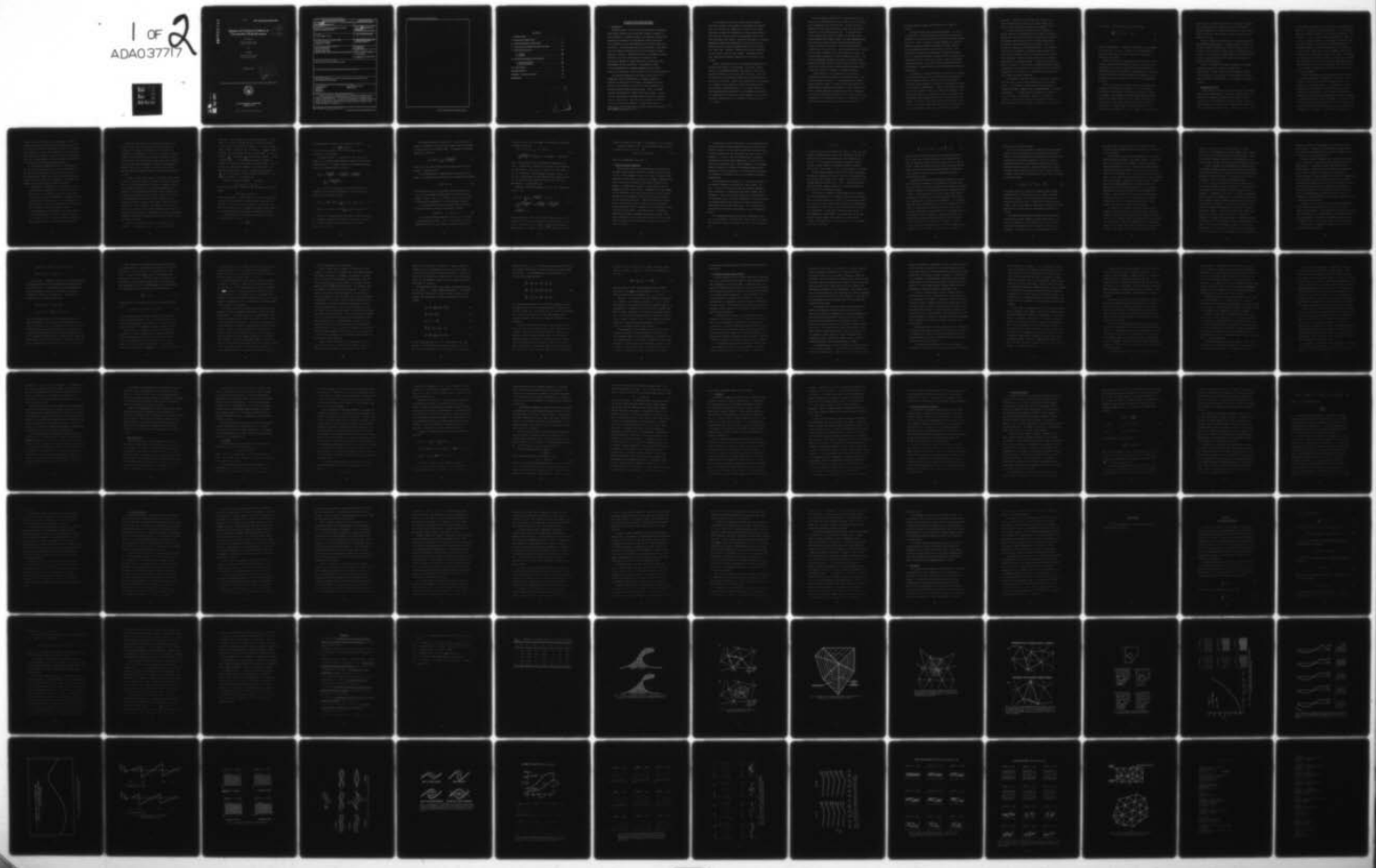
AD-A037 717 NAVAL RESEARCH LAB WASHINGTON D C F/G 20/4
SOLUTION OF TRANSIENT PROBLEMS IN FREE SURFACE HYDRODYNAMICS.(U)
FEB 77 J P BORIS, M J FRITTS

