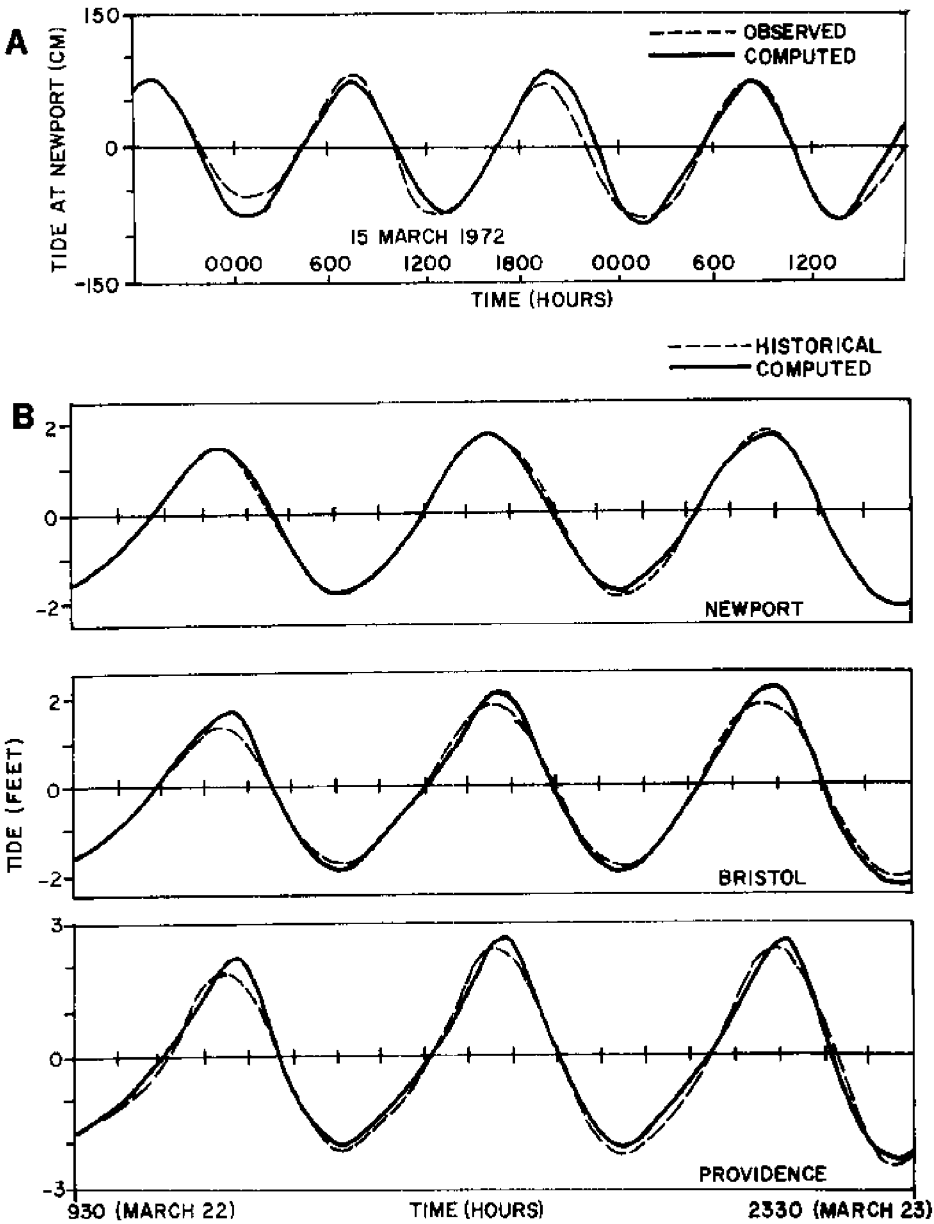


Fig. 16. Comparison of observed tide at the Newport gauge and that computed by the model (A); and comparison of historical and computed tide at the Newport (top), Bristol (middle), and Providence (bottom) stations (B).

4.2 COMPUTED VELOCITIES AND FLOWRATES

A number of current velocity observations were used to verify the model. These include near-surface and near-bottom current meter data, and drifting pole data. The observations were in the lower bay in the East and West Passages.

Currents in the West Passage were studied by Sturges and Weisberg (19), who used a string of Savonious rotor-type meters anchored to the bottom. The model-predicted velocity (averaged at n = 8, m = 36 and 37) is plotted against the surface and bottom currents for the period studied (Fig. 17a). (The observed values are two meters above the bottom and two meters below the mean surface). The model velocities seem to follow the phase of the near-bottom current and the amplitude of the near-surface current. The phase difference between the two is probably due to the effects of viscosity (see Lamb (20)). In several cases, the near-surface velocity is 50 percent greater than the near-bottom velocity, a fact which must be taken into account when reducing data from pole-type current measurements. The numerous small-scale variations may be due to wind effects, which were not included in this model run.

Another type of velocity observation was made with drifting poles (15 and 21) in connection with a geomagnetic electrokinetograph (GEK) feasibility study (22).

In that investigation, several poles were allowed to drift with the current under the Jamestown Bridge, and the total rate of flow was calculated (21). The computed flowrate (Fig. 17b) is generally less than the observed; the relationship between the pole velocity and the average velocity over the entire vertical section is difficult to assess, since the velocity varies with the depth. A Marine Research, Inc. study (23) established that the average velocity over the whole depth (65 feet) was about 93 percent of the velocity measured in the top 45 feet, during the ebb. At present, little is known about the variations during the flood.

The last series of observations (18) were made by E. Levine in the East Passage with a Savonious rotor-type meter, mounted at eight feet from the bottom in 42 feet of water near the Newport tide gauge (Fig. 14). The computed velocity (average at $n = 14$, $m = 38$ and 39) compares favorably with the observed (Fig. 17c), although not quite as well as in the West Passage. The computed ebb is again greater than that observed, but the computed flood is less. The first flood is larger than the first, a feature not seen in the bottom current.

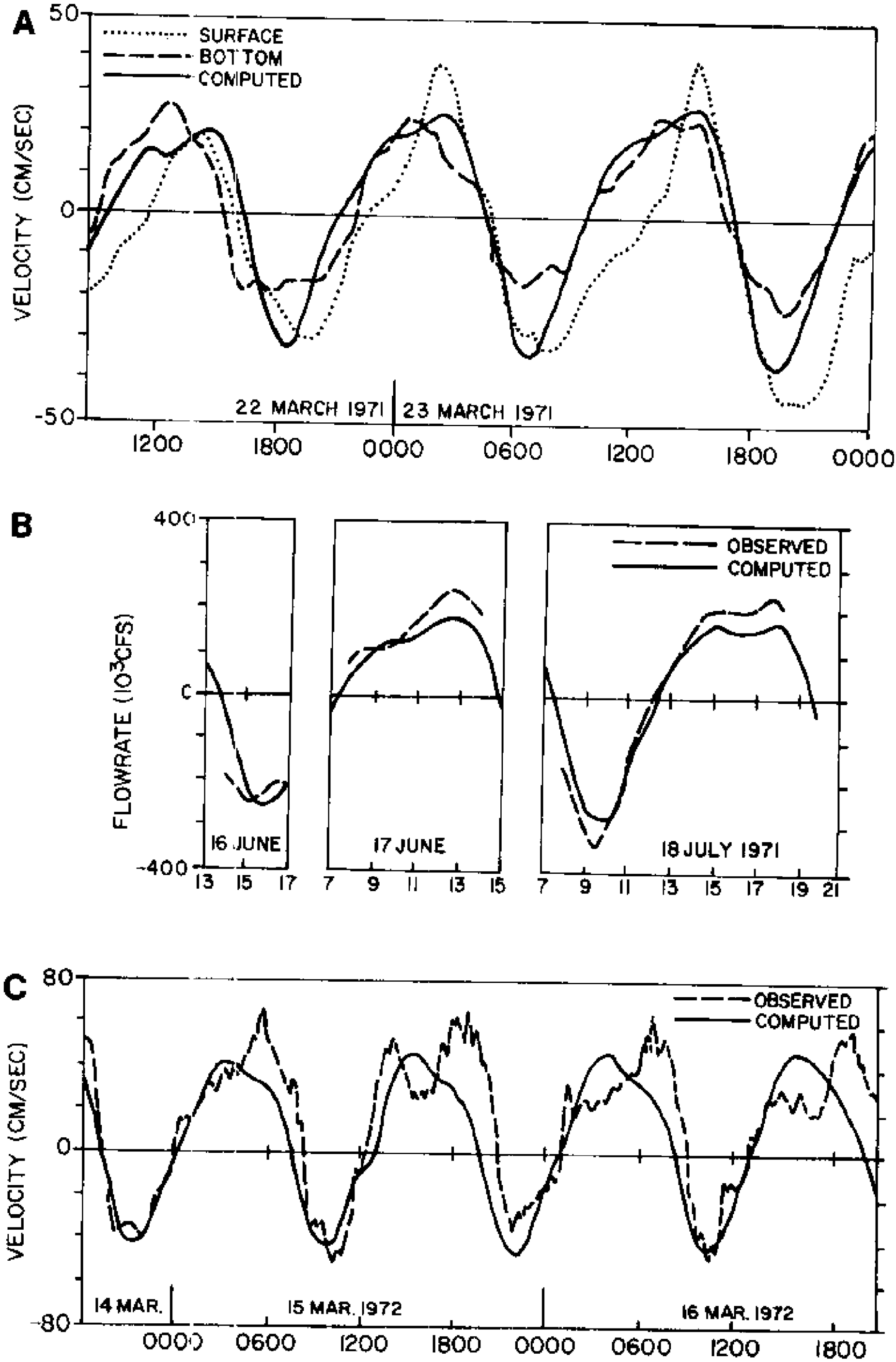


Fig. 17. Comparison of observed and predicted current velocity in the west passage (A), flowrate at the Jamestown Bridge (B), and current velocity in the east passage (C).

4.3 APPLICATION: NON-TIDAL FLOW

The numerical model provides an unique opportunity to study mean flow patterns of river discharge. In this study, a constant flowrate of 1000 c.f.s. was introduced at the Providence River; other river inputs and tidal variations were suppressed. The resultant current vectors (Fig. 18a) indicate that the Coriolis acceleration is important in determining the direction of the flow. The current tends toward the rightward shore in the narrow passages. Of particular interest is the counter clockwise circulation in Greenwich Bay, and the circulation around Hog Island near Bristol Harbor.

The total net flow past several sections was also calculated (Fig. 18b). About two-thirds of the water moved rightward from the Providence River into the upper West Passage, apparently under the influence of the Coriolis acceleration. A sizable fraction, however, flowed back into the East Passage just south of Prudence Island; as a consequence, the net flow out of the bay was greater in the East Passage.

The proportions of the net flow entering each channel compare favorably with those obtained by Hicks (16), when the Taunton River contribution is eliminated. His estimate of 74 percent leaving the East Passage is higher than both the value obtained in this study (60 percent)

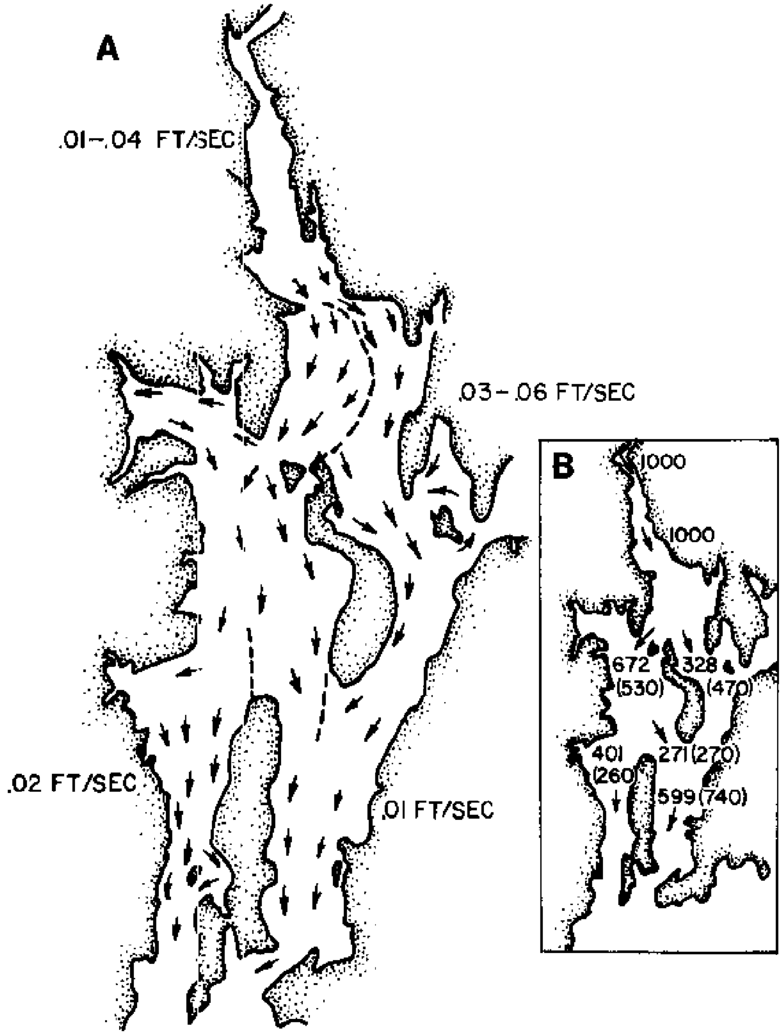


Fig. 18. Predicted non-tidal current vectors for a Providence River discharge of 100 c.f.s. Inset: predicted flowrates (c.f.s.) through each passage (those estimated by Hicks are shown in parentheses).

and the value from the tidal flowrate study (following section) of 71 percent.

A simulation involving only Taunton River discharge proved to be unstable; water entering eastward at the Mt. Hope Bridge tends northward due to Coriolis force, but must eventually turn southward to leave the Bay. The computed solution indicated that neither of these tendencies were dominant, so that a steady flow regime was not established.

4.4 APPLICATION: EAST AND WEST PASSAGE FLOWRATES

Another rather simple model task is the estimation of flowrates past any section in the Bay. This type of information is useful for many application; a biological model of a segment of the estuary is one example.

Flowrates are computed at each time step by an equation similar to 3.5.1 for the Jamestown Bridge. The Newport section in the East Passage was taken at m = 38, n = 12, 13, and 14. The maximum rate of flow in the East Passage was found to be about 2.4 times the West Passage flow during both ebb and flood (Fig. 19). The total flow in the flood and ebb portions (flowrate integrated over time between successive slack waters) was greater in the East Passage by a factor of 2.37 in the ebb and 2.48 in the flood. The average volume entering and leaving in each tidal cycle was 13.97 billion cubic feet.

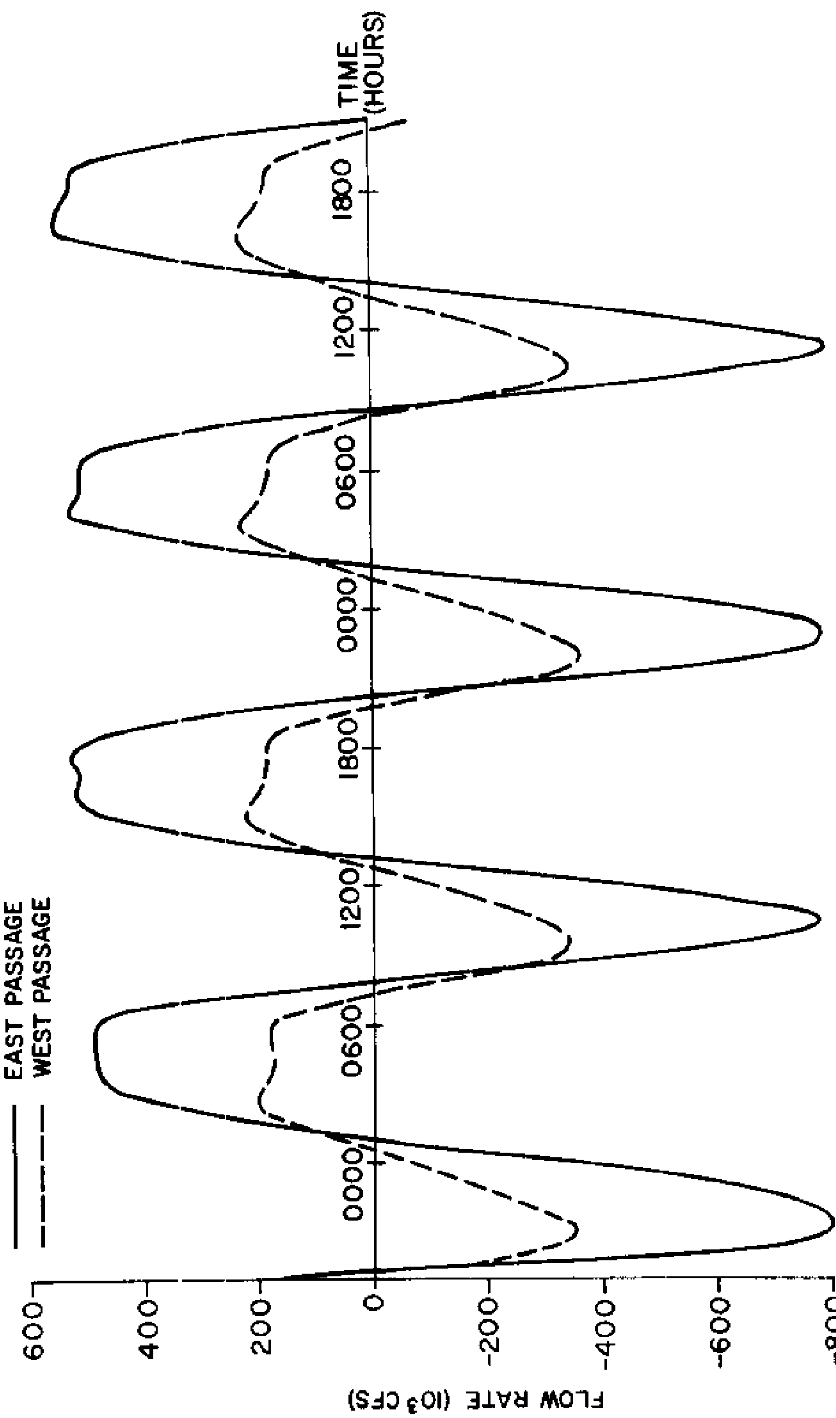


Fig. 19. Predicted flowrates through the lower east and west passages for a period beginning at 1900 E.S.T., March 15, 1972.

This is probably due to the nearly uniform range of the tide at this time (Fig. 16). As noted before, the computed results show that 71 percent of the total volume of water entering and leaving Narragansett Bay during a tidal cycle makes it through the East Passage.

4.5 APPLICATION: CURRENT VECTORS AND TIDAL CO-RANGE LINES

The numerical solution of the tide and current is readily available for inspection at specific locations as well as for the entire bay. The area chosen was the West Passage adjacent to Wickford Harbor (Fig. 1) which has interest because of a proposed nuclear power plant at nearby Rome Point.

Local tide and currents were taken from a simulation of the time around the first high water. Before H.W., the tide is increasing up the Bay, being 1.35 feet at the mouth and increasing to 1.70 feet north of Conanicut Island. (Fig. 20). The current is in the flood stage and is northward (into the Bay).

At four minutes after H.W., the tide is 1.39 feet at the mouth and ebbing. The tide north of Conanicut Island is now about 1.80 feet. The current is now beginning to ebb, except in the deep center section of the West Passage, where it is still in the flood stage. This is consistent with the results of Jones (21) and Krabach (22),

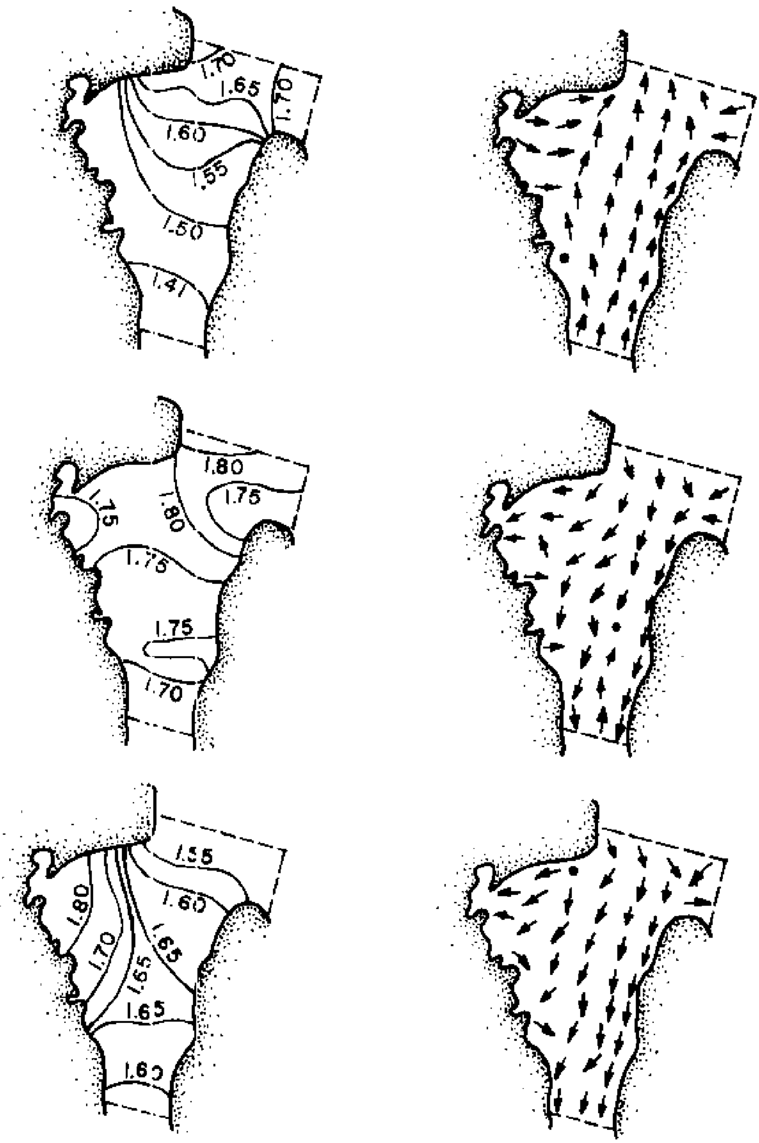


Fig. 20. Predicted co-range lines (left column), and current vectors (right column) in a portion of the west passage adjacent to Wickford Harbor. The times relative to Newport high water are 28 minutes before (top), 4 minutes after (middle), and 32 minutes after (bottom).

who found the near-shore shallow water reversing sooner than the central deep water.

Thirty-two minutes after H.W. the current is uniformly ebbing. The tide is interesting because it is now near its maximum at Wickford Harbor, quite a bit later than the Newport H.W. It is possible that Coriolis acceleration causes water to pile up in the harbor in the presence of the southward flowing current.

Current vectors for November 8, 1970, in the same region (Fig. 21) shows an even more complex pattern. The current is flooding near the shore, but ebbing in the center of the channel. A clockwise circulation near the harbor is evident, lending support to the observed eddy structure proposed by Polgar (24) and Marine Research, Inc. (23).

4.6 APPLICATION: HURRICANE SURGE

One of the more interesting applications of the tidal model is the simulation of hurricane surge. The devastation caused by the hurricane of 1938 has initiated extensive study of Narragansett Bay physical oceanography, primarily for the effects of a proposed hurricane barrier project (25). Wind setup has also been studied at Narragansett Pier (26).

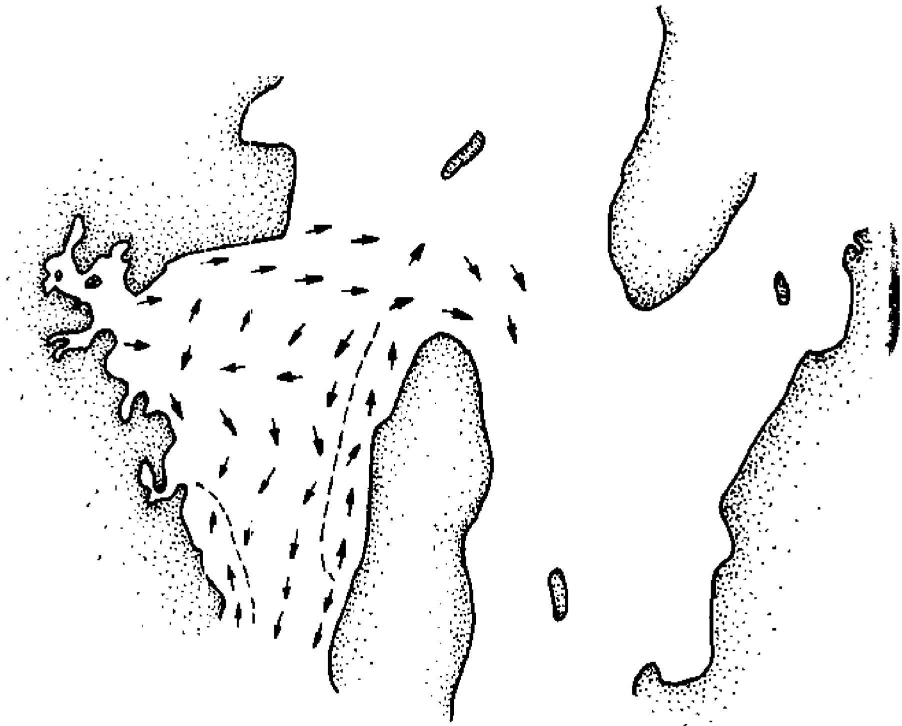


Fig. 21. Predicted current vectors in a portion of the west passage adjacent to Wickford Harbor at 70 minutes before low water at Newport. Note reverse flow in the shallow near-shore water, and clockwise motion east of Wickford Harbor.

For the purposes of this preliminary investigation, a hurricane will be modeled from available data on the storm of September 14, 15, 1944 (27, 28, 29). That storm, with maximum winds of 90 mph, moved northeast over the Atlantic coast and passed directly over the Bay, causing tides 9.9 feet higher than normal at Providence. The storm had a radius of about 200 n.m., and traveled at an average rate of 30 kts. The observed high water occurred as the storm crossed the Bay.

According to Bodine (30), the total surge at an open coast has several components. That is

$$S_T = S_x + S_y + S_{\Delta p} + S_w \qquad 4.6.1$$

where the total surge above the tidal effects, S_T, is the sum of the x- and y-components of wind setup, (S_x, S_y), the atmospheric pressure setup ($S_{\Delta p}$), and the breaking wave setup (S_w). In addition, the local wind effects over the Bay will contribute to the total surge at any location in the Bay. The model may be used to predict water levels if the surge at the mouth and the wind distribution over the Bay are given as input.

The total problem is quite complex, due to the time-dependence of the inputs, and the effects of the continental shelf on the surge. For this reason, several assumptions will be made in the formulation of the inputs, and these will be mentioned in the development. It should also be noted that this study does not attempt to be the final word in surge modeling, but an engineering approach with emphasis on obtaining a practical solution. The results show that many of the simplifications are realistic.

The wind distribution over a section of the hurricane parallel to the direction of propagation and through the location of maximum wind speed was taken from data on the storm given by Wilson (27). The analytic expression

$$W_H = 90 \exp\left\{-0.2|T - T_{eye}|\right\} \qquad 4.6.2$$

where T_{eye} is the time the eye of the storm intersects the coastline, is a good approximation to the wind at 1800 and 2200 E.S.T. on September 14, (the storm passed directly over the Bay at about 2340 EST) (Fig. 22a).

The wind direction changes as the storm approaches the coast, blowing first to the west, then swinging around to the east as the storm passes. For the coordinate system with X_H northward and Y_H westward (Fig. 23) the wind direction, θ_1, from the x-axis was taken to be

$$\theta_1 = 20° + C_H \cdot 90°\left[1-\exp(-.023|T-T_{eye}|V_p\right]-A_{Hurr} \qquad 4.6.3$$

where

$$C_H = 1, \ T < T_{eye}$$

$$= -1, \ T > T_{eye},$$

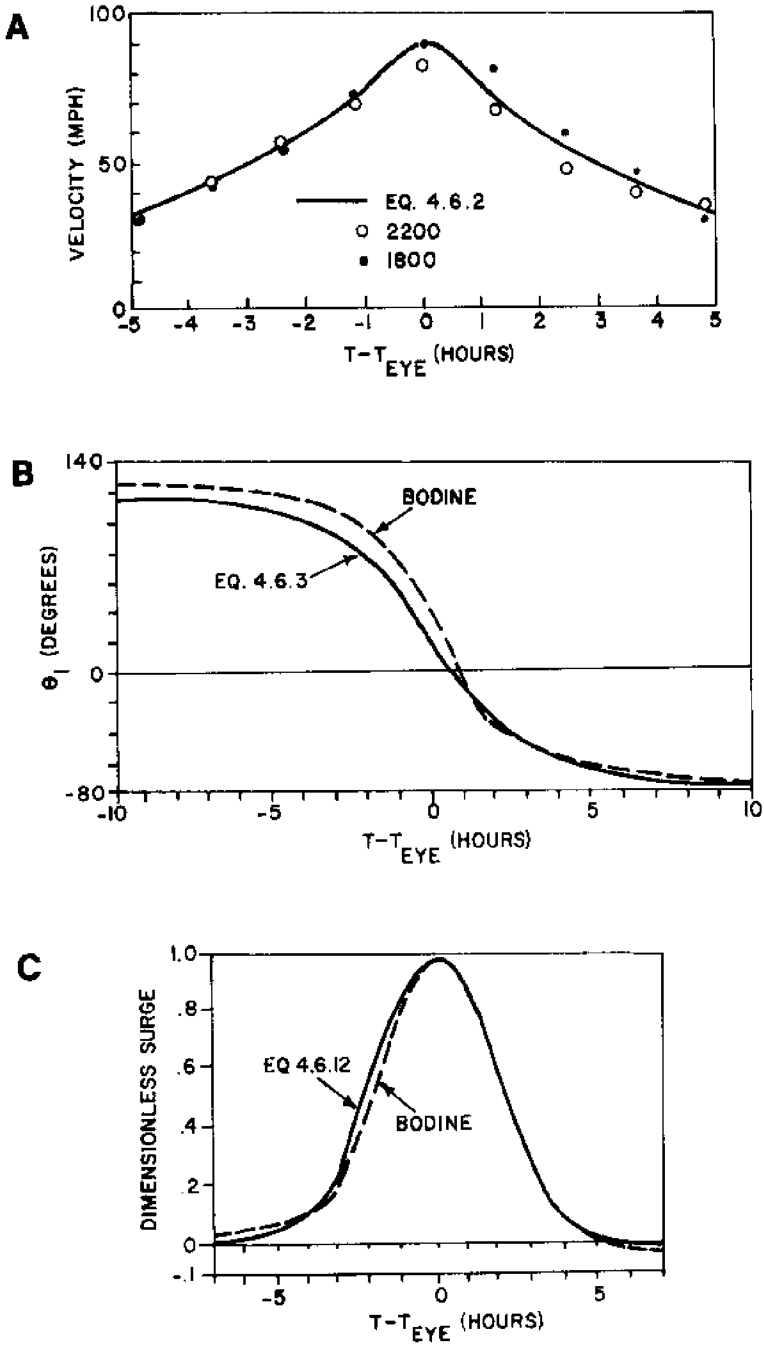


Fig. 22. Comparison of data and analytic representation for hurricane wind speed (A), wind direction (B), and coastal surge (C).

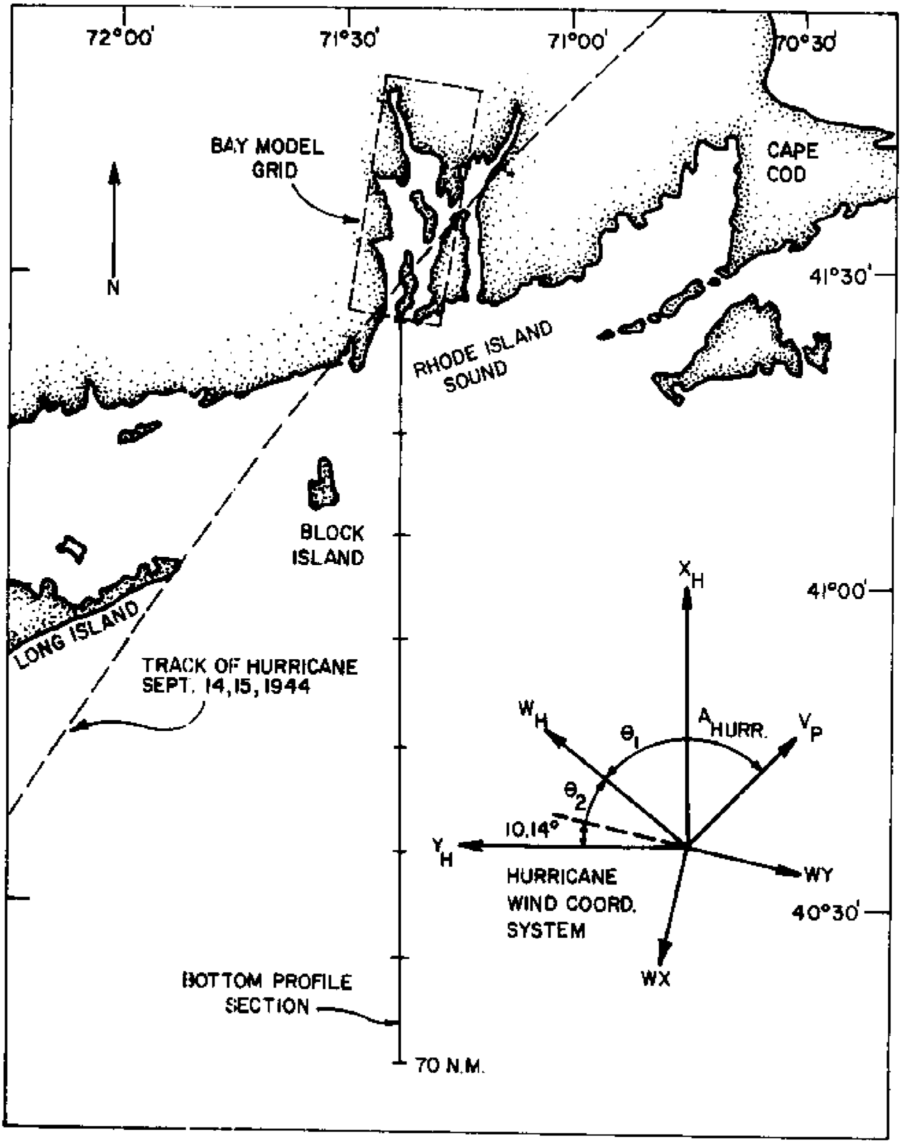


Fig. 23. Geography of offshore region used in hurricane study and coordinate system for approaching storm.

A_{Hurr} is the angle to the right of north at which the storm approaches and V_p is the propagation speed of the hurricane. This expression was obtained from data used by Bodine (30) (Fig. 22b). Now the coordinate system for the Bay is inclined about 10.14° to the right of North and the x- and y-directions in the model will then be

$$WX = -W_H \sin \theta_2 \qquad 4.6.4$$

$$WY = -W_H \cos \theta_2 \qquad 4.6.5$$

where $$\theta_2 = 90° - 10.14° - \theta_1 \qquad 4.6.6$$

The surge at the coast due to wind setup is a dynamic problem extensively treated by Bodine (28). The perpendicular components, S_x, will be calculated in a simple fashion. The surge due to a constant wind of magnitude U is given by Ippen (17) as

$$S_x = g \frac{K U^2 \ell}{(h_1 - h - S_x)} \ln \left(\frac{h_1}{h + S_x}\right) \qquad 4.6.7$$

(Fig. 24a) for the case of depth increasing linearly from h at the coast to h_1 at a distance ℓ. The coefficient, k, is usually taken as 3.0×10^{-6}. For variable wind speed, U^2 may be replaced (17) by its equivalent:

$$U_e^2 = \frac{1}{\ell} \int_o |U| U_x d_x \qquad 4.6.8$$

The integral is evaluated in Ref. 31. For the region just off Narragansett Bay, ℓ = 70 n.m., h = 60 ft, h_1 = 270 ft (Fig. 24b) and for $|U|U_x$ maximum of (90 mph)^2, we have

$$S_x = \frac{320}{(210 - S_x)} \ln\left(\frac{270}{60 + S_x}\right) \qquad 4.6.9$$

for which S_x = 2.23 feet.

The y-component, S_y, is more difficult to calculate directly since it depends on the local geometry and Coriolis effect on x-velocities. However, the numerical study of Bodine (30) gave a value of S_y = 52% of S_x, for a storm approaching perpendicularly to the coast. In another study, (17), the Coriolis component was found to be 65%. For our study, the former value was used. Thus

$$S_x + S_y = 1.52\ S_x \qquad 4.6.10$$

The contribution due to a decrease in atmospheric pressure is given by Bodine (30) as

$$S_{\Delta p} = 1.14\ \Delta P\ (1 - e^{-R/r})\ \text{ft} \qquad 4.6.11$$

where ΔP is the pressure difference between ambient and the minimum, in inches of mercury, R the radius of maximum winds, and r the distance from storm center to the point of calculation of the surge. For the 1944 storm, ΔP = 1.34", R = 30 n.m., and r = 35 n.m. (27) giving a value of $S_{\Delta p}$ of .89 feet.

The breaking wave setup is the smallest component, and has been neglected here. The maximum surge experienced

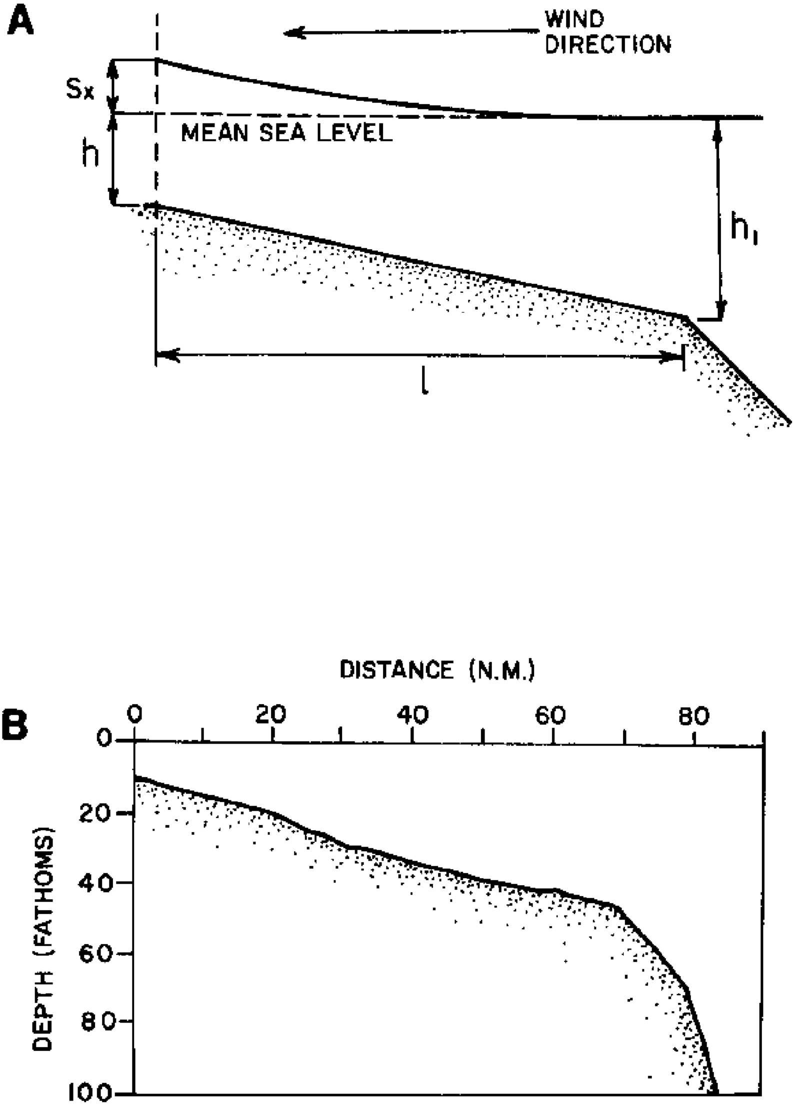


Fig. 24. Definition sketch of coastal surge (A), and bathymetry of a section due south of Narragansett Bay (B).

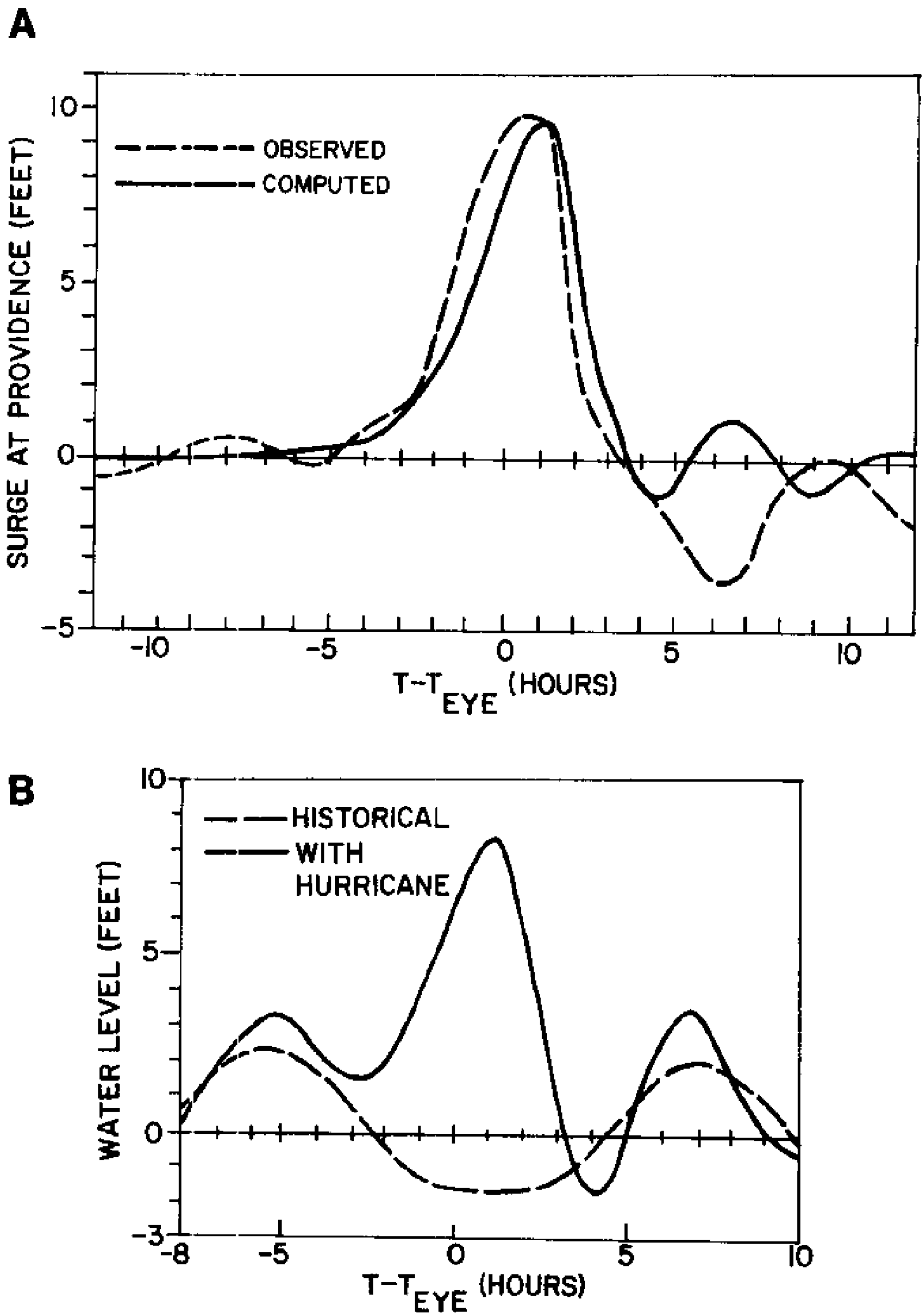


Fig. 25. Comparison of predicted and observed hurricane surge at Providence (A), and comparison of historical (without hurricane) and computed (with hurricane) water level at Providence Harbor (B).

at the mouth of the Bay is then $1.52 \times 2.23 + .89 = 4.28$ feet.