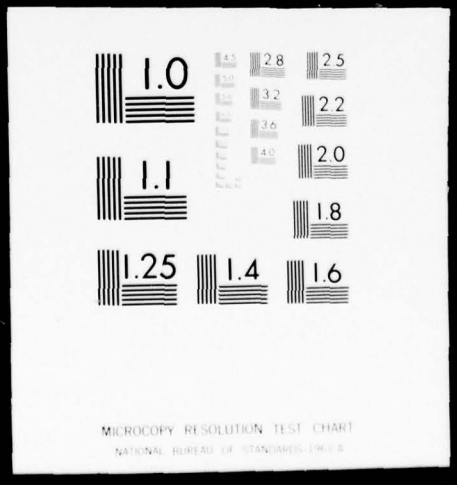
UNCLASSIFIED NRL-MR-3446 NL

1 OF 2
ADA037717



1 OF 2

AD A037717



NRL Memorandum Report 3446

Solution of Transient Problems in Free Surface Hydrodynamics

M. J. FRITTS

Science Applications, Inc.
McLean, Virginia 22101

and

J. P. BORIS

Plasma Dynamics Branch
Plasma Physics Division

February 1977

This work was sponsored by the Office of Naval Research under Contract RR011-09-41

NAVAL RESEARCH LABORATORY
Washington, D.C.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL ~~Memorandum Report~~ 3446	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) SOLUTION OF TRANSIENT PROBLEMS IN FREE SURFACE HYDRODYNAMICS		5. TYPE OF REPORT & PERIOD COVERED Interim report on a continuing NRL problem. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J. P. Boris and M. J. Fritts		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem H02-51
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research 800 North Quincy Street Arlington, Virginia 22217		12. REPORT DATE February 1977 13. NUMBER OF PAGES 115
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This report was sponsored by the Office of Naval Research under Contract RR011-09-41, title Computational Fluid Dynamics.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

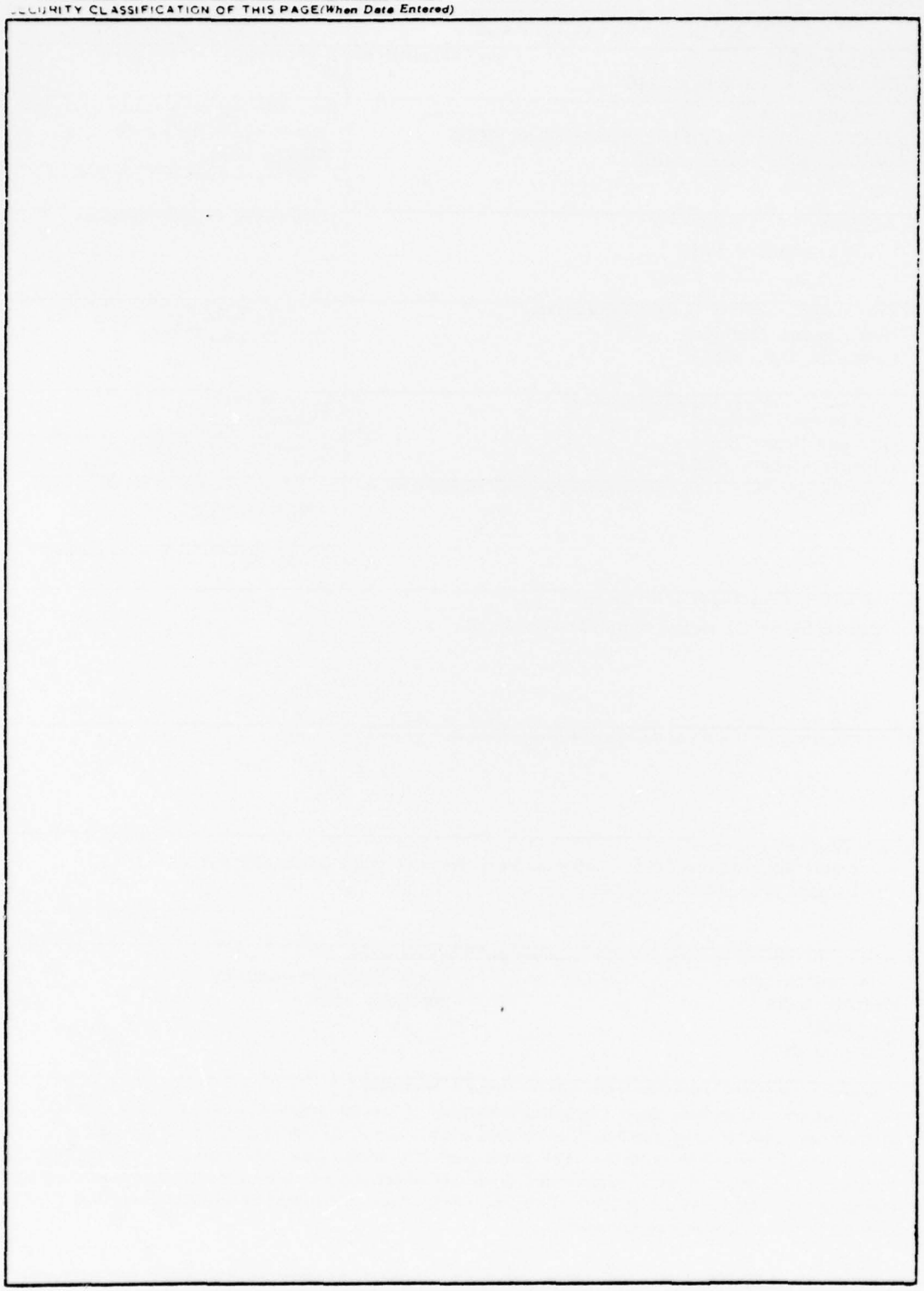
Numerical methods
Hydrodynamics
Lagrangian
Triangular grid
Kelvin-Helmholtz instability
Nonlinear waves