The surge variation with time is taken from the computed results of Bodine, and is approximated by

$$S_T = S_{T,\ max} \exp(-.148 \left[T - T_{eye}\right]^2) \qquad 4.6.12$$

(see Fig. 22c).

The wind distribution and coastal surge were used in the modeling of the 1944 storm. The surge computed at Providence Harbor is seen in Fig. 25a. The peaks of each curve are very close, within about one-fourth of a foot, although the computed peak is about one hour after the observed. This is probably due to the difficulty of determining the exact time the storm passed. The computed water level and the historical tide are shown in Fig. 25b; the peak surge occurred near low water, thus saving Providence from a flood stage 12.5 feet above mean sea level.

The surges computed at Newport, Bristol, and Providence are seen in Fig. 26; the maximum surge in the upper Bay is approximately five feet greater than in the lower Bay, indicating the effects of wind and bay geometry. The surge at one hour after T_{eye} (Fig. 27) also shows this. Note that the higher water levels occur on the west shore of the Bay.

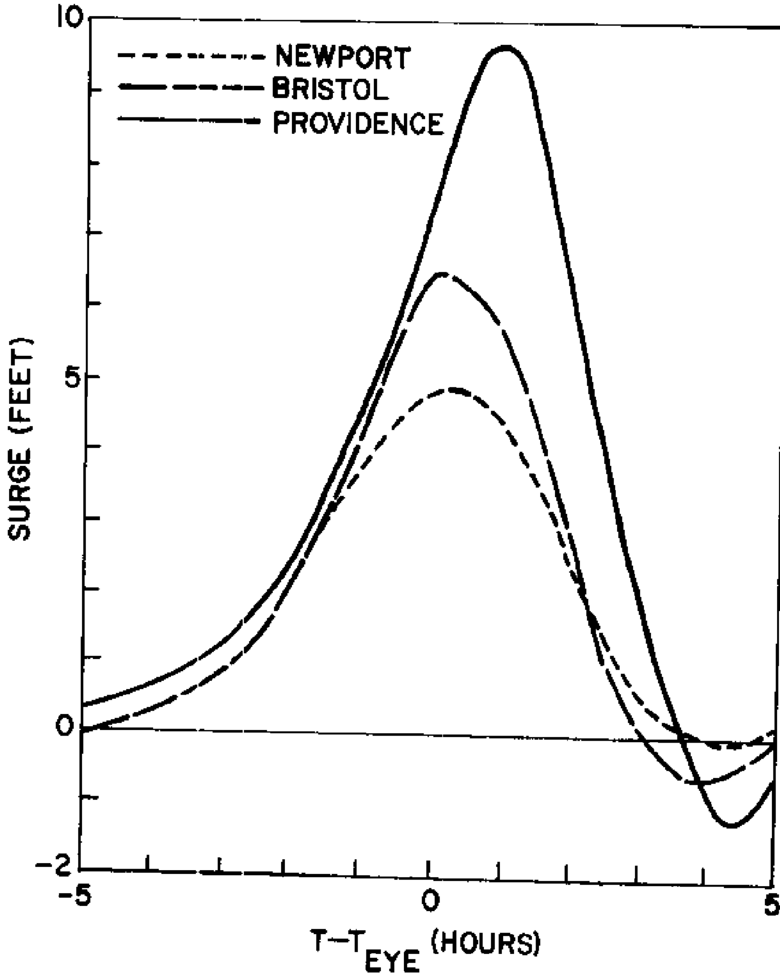


Fig. 26. Comparison of computed surge at the Newport, Bristol, and Providence tide stations.

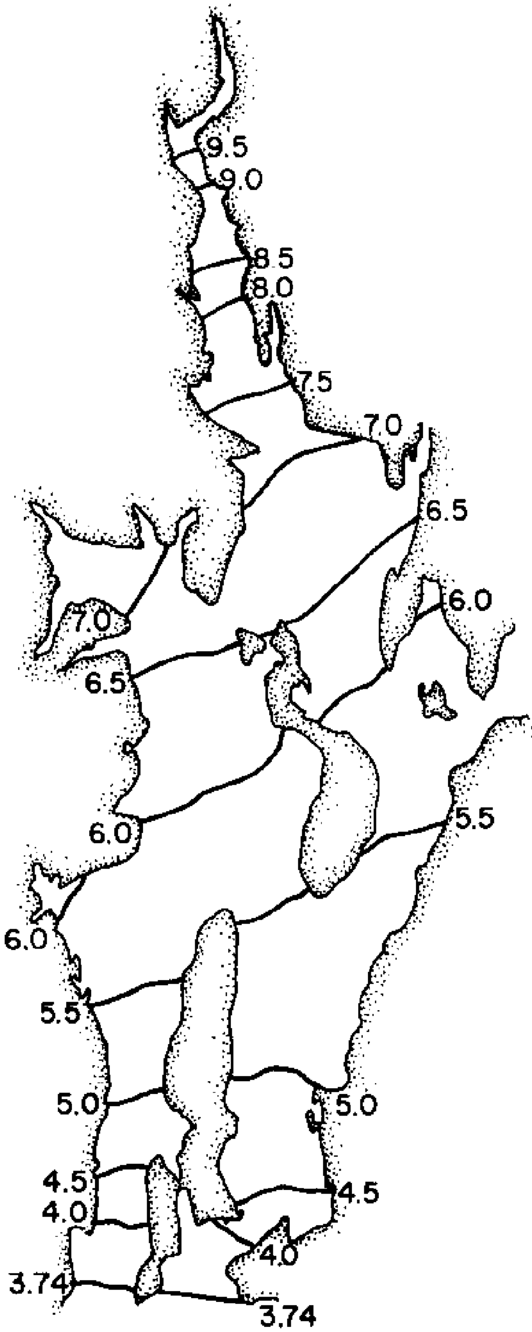


Fig. 27. Water-level isometry (in feet) in Narragansett Bay for simulated hurricane at the time of maximum surge at Providence Harbor.

V. OUTLINE OF THE MODEL PROGRAM

5.0 INTRODUCTION

The main program, run in Fortran IV, on the University's IBM 360/50 digital computer, consists of two basic phases. In the first phase, the initial data is read in, and all parameters and constants are initialized. A section imbedded in this phase sets all boundary conditions each time they are called. The second phase consists of the multistep operations which compute the water levels and two velocity components. These operations are repeated in a cyclical fashion to obtain a time-history.

At every time step, five distinct sections of phase two are entered. In the first half time step, U and S are claculated implicitly for each gridline parallel to the Y-axis. The printing section is entered, and the process continues to the second half time step, where V and S are calculated, then U. Each of the sections is described below.

5.1 INITIALIZATION

In this section of the program, the physical body of water is described in terms of the computational scheme. All arrays are dimensioned, and several are put in common storage for the subroutines. The arrays A,B,P,Q,R, and S are used in the matrix inversion portion of the implicit computations. The array F stores values of the Coriolis parameter, herein taken as a constant. KONVRT is used in the printing section. NH is used in the first phase to write initial depth values. NPRINT stores the steps at which a printout is desired, and DAVG is used in the calculation of average depths.

Arrays in common are SE, SEP, U, UP, and VP, the water levels and velocities at two time levels. C denotes Chezy coefficients, and H the depths. NBD and MBD are the grid identification arrays, and NOBD and MOBD store information on the open boundaries. The arrays W, F2, Z, E, HP, EP, HB, EB, ARN, ARGP, ARGB, ARGLB, HL, and EL Store tidal data. ZIA, ZIB, ZIC, ZID, ZIE, YIA, YIB, are utility arrays which store computational parameters at each timestep. UAVG and VAVG store average velocities, and IFIELD stores the computational field.

The next group of statements set the initial values of several parameters such as grid dimensions, time step, and certain boundary conditions. These will be explained in more detail later.

After the constants of the difference equation, C_i, are set, all arrays describing the physical system are set at zero. Then the subroutines KURIH, DIVE, FIND, DEPTH, CHEZY, and CHECK, are called, and the relevant data read in. These subroutines are described elsewhere in this paper. Values of NPRINT are read.

Initial values of velocity and water level are then set. A starting value of tide can be specified with the parameters HINV and SEINV. Initial values of velocity and water level can be introduced by reading in the appropriate matrices.

Finally, the initial values are printed out before computation.

5.2 BOUNDARY CONDITIONS

The statements about the boundaries are included in the first phase for accessibility to the user. The velocities at the Providence and Pawtuxet Rivers are introduced after the model time is calculated. Then the velocity at the Mt. Hope Bridge is specified as in the method of Section 2.7.

Several conditions at the mouth of the Bay can be specified, using the parameter IMODE 1. For the tidal input, the water levels at both the most eastward and westward grids are computed; the tide in the other grids is determined by linear interpolation.

5.3 THE IMPLICIT OPERATIONS

The first portion of each half time step, one velocity component (U in the first half, V in the second) and the water level are solved implicitly. The method is that described in Section 1.3.

In the first half time step, U and SE are solved from equations A.1 and A.2. Choosing the first element in NBD, the program identifies the gridline parallel to the x-axis. The R and S are initialized, depending on the lower bound. The process continues, marching up the gridline, calculating A, P, Q. B, R and S for each grid, until the upper bound is reached. Marching backward down the gridline, UP and SEP are alternately computed, by the method of Section 1.3. The next gridline is identified, and the operation continues until all gridlines have been traversed. The explicit operations are then begun.

The implicit operation in the second half time step is analogous to the first, except that VP and SEP are computed along gridlines parallel to the y-axis.

5.4 THE EXPLICIT OPERATIONS

After the water level and first velocity component have been computed, the other velocity component is computed explicitly by equations A.3 or A.6. Gridline selection proceeds as before, but the direction is perpendicular to that used in the preceding implicit operation. The velocity is computed at each grid, without storage of any other parameters, since the operation is explicit.

5.5 THE PRINTING OPERATION

Between the first and second half time step, the printing section is entered. A check is made on the print-out stepnumbers (containing in NPRINT), and if specified, the matrices containing SE, U, and V are printed or punched. Then regardless of printing, the lower level values (SE, U, V) are replaced by the newer ones (SEP, UP, VP), and the next implicit operation is entered.

Punched card output, if desired, is also produced in this section of the program.

VI. THE SUBROUTINES

6.0 INTRODUCTION

In this section the subroutines are outlined. The subroutines DIVE and FIND are essential to the formulation of gridline information, and will be practically identical for all program uses. The subroutines DEPTH and CHEZY may be altered to reflect the geography of the water body being modeled. In our model, the depths are read in on cards and stored for use, but it is possible that analytic expressions can be used to calculate depths, especially read in on cards, or calculated analytically from the depths as is done in our program. The surbroutine KURIH may be used to calculate the tide analytically as is done here, or to read in tabulated values when no Coast and Geodetic Survey tidal constants are available. Subroutine CHECK coefficients are specified at all computational grid squares. ANALYZE prints out a comparison between historical and computed tides at three stations around the Bay.

6.1 SUBROUTINE KURIH

This subroutine calculates, given the minute, hour, day, and year of the model starting time, the height of the tide at Newport, Providence, Bristol, and the lower boundary. The general approach is to calculate the value of the equilibrium arguments at the beginning of the year being modeled, and then the arguments at the start of the model time. The data for Rhode Island locations (10) is assumed to be accurate, and the effect of wind, rainstorms, etc., is not taken into account.

The equation for the height of the tide above some reference place is given by

$$h(t) = H_o + \sum f_n(t)\, H_n \cos\left[W_n t + (V_o + u) - K_n\right] \qquad 6.1.1$$

(see Section 2.5 for definitions of the parameters). The epoch of the constituent, K_n, is usually combined with a longitude and time correlation (relative to Greenwich) so that the constituent argument is given by

$$W_n t + \text{Greenwich } (V_o + u)_n - K'_n$$

Thus with H_o, and K'_n available in the tidal data sheets, (10) and W_n tabulated (12), it remains to determine the node factor f(n), and Greenwich $(V_o + u)$ for any time. The method is given in Schureman, Ref. 12. The equilibrium argument at Greenwich $(V_o + u)$ can be expanded into the sum of several astronomical angles.

$$V_o = C_\tau \cdot T + C_s \cdot S + C_h \cdot h + C_p \cdot P + C_{p_1} \cdot p_1 + C_\pi \cdot \frac{\pi}{2} \qquad 6.1.2$$

$$u = C\xi \cdot \xi + C_{\nu} \cdot \nu + C_R \cdot R \qquad 6.1.3$$

where T = hour angle of the sun

S = mean longitude of the moon

h = mean longitude of the sun

p = longitude of lunar perigee

p_1 = longitude of solar perigee

For definition of the angles ξ and ν, refer to Schurman, (12), Fig. 1. R is an augmenting angle of only the L_2 constituent argument. The associated coefficients, C, take on the values 0, $\pm$ 1, $\pm$ 2, $\pm$ 3, $\pm$ 4, $\pm$ 6, depending upon the particular constituent.

Several of the terms in 6.1.2 and 6.1.3 can be determined by evaluation of a series

$$S = 270.454 + (1336 \text{ rev.} + 307.892T) + .00252T^2 \qquad 6.1.4$$

$$h = 279.697 + (100 \text{ rev.} + 0.7690T) + .00030T^2 \qquad 6.1.5$$

$$p = 334.328 + (11 \text{ rev.} + 109.032T) - 0.010344T^2 \qquad 6.1.6$$

$$p_1 = 281.221 + 1.719T + .00045T^2 \qquad 6.1.7$$

where rev. is the number of revolutions (at 360° per year) and T the time, in Julian centuries (36525.0 days), reckoned from the reference time (12.00 noon on December 31, 1899 by the Gregorian calendar).

In addition, the longitude of the moon's node N, which appears in subsequent calculations, can be represented by

$$N = 259.182 - (31.41593 \text{ rev.} + 134.142T) + .0021T^2 \qquad 6.1.8$$

From this information, the following angles can be computed:

I = Inclination of moon's orbit to plane of earth's equator.

p = Mean longitude of lunar perigee reckoned from the lunar intersection.

by the equations:

$$\cos(I) = .91370 - 0.03569 \cos(N) \qquad 6.1.9$$

$$\xi = N - \arctan(1.01883 \tan(N/2)) - \arctan(0.64412 \tan(N/2)) \qquad 6.1.10$$

$$\nu = \arctan(1.01883 \tan(N/2)) - \arctan(0.64412 \tan(N/2)) \qquad 6.1.11$$

$$P = p - \xi \qquad 6.1.12$$

$$1/R_a{}^2 = 1. - 12 \tan^2(I/2) \cos(2P) + 36 \tan^4(I/2)) \qquad 6.1.13$$

$$R = \arctan\left(\frac{\sin(2P)}{\frac{\cot an^2(I/2)}{6} - \cos(2P)}\right) \qquad 6.1.14$$

where R_a appears only in the L_2 constituent amplitude. Furthermore, for the constituent K_1,

$$\nu' = \arctan\left(\frac{\sin(2I)\sin(2\nu)}{\sin^2(I)\cos(2\nu) + 0.0727}\right) \qquad 6.1.15$$

and for constituent K_2,

$$\nu'' = \frac{1}{2}\arctan\left(\frac{\sin^2(I)\sin(2\nu)}{\sin^2(I)\cos(2\nu) + 0.0727}\right) \qquad 6.1.16$$

It should be noted that for any particular year, the angles composing V_o are computed for the beginning of that year (hour 0, January 1), which the angles composing u are for the middle of the year (hour 12 on July 1, or hour 0 on July 2 in leap years). For this reason, the equilibrium angle at Greenwich is first computed for hour 0, January 1, and then advanced to the day, hour, and minute, using the angular velocity of the constituent.

After the constituent arguments the node factors are calculated by one of the following node formulae:

$$f^{(1)} = 1.000 \qquad 6.1.17$$

$$f^{(2)} = \cos^4(I/2)/0.9154 \qquad 6.1.18$$

$$f^{(3)} = (f^{(2)})^2 \qquad 6.1.19$$

$$f^{(4)} = (f^{(2)})^3 \qquad 6.1.20$$

$$f^{(5)} = f^{(2)}/R_a \qquad 6.1.21$$

$$f^{(6)} = \frac{\sin(I)\cos^2(I/2)}{0.3800} \qquad 6.1.22$$

$$f^{(7)} = (.8965 \sin^2(2I) + .6001 \sin(2I) \cdot \cos(\nu) + 0.1006)^{1/2}$$

$$f^{(8)} = (19.044 \sin^4(I) + 2.7702 \sin^2(I) \cdot \cos(2\nu) + 0.0981)^{1/2}$$

where I, R_a, and ν are calculated at the nearest day, (the node factors vary only slightly over the year).

The program proceeds in the following order.

1. The year, day, hour, and minute of model time, as well as the number of tidal constituents being empolyed, are read in.
2. The names, then the amplitudes and epoches for Newport, Providence, Bristol, and the mouth, of each constituent are read in.
3. The values of the coefficients, C, and the node formulae numbers are read in.
4. The number of Julian days to both the beginning and middle of the year is determined, then converted to Julian centuries.
5. The values of N, h, p, p_1, s, I, ξ, ν, P, R_a^{-1}, and R are determined for both times.
6. The equilibrium arguments are calculated, then advanced to the model time.

7. N, p, I, ξ, ν, P, R_a^{-1}, and the node factors are computed for the day and year.
8. The tide is computed for Newport, Providence, and Bristol, and a graphical display printed.

At the end of the subroutine, the approximate time of high water at Newport is determined from the M_2 constituent. This value, TS, relative to the model starting time, is used in the Mt. Hope boundary condition.

6.2 SUBROUTINE DIVE

In this subroutine, the computation field is simply read in, and stored in the matrix IFIELD for subsequent use. At computational grids, IFIELD has a value of 1; at water boundary grids it has a value of 2. At land grids, it is 0.

6.3 SUBROUTINE FIND

Here the field of computation is examined, and the gridlines in both the x- and y-directions are flagged. The vector NBD consists of integer elements, which contain information on the gridlines parallel to the x-axis, is similar.

The elements have the general form

ab/cd/ef/gh

where ab indicates the type of boundary, cd the column (NBD) or row (MBD) of the gridline, and ef and gh are the lower and upper computation grid numbers.

The boundary code is

ab	=	00	solid boundaries
ab	=	20	lower boundary velocity
ab	=	10	lower bound tide height
ab	=	2	upper bound velocity
ab	=	1	lower bound tide height

6.4 SUBROUTINE DEPTH

This simply reads in the initial mean low water depths, in feet, then adds a linear mean tide, and converts to yards.

6.5 SUBROUTINE CHEZY

This subroutine scans each grid square, and if at least one depth is non-zero, calculates the Chezy coefficient by the relationship

$$C = \frac{1.49}{N} (H)^{1/6}$$

A Manning factor, N, is determined for each grid from Eq. 2.4.1.

6.6 SUBROUTINE ANALYZE

This is used to compare series and computed values of the tide at Newport, Bristol, or Providence. The stored value is displayed numerically and graphically alongside the series value. The subroutine can be generalized to store and print any quantity at the end of the run.

6.7 SUBROUTINE CHECK

When boundary grids are altered, there exists the possibility that a depth or Chezy value will be omitted in the data set. This subroutine checks that all parameters are specified before computation, and indicates which values may be missing in the diagnostic printout.

VII. PROGRAM USER'S GUIDE

7.0 SYSTEM DIMENSIONS

All arrays in the COMMON and DIMENSION statements must be given suitable storage. Values must be assigned the following:

NMAX = Maximum grid size in the y-direction (Fig. 3) which will require changes in format of output if it exceeds 32.

MMAX = Maximum grid size in the x-direction not to exceed 99.

```
DIMENSION A(  ), B(  ), P(  ), Q(  ), R(  ), S(  ),
F(  ), KONVRT(D), NH(  ), NPRINT(  ), DAVG(  )
```

The vectors A, B, P, Q, R, S are used in the implicit computation, and should have the dimension equal to MMAX or NMAX, whichever is larger. Similarly, with F, KONVRT and NH have the dimension NMAX. NPRINT is arbitrary, and DAVG should have the dimension MMAX.

```
COMMON SE(  ), SEP(  ), V(  ), VP(  ), U(  ), UP(  ),
C(  ), H(  ), VAVG(  ), UAVG(  ), IFIELD(  ).
```

These are two-dimensional arrays, with the general dimensions

SE(NMAX, MMAX)

NBD, ( ) MBD ( ) These vectors are used in storing gridline information, and therefore have as dimensions the number of gridlines in the x- and y-directions, respectively. Both should have dimensions about one and a half times MMAX or NMAX , whichever is larger.

MOBD ( ), NOBD ( ), These store information on the gridlines which have either upper or lower boundaries. The dimension of MOBD is the number of open bounds on grids in the M direction, plus. Similarly for NOBD.

W ( ), FZ ( ), Z ( ), E ( ), HP( ), EP( ), HB( ), EB ( ), ARN ( ), ARGP ( ), ARGB( ), ARGLB,()HL( ), EL( ). These store information on the tidal data, and should have the dimension of the number of tidal constituents employed.