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

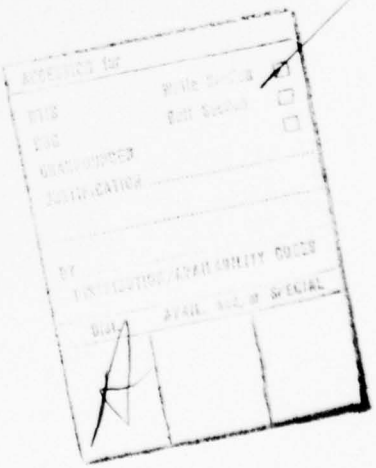
Numerical algorithms for a Lagrangian treatment of incompressible hydrodynamics with free surfaces are developed and applied. The method centers about the use of a triangular, general-connectivity finite-difference mesh. The added flexibility arising from this triangular mesh permits accurate long-time solutions of complicated flows with shear and waves in free-surface problems. The techniques discussed here are applied to nonlinear free-surface waves and to unstable shear flow beneath a free surface in two dimensions.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)



CONTENTS



SOLUTION OF TRANSIENT PROBLEMS IN FREE SURFACE HYDRODYNAMICS

I INTRODUCTION

Lagrangian methods offer the most natural approach to transient hydrodynamics problems which contain free surfaces, interfaces or sharp boundaries. In practice, their use in numerical solutions has generally been restricted to "well-behaved" flows, since shear, fluid separation or even large amplitude motion produce severe grid distortions. The distortions arise from the migration of mesh points which were formerly neighboring but which have crossed or become separated in the flow. Numerical solutions of the physical equations differenced over such a mesh quickly fail because of the inaccuracies inherent in approximations which ignore mesh point rearrangement and grid distortion. This problem is solved by the implementation of a general-connectivity grid in which local mesh reconnections are made whenever the grid distorts appreciably.

The techniques described in this paper involve the use of a Lagrangian, finite-difference mesh of connected triangles to represent the fluid motion, the various interfaces, and the free surfaces. This approach is a fruition of efforts carried out at NRL and elsewhere during the past few years. Some of the basic concepts were developed by Crowley[1] using the code FLAG. Our own work has concentrated on the free-surface and physical consistency aspects of the problem. Early NRL efforts centered on LINUS2[2-4], a 2D MHD code with axial symmetry. Attention more recently has turned toward solving free-surface problems in naval hydrodynamics.[9,10,11]

Note: Manuscript submitted January 17, 1977.

A good example of the type of problem requiring improved numerical techniques is the problem of a large amplitude breaking wave. When the wave top separates from the wave due to bottom shallowing, nonlinear wave interaction, or strong surface winds, it falls back into the wave trough in a complicated flow which the usual techniques cannot handle. One goal of this paper is to present Lagrangian triangular-cell algorithms in which the crest of the numerical wave will be able to separate from the body of the wave, fall under the influence of gravity and any strong winds back into the trough of the wave, and become reabsorbed smoothly while following the numerical representation of the fluid at the surface.

Figure 1 illustrates the use of a Lagrangian triangular mesh to facilitate the solution of this problem. As the neck of fluid narrows, the corresponding triangle sides shrink to zero and the connection between the two bodies of fluid is broken. In this way the topology of the problem has changed self-consistently with the evolution of the system. Clearly, any representation of the problem, to be adequate, must be capable of drastic local changes such as this to reflect complicated motions. It is equally clear that a numerical free-surface capability which incorporates such reconnection algorithms for the solution of fully nonlinear flows will allow an almost inexhaustible list of important hydrodynamics problems to be solved.