ZIA( ), ZIB( ), ZIC( ), ZID( ), ZIE( ), YIA( ), YIB( ). These utility vectors may have any dimension, usually the largest number of time steps (MAXST) ever used.

LOGICAL READIN The logical variable, READIN, has either of two values. (.TRUE. or .FALSE.), indicating whether initial values of the matrices SE, U, and V, are read in from data.

7.1 EXECUTION PARAMETERS

These are the input constants which are most likely to change over several program runs, so appear first for convenience to the user.

AT	AT is the length of one-half time step, or the time step of each of the two implicit-explicit operations. The total modeled time is, therefore, twice AT times MAXST. Printouts occur at the end of a full operation, so consecutive outputs occur twice AT apart in model time.
MAXST	This is the total number of full time steps to be executed. The utility vectors ZIA, ZIB, should be dimensioned equal to the largest MAXST the programmer is likely to use.
READIN	This logical variable is set either .TRUE. or .FALSE. See explanation in Section 6.1.
IPUNCH	The arrays SE, U, and V can be stored at any time step by setting

IPUNCH = (NST) where NST is the particular time step. When IPUNCH is set greater than MAXST, no punched output is generated.

IRMS For step numbers greater than IRMS, the average U and V velocities are calculated.

TEYE, VHURR, ANHURR, SURGE, WMHURR are the hurricane parameters, described in Section 4.6. TEYE is the time, in hours, from model start time at which the hurricane intersects the coastline. VHURR is its propagation speed (kts) and ANHURR its approach angle (degrees) to the right of north. SURGE is the maximum surge at the coast, and WMHURR the storm's maximum wind velocity (mph).

7.2 COMPUTATION PARAMETERS

These are several physical and program parameters which remain constant for most program runs. In the Bay model, the length unit is yards.

AL	is the length of each square grid, or, equivalently, the distance between each point of water level calculation.
AG	is the acceleration due to gravity, g.
CMANN	is the Manning friction parameter, N.
NMAX	is the max number of grid squares in the x-direction.
ANGLAT	is the angle of latitude of the body of water, in degrees. For small areas, the latitude of the center is sufficient, but the vector F can be used to incorporate latitude variation.
NI	is the number of iteration performed during the matrix inversion of the implicit step. For small

time steps, a value of 1 is sufficient.

MOBD ( ), NOBD( ) are vectors storing gridline data on the open boundaries. For example,

$$MOBD(i) = ab/cd/ef/g$$

is the i^{th} MOBD vector. The value ab is the row number M of the boundary, which runs from N = cd to N = ef (ef > cd). The value of g is 1 for a water level specification, or 2 for a velocity specification.

MINDO, NINDO — are equal to the total number of MOBD and NOBD vectors plus one, respectively.

IMODE1, IMODE2 — These mode parameters are used to control the type of boundary conditions, which are changed during experimentation. Each mode value corresponds to a specific boundary condition type to be employed. IMODE1 refers to the boundary at Rhode Island Sound, and IMODE2 the Mt. Hope Bay boundary. The conditions are given below.

IMODE1 =1 Normal astronomical tide

2 Zero tide

3 Water level extrapolated from interior field

4 Flowrate continuity

5 Hurricane surge

6 Hurricane surge plus tide

IMODE2 =1 Tidal and River flow

2 River flow only

3 Flowrate continuity

QPROV, QBLACK, QPAWT, QTAUNT are the flowrates (cubic feet per second) for the Providence, Blackstone, Pawtucket, and Taunton Rivers, respectively.

HINV, SEINV are parameters describing the initial water level configuration, by the equation

$$SE(N,M) = HINV + SEINV\left(1 - \frac{M-1}{MMAX}\right) \qquad 7.2.1$$

NSECT is the maximum number of gridlines in either the x- or y-directions. See NBD, MBD.

CDRAG is the wind friction drag coefficient, in the stress equation.

CRHO is the ratio of the densities of air/water.

VIII. FURTHER APPLICATIONS

8.1 HYDRODYNAMIC MODEL

Many applications of the numerical model involving the computed water levels and velocities are feasible. The most apparent are studies similar to those described in Chapter 4, focusing upon different locations in the Bay. Several such studies are presently in progress.

The tidal flow through a segment of the Bay is the subject of one current project. Flowrates are computed through the boundaries of a portion of the upper West Passage by the method described in Section 3.5. From these, the net flow at any time can be determined. An attempt is being made to correlate the net flow with the tide at Newport, the object being to obtain an analytic function for the flow as a function of the tidal range. The relationship, which has the nature of a transfer function, can be used as input to a phytoplankton-zooplankton model being developed by the Graduate School of Oceanography at the University.

This approach is easily adaptable for use in the future development of a finite-element model (one which employs a small number of non-uniform elements, rather than a large number of square grids, and makes use of transfer functions). Computed flowrates across the boundaries would form the input to such a model.

Another closely-related future project would be the modeling of a small portion of the Bay with a square-element gridnet. The grid squares, however, could be much smaller than the one-half nautical mile dimension of the present model. Water levels, for example, computed from the Bay model, would be boundary condition data for this fine-grid model. Such a "localized" model would be useful in studies of small-scale circulation patterns in the vicinity of waste water outfalls or breakwaters.

An investigation involving Bay tidal dynamics is a possible project. One topic would be a comparison of model predicted currents with those given in the tidal current charts adapted from Haight(14). Of special interest is the method of conversion of chart velocities for the range of the tide at Newport. The variations of current directions over several tidal cycles could be examined. Another possibility is a study of time of high and low water relative to Newport for specific coastal locations around the Bay.

A project presently underway is the determination of particle paths, which are the solutions of the approximate equations

$$x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} u\ (x_i, y_i, t)\ dt \qquad 8.1.1$$

$$y_{i+1} = y_i + \int_{t_i}^{t_{i+1}} v\ (x_i, y_i, t)\ dt \qquad 8.1.2$$

where (x_i, y_i), (x_{i+1}, y_{i+1}) are the coordinates of the particle position at times t_i and t_{i+1}, respectively. The solution accuracy increases as the time difference

$$t_{i+1} - t_i = \Delta t$$

decreases. The paths are presently being used to study the motion of oil spills, but may be used in flushing studies, whereby the time for particle introduced at any point in the Bay to reach the mouth can be determined.

8.2 SALT CONCENTRATION MODEL

An obvious use for the hydraulic flow predicted by the model is the velocity data for the generalized concentration equation,

$$\frac{\partial c}{\partial t} + u\ \frac{\partial c}{\partial x} + v\ \frac{\partial c}{\partial y} - \frac{\partial}{\partial x}\ (D\ \frac{\partial c}{\partial x}) - \frac{\partial}{\partial y}\ (D\ \frac{\partial c}{\partial y}) = C_t \qquad 8.2.1$$

where C is the concentration of any conservative property, D is a dispersion coefficient, and C_t a source term. The easiest concentration to model is salt, since it is conservative and much data on its distribution is available. The model would predict the two-dimensional salinity distribution from either boundary conditions or source terms. The dispersion coefficient can be approximated analytically by the Elder dispersion expression (32), and checked against data.

The concentration of any species can also be modeled with the above equation. Specifically, dye studies can provide information about concentration dynamics in the Bay.

8.3 TEMPERATURE MODEL

Once the salinity model is operational, it is but a small step to simulating the concentration of heat. The equation is identical to (8.2.1), with the added complexity of heat transfer across the upper surface (imbedded in the term C_t). The heat model could be used to investigate the effects of adding heated water to the Bay. Another application is the prediction of temperature as an input to models of biological growth rates and populations.

8.4 WATER QUALITY MODEL

With temperature and conservative concentration models, the dynamics of non-conservative species, such as dissolved oxygen (DO) and biochemical oxygen demand (BOD), using the relations

$$\frac{\partial(BOD)}{\partial t} + u \frac{\partial}{\partial x} (BOD) + v \frac{\partial}{\partial y} (BOD) - \frac{\partial}{\partial x} \left(D \frac{\partial(BOD)}{\partial x}\right)$$
$$- \frac{\partial}{\partial y} \left(D \frac{\partial}{\partial y} (BOD)\right) + d\ (BOD) - J = 0 \qquad (8.4.1)$$

$$\frac{\partial}{\partial t} (DO) + u \frac{\partial}{\partial x} (DO) + v \frac{\partial}{\partial y} (DO) - \frac{\partial}{\partial x} \left(D \frac{\partial(DO)}{\partial x}\right)$$
$$- \frac{\partial}{\partial t} \left(D \frac{\partial}{\partial y} (DO)\right) - r\ ((DO_s)-(DO)) - P + d\ (BOD) = 0 \qquad (8.4.2)$$

where d is the BOD decay coefficient, J the BOD source, r the aeration coefficient, DO_s the saturation value, and P the rate of DO increase due to photosynthesis or other causes.

APPENDIX A: The Model Equations in Finite-Difference Notation

The three basic equations, 1.1.22, 1.1.23, and 1.1.24, may be expressed in finite difference form, using the notation outlined in Equations 1.2.8 through 1.2.15. The results are:

I. First Half Time Step

X-Momentum:

$$U^{t+1/2} = U^t + \frac{1}{2}\Delta T\, f\, \overline{\overline{V}}^t - \frac{1}{2}\frac{\Delta T}{\Delta L} U^{t+1/2}\, \delta^*_x U^t - \frac{1}{2}\frac{\Delta T}{\Delta L} U^{t+1/2}\, \overline{\overline{V}}^t \delta^*_y U^t$$

$$- \frac{1}{2}\frac{\Delta T}{\Delta L}\, g\, \delta_x \eta^{t+1/2} - \frac{1}{2}\Delta T\, R^t_{(x)} - \frac{1}{2}\Delta T\, F^{t+1/2}_{(x)} , \qquad (A.1)$$

$$\text{at } X_c + \frac{1}{2}\Delta L,\ Y_c .$$

Conservation of Mass:

$$\eta^{t+1/2} = \eta^t - \frac{1}{2}\frac{\Delta T}{\Delta L}\delta_x \left[(\overline{h}^y + \overline{\eta}^x)^{t+1/2} U^{t+1/2}\right]$$

$$- \frac{1}{2}\frac{\Delta T}{\Delta L}\delta_y \left[(\overline{h}^x + \overline{\eta}^y)^t V^t\right] , \qquad (A.2)$$

$$\text{at } X_c,\ Y_c .$$

Y-Momentum:

$$V^{t+1/2} = V^t - \frac{1}{2}\frac{\Delta T}{\Delta L}\delta^*_x V^t\, \overline{\overline{U}}^{t+1/2} - \frac{1}{2}\frac{\Delta T}{\Delta L}\delta^*_y V^t\, V^{t+1/2}$$

$$- \frac{1}{2}\frac{\Delta T}{\Delta L}\, g\, \delta_y \eta^t - \frac{1}{2}\Delta T\, R^{t+1/2}_{(y)} - \frac{1}{2}\Delta T\, F^t_{(y)} , \qquad (A.3)$$

$$\text{at } X_c,\ Y_c + \frac{1}{2}\Delta L .$$

II. <u>Second Half Time Step</u>

X-Momentum

$$u^{t+1} = u^{t+\frac{1}{2}} + \tfrac{1}{2}\Delta T\, f\, \bar{\bar{v}}^{t+\frac{1}{2}} - \tfrac{1}{2}\frac{\Delta T}{\Delta L}\, u^{t+\frac{1}{2}}\, \delta_x^{*} u^{t+\frac{1}{2}}$$

$$-\tfrac{1}{2}\frac{\Delta T}{\Delta L}\, \bar{\bar{v}}^{t+1}\, \delta_y^{*} u^{t+\frac{1}{2}} - \tfrac{1}{2}\frac{\Delta T}{\Delta L}\, g\, \delta_x \eta^{t+\frac{1}{2}}$$

$$-\frac{\Delta T}{\Delta L}\, R_x^{t+1} - F_y^{t+\frac{1}{2}} \qquad (A.4)$$

at $X_c + \frac{1}{2}\Delta L,\ Y_c$

Conservation of Mass

$$\eta^{t+1} = \eta^{t+\frac{1}{2}} - \frac{\Delta T}{\Delta L}\delta_x\left[(\bar{h}^{y} + \bar{\eta}^{x})^{t+\frac{1}{2}}\right] u^{t+\frac{1}{2}}$$

$$-\tfrac{1}{2}\frac{\Delta T}{\Delta L}\delta_y\left[(\bar{h}^{x} + \bar{\eta}^{y})^{t+1}\right] v^{t+1} \qquad (A.5)$$

at $X_c,\ Y_c$

Y-Momentum

$$v^{t+1} = v^{t+\frac{1}{2}} - \tfrac{1}{2}\frac{\Delta T}{\Delta L}\, f\, \bar{\bar{u}}^{t+\frac{1}{2}} - \tfrac{1}{2}\frac{\Delta T}{\Delta L}\, \bar{u}^{t+\frac{1}{2}}\, \delta_x^{*} v^{t+\frac{1}{2}}$$

$$-\tfrac{1}{2}\frac{\Delta T}{\Delta L}\, v^{t+1}\, \delta_y^{*} v^{t+1} - \tfrac{1}{2}\frac{\Delta T}{\Delta L}\, g\, \delta_y \eta^{t+1}$$

$$-\tfrac{1}{2}\frac{\Delta T}{\Delta L}\, R_y^{t+\frac{1}{2}} - \tfrac{1}{2}\frac{\Delta T}{\Delta L}\, F_y^{t+1} \qquad (A.6)$$

at $X_c,\ Y_c + \frac{1}{2}\Delta L$

where the bottom stress term, R, is defined as

$$R_x^t = g\,u^t \frac{\left[(u^t)^2 + (\bar{\bar{v}}^t)^2\right]^{\frac{1}{2}}}{(\bar{h}^y + \bar{\eta}^x)^t\,(\bar{C}^x)^2} \qquad (A.7)$$

$$R_y^{t+\frac{1}{2}} = g\,v^{t+\frac{1}{2}} \frac{\left[(\bar{\bar{u}}^{t+\frac{1}{2}})^2 + (v^t)^2\right]^{\frac{1}{2}}}{(\bar{h}^x + \bar{\eta}^y)^{t+\frac{1}{2}}\,(\bar{C}^y)^2} \qquad (A.8)$$

$$R_x^{t+1} = g u^{t+1} \frac{\left[(u^{t+\frac{1}{2}})^2 + (\bar{\bar{v}}^{t+1})^2\right]^{\frac{1}{2}}}{(\bar{h}^y + \bar{\eta}^x)^{t+\frac{1}{2}}\,(\bar{C}^x)^2} \qquad (A.9)$$

$$R_y^{t+\frac{1}{2}} = g\,v^{t+\frac{1}{2}} \frac{\left[(\bar{\bar{u}}^{t+\frac{1}{2}})^2 + (v^{t+\frac{1}{2}})^2\right]^{\frac{1}{2}}}{(\bar{h}^x + \bar{\eta}^y)^{t+\frac{1}{2}}\,(\bar{C}^y)^2} \qquad (A.10)$$

and the surface stress terms, f , are defined as

$$F_x^{t+\frac{1}{2}} = \frac{K\,(w_x^{t+\frac{1}{2}})^2}{(\bar{h}^y + \bar{\eta}^x)^t} \qquad (A.11)$$

$$F_y^t = \frac{K\,(w_y^t)^2}{(\bar{h}^x + \bar{\eta}^y)^t} \qquad (A.12)$$

$$F_x^{t+\frac{1}{2}} = \frac{K\,(w_x^{t+\frac{1}{2}})^2}{(\bar{h}^y + \bar{\eta}^x)^{t+\frac{1}{2}}} \qquad (A.13)$$

$$F_y^{t+1} = \frac{K\,(W_y^{t+1})^2}{(\bar{h}^x + \bar{h}^y)^{t+\frac{1}{2}}} \qquad (A.14)$$

where $K = \dfrac{k\,\rho_{air}}{\rho_{water}}$

APPENDIX B: The Solution of the Implicit Equations

The implicit method of solution for η and U in the first half of the time step is presented herein. The solution of of η and V in the second is analogous. Starting with equations A.2 and A.1 (in Appendix A), and writing out the finite-difference approximations, we have

$$-r_{m-\frac{1}{2}} U_{m-\frac{1}{2}} + \eta_m + r_{m+\frac{1}{2}} U_{m+\frac{1}{2}} = A_m \tag{B.1}$$

$$-r_m \eta_m + U_{m+\frac{1}{2}} + r_{m+1} \eta_{m+1} = B_{m+\frac{1}{2}} \tag{B.2}$$

where the coefficients r are

$$r_{m\pm\frac{1}{2}} = \frac{1}{2}\frac{\Delta T}{\Delta L} (\bar{h}^y + \bar{\eta}^x)_{m\pm\frac{1}{2}} \tag{B.3}$$

$$r_m = \frac{1}{2}\frac{\Delta T}{\Delta L} g \tag{B.4}$$

and A_m, B_m are the remaining terms in equations A.2 and A.1, respectively. Both η and U are at the $t+\frac{1}{2}$ time level (except for $\bar{\eta}^x$ in B.3, which is at time t).

Suppose the first computational grid is at m=2, and the last is m=J. Then the values of η occur with subscripts m=2,3,...J, while U values have subscripts $m=1\frac{1}{2}, 2\frac{1}{2}, \ldots J+\frac{1}{2}$ (see Figure B.1).

Solving eq. B.1 for η_m at m=2 gives

$$\eta_2 = A_2 + r_{1\frac{1}{2}} U^*_{1\frac{1}{2}} - r_{2\frac{1}{2}} U_{2\frac{1}{2}} \tag{B.5}$$

where $U^*_{1\frac{1}{2}}$ is the velocity at the boundary. For the case of a land boundary, $U^*_{1\frac{1}{2}}$ is zero. Equation B.5 may be rewritten as

$$\eta_2 = -P_2 U_{2\frac{1}{2}} + Q_2 \tag{B.6}$$

where $P_2 = r_{2\frac{1}{2}}$ (B.7)

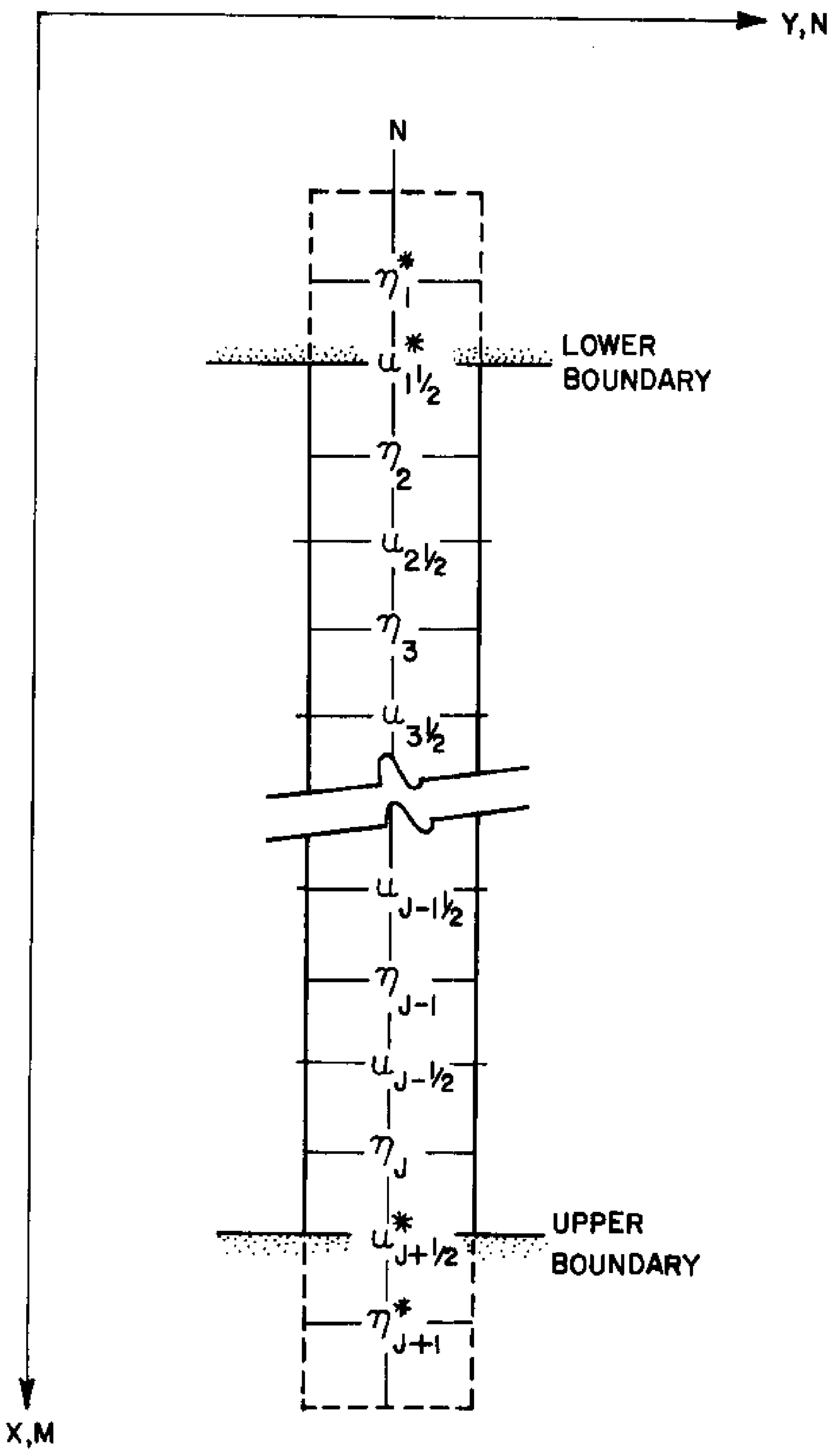


Fig. B-1. Definition sketch showing placement of water-level (η) and velocity (u) along a grid row example in the x-direction.

and $W_2 = A_2 + r_{1\frac{1}{2}} U^*_{1\frac{1}{2}}$ (B.8)

Equation B.2 at m=2 is

$$U_{2\frac{1}{2}} = B_{2\frac{1}{2}} + r_2 \eta_2 - r_3 \eta_3 \quad (B.9)$$

Taking the expression for η_2 from eq. B.6, and substituting into the above ,

$$U_{2\frac{1}{2}} = B_{2\frac{1}{2}} + r_2 (-P_2 U_{2\frac{1}{2}} + W_2) - r_3 \eta_3 \quad (B.10)$$

or $U_{2\frac{1}{2}} = -R_2 \eta_3 + S_2$ (B.10a)

where $R_2 = \dfrac{r_3}{1+r_2P_2}$ (B.11)

$$S_2 = \frac{B_{2\frac{1}{2}} + r_2W_2}{1 + r_2P_2} \quad (B.12)$$

The next water level, η_3, is (from eq. B.1 at m=3)

$$\eta_3 = A_3 + r_{2\frac{1}{2}}U_{2\frac{1}{2}} - r_{3\frac{1}{2}}U_{3\frac{1}{2}} \quad (B.13)$$

and substituting the expression for $U_{2\frac{1}{2}}$ from eq. B.10a,

$$\eta_3 = A_3 + r_{2\frac{1}{2}}(-R_2\eta_3 + S_2) - r_{3\frac{1}{2}}U_{3\frac{1}{2}}$$

or $\eta_3 = -P_3U_{3\frac{1}{2}} + Q_3$ (B.14)

where $P_3 = \dfrac{r_{3\frac{1}{2}}}{1 + r_{2\frac{1}{2}}R_2}$ (B.15) and $$U_3 = \frac{A_3 + r_{2\frac{1}{2}} S_2}{1 + r_{2\frac{1}{2}} R_2} \tag{B.16}$$

The velocity $U_{3\frac{1}{2}}$ is obtained from eq B.2 at m=3 ;

$$U_{3\frac{1}{2}} = B_{3\frac{1}{2}} + r_3 \eta_3 - r_4 \eta_4 \tag{B.17}$$

Or $$U_{3\frac{1}{2}} = -R_3 \eta_4 + S_3 \tag{B.18}$$

where $$R_3 = \frac{r_4}{1 + r_3 P_3} \tag{B.19}$$

$$S_3 = \frac{B_{3\frac{1}{2}} + r_3 Q_3}{1 + r_3 P_3} \tag{B.20}$$

This proceedure (calculation of P_m, Q_m, R_m, and S_m) is repeated for all m up to m=J, where, for a land boundary at J+½,

$$\eta_J = -P_J U^*_{J+\frac{1}{2}} + Q_J \tag{B.21}$$

and η_J is easily computed since $U^*_{J+\frac{1}{2}}$ is zero.