Previous Lagrangian treatments[5,6,7] have been restricted to simple flows or small amplitude motions because they use a topologically rectangular finite-difference mesh. The major practical problem with these rectangular methods arises from the inflexible connectivity of the various mesh points.[8] In complicated and strongly sheared flows, where one element of fluid may become widely separated from a nearby element, the usual Lagrangian treatments break down because a simply structured rectangular mesh becomes too severely distorted to allow an adequate numerical representation of the fluid flow. When the mesh becomes so distorted that no greater Lagrangian motion can be permitted, a process of continual rezoning amounts to a form of numerical diffusion. Thus the usual Lagrangian treatments are capable of extending linear models significantly into the nonlinear regime but are not capable of providing a satisfactorily accurate longtime representation of highly complicated flows bordering on turbulent phenomena.

Another drawback of such approaches is a difficulty in representing complicated boundaries and structures because of the limited topology of the mesh. It is often necessary to require greater resolution simply to obtain a satisfactory initial grid. Rectangular mesh approaches also appear to suffer a serious "even-odd" or computational-mode instability which must be overcome by some form of added numerical damping.[5,6] This damping destroys the reversibility of the algorithm and limits its usefulness for high Reynolds-number flows even though some care has been given to

developing damping algorithms which minimize this nonphysical diffusion.

A triangular-element mesh has several advantages. The grid can be restructured. Individual triangles and sides can be bisected or rearranged to give new grid structures which better represent changing fluid flows. Since the number of triangles meeting at a vertex is variable, increased accuracy in one region of the flow does not force unnecessary resolution in other areas of the flow. This versatility also permits both regular and irregular tessellations of the x-y plane with triangles. Triangles, unlike rectangles, can symmetrically cover a circle without cusps or other local representation irregularities. Thus free surfaces, complicated interfaces and boundaries of immersed objects can be represented accurately and economically.

The triangle is a much less ambiguous structure than a rectangle or higher-order polygon and hence interpolations and integrals are usually simpler to perform. Moreover, the even-odd problems of rectangular schemes appear to be absent, or at least greatly subdued, in the triangular representation. Because of the three-sided figures (rather than four-sided), there is no unambiguous labelling of grid points as even and odd. A vertex, which is one point removed from its neighbor along a particular path, will be two points removed by another. This does not mean that an even-odd problem cannot occur, only that it is not a topologically natural mode of instability and hence is considerably slower growing than its rectangular-mesh

counterpart. Experience with the SPLISH code supports this; a reversible algorithm, or at least a very weakly damped one, is feasible using the Lagrangian triangular mesh approach.[9,10,11]

Of course there are also problems with the triangles. A general connectivity triangular mesh has non-trivial bookkeeping problems associated with the grid and its connections, the sides of the triangles, and the basic cells of the fluid dynamic system. Furthermore, numerical experience with triangles is much more limited than experience with rectangular meshes. The statement of the hydrodynamic equations in triangular systems requires that especially good attention be paid to the spatial derivative terms which are needed. There are, as well, some intrinsic numerical complications in triangular systems. The major one, discussed in detail in Section 3, is the counting of equations and free unknowns. This difficulty can be argued to be the price we pay for the relative suppression of the "even-odd" or "computational" modes observed in more standard rectangular approaches.

In this paper, we restrict ourselves to the study of systems in which the fluid is inviscid and incompressible but has variable density. The conservation integral approach and definitions of divergence we employ, however, allow a natural extension to compressible systems. We also restrict consideration to problems in which the gravity is constant and directed in the negative $\hat{y}$ direction. These are not necessary restrictions but simplify the analyses and allow the full spectrum of problems of current interest

to be solved. The basic equations of the system are:

$$\rho \frac{d\underset{\sim}{V}}{dt} = - \underset{\sim}{\nabla} P - \rho g \hat{y} \qquad \text{and} \tag{1}$$

$$\underset{\sim}{\nabla} \cdot \underset{\sim}{V} = 0, \tag{2}$$

where the fluid density ρ, pressure P and velocity $\underset{\sim}{V}$, are assumed to vary only with x and y. Equation (2), incompressibility, removes the sound waves. We will assume that P = constant or is given along free surfaces.

The basic discussion of triangular grids and connection of grid vertices, triangle sides and triangle volumes is found in Section 2. Tessellation of the plane is considered and the representation of complicated surfaces and interfaces by a triangular mesh is described. Concepts and a notation for this triangular mesh representation are developed and the vector aspects which can be utilized numerically are given.