Suppose, however, that instead of land boundaries, the first (m=1) and last (m=J+1) are water boundaries, with either velocity or water level values given. For a first grid water level value, η^*_1, eq. B.2 gives

$$U_{1\frac{1}{2}} = B_{1\frac{1}{2}} + r_1 \eta^*_1 - r_2 \eta_2 = -R_1 \eta_2 + S_1 \tag{B.22}$$

where $$R_1 = r_2 \tag{B.23}$$

and $$S_2 = B_{1\frac{1}{2}} + r_1\eta_1^* \tag{B.24}$$

For a first grid velocity, $U^*_{1\frac{1}{2}}$, eq. B.5 will suffice. For the case of a last grid water level value, η^*_{J+1}, eq. B.2 leads to

$$\begin{aligned} U_{J+\frac{1}{2}} &= B_{J+\frac{1}{2}} + r_J\eta_J - r_{J+1}\eta^*_{J+1} \\ &= -R_J\eta^*_{J+1} + S_J \end{aligned} \tag{B.25}$$

There are three methods of specifying the last grid $(m=J+1)$ velocity. The first is to specify the value $U^*_{J+1+\frac{1}{2}}$ and apply eq. B.1 to obtain

$$\eta_{J+1} = -P_{J+1}U^*_{J+1+\frac{1}{2}} + Q_{J+1} \tag{B.26}$$

which involves the calculation of η at the boundary grid $(m=J)$. Secondly, it is possible to calculate $U_{J+\frac{1}{2}}$ from $U^*_{J+1+\frac{1}{2}}$ using a flowrate conservation law. Finally, the velocity at $m=J+\frac{1}{2}$ could be specified, and eq. B.21 used directly. This last method is the most efficient, and is the one used in the present model calculations.

In general, the coefficients can be written as

$$P_m = \frac{r_{m+\frac{1}{2}}}{1 + r_{m-1}R_{m-1}} \tag{B.27}$$

$$Q_m = \frac{A_m + r_{m+\frac{1}{2}}S_{m-1}}{1 + r_{m-\frac{1}{2}}R_{m-1}} \tag{B.28}$$

$$R_m = \frac{r_m}{1 + r_{m-1}P_m} \tag{B.29}$$

$$S_m = \frac{\theta_{m+\frac{1}{2}} + r_m Q_m}{1 + r_{m-1} P_m} \tag{B.30}$$

Starting at the lower boundary ($m=1$), R_m and S_m are calculated (from B.23 and B.24 for a water level boundary; $R_1 = S_1 = 0$ for a land boundary; $R_1 = 0$, $S_1 = U^*_{1+\frac{1}{2}}$ for a velocity boundary). Then at the computational grids ($m=2$ to $m=J$) A_m, P_m, Q_m, θ_m, R_m, and S_m are calculated in that order for each m. At $m = J$, $U_{J+\frac{1}{2}}$ assumes its appropriate value (zero for a land boundary; the specified value for a velocity boundary; or computed from eq. B.25 for a water level boundary). The remaining values of η and U are then obtained from the recursive relations

$$\eta_m = -P_m U_{m+\frac{1}{2}} + Q_m \tag{B.31}$$

$$U_{m-\frac{1}{2}} = -R_m \eta_m + S_{m-1} \tag{B.32}$$

for m decreasing from $m=J$ to $m=2$.

APPENDIX C: The Model Program Listing

```
C          SET DIMENSIONS OF THE SYSTEM
C
      DIMENSION A(48),B(48),P(48),Q(48),R(48),S(48),F(48),
     1KONVRT(21),NH(21),NPRINT(100),DAVG(80)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBC(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DIMENSION WAVG(19,48),ZID(1000)
      LOGICAL READIN,TEST
C
C     SET EXECUTION PARAMETERS
C
      DATA YR,DAY,THR,TMIN/44.,258.,04.,40./
C   IMODE1=1(TIDE),2(0 TIDE),3(EXTRAP SE),4(Q),5(SURGE),6(SURGE+TIDE)
      DATA IMODE1,IMODE2/4,2/
      DATA AT,MAXST,CMANN,IPUNCH,IRMS/120.,200,.020,1000,150/
      DATA QPROV,QTAUNT,QBLACK,QPAWT/1000.,0000.,0000.,0000./
      DATA TEYE,VHURR,ANHURR,SURGE,WMHURR/20.,30.,50.,4.28,90./
      DATA HINV,SEINV,WX,WY,CMAG,CMAGSE/0.,0.,0.,0.,1.,1./
      DATA AL,AG,CRHO,CDRAG,ANGLAT/1012.7,10.73,.00114,.0025,41.6/
      DATA NMAX,MMAX,NI,NINDO,MINDO,NSECT/19,48,1,3,4,80/
      NOBD(1)=1923242
      NOBD(2)=0410112
      MOBD(1)=0103042
      MOBD(2)=4808091
      MOBD(3)=4811131
      READIN=.FALSE.
      WY=1.
      TEST=.FALSE.
      IF(IMODE1.EQ.4) CMAG=1000.
      IF(IMODE1.EQ.4) CMAGSE=1000.
      IF(IMODE1.EQ.4)MOBD(2)=MOBD(2)+1
      IF(IMODE1.EQ.4)MOBD(3)=MOBD(3)+1
      GO TO 87
C
C     SET OPEN BOUNDS
C
   89 CONTINUE
      T1=K-1
      T1=T1*AT/3600.
      T2=T1+7./60.
      T3=T1+8./60.
      T4=T1-TS
      THURR=ABS(T1-TEYE)
C     SET LOWER BOUNDARY
      GO TO (1080,1080,1082,1083,1084,1084),IMODE1
 1080 SLB=0.
      SLBB=0.
      IF(IMODE1.EQ.2) GO TO 1081
   79 CONTINUE
      DO 83 I=1,NTERM
      SLBB=F2(I)*HL(I)*COS(W(I)*T3+ARN(I)   )+SLBB
   83 SLB=F2(I)*HL(I)*COS(W(I)*T2+ARN(I)   )+SLB
 1081 SEP(08,48)=SLB/3.
      SEP(09,48)=(5.*SLB+1.*SLBB)/18.
```

```
      SEP(11,48)=(2.*SLB+4.*SLBB)/18.
      SEP(12,48)=(1.*SLB+5.*SLBB)/18.
      SEP(13,48)=SLBB/3.
      GO TO 1090
 1082 SEP(08,48)=2.*SEP(08,47)-SEP(08,46)
      SEP(09,48)=2.*SEP(09,47)-SEP(09,46)
      SEP(11,48)=2.*SEP(11,47)-SEP(11,46)
      SEP(12,48)=2.*SEP(12,47)-SEP(12,46)
      SEP(13,48)=2.*SEP(13,47)-SEP(13,46)
      GO TO 1090
 1083 CONTINUE
      TEMP2=0.0
      DO 1183 L=8,9
      LL=L-1
 1183 TEMP2=TEMP2 +(H(L,46)+H(LL,46)+SE(L,46)+SE(L,47))*UP(L,46)
      Q1=TEMP2/2.
      UP(8,47)=Q1/(H(8,47)+H(7,47)+SE(8,47)*2.)
      UP(9,47)=Q1/(H(9,47)+H(8,47)+2.*SE(9,47))
      TEMP2=0.0
      DO 70 L=11,13
      LL=L-1
   70 TEMP2=TEMP2+(H(L,46)+H(LL,46)+SE(L,46)+SE(L,47))*UP(L,46)
      Q1=TEMP2/3.
      UP(11,47)=Q1/(H(11,47)+H(10,47)+2.*SE(11,47))
      UP(12,47)=Q1/(H(12,47)+H(11,47)+2.*SE(11,47))
      UP(13,47)=Q1/(H(13,47)+H(12,47)+2.*SE(13,47))
      GO TO 1090
 1084 CONTINUE
      SLB=SURGE*EXP(-.148*THURR**2)
      RAD=180./3.14157
C     WX=WIND SPEED (KNOTS) TO SOUTH, WY IS TO EAST
      CHURR=1.0
      IF(T1.GT.TEYE) CHURR=-1.
      WHURR=WMHURR*1.15*EXP(-.20*THURR)
      THETA1=CHURR*90.*(1.-EXP(-.023*THURR*VHURR))+20.-ANHURR
      THETA2=(90.-10.14-THETA1)/RAD
      WX=-WHURR*SIN(THETA2)
      WY=-WHURR*COS(THETA2)
      SLBB=SLB
      IF(IMODEL.EQ.6) GO TO 79
      GO TO 1081
 1090 CONTINUE
C
C     SET MT. HOPE BAY BOUNDARY
      QMHB=0.
      GO TO (1091,1092,1093),IMODE2
 1091 QMHB=150.5*COS(2.*PI*(T4-9.82)/12.42)+33.2*COS(2.*PI*(T4-6.29)/
     1 6.21)+36.4*COS(2.*PI*(T4-3.32)/4.14)
      QMHB=QMHB*1000.
 1092 QMHB=QMHB-.72*QTAUNT
      QMHB=2.*QMHB/(27.*AL*(H(18,23)+H(18,22)+2.*SE(18,23)))
      VP(18,23)=QMHB
      GO TO 1098
 1093 VP(19,23)=VP(18,23)*(H(18,23)+H(18,22))/(H(19,23)+H(19,22))
     1          +VP(18,24)*(H(18,24)+H(18,23))/(H(19,24)+H(19,23))
 1098 VP(18,24)=VP(18,23)/1000.
C
C     SET PROVIDENCE AND PAWTUXET RIVER BOUNDARIES
      A1=.5*AL*(H(3,1)+H(2,1)+2.*SE(3,1))
      A2=.5*AL*(H(3,1)+H(4,1)+2.*SE(4,1))
```

```
      A3=A1+A2
      U(3,1)=(QPROV+QBLACK)/(27.*A3)
      U(4,1)=(QPROV+QBLACK)/(27.*A3)
      V(4,11)=QPAWT/((H(4,11)+SE(4,11))*C5)
      V(4,12)=QPAWT/((H(4,11)+SE(4,11))*C5*1000.)
      IF(ISTEP-1)301,96,301
C
C
C     STORE VALUES
C
  81  CONTINUE
      Z1A(NST)=SEP(15,40)*3.
      Z1B(NST)=SEP(16,20)*3.
      Z1C(NST)=SEP(02,05)*3.
      GO TO 82
   87 CONTINUE
C
C
C      INITIALIZE ALL VARIABLES
C
      PI=3.1415927
      ARG=ANGLAT*3.1415927/180.
      FF=3.1415927*SIN(ARG)/21600.
 2080 NST=0
      ICH=1
      IP=1
      C1=AT*AG/AL
      C2=AT/AL
      C3=AT/4.
      C4=8.*AT*AG
      C5=54.*AL
      C6=2.*CDRAG*CRHO*AI*(1.667/3.)**2
      DO 6 M=1,MMAX
      DO 6 N=1,NMAX
    4 SE(N,M)=0.0
      SEP(N,M)=0.0
      WAVG(N,M)=0.
      VAVG(N,M)=0.0
      UAVG(N,M)=0.0
      VP(N,M)=0.0
      UP(N,M)=0.0
      V(N,M)=0.
      U(N,M)=0.
      C(N,M)=0.
      H(N,M)=0.0
    6 F(N)=FF
    8 CONTINUE
C
C     CALL SUBROUTINES
C
      CALL KURIH(MAXST,AT,NTERM,TS,YR,DAY,THR,TMIN)
      CALL DIVE(NMAX,MMAX)
      CALL FINC(MIND,NIND,MMAX,NMAX,MINDC,NINDC,NSECT)
      CALL DEPTH(NMAX,MMAX)
      CALL CHEZY(NMAX,MMAX,CMANN)
      CALL CHECK(NMAX,MMAX)
      READ(5,25) (NPRINT(N),N=1,26)
  25  FORMAT(26I3)
C
C
```

```
C     READ IN INITIAL VALUES OF TIDE, U VELOCITY, AND V VELOCITY
      IF(READIN) GO TO 36
      GO TO 37
   36 CONTINUE
      DO 6030 M=1,MMAX
 6030 READ(5,6039) (SE(N,M),N=1,12)
      DO 6031 M=1,MMAX
 6031 READ(5,6039) (SE(N,M),N=13,NMAX)
      DO 6032 M=1,MMAX
 6032 READ(5,6039) ( U(N,M),N=1,12)
      DO 6033 M=1,MMAX
 6033 READ(5,6039) ( U(N,M),N=13,NMAX)
      DO 6034 M=1,MMAX
 6034 READ(5,6039) ( V(N,M),N=1,12)
      DO 6035 M=1,MMAX
 6035 READ(5,6039) ( V(N,M),N=13,NMAX)
 6039 FORMAT(12F5.2)
      DO 6040 M=1,MMAX
      DO 6040 N=1,NMAX
      SEP(N,M)=SE(N,M)
      UP(N,M)=U(N,M)
 6040 VP(N,M)=V(N,M)
   37 CONTINUE
C
C                 WRITE INITIAL VALUES
C
      WRITE(6,12)
   12 FORMAT(1H1,/1X,24HINITIAL DEPTHS IN .1 YD./)
      DO 9 M=1,MMAX
      DO 40 N=1,NMAX
   40 NH(N)=H(N,M)*10.+.01
    9 WRITE(6,6001)M,(NH(N),N=1,NMAX)
      WRITE(6,15) CMANN
   15 FORMAT(1H1,/1X,'CHEZY VALUES (YDS) FOR MANNING N =',F5.3/)
   10 FORMAT(1X, I2,1X,32F4.0)
      DO 16 JA=1,MMAX
   16 WRITE(6,10) JA,(C(N,JA),N=1,NMAX)
      WRITE(6,17)
   17 FORMAT(1H1,/1X,28HINITIAL WATER LEVELS IN FEET)
      DO 18 JC=1,MMAX
      DO 13 N=1,NMAX
   13 KONVRT(N)=SEP(N,JC)*300.
   18 WRITE(6,6001) JC,(KONVRT(N),N=1,NMAX)
 6018 FORMAT(5X,10(I2,1X,F6.2,2X))
      IF(READIN) GO TO 6020
      GO TO 6028
 6020 WRITE(6,6021)
 6021 FORMAT(1H1,10X,16HINITIAL U IN FPS)
      DO 6023 M=1,MMAX
      DO 6022 N=1,NMAX
 6022 KONVRT(N)=300.*U(N,M)
 6023 WRITE(6,6001) M,(KONVRT(N),N=1,NMAX)
      WRITE(6,6024)
 6024 FORMAT(1H1,10X,16HINITIAL V IN FPS)
      DO 6026 M=1,MMAX
      DO 6025 N=1,NMAX
 6025 KONVRT(N)=300.*V(N,M)
 6026 WRITE(6,6001) M,(KONVRT(N),N=1,NMAX)
 6028 CONTINUE
      ISTEP=2
```

```
      GO TO 500
 6001 FORMAT(1H ,I2,1X,32I4)
C
C
C       COMPUTE UP AND SEP ON ROW N ( FIRST HALF TIMESTEP)
C
   88 ISTEP=1
      NST=NST+1
      K=2*NST-1
 2001 IF(NST.GT.MAXST) GO TO 410
C
C     CONTINUE TO SET OPEN BOUND
      GO TO 89
   90 NUM=1
  100 IF(NUM.EQ.NIND) GO TO 190
      NSRCH=NBD(NUM)/1000000
      N    =NBD(NUM)/10000  -NSRCH*100
      MF   =NBD(NUM)/100-NSRCH*10000-N*100
      L    =NBD(NUM)-NSRCH*1000000-N*10000-MF*100
      LL=L-1
      LLL=L+1
      IA=NSRCH/10
      IB=NSRCH-10*IA
      MFF  =MF-1
      NNN=N+1
      NN=N-1
      IT=1
      R(MFF)=0.0
      S(MFF)=0.0
      GAMMA=0.5
      IF(IA.EQ.1) GO TO 993
      IF(IA.EQ.2) GO TO 992
      GO TO 99
  992 R(MFF)=0.
      S(MFF)=UP(N,MFF)
      GO TO 99
  993 MFF=MF-1
      TEMP10=U(NNN,MFF)
      IF(TEMP10.EQ.0.) TEMP10=   U(NN,MFF)
      TEMP11=U(NN,MFF)
      IF(TEMP11.EQ.0.) TEMP11=   U(NNN,MFF)
      M=MFF
      MMM=MF
      ICHECK=10
 6100 FORMAT(10X,'N=',I2,2X,'M=',I2,2X,'NST=',I4,2X,'CHECK POINT ',I4)
      TEMP12=-C6*WX*ABS(WX)/(SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))
  990 ALPHA=1.
      R(MFF)=C1/(1.+C2*(U(N,MF)-U(N,MFF))*(1.-ALPHA))
      S(MFF)=(U(N,MFF)+C1*SEP(N,MFF)-TEMP12
     1 -U(N,MFF)*SQRT(U(N,MFF)**2+(((V(N,MF)+ V(NN,MF))**2)/16.))/
     2((SE(N,MFF)+SE(N,MF)+H(N,MFF)+H(NN,MFF))*((C(N,MFF)+C(N,MF))
     3**2))*C4+(V(N,MF)+V(NN,MF))*.25*(AT*F(N)-(1.-GAMMA)*C2*
     4(TEMP10   -U(N,MFF))-GAMMA*C2*(U(N,MFF)-TEMP11)))/
     5(1. + C2*(U(N,MF)-U(N,MFF))*(1.-ALPHA))
   99 CONTINUE
      K=MF
  101 DO 102 M=K,L
      ICHECK=20
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      MM=M-1
```

```
      MMM=M+1
      TEMP9=SE(N,M)
      IF(IT.GT.1) TEMP9=SEP(N,M)
  988 TEMP1=SE(NNN,M)
      IF(TEMP1.EQ.0.) TEMP1=2.*SE(N,M)-SE(NN,M)
  986 TEMP2=SE(NN,M)
      IF(TEMP2.EQ.0.) TEMP2=2.*SE(N,M)-SE(NNN,M)
  984 TEMP3=SE(N,MMM)
      IF(TEMP3.EQ.0.) TEMP3=2.*SE(N,M)-SE(N,MM)
      IF(IT.GT.1) TEMP3=SEP(N,MMM)
      IF(IT.GT.1.AND.TEMP3.EQ.0.) TEMP3=2.*SEP(N,M)-SEP(N,MM)
  906 TEMP4=SE(N,MM)
      IF(TEMP4.EQ.0.) TEMP4=2.*SE(N,M)-SE(N,MMM)
      IF(IT.GT.1) TEMP4=SEP(N,MM)
       IF(IT.GT.1.AND.TEMP4.EQ.0.) TEMP4=2.*SEP(N,M)-SEP(N,MMM)
  913 A(M)= SE(N,M) -.5*C2*(H(N,M)+H(N,MM)+SE(N,M) +TEMP1 )*
     1V(N,M)+.5*C2*(H(NN,MM)+H(NN,M)+SE(N,M)+TEMP2 )*V(NN,M)
      P(M)=.5*C2*(H(N,M)+H(NN,M)+TEMP9 +TEMP3 )/( 1.+.5*C2*
     1(H(N,MM)+H(NN,MM)+TEMP4 +TEMP9 )*R(MM))
      Q(M)=(A(M)+.5*C2*(H(N,MM)+H(NN,MM)+TEMP4 +TEMP9 )*S(MM)
     1)/(1.+.5*C2*(H(N,MM)+H(NN,MM)+TEMP4 +TEMP9)*R(MM))
      IF(M.EQ.L)GO TO 102
  914 GAMMA= 0.5
      TEMP10=U(NNN,M)
      IF(TEMP10.EQ.0.) TEMP10=  U(NN,M)
  916 TEMP11=U(NN,M)
      IF(TEMP11.EQ.0.) TEMP11=  U(NNN,M)
  918 TEMP6=AT*F(N)-(1.-GAMMA)*C2*(TEMP10-U(N,M))-GAMMA*C2*
     1(U(N,M)-TEMP11)
      TEMP6=.25*TEMP6
      TEMP12=-C6*WX*ABS(WX)/(SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))
      B(M)=U(N,M)+TEMP6 *(V(N,M)+V(N,MMM)+V(NN,M)+V(NN,MMM))
     1 -U(N,M)*SQRT( U(N,M)**2 +(((V(N,M)+V(N,MMM)+V(NN,M)+V(NN,MMM)
     2)**2)/16.))/((SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))*((C(N,M)+C(N,MM
     3M))**2))*C4-TEMP12
      ALPHA=0.5
      TEMP1=1.+C2*(AG*P(M)+(1.-ALPHA)*(U(N,MMM)-U(N,M))+
     1ALPHA*(U(N,M)-U(N,MM)))
      R(M)=  C1/TEMP1
      S(M)=(B(M)+  C1*Q(M))/TEMP1
  102 CONTINUE
      ICHECK=30
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
C
      IF(IB.EQ.0) TEMP1=0.
      IF(IB.EQ.2) TEMP1=UP(N,L)
      IF(IB.EQ.1) GO TO 103
      GO TO 104
  103 CONTINUE
      TEMP10=U(NNN,L)
      IF(TEMP10.EQ.0.) TEMP10=  U(NN,L)
  921 TEMP11=U(NN,L)
      IF(TEMP11.EQ.0.)  TEMP11=  U(NNN,L)
  923 LLL =L+1
      LL  =L-1
      MMM=LLL
      M=L
      ICHECK=40
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      TEMP12=-C6*WX*ABS(WX)/(SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))
```

```
      ALPHA=0.
      TEMP1  =(-C1*SEP(N,LLL)+U(N,L)*(1.-C4*SQRT(U(N,L)**2+(((V(N,L)+
     1V(NN,L))**2)/16.))/((SE(N,L)+SE(N,LLL)+
     2H(N,L)+H(NN,L))*((C(N,L)+C(N,LLL))**2))-TEMP12)+
     3.25*(AT*F(N)-GAMMA*C2*(U(N,L)- TEMP11)-(1.-GAMMA)*C2*
     4(TEMP10-U(N,L))))*(V(N,L)+V(NN,L))
     5+C1*Q(L))/(1.+C2*(AG*P(L)+(U(N,L)-U(N,LL))*ALPHA))
  104 CONTINUE
      UP(N,L)=TEMP1
      M=L
      DO 106 J=K,L
      MM=M-1
      SEP(N,M)=-P(M)*UP(N,M)+Q(M)
      UP(N,MM)=-R(MM)*SEP(N,M)+S(MM)
  106 M=M-1
      IF(NSRCH.EQ.20) GO TO 110
      GO TO 111
  110 A1=SEP(NNN,MFF)
      IF(A1.EQ.0.) A1=SEP(NN,MFF)
      SEP(N,MFF)=(SE(N,MFF)+SEP(N,MF)+A1)/3.0
  111 IF(NSRCH.EQ.2) GO TO 115
      GO TO 120
  115 A1=SEP(NNN,LLL)
      IF(A1.EQ.0.) A1=SEP(NN,LLL)
      SEP(N,LLL)=(SE(N,LLL)+SEP(N,L)+A1)/3.0
  120 IT=IT+1
      IF(IT.LE.NI) GO TO 101
 1020 NUM=NUM+1
      GO TO 100
  190 CONTINUE
      NUM=1
C
C              COMPUTE VP ON COLUMN M (FIRST HALF TIMESTEP)
C
  201 IF(NUM.EQ.MINO) GO TO 202
  924 MSRCH=MBC(NUM)/1000000
      M     =MBC(NUM)/10000 -MSRCH*100
      NF    =MBC(NUM)/100    -MSRCH*10000 -M*100
      L     =MBC(NUM)         -MSRCH*1000000-M*10000-NF*100
      LL=L-1
      LLL=L+1
      NFF=NF-1
      MMM=M+1
      MM=M-1
      IA=MSRCH/10
      IB=MSRCH-10*IA
      DO 204 N=NF,LL
      ICHECK=50
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      NN=N-1
      NNN=N+1
      BETA=0.5
10018 TEMP4=C2*((1.-BETA)*(V(NNN,M)-V(N,M))+BETA*(V(N,M)-V(NN,M)))
      TEMP1=V(N,M)**2+(((UP(N,M)+UP(NNN,M)+UP(N,MM)+UP(NNN,MM))**2)/
     116.)
      TEMP2=(SEP(N,M)+SEP(NNN,M)+H(N,MM)+H(N,M))*(C(N,M)+
     1C(NNN,M))**2
      TEMP12=-C6*WY*ABS(WY)/(SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))
      TEMP3=1.+C4*SQRT(TEMP1)/TEMP2+TEMP4+TEMP12
      TEMP3=1./TEMP3
```

```
2040 DELTA=0.5
      TEMP10=V(N,MMM)
      IF(TEMP10.EQ.0.) TEMP10=  V(N,MM)
 926  TEMP11=V(N,MM)
      IF(TEMP11.EQ.0.) TEMP11=  V(N,MMM)
 928  TEMP1=(AT*F(N)+(1.-DELTA)*C2*(TEMP10-V(N,M))+DELTA*C2*
     1(V(N,M)-TEMP11))*.25
 204  VP(N,M)=TEMP3*
     1(V(N,M)-TEMP1*(UP(N,M)+UP(NNN,M)+UP(N,MM)+UP(NNN,MM))
     2- C1*(SE(NNN,M)-SE(N,M)))
      ICHECK=60
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      IF(IB.EQ.0) TEMP1=0.
      IF(IB.EQ.2) TEMP1=VP(L,M)
      IF(IB.EQ.1) GO TO 205
      GO TO 206
 205  CONTINUE
      TEMP10=V(L,MMM)
      IF(TEMP10.EQ.0.) TEMP10=  V(L,MM)
 931  TEMP11=  V(L,MM)
      IF(TEMP11.EQ.0.) TEMP11=  V(L,MMM)
 933  LLL=L+1
      BETA  =0.
      LL =L-1
      TEMP4=C2*BETA*(V(L,M)-V(LL,M))
      TEMP1=V(L,M)**2+(((UP(L,M)+UP(L,MM))**2)/16.)
      TEMP2=(SEP(L,M)+SEP(LLL,M)+H(L,MM)+H(L,M))*(C(L,M)+C(LLL,M))**2
      N=L
      NNN=LLL
      TEMP12=-C6*WY*ABS(WY)/(SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))
      TEMP3=1.+C4*SQRT(TEMP1)/TEMP2+TEMP4+TEMP12
      TEMP3=1./TEMP3
      DELTA=0.5
      TEMP1=.25*(AT*F(N)+(1.-DELTA)*C2*(TEMP10-V(L,M))+DELTA*C2*
     1(V(L,M)-TEMP11))
      TEMP1  =TEMP3*(V(L,M)-TEMP1*(UP(L,M)+UP(L,MM))
     1-C1*(SE(LLL,M)-SE(L,M)))
 206  VP(L,M)=TEMP1
      IF(IA.EQ.0) TEMP1=0.
      IF(IA.EQ.2) TEMP1=VP(NFF,M)
      IF(IA.EQ.1) GO TO 207
      GO TO 208
 207  NFF=NF-1
      TEMP10=V(NFF,MMM)
      IF(TEMP10.EQ.0.) TEMP10=  V(NFF,MM)
 936  TEMP11=V(NFF,MM)
      IF(TEMP11.EQ.0.) TEMP11=  V(NFF,MMM)
 938  BETA=1.
      TEMP4=C2*(1.-BETA)*(V(NF,M)-V(NFF,M))
      TEMP1=V(NFF,M)**2+(((UP(NF,M)+UP(NF,MM))**2)/16.)
      TEMP2=(SEP(NFF,M)+SEP(NF,M)+H(NFF,M)+H(NFF,MM))*
     1(C(NF,M)+C(NFF,M))**2
      N=NFF
      NNN=NF
      ICHECK=70
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      TEMP12=-C6*WY*ABS(WY)/(SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))
      TEMP3=1.+C4*SQRT(TEMP1)/TEMP2+TEMP4+TEMP12
      TEMP3=1./TEMP3
      DELTA=0.5
```

```
      TEMP1=.25*(AT*F(N)+(1.-DELTA)*C2*(TEMP10-V(NFF,M))
     1  +DELTA*C2*(V(NFF,M)-TEMP11))
      TEMP1    =TEMP3*(V(NFF,M)-TEMP1*(UP(NF,M)+UP(NF,MM))
     1          -C1*(SE(NF,M)-SE(NFF,M)))
  209 VP(NFF,M)=TEMP1
      NUM=NUM+1
      GO TO 201
  202 CONTINUE
C
C
C              PRINT INSTRUCTIONS
C
C
  500 IF(ISTEP-2)297,296,297
  296 CONTINUE
C     STORE VALUES
      IF(NST.EQ.0) GO TO 82
      GO TO 81
  82  CONTINUE
      IVL=NST
      IF(IVL.GE.IRMS) CALL VELANA(IVL,IRMS,MAXST,CMAG,WAVG)
      IF(NST.EQ.NPRINT(IP)) GO TO 295
      GO TO 297
  295 IP=IP+1
      CALL        PRINTO(NST,MMAX,NMAX,CMAG,COIM,AT,CMAGSE)
  297 NUM=1
      DO 292 N=1,NMAX
      DO 292 M=1,MMAX
      U(N,M)=UP(N,M)
      V(N,M)=VP(N,M)
  292 SE(N,M)=SEP(N,M)
      IF(NST.EQ.IPUNCH) GO TO 6045
      GO TO 6060
 6045 IF(ISTEP.EQ.2) GO TO 6050
      GO TO 6060
 6050 CONTINUE
      DO 6051 M=1,MMAX
 6051 WRITE(7,6059) (SE(N,M),N=1,12)
      DO 6052 M=1,MMAX
 6052 WRITE(7,6059) (SE(N,M),N=13,NMAX)
      DO 6053 M=1,MMAX
 6053 WRITE(7,6059) ( U(N,M),N=1,12)
      DO 6054 M=1,MMAX
 6054 WRITE(7,6059) ( U(N,M),N=13,NMAX)
      DO 6055 M=1,MMAX
 6055 WRITE(7,6059) ( V(N,M),N=1,12)
      DO 6056 M=1,MMAX
 6056 WRITE(7,6059) ( V(N,M),N=13,NMAX)
 6059 FORMAT(12F5.2)
 6060 CONTINUE
      GO TO(299,88),ISTEP
  299 ISTEP=2
      K=2*NST
C              SET OPEN BOUNDS
      GO TO 89
C
C          COMPUTE VP AND SEP ON COLUMN M (SECOND HALF TIMESTEP)
C
  301 IF(NUM.EQ.MIND) GO TO 390
  939 MSRCH=MBD(NUM)/1000000
```

```
      M      =MBD(NUM)/10000  -MSRCH*100
      NF     =MBD(NUM)/100    -MSRCH*10000   -M*100
      L      =MBD(NUM)        -MSRCH*1000000-M*10000-NF*100
      MM=M-1
      MMM=M+1
      LL=L-1
      LLL=L+1
      IA=MSRCH/10
      IB=MSRCH-10*IA
      NFF=NF-1
      R(NFF)=0.0
      S(NFF)=0.0
      IF(IA.EQ.1) GO TO 940
      IF(IA.EQ.2) GO TO 941
      GO TO 319
  940 NFF=NF-1
      TEMP10=V(NFF,MMM)
      IF(TEMP10.EQ.0) TEMP10=V(NFF,MM)
  943 TEMP11=V(NFF,MM)
      IF(TEMP11.EQ.0.) TEMP11=  V(NFF,MMM)
      NNN=NF
C     N=NF
      N=NFF
      ICHECK=80
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      TEMP12=-C6*WY*ABS(WY)/(SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))
  945 DELTA=0.5
      BETA=1.
      R(NFF)=C1/(1.+C2*(V(NF,M)-V(NFF,M))*(1.-BETA))
      S(NFF)=(V(NFF,M)+C1*SEP(NFF,M)-TEMP12
     1-V(NFF,M)*SQRT(V(NFF,M)**2+(((U(NF,M)+U(NF,MM))**2)/16.))/
     2((SE(NFF,M)+SE(NF,M)+H(NFF,M)+H(NFF,MM))*((C(NFF,M)+C(NF,M))
     3**2))*C4-.25*(AT*F(N)+(1.-DELTA)*C2*(TEMP10-V(NFF,M))
     4+DELTA*C2*(V(NFF,M)-TEMP11))*(U(NF,M)+U(NF,MM)))/
     5(1.+C2*(1.-BETA)*(V(NF,M)-V(NFF,M)))
      GO TO 319
  941 R(NFF)=C.
      S(NFF)=VP(NFF,M)
  319 CONTINUE
      K=NF
      IT=1
  303 DO 302 N=K,L
      ICHECK=90
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      NN=N-1
      NNN=N+1
      TEMP9=SE(N,M)
      IF(IT.GT.1) TEMP9=SEP(N,M)
  947 TEMP1=SE(N,MMM)
      IF(TEMP1.EQ.0.) TEMP1=2.*SE(N,M)-SE(N,MM)
  949 TEMP2=SE(N,MM)
      IF(TEMP2.EQ.0.) TEMP2=2.*SE(N,M)-SE(N,MMM)
  951 TEMP3=SE(NNN,M)
      IF(TEMP3.EQ.0.) TEMP3=2.*SE(N,M)-SE(NN,M)
      IF(IT.GT.1) TEMP3=SEP(NNN,M)
      IF(IT.GT.1.AND.TEMP3.EQ.0.) TEMP3=2.*SEP(N,M)-SEP(NN,M)
  958 TEMP4=SE(NN,M)
      IF(TEMP4.EQ.0.) TEMP4=2.*SE(N,M)-SE(NNN,M)
      IF(IT.GT.1) TEMP4=SEP(NN,M)
      IF(IT.GT.1.AND.TEMP4.EQ.0.) TEMP4=2.*SEP(N,M)-SEP(NNN,M)
```

```
965 A(N)=SE(N,M)-.5*C2*(H(N,M)+H(NN,M)+SE(N,M)+TEMP1 )*U(N,M)
   1+ .5*C2*(H(N,MM)+H(NN,MM)+TEMP2  +SE(N,M))*U(N,MM)
    P(N)= .5*C2*(H(N,M)+H(N,MM)+TEMP9+TEMP3  )/(1.+ .5*C2*
   1(H(NN,M)+H(NN,MM)+TEMP4 +TEMP9 )*R(NN))
    Q(N)=(A(N)+.5*C2*(H(NN,M)+H(NN,MM)+TEMP4+TEMP9   )*S(NN)
   1)/(1.+     .5*C2*(H(NN,M)+H(NN,MM)+TEMP4+TEMP9 )*R(NN))
    IF(N.EQ.L) GO TO 302
966 DELTA=0.5
    TEMP10=V(N,MMM)
    IF(TEMP10.EQ.0.) TEMP10=  V(N,MM)
968 TEMP11=V(N,MM)
    IF(TEMP11.EQ.0.) TEMP11=  V(N,MMM)
970 TEMP6=AT*F(N)+(1.-DELTA)*C2*(TEMP10-V(N,M))
   1    +DELTA*C2*(V(N,M)-TEMP11)
    TEMP6=.25*TEMP6
    TEMP12=-C6*WY*ABS(WY)/(SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))
    B(N)=V(N,M)-TEMP6  *(U(N,M)+U(NNN,M)+U(NNN,MM)+U(N,MM))
   1- V(N,M)*SQRT(V(N,M)**2+(((U(N,M)+U(NNN,M)+U(N,MM)+U(NNN,MM)
   2)**2)/16.))/((SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))*((C(N,M)+
   3C(NNN,M))**2))*C4 -TEMP12
     BETA=0.5
    TEMP1=1.+C2*(AG*P(N)+(1.-BETA)*(V(NNN,M)-V(N,M))+
   1       BETA*(V(N,M)-V(NN,M)))
    R(N)= C1/TEMP1
    S(N)=(B(N)+  C1*Q(N))/TEMP1
302 CONTINUE
    LLL=L+1
C
    IF(IB.EQ.0) TEMP1=0.
    IF(IB.EQ.2) TEMP1=VP(L,M)
    IF(IB.EQ.1) GO TO 307
    GO TO 305
307 CONTINUE
     TEMP10=V(L,MMM)
    IF(TEMP10.EQ.0.) TEMP10=  V(L,MM)
973 TEMP11=V(L,MM)
    IF(TEMP11.EQ.0.) TEMP11=  V(L,MMM)
975 LLL=L+1
    LL=L-1
    BETA=0.
    N=L
    NNN=LLL
    ICHECK=100
    IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
    TEMP12=-C6*WY*ABS(WY)/(SE(N,M)+SE(NNN,M)+H(N,M)+H(N,MM))
    TEMP1  =(-C1*SEP(LLL,M)+V(L,M)*(1.-C4*SQRT(V(L,M)**2+(((U(L,M)+
   1U(L,MM))**2)/16.))/((SE(L,M)+SE(LLL,M)+
   2H(L,M)+H(L,MM))*((C(L,M)+C(LLL,M))**2))-TEMP12)+
   3.25*(AT*F(N)+(1.-DELTA)*C2*(TEMP10-V(L,M))+DELTA*C2*
   4(V(L,M)-TEMP11))*(U(L,M)+U(L,MM))
   5+C1*Q(L))/(1.+C2*(AG*P(L)+BETA*(V(L,M)-V(LL,M))))
305 CONTINUE
    VP(L,M)=TEMP1
    N=L
    DO 306 J=K,L
    NN=N-1
    SEP(N,M)=-P(N)*VP(N,M)+Q(N)
    VP(NN,M)=-R(NN)*SEP(N,M)+S(NN)
306 N=N-1
    IF(MSRCH.EQ.20) GO TO 310
```

```
      GO TO 311
  310 A1=SEP(NFF,MM)
      IF(A1.EQ.0.) A1=SEP(NFF,MMM)
      SEP(NFF,M)=(SE(NFF,M)+SEP(NF,M)+A1)/3.0
  311 IF(MSRCH.EQ.2) GO TO 312
      GO TO 315
  312 A1=SEP(LLL,MM)
      IF(A1.EQ.0.) A1=SEP(LLL,MMM)
      SEP(LLL,M)=(SE(LLL,M)+SEP(L,M)+A1)/3.0
  315 IT=IT+1
      IF(IT.LE.NI) GO TO 303
  976 NUM=NUM+1
      GO TO 301
C
C         COMPUTE UP ON ROW N (SECOND HALF TIMESTEP)
C
  390 NUM=1
  340 IF(NUM.EQ.MIND) GO TO 402
 1021 NSRCH=NBD(NUM)/1000000
      N     =NBD(NUM)/10000  -NSRCH*100
      MF    =NBD(NUM)/100-NSRCH*10000-N*100
      L     =NBD(NUM)-NSRCH*1000000-N*10000-MF*100
      IA=NSRCH/10
      IB=NSRCH-10*IA
      NN=N-1
      NNN=N+1
      LL=L-1
      LLL=L+1
      MFF   =MF-1
      DO 404 M=MF,LL
      ICHECK=110
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      MMM=M+1
      MM=M-1
      ALPHA=0.5
      TEMP4=C2*((1.-ALPHA)*(U(N,MMM)-U(N,M))+ALPHA*(U(N,M)-U(N,MM)))
      TEMP1 =U(N,M)**2+(((V(N,M)+V(N,MMM)+V(NN,M)+V(NN,MMM))**2)/16.)
      TEMP2=(SEP(N,M)+SEP(N,MMM)+H(N,M)+H(NN,M))*(C(N,M)+C(N,MMM))**2
      TEMP12=-C6*WX*ABS(WX)/(SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))
      TEMP3=1.+C4*SQRT(TEMP1)/TEMP2+TEMP4+TEMP12
      TEMP3=1./TEMP3
      GAMMA=0.5
      TEMP10=U(NNN,M)
      IF(TEMP10.EQ.0.) TEMP10=  U(NN,M)
  978 TEMP11=U(NN,M)
      IF(TEMP11.EQ.0.) TEMP11=  U(NNN,M)
  980 TEMP1=AT*F(N)-(1.-GAMMA)*C2*(TEMP10-U(N,M))
     1-GAMMA*C2*(U(N,M)-TEMP11)
      TEMP1=.25*TEMP1
  404 UP(N,M)=TEMP3*
     1    (U(N,M)+TEMP1*(VP(N,M)+VP(N,MMM)+VP(NN,M)+VP(NN,MMM))
     2- C1*(SE(N,MMM)-SE(N,M)))
      IF(IB.EQ.1) GO TO 405
      IF(IB.EQ.2) GO TO 420
      GO TO 406
  405 CONTINUE
      TEMP10=U(NNN,L)
      IF(TEMP10.EQ.0.) TEMP10=  U(NN,L)
 1001 TEMP11=U(NN,L)
      TEMP4=C2*ALPHA*(U(N,L)-U(N,LL))
```

```
      TEMP1=U(N,L)**2+((((V(N,L)+V(NN,L))**2)/16.)
      TEMP2=(SEP(N,L)+SEP(N,LLL)+H(N,L)+H(NN,L))*(C(N,L)+C(N,LLL))**2
      M=L
      MMM=LLL
      ICHECK=120
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      TEMP12=-C6*WX*ABS(WX)/(SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))
      TEMP3=1.+C4*SQRT(TEMP1)/TEMP2+TEMP4 +TEMP12
      TEMP3=1./TEMP3
      GAMMA=0.5
      TEMP1=.25*(AT*F(N)-(1.-GAMMA)*C2*(TEMP10-U(N,L))-GAMMA*C2*
     1(U(N,L)-TEMP11))
      UP(N,L)=TEMP3*(U(N,L)+TEMP1*(VP(N,L)+VP(NN,L))
     1-C1*(SE(N,LLL)-SE(N,L)))
       GO TO 406
  420 LLL=L+1
      LL=L-1
      SEP(N,LLL)=2.*SEP(N,L)-SEP( N,LL)
  406 IF(IA.EQ.1) GO TO 407
      GO TO 408
 407  MFF=MF-1
      TEMP10=U(NNN,MFF)
      IF(TEMP10.EQ.0.) TEMP10=  U(NN,MFF)
 1006 TEMP11=U(NN,MFF)
      IF(TEMP11.EQ.0.) TEMP11=  U(NNN,MFF)
 1008 ALPHA=1.
      TEMP4=C2*(1.-ALPHA)*(U(N,MF)-U(N,MFF))
      TEMP1=U(N,MFF)**2+(((V(N,MF)+V(NN,MF))**2)/16.)
      TEMP2=(SEP(N,MFF)+SEP(N,MF)+H(N,MFF)+H(NN,MFF))*(C(N,MF)+C(N,MFF)
     1)**2
      M=MFF
      MMM=MF
      ICHECK=130
      IF(TEST) WRITE(6,6100) N,M,NST,ICHECK
      TEMP12=-C6*WX*ABS(WX)/(SE(N,M)+SE(N,MMM)+H(N,M)+H(NN,M))
      TEMP3=1.+C4*SQRT(TEMP1)/TEMP2+TEMP4 +TEMP12
      TEMP3=1./TEMP3
      GAMMA=0.5
      TEMP1=.25*(AT*F(N)-(1.-GAMMA)*C2*(TEMP10-U(N,MFF))-GAMMA*C2*
     1(U(N,MFF)-TEMP11))
      UP(N,MFF)= TEMP3*(U(N,MFF)+TEMP1*(VP(N,MF)+VP(NN,MF))
     1-C1*(SE(N,MF)-SE(N,MFF)))
  408 CONTINUE
      NUM=NUM+1
      GO TO 340
  402 CONTINUE
      GO TO 500
  410 CONTINUE
      CALL ANLYZE(MAXST,NTERM,AT)
      STOP
      END
C
C
      SUBROUTINE PRINTO(NST,MMAX,NMAX,CMAG,CDIM,AT,CMAGSE)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48)
      DIMENSION KONVRT(25)
      TIME=NST
      TIME=TIME*2.*AT/3600.
      CDIM=3.0000
```

```
      WRITE(6,5020) NST,TIME
 5020 FORMAT(1H1,43HAVERAGED SE AND SEP FOR SECOND HALF OF STEP,I5,
     1 5X,'TIME = ',F6.2,' HRS')
      DO 6000 M=1,MMAX
      DO 6006 N=1,NMAX
 6006 KONVRT(N)=(SE(N,M)+SEP(N,M))*50.*CDIM*CMAGSE
 6000 WRITE(6,6001)M,(KONVRT(N),N=1,NMAX)
 6001 FORMAT(1H ,I2,1X,32I4)
      WRITE(6,5021) NST,TIME
 5021 FORMAT(1H1,41HAVERAGED V AND VP FOR SECOND HALF OF STEP,I5,
     1 5X,'TIME = ',F6.2,' HRS')
      DO 6003 M=1,MMAX
      DO 6007 N=1,NMAX
 6007 KONVRT(N)=(V(N,M)+VP(N,M))*50.*CDIM*CMAG
 6003 WRITE(6,6001) M,(KONVRT(N),N=1,NMAX)
      WRITE(6,5022) NST,TIME
 5022 FORMAT(1H1,41HAVERAGED U AND UP FOR SECOND HALF OF STEP,I5,
     1 5X,'TIME = ',F6.2,' HRS')
      DO 6004 M=1,MMAX
      DO 6008 N=1,NMAX
 6008 KONVRT(N)=(U(N,M)+UP(N,M))*50.*CDIM*CMAG
 6004 WRITE(6,6001) M,(KONVRT(N),N=1,NMAX)
      RETURN
      END
C     SUBROUTINE KURIH
      SUBROUTINE KURIH(MAXST,AT,NTERM,TS,YR,DAY,THR,TMIN)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARGN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DIMENSION FA(20),CT(20),CS(20),CH(20),CP(20),CP1(20),C90(20),
     1 CXI(20),CNU(20),CR(20),G(20),NODE(20),TITLE(20)
      REAL LINEN(40),LINEP(40),LINEB(40)
      DATA BLANK,DOT,STAR/' ','.','*'/
      NTERM=17
      DO 16 J=1,NTERM
   16 READ(5,18) I,TITLE(J),Z(J),HP(J),HB(J),E(J),EP(J),EB(J)
   18 FORMAT(I2,A4,3F5.3,3F5.1)
      DO 30 J=1,NTERM
   30 READ(5,101)I,W(J),CT(J),CS(J),CH(J),CP(J),CP1(J),C90(J),CXI(J),
     2CNU(J),CR(J),NODE(J)
  101 FORMAT(I2,F10.7,9F3.0,I1)
      DO 175 J=1,NTERM
  175 READ(5,176)I,TITLE(J),HL(J),EL(J)
  176 FORMAT(I2,A4,F6.4,F7.1)
      WRITE(6,29)
   29 FORMAT(1H1,/,5X,'T I D A L   C O N S T I T U A N T   P A R A M E T',
     1' E T E R S')
      WRITE(6,99)
   99 FORMAT( / ,15X,'NEWPORT',6X,'PROVID',9X,'BRISTOL',15X,'ARGUEMENT',
     1 35X,'NODE')
      WRITE(6,120)
  120 FORMAT(5X,'NAME',5X,3('H',7X,'E',5X),'SPEED',5X,'T',4X,'S',4X,'H',
     1 4X,'P',4X,'P1',3X,'90',3X,'XI',3X,'NU',3X,'R',2X,'FORMULA')
      DO 121 J=1,NTERM
  121 WRITE(6,122) J,TITLE(J),Z(J),E(J),HP(J),EP(J),HB(J),EB(J),W(J),
     1 CT(J),CS(J),CH(J),CP(J),CP1(J),C90(J),CXI(J),CNU(J),CR(J),NODE(J)
```

```
 122  FORMAT(2X,I2,1X,A4,2X,3(F6.2,1X,F6.1,1X),2X,F8.5,2X,9(F3.0,2X),I2)
       WRITE(6,177)
177    FORMAT(//,10X,'LOWER BOUNDARY')
       WRITE(6,178)
178    FORMAT(5X,'NAME',5X,'H',7X,'E')
       DO 179 J=1,NTERM
179    WRITE(6,180)J,TITLE(J),HL(J),EL(J)
180    FORMAT(2X,I2,1X,A4,2X,F6.4,1X,F6.1)
      RAD=57.2957795
C
      IYR=YR/4.
      HDAY=183.0
      FIYR=4*IYR
      GDAY=1461*IYR
      IF(FIYR.EQ.YR) GO TO 57
      HDAY=182.5
  55  GDAY=GDAY+365.
      FIYR=FIYR+1.
      IF(FIYR.GT.99.) STOP
      IF(FIYR.EQ.YR) GO TO 56
      GO TO 55
  56  GDAY=GDAY+1.
  57  CONTINUE
      DAY=DAY-1.00
      GDAY=GDAY-1.00+.500
      ZS=(GDAY+HDAY)/36525.
      ZT=GDAY/36525.
      T1=24.000*DAY+THR+TMIN/60.
      T2=THR+TMIN/60.
      T3=180.0/RAD
      T4=T1-1./60.
C
      AN=(259.182-(134.142-.0021*ZT)*ZT)/RAD-31.41592*ZT
      AAH=ZT*100.
      IH=AAH
      AAH=AAH-IH
      AH=(279.697+0.7690*ZT+(.00030    )*ZT**2)/RAD+AAH*(360./RAD)
      AP=(334.328+(109.032-.01034*ZT)*ZT)/RAD+69.11503*ZT
      AP1=(281.221+(1.719+.00045*ZT)*ZT)/RAD
      AAS=ZT*1336.
      IS=AAS
      AAS=AAS-IS
      AS=( 270.434+307.892*ZT+(.00252   )*ZT**2)/RAD+AAS*(360./RAD)
      CSI=0.91370-0.03569*COS(AN)
      SNI=SQRT(1.00-CSI**2)
      TNI=SNI/CSI
      AI=ATAN(TNI)
      TNI2=SIN(AI/2.)/COS(AI/2.)
      CTNI2=1./TNI2
      TNN2=SIN(AN/2.)/COS(AN/2.)
      AXI=AN-ATAN(1.01883*TNN2)-ATAN(0.64412*TNN2)
      ANU=ATAN(1.01883*TNN2)-ATAN(0.64412*TNN2)
      APCAP=AP-AXI
      ARAINV=SQRT(1.-12.*(TNI2**2)*COS(2.*APCAP)+36.*TNI2**4)
      ARCAP=ATAN(SIN(2.*APCAP)/(CTNI2**2/6.      -COS(2.*APCAP)))
C
      BN=(259.182-(134.142-.0021*ZS)*ZS)/RAD-31.41592*ZS
      BP=(334.328+(109.032-.01034*ZS)*ZS)/RAD+69.11503*ZS
      CSI=0.91370-0.03569*COS(BN)
      SNI=SQRT(1.00-CSI**2)
```

```
      TNI=SNI/CSI
      BI=ATAN(TNI)
      TNI2=SIN(BI/2.)/COS(BI/2.)
      CTNI2=1./TNI2
      TNN2=SIN(BN/2.)/COS(BN/2.)
      BXI=BN-ATAN(1.01883*TNN2)-ATAN(0.64412*TNN2)
      BNU=ATAN(1.01883*TNN2)-ATAN(0.64412*TNN2)
      BPCAP=AP-BXI
      BRCAP=ATAN(SIN(2.*BPCAP)/(CTNI2**2/6.-COS(2.*BPCAP)))
C
      BBNU=BNU
      DO 32 J=1,NTERM
      G(J)=0.00
      IF(J.EQ.5) GO TO 60
      GO TO 65
   60 BBNU=BNU
      BNU=ATAN((SIN(2.*BI)*SIN(BNU))/(SIN(2.*BI)*COS(BNU)+0.3347))
   65 IF(J.EQ.8) GO TO 66
      GO TO 70
   66 BBNU=BNU
      BNU=0.50*ATAN((SIN(BI)**2*SIN(2.*BNU))/(SIN(BI)**2*COS(2.*BNU)+
     1 0.0727))
   70 G(J)=CT(J)*T3+CS(J)*AS+CH(J)*AH+CP(J)*AP+CP1(J)*AP1+C90(J)*90./
     2RAD+CXI(J)*BXI+CNU(J)*BNU+CR(J)*BRCAP
      W(J)=W(J)/RAD
      ARN(J)=W(J)*T1+G(J)-E(J)/RAD
       ARGLB(J)=W(J)*T4+G(J)-EL(J)/RAD
      ARGP(J)=W(J)*T1+G(J)-EP(J)/RAD
      ARGB(J)=W(J)*T1+G(J)-EB(J)/RAD
      BNU=BBNU
   32 CONTINUE
C
      ZT=(GDAY+DAY)/36525.
      AP=(334.328+(109.032-.01034*ZT)*ZT)/RAD+69.11503*ZT
      AN=(259.182-(134.142-.0021*ZT)*ZT)/RAD-31.41592*ZT
      CSI=0.91370-0.03569*COS(AN)
      SNI=SQRT(1.00-CSI**2)
      TNI=SNI/CSI
      AI=ATAN(TNI)
      TNI2=SIN(AI/2.)/COS(AI/2.)
      TNN2=SIN(AN/2.)/COS(AN/2.)
      ANU=ATAN(1.01883*TNN2)-ATAN(0.64412*TNN2)
      AXI=AN-ATAN(1.01883*TNN2)-ATAN(0.64412*TNN2)
      APCAP=AP-AXI
      ARAINV=SQRT(1.-12.*(TNI2**2)*COS(2.*APCAP)+36.*TNI2**4)
C
C            S2,SA,P1,S1,T2
      FA(1)=1.000
C
C     ( 78) M2,(2N2),N2,MU2,NU2,
      FA(2)=COS(AI/2.)**4/0.9154
C
C            M4
      FA(3)=FA(2)**2
C
C            M6
      FA(4)=FA(2)**3
C
C     (2,15) L2
      FA(5)=FA(2)*ARAINV
```

```
C
C       ( 75) 01,01
        FA(6)=SIN(AI)*COS(AI/2.)**2 /C.3800
C
C       (227) K1
        FA(7)=SQRT(.8965*SIN(2.*AI)**2+0.6001*SIN(2.*AI)*COS(ANU)+0.1006)
C
C       (235) K2
        FA(8)=SQRT(19.0444*SIN(AI)**4+2.7702*SIN(AI)**2*COS(2.*ANU)+.0981)
        DO 40 J=1,NTERM
        JJ=NODE(J)
   40   F2(J)=FA(JJ)
        C2=1.00
        C3=60.00
        YP=3.0
        T=0.0000
        DAY=DAY+1.
        WRITE(6,28)
   28 FORMAT(1H1,5X,'T I D A L   C U R V E S ')
        WRITE(6,22) C3,YR,DAY,THR,TMIN
   22   FORMAT(/,5X,' TIME INTERVAL=',F3.0,5X,' YEAR=',F3.0,' DAY=',F4.0,
      1 ' HR=',F3.0,' MIN=',F3.0)
        WRITE(6,23)
   23   FORMAT(/,18X,'NEWPORT',33X,'BRISTOL',28X,'PROVIDENCE')
        WRITE(6,24) YP
   24   FORMAT(33X,F4.1,' FT')
        WRITE(6,26)
   26   FORMAT(2X,3('I',36('.'),'I'))
        DO 50 K=1,39
        LINEN(K)=BLANK
        LINEP(K)=BLANK
   50   LINEB(K)=BLANK
        NHR=180C./AT
        DO 25 J=1,MAXST,NHR
        S=0.0
        SLB=0.000
        SP=0.0
        SB=0.0
        T2=T-1./60.
        DO 21 I=1,NTERM
        SP=F2(I)*HP(I)*COS(W(I)*T+ARGP(I))+SP
        SB=F2(I)*HB(I)*COS(W(I)*T+ARGB(I))+SB
        SLB=F2(I)*Z(I)*COS(W(I)*T2+ARGLB(I))+SLB
   21   S= F2(I)*Z(I)*COS(W(I)*T+ARN(I))+S
        TIDEN   =S
        TIDELB    =SLB/3.0
        TIDEP    =SP
        TIDEB    =SB
        LINEN(20)=DOT
        LINEP(20)=DOT
        LINEB(20)=DOT
        KN=(TIDEN/YP)*19.+20.
        KP=(TIDEP/YP)*19.+20.
        KB=(TIDEB/YP)*19.+20.
        LINEN(KN)=STAR
        LINEP(KP)=STAR
        LINEB(KB)=STAR
        T=T+C2
        L=J-1
        WRITE(6,52)(LINEN(K),K=1,38),(LINEB(K),K=1,38),(LINEP(K),K=1,38),L
```

```
      LINEN(KN)=BLANK
      LINEP(KP)=BLANK
      LINEB(KB)=BLANK
   25 CONTINUE
  52  FORMAT(2X,114A1,I3)
C
      TS=0.
      CCHECK=COS(ARN(1))
      IF(CCHECK.LT.0.) GO TO 11
      GO TO 12
   11 TS=6.21
   12 CCHECK=COS(W(1)*TS+ARN(1))
      SCHECK=SIN(W(1)*TS+ARN(1))
      FCHECK=ATAN(SCHECK/CCHECK)*12.42/(6.28318)
      TS=TS-FCHECK
      RETURN
      END
C
C     SUBROUTINE DIVF
      SUBROUTINE DIVF(NMAX,MMAX)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DIMENSION NO(40)
      WRITE(6,5)
      DO 1 N=1,NMAX
    1 NO(N)=N
      WRITE(6,6) (NO(N),N=1,NMAX)
      DO 2 M=1,MMAX
      READ(5,3) (IFIELD(N,M),N=1,NMAX)
      DO 10 N=1,NMAX
      NBD(N)=IFIELD(N,M)
      IF(NBD(N).EQ.2) NBD(N)=0
   10 CONTINUE
      WRITE(6,4) M,(IFIELD(N,M),N=1,NMAX)
      DO 2 N=1,NMAX
    2 H(N,M)=FLOAT(NBD(N))
      RETURN
    3 FORMAT(32I2)
    4 FORMAT(1H ,I2,3X,32I2)
    5 FORMAT(1H1,10X,21HWATER LEVELS IN FIELD)
    6 FORMAT(1H0,2H M,3X,32I2)
      END
C
C           SUBROUTINE  FIND
      SUBROUTINE FIND(MIND,NIND,MMAX,NMAX,MINDO,NINDO,NSECT)
      LOGICAL START
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DO I J=1,NSECT
      NBD(J)=0
```

```
    1 MBD(J)=0
      MIND=1
      NIND=1
      DO 2 N=2,NMAX
       START=.TRUE.
      DO 3 M=2,MMAX
      IF(.NOT.START) GO TO 4
      IF(H(N,M).EQ.0.)GO TO 3
      NBD(NIND)=M*100+NBD(NIND)
      START=.FALSE.
      GO TO 3
    4 IF(H(N,M).NE.0.)GO TO 5
      NBD(NIND)=M-1+NBD(NIND)+10000*N
      GO TO 6
    5 IF(M.NE.MMAX) GO TO 3
      NBD(NIND)= M+NBD(NIND)+10000*N
    6 NIND=NIND+1
      START=.TRUE.
    3 CONTINUE
    2 CONTINUE
      DO 12 M=2,MMAX
      START=.TRUE.
      DO 13 N=2,NMAX
      IF(.NOT.START)GO TO 14
      IF( H(N,M).EQ.0.) GO TO 13
      MBD(MIND)=N*100+MBD(MIND)
      START=.FALSE.
      GO TO 13
   14 IF(H(N,M).NE.0.) GO TO 15
      MBD(MIND)=N-1+MBD(MIND)+10000*M
      GO TO 16
   15 IF(N.NE.NMAX) GO TO 13
      MBD(MIND)=N+MBD(MIND)+10000*M
   16 MIND=MIND+1
      START=.TRUE.
   13 CONTINUE
   12 CONTINUE
      NUM=1
  100 IF(NUM.EQ.NIND) GO TO 300
      N=  NBD(NUM)/10000
      MF = NBD(NUM)/100  -N*100
      L  =NBD(NUM)-N*10000-MF*100
      MFLEF=MF-1
      LRIG=L+1
      NA=1
  200 IF(NA.EQ.MINDO)GO TO  210
      M=MOBD(NA)/100000
      NTOP=MOBD(NA)/10  -M*10000 -NBOT*100
      NBOT=MOBD(NA)/1000 -M*100
      IF(NBERN.GT.1) GO TO 501
      IF(((N.GE.NBOT).AND.(N.LE.NTOP)).AND.(MFLEF.EQ.M)) NBD(NUM)=
     1NBD(NUM) +10000000
      IF(((N.GE.NBOT).AND.(N.LE.NTOP)).AND.(LRIG.EQ.M))  NBD(NUM)=
     1NBD(NUM)+1000000
      GO TO 205
  501 IF(((N.GE.NBOT).AND.(N.LE.NTOP)).AND.(MFLEF.EQ.M)) NBD(NUM)=
     1NBD(NUM) +20000000
      IF(((N.GE.NBOT).AND.(N.LE.NTOP)).AND.(LRIG.EQ.M))  NBD(NUM)=
     1NBD(NUM)+2000000
  205 NA=NA+1
```

```
      GO TO 200
  210 NUM=NUM+1
      GO TO 100
  300 CONTINUE
      NUM=1
  101 IF(NUM.EQ.MIND) GO TO 301
      M=MBD(NUM)/10000
      NF=MBD(NUM)/100  -M*100
      L=MBD(NUM)-M*10000-NF*100
      NFBOT=NF-1
      LTOP=L+1
      NA=1
  201 IF(NA.EQ.NIND)GO TO 211
      N=NOBD(NA)/100000
      MLEF=NOBD(NA)/1000-N*100
      MRIG=NOBD(NA)/10-N*10000-MLEF*100
      MBERN=NOBD(NA)-N*100000-MLEF*1000-MRIG*10
      IF(MBERN.GT.1) GO TO 502
      IF(M.GE.MLEF.AND.M.LE.MRIG.AND.NFBOT.EQ.N) MBD(NUM)=MBD(NUM)
     1     +  10000000
       IF(M.GE.MLEF.AND.M.LE.MRIG.AND.LTOP.EQ.N) MBD(NUM)=MBD(NUM)
     1  +1000000
      GO TO 206
  502 IF(M.GE.MLEF.AND.M.LE.MRIG.AND.NFBOT.EQ.N) MBD(NUM)=MBD(NUM)
     1     +  20000000
       IF(M.GE.MLEF.AND.M.LE.MRIG.AND.LTOP.EQ.N) MBD(NUM)=MBD(NUM)
     1  +2000000
  206 NA=NA+1
      GO TO 201
  211 NUM=NUM+1
      GO TO 101
  301 CONTINUE
      WRITE(6,20)
      NHAF=NSECT/2
      DO 22 J=1,NHAF
      JJ=J+NHAF
   22 WRITE(6,21) J,NBD(J),MBD(J),JJ,NBD(JJ),MBD(JJ)
      WRITE(6,30) NIND,MIND
   30 FORMAT(/,2X,'NIND = ',I2,5X,'MIND = ',I2)
   21 FORMAT(2X,I4,2X,I9,1X,I9,10X,I4,2X,I9,1X,I9)
   20 FORMAT(1H1,3X,3HNUM,6X,3HNBD,7X,3HMBD,13X,3HNUM,6X,3HNBD,7X,3HMBD)
      RETURN
      END
C
C     SUBROUTINE DEPTH
      SUBROUTINE DEPTH(NMAX,MMAX)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19-48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DIMENSION NH(21)
      WRITE(6,5)
    5 FORMAT(1H1,/,2X,'INITIAL DEPTHS (FEET) AT MEAN LOW WATER'/)
      DO 10 M=1,MMAX
      READ(5,20) (H(N,M),N=1,NMAX)
   20 FORMAT(20F4.0)
      DO 15 N=1,NMAX
```

```
   15 NH(N)=H(N,M)      +.01
   10 WRITE(6,30) M,(NH(N),N=1,NMAX)
   30 FORMAT(1X,I2,1X,21I4)
C     ADD MEAN TIDE
      WRITE(6,35)
   35 FORMAT(1H1,/5X,'INITIAL DEPTHS (FEET TIMES 10) AT MSL ',40X,'AVG.
     1DEPTH'/)
      GNT=0.
      TAV=0.0
      DO 50 M=1,MMAX
      B=MMAX
      CX=M
      A=1.8+0.8*(B-CX)/B
      GN=0.
      DAV=0.0
      DO 45 N=1,NMAX
      IF(H(N,M).EQ.0.0) GO TO 41
      H(N,M)=H(N,M)+A
   41 IF(IFIELD(N,M).EQ.0) GO TO 45
      NN=N-1
      DAV=DAV+.5*(H(N,M)+H(NN,M))
      GN=GN+1.
   45 CONTINUE
      TAV=TAV+DAV
      IF(GN.GT.0.) DAV=DAV/GN
      IF(GN.EQ.0.) DAV=0.
      GNT=GNT+GN
      DO 46 N=1,NMAX
   46 NH(N)=H(N,M)*10.+.01
      WRITE(6,47) M,(NH(N),N=1,NMAX),DAV
   47 FORMAT(1X,I2,1X,19I4,5X,F5.1)
C     CONVERT TO YARDS
      DO 48 N=1,NMAX
   48 H(N,M)=H(N,M)/3.
   50 CONTINUE
      WRITE(6,60) GNT
   60 FORMAT(/4X,'TOTAL NUMBER OF GRIDS (EXCLUDING BOUNDS)',3X,F4.0)
      TAV=TAV/GNT
      WRITE(6,70) TAV
   70 FORMAT(/,4X,'AVG. DEPTH OF BAY',2X,F5.2)
      RETURN
      END
C
C     SUBROUTINE CHEZY
      SUBROUTINE CHEZY(NMAX,MMAX,CMANN)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      F1=.3
      DO 50 M=1,MMAX
      F3=CMANN*(1. + F1*(1.-(2.*M)/(1.*MMAX)))
      DO 40 N=1,NMAX
      NN=N-1
      MM=M-1
      IF(N.EQ.1) GO TO 10
      IF(IFIELD(N,M).EQ.0) GO TO 10
```

```
   20   IF(M.EQ.1) GO TO 30
        A=H(N,MM)+H(NN,MM)
        GO TO 35
   30   A=H(N,M)+H(NN,M)
   35   A=(A+H(N,M)+H(NN,M))*.25
        C(N,M)=1.49*A**(1./6.)/(F3*1.732)
        GO TO 37
   10   C(N,M)=0.0
   37   CONTINUE
   40   CONTINUE
   50   CONTINUE
        RETURN
        END
C
C        SUBROUTINE VELANA
        SUBROUTINE VELANA(IVL,IRMS,MAXST,CMAG,WAVG)
        COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
       1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
       2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
       3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
       4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
       5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
       6 ZIC(1000)
        DIMENSION KONVRT(19), WAVG(19,48)
        DATA NMAX,MMAX/19,48/
        DO 10 M=1,MMAX
        DO 10 N=1,NMAX
        NM=N-1
        MM=M-1
        IF(M.EQ.1) MM=M
        UAVG(N,M)=UAVG(N,M)+U(N,M)
        VAVG(N,M)=VAVG(N,M)+V(N,M)
   10   CONTINUE
        IF(IVL.EQ.MAXST) GO TO 25
        GO TO 60
   25   CONTINUE
        F1=MAXST-IRMS+1
        WRITE(6,102)   CMAG
        DO 35 M=1,MMAX
        DO 30 N=1,NMAX
        UAVG(N,M)=UAVG(N,M)*3./F1
   30   KONVRT(N)=UAVG(N,M)*CMAG
        WRITE(6,100) M,(KONVRT(N),N=1,NMAX)
        WRITE(7,101)(UAVG(N,M),N=01,08)
        WRITE(7,101)(UAVG(N,M),N=09,15)
        WRITE(7,101)(UAVG(N,M),N=16,19)
   35   CONTINUE
        WRITE(6,103) CMAG
        DO 45 M=1,MMAX
        DO 40 N=1,NMAX
        VAVG(N,M)=VAVG(N,M)*3./F1
   40   KONVRT(N)=VAVG(N,M)*CMAG
        WRITE(6,100) M,(KONVRT(N),N=1,NMAX)
        WRITE(7,101)(VAVG(N,M),N=01,08)
        WRITE(7,101)(VAVG(N,M),N=09,15)
        WRITE(7,101)(VAVG(N,M),N=16,19)
   45   CONTINUE
   60   CONTINUE
  100   FORMAT(5X,I2,5X,20I5)
  101   FORMAT(8E10.4)
```

```
  102 FORMAT(1H1,//,5X,'U MEAN TIMES ',E10.4,/)
  103 FORMAT(1H1,//,5X,'V MEAN TIMES ',E10.4,/)
      RETURN
      END
C
C
C     SUBROUTINE ANLYZE(MAXST,NTERM,AT)
      SUBROUTINE ANLYZE(MAXST,NTERM,AT)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DIMENSION XIA(1000),ALINE(65)
      DATA BLANK,DOT,STAR/' ','.','*'/
      CT=.33333333
       NSTEP=600./AT
      DO 10 K=1,61
   10 ALINE(K)=BLANK
C
      T=0.
       DO 30 N=1,MAXST,NSTEP
      S=0.
      DO 20 I=1,17
   20 S= F2(I)*Z(I)*COS(W(I)*T+ARN(I))+S
      XIA(N)=S
   30 T=T+CT
      ZA=0.
      DO 40 N=1,MAXST,NSTEP
      IF(ABS(ZIA(N)).GT.ZA) ZA=ABS(ZIA(N))
   40 IF(ABS(XIA(N)).GT.ZA) ZA=ABS(XIA(N))
      ZS=ZA
      WRITE(6,45)
      WRITE(6,46)
      DO 60 N=1,MAXST,NSTEP
      ALINE(31)=DOT
      JM=31.+(ZIA(N)/ZS)*30.
      JS=31.+(XIA(N)/ZS)*30.
      ALINE(JM)=STAR
      ALINE(JS)=DOT
      WRITE(6,50) N,ZIA(N),XIA(N),(ALINE(J),J=1,61)
      ALINE(JM)=BLANK
      ALINE(JS)=BLANK
   60 CONTINUE
C
      T=0.0
      DO 90 N=1,MAXST,NSTEP
      SB=0.0
      DO 80 I=1,17
   80 SB=F2(I)*HB(I)*COS(W(I)*T+ARGB(I))+SB
      DO 100 N=1,MAXST,NSTEP
      IF(ABS(ZIB(N)).GT.ZA) ZA=ABS(ZIB(N))
  100 IF(ABS(XIA(N)).GT.ZA) ZA=ABS(XIA(N))
   90 T=T+CT
      ZS=ZA
      WRITE(6,45)
      WRITE(6,65)
      DO 110 N=1,MAXST,NSTEP
```

```
      ALINE(31)=DOT
      JM=31.+(ZIB(N)/ZS)*30.
      JS=31.+(XIA(N)/ZS)*30.
      ALINE(JM)=STAR
      ALINE(JS)=DOT
      WRITE(6,50) N,ZIB(N),XIA(N),(ALINE(J),J=1,61)
      ALINE(JS)=BLANK
      ALINE(JM)=BLANK
  110 CONTINUE
C
      T=0.
      DO 150 N=1,MAXST,NSTEP
      SP=0.0
       DO 140 I=1,17
  140 SP=F2(I)*HP(I)*COS(W(I)*T+ARGP(I))+SP
      XIA(N)=SP
  150 T=T+CT
      ZA=0.0
      DO 160 N=1,MAXST,NSTEP
      IF(ABS(ZIC(N)).GT.ZA) ZA=ABS(ZIC(N))
  160 IF(ABS(XIA(N)).GT.ZA) ZA=ABS(XIA(N))
      ZS=ZA
      WRITE(6,45)
      WRITE(6,48)
      DO 170 N=1,MAXST,NSTEP
      ALINE(31)=DOT
      JS=31.+(XIA(N)/ZS)*30.
      JM=31.+(ZIC(N)/ZS)*30.
      ALINE(JM)=STAR
      ALINE(JS)=DOT
      WRITE(6,50) N,ZIC(N),XIA(N),(ALINE(J),J=1,61)
      ALINE(JS)=BLANK
      ALINE(JM)=BLANK
  170 CONTINUE
C
   45 FORMAT(1H1,/,12X,'MODEL',5X,'SERIES',10X,'WATER LEVEL CURVE AT')
   46 FORMAT(52X,'NEWPORT'/)
   48 FORMAT(52X,'PROVIDENCE')
   65 FORMAT(52X,'BRISTOL')
   50 FORMAT(5X,I3,5X,2(F6.2,3X),'I',61A1,'I')
      RETURN
      END
C     SUBROUTINE CHECK(NMAX,MMAX)
      SUBROUTINE CHECK(NMAX,MMAX)
      COMMON SE(19,48),SEP(19,48),V(19,48),VP(19,48),
     1 U(19,48),UP(19,48),C(19,48),H(19,48),IFIELD(19,48),
     2 NBD(85),MBD(85),NOBD(4),MOBD(4),UAVG(19,48),
     3 VAVG(19,48),ARGLB(20),ARN(20),ARGB(20),ARGP(20),
     4 HL(20), Z(20),HB(20),HP(20),EL(20), E(20),
     5 EB(20),EP(20),F2(20),W(20),ZIA(1000),ZIB(1000),
     6 ZIC(1000)
      DIMENSION ALINE(100)
      DATA BLANK,DOT,ZERO,AH,AC,AB/' ','.','O','H','C','B'/
      WRITE(6,5)
    5 FORMAT(1H1,/,5X,'FIELD DIAGNOSTIC ')
      DO 40 M=1,MMAX
      DO 20 N=1,NMAX
      ALINE(N)=BLANK
      IF(IFIELD(N,M).NE.0) GO TO 10
      IF(H(N,M).NE.0.0) ALINE(N)=DOT
```

```
      GO TO 20
   10 CONTINUE
      ALINE(N)=ZERO
      IF(H(N,M).EQ.0.0) ALINE(N)=AH
      IF(C(N,M).EQ.0.0) ALINE(N)=AC
      IF(H(N,M).EQ.0.0.AND.C(N,M).EQ.0.0) ALINE(N)=AB
   20 CONTINUE
      WRITE(6,30) M,(ALINE(N),N=1,NMAX)
 30   FORMAT(1X,I3,2X,50(A1,1X))
   40 CONTINUE
      RETURN
      END
```

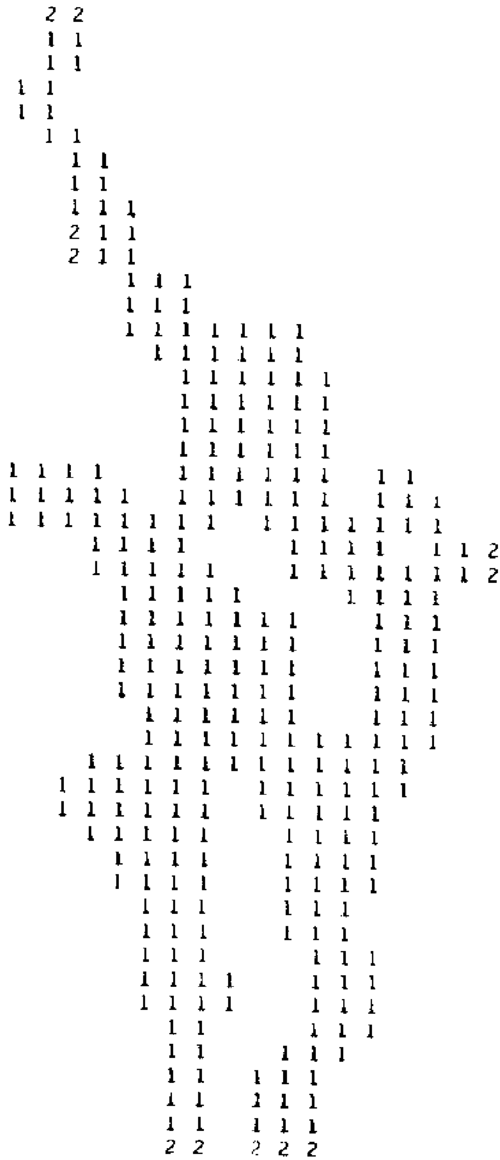


Fig. C-1. The interior computational field for Narragansett Bay. Grids assigned the number 1 represent water; those assigned the number 2, the presence of an open boundary.

```
             4.   4.
             4.   4.
             4.   4.
       30.   4.
        4.  30.
             4.  35.
                  30.   4.
                   6.  35.
                   8.  35.   4.
                   4.  35.   5.
                   4.  10.  20.
                            35.   5.   4.
                            27.  20.  12.
                             9.  28.  35.  20.  11.  15.   4.
                                  4.   7.  28.  16.  16.  17.  20.
                                       9.  12.  23.  20.  19.  15.
                                      15.  20.  20.  16.  24.  19.
                                      15.  21.  17.  17.  27.  22.
                                  8.  17.  28.  24.  19.  21.   4.
  10.  12.  12.   5.   4.       10.  15.  37.  20.  19.  22.  35.       16.  20.
   4.  10.   9.  11.   9.  40.  15.  40.  18.   4.  21.  22.  40.       25.  20.   5.
   4.   4.   4.   4.   7.  14.  65.   4.   4.       15.  20.  30.  32.  17.   5.  25.   4.
                   4.  10.  15.  16.  18.  10.            21.  27.  37.  21.  10.  35.  43.  43.
                        9.  21.  19.  20.  23.            13.  15.  37.  25.  30.  80.  10.  10.
                        5.  22.  22.  21.  23.  26.  10.            5.  42.  44.  20.
                            15.  32.  21.  22.  22.  21.  20.            40.  45.  18.
                            17.  25.  23.  21.  27.  23.  26.            80.  40.  25.
                            27.  22.  24.   5.  44.  26.  30.            70.  55.   5.
                        8.  15.  24.  30.   9.  27.  28.  40.           100.  42.   5.
                            10.  23.  22.  22.  24.  30.  45.       40.  51.  52.  30.
                            35.  33.  21.  21.  24.  28.  29.  21.  90.  23.  22.  30.
                   5.   7.  17.  24.  22.  20.  27.  34.  30.  37. 100.  53.   5.
             14.  13.  15.  21.  27.  30.   4.       44.  35.  35.  90.  48.   5.
                  13.  18.  22.  27.  35.   4.       33.  45.  58.  67.  54.
                       15.  27.  25.  40.   4.       50.  53.  50.  65.  25.
                        8.  20.  25.  30.   4.       35.  60.  80.  60.  25.
                             8.  30.  30.   4.            70.  81.  70.  26.
                                 15.  45.   4.            73. 100.  25.
                                 25.  50.   4.            65.  90.  33.
                                 33.  42.  20.            70.  95.  36.   6.
                                 46.   5.  20.   4.       55. 138.  35.  40.
                                 35.  39.  10.   4.       40. 130.  65.  20.
                                 20.  45.   4.            90. 100.  50.  16.
                                      33.  25.       50.  85. 150.   4.
                                      45.  24.       90. 100.   4.
                                 31.  42.  45.      105. 105.   1.
                                 40.  50.  50.  60. 110.  83.  30.
                                 40.  60.  70.  81.  70.  85.  40.
```

Fig. C-2. Depth field for Narragansett Bay. Numbers are mean low water depths in feet.

REFERENCES

1. Leendertse, J.J., Aspects of a Computational Model for Long-Period Water-Wave Propagation, Rand Corporation, Santa Monica, California, Memorandum RM 5294-PR, May, 1967.

2. Reid, R.O. and Bodine, B.R., "Numerical Model for Storm Surges in Galveston Bay", Proc., A.S.C.E. (Wat. and Harb. Div.), V. 94, (WW1), pp 33-57, 1968.

3. Masch, F.D., and Brandes, R.J., Tidal Hydrodynamic Simulation in Shallow Estuaries, University of Texas at Austin, Hydraulic Engin, Lab. Tech. Rept. HYD 12-7102, August, 1971.

4. Mungall, J.C.H., and Matthews, J.B., A Variable-Boundary Numerical Tidal Model, Institute of Marine Science, University of Alaska, College Alaska, Report R70-4, March, 1970.

5. Pritchard, D.W., Two-Dimensional Models, Estuarine Modeling: An Assessment, Tracor, Inc., Austin Texas Report for the Water Quality Office, Environmental Protection Agency, Water Pollution Control Research Series 16070 DZV, Feb., 1971.

6. Sobey, R.J., Finite-Difference Schemes Compared for Wave-Deformation Characteristics in Mathematical Modeling of Two-Dimensional Long-Wave Propagation, Coastal Engineering Research Center, Tech. Memo No. 32, Oct., 1970.

7. Grimsrud, G.P., Mathematical Modeling of Tidal Heights andCurrents in the Point Judith Harbor of Refuge, M.S. Thesis, University of Rhode Island, 1970.

8. Mitchell, A.R., Computational Methods in Partial Differential Equations, J. Wiley, 1969, p 255.

9. Henderson, F.M., Open Channel Flow, MacMillan Co., 1966.

10. N.O.A.A. (private communication).

11. U.S. Dept. of Commerce, Tidal Current Tables, Atlantic Coast of North America, N.O.A.A., 1971.

12. Schureman, P., Manual of Tide Observations, U.S. Coast and Geodetic Survey, Sp. Pub. No. 198, 1941.

13. U.S. Geological Survey, Surface Water Records for Mass., New Hampshire, Rhode Island, and Vermont.

14. Haight, F.J., Currents in Narragansett Bay, Buzzards Bay, and Nantucket and Vineyard Sounds, U.S.C. and G.S. Sp. Pub. No. 208, 1938.

15. Binkerd, R., Some Measurements of Currents on Narragansett Bay Using Floats, with a New Application for Pole, Floats, M.S. Thesis, University of Rhode Island, 1972.

16. Hicks, S.D., The Physical Oceanography of Narragansett Bay, Limnology and Oceanography, V. 4(3), pp 316-327, 1959.

17. Ippen, A.T., Estuary and Coastline Hydrodynamics, McGraw-Hill, 1966, p 744.

18. Levine, E., Tidal Energetics of Narragansett Bay, (Manuscript in preparation), University of Rhode Island Graduate School of Oceanography, 1972.

19. Sturges, W. and Weisberg, R.H., Non-Tidal Transport, West Passage of Narragansett Bay near Rome Point, Rome Point Circulation Study, Final Report, Univ. of R.I., Contract 98-20-7057, July, 1971.

20. Lamb, Sir Horace, Hydrodynamics, Dover Publications, 1945, Section 347.

21. Jones, D., Prediction of Mass Transports in Narragansett Bay Using One Vertical Array of Velocity Meters, M.S. Thesis, University of Rhode Island, 1972.

22. Krabach, M.H., The Feasibility of Measuring Water Transport Through the Entrances of Narragansett Bay, With a Geomagnetic Electrokinetograph, M.S. Thesis, University of Rhode Island, 1972.

23. Rome Point Investigations, Quart. Prog. Rept. of Marine Research, Inc., E. Wareham, Mass., Jan., 1972.

24. Polgar, T.T., Physical Characteristics of Narragansett Bay Relating to the Dispersion of Heated Effluents in the Vicinity of Rome Point. Rome Point, Prelim. Site Evaluation, U.R.I. Contract 98-20-7081, 1970.

25. Hicks, S.D., Frazier, D.E., and Garrison, L.E., Physical Oceanographic Effects of Proposed Hurricane Protection Structures on Narragansett Bay Under Normal Conditions, Hurricane Protection Project, Ref. No. 56, 12, 1956.

26. Fairchild, J.C., Model Study of Wave Setup Induced by Hurricane Waves at Narragansett Pier, Rhode Island, Beach Erosion Board, Bulletin V. 12, July, 1958.

27. Wilson, B.W., The Prediction of Hurricane Storm-Tides in New York Bay, Beach Erosion Board Tech. Memo No. 120, August, 1960.

28. McAleer, J.B., and Townsend, G.E., Hurricane Protection Planning in New England, Proc. of the A.S.C.E., Hydraulics Div., Paper 1726, August, 1958.

29. Neumann, G., and Pierson, W.J., Principles of Physical Oceanography, Prentice-Hall, Inc., 1966, p 545.

30. Bodine, B.R., Storm Surge on the Open Coast: Fundamentals and Simplified Prediction, Coastal Engin. Research Center, Tech. Memo No. 35, May, 1971.

31. Shore Protection, Planning and Design, Coastal Engineering Research Center, Tech. Rept. No. 4, June, 1966.

32. Elder, J.W., The Dispersion of a Marked Fluid in a Turbulent Shear Flow, Journal of Fluid Mechanics, V. 5, 1959.