Section 3 is devoted to describing methods for integrating the equations of incompressible hydrodynamics. A counting problem encountered in some schemes is described and the motion of the mesh is discussed. Two essentially reversible algorithms are possible, leap-frog and centered implicit, but both require iteration. Each can be coupled to any of the several possible spatial treatments. Reversibility is desired because it reflects a property of the inviscid physical equations, and if it is demanded of the algorithms,

major sources of numerical diffusion can be eliminated. Throughout the discussion emphasis is placed on the choice of algorithms actually implemented in the Cartesian SPLISH code. We also discuss briefly questions of mass, momentum and energy conservation which enter into the code.

Section 4 treats the important aspects of adjusting and reversibly restructuring the mesh. It describes methods of setting up the grid structure initially and the various techniques which can be used to modify the grid to better follow the Lagrangian motion of the fluid. One of the most important aspects of this restructuring is that strong shear flows can be followed for long times with minimal numerical diffusion.

Section 5 is devoted to an extensive treatment of nonlinear free-surface waves in which detailed comparison with theory is made. Section 6 applies the full numerical formalism with reconnections to the problem of a Kelvin-Helmholtz unstable shear flow beneath a free surface. A short summary is contained in Section 7.

II. Triangular Mesh Logic

In this section we describe some of the features of triangular meshes and the representation of the vertex interconnections. The techniques for adjusting and restructuring the mesh during the actual course of a calculation will be attacked later in Section 4. Figures 2a and 2b show a section of a triangular mesh representation with an interface of a fluid of type I connected to a fluid of type II.

The basic elements involved in the construction of a triangular mesh are shown. In figure 2a a particular triangle j, is shown in heavy lines and the various elements of that triangle are labelled. Three vertices, V_1, V_2, and V_3, are connected consecutively by sides S_1, S_2, and S_3. Clearly, triangle j shares these vertices and sides with other triangles. The direction of labelling around each triangle is taken to be counter-clockwise and the $\hat{z}$ direction is out of the page. Since the mesh can be irregularly connected, an arbitrary number of triangles can meet at each vertex. For example, five triangles and sides meet at vertex V_1. The number of triangles and sides meeting at a vertex is equal except near free-surfaces where there may be no grid above the surface. Each side is bounded by two triangles (in general) and each triangle shares sides with three other triangles.

Figure 2b illustrates several important features of triangles which are used throughout the remainder of this paper. It is convenient to define a cell surrounding a vertex as shown by the shaded region surrounding vertex 3. The borders of such vertex-centered cells are determined by constructing all of the side bisectors for each triangle. Since the three side bisectors all intersect at a point, as shown for triangle j, there is no ambiguity in constructing the vertex cells as indicated. The point of intersection of the side bisectors, the centroid of the triangle, is the center of gravity for the figure and this is true in r-z as well as Cartesian (x-y) coordinates. The three side bisectors of any

triangle divide the triangle into six subtriangles. These six subtriangles all have the same area and, therefore, each of the three vertex cells receives one-third of the area of the triangle. When a quantity is constant over a triangle, the contributions of that quantity from a triangle to each of the three vertex cells are also equal. These features of side bisectors make the calculation of cell volumes and cell areas particularly simple.

Any computational representation of such a triangular mesh must record all important aspects of the mesh interconnection. If each vertex, each side and each triangle is numbered, lists of interconnections can simply take the form of an ordered series of integers which can be stored quite compactly in a computer. For example, the information might be stored in the form:

1. each vertex list enumerates (A) the vertices to which the given vertex is connected; (B) the sides to which it is connected; and (C) the triangles which meet at that vertex;
2. for each side, lists could record (A) the starting and the finishing vertex of that side, and (B) the triangle to the right and the left of that side considered as a directed vector from the starting vertex to the finishing vertex;
3. each triangle list could contain (A) the three vertices of the triangle ordered in a counter-clockwise direction, (B) the three sides of the triangle between the corresponding vertices, again ordered in a counter-clockwise direction.

Since these lists are not all independent, they are not all needed. Knowing only the vertices to which a given vertex is connected, for example, still permits all of the other vertex information to be determined. If vertex 2 is connected to vertex 7, a search of the side lists can be performed to determine which side connects vertex 7 and vertex 2,and thus the various sides connected to a given vertex can be determined. However, while maintaining redundant lists is costly in terms of storage, there are real advantages gained in increased computation speed, clarity and ease of coding.

It should be clear that the arrays of quantities used to define the grid and its motion involve the storage of only local information. There is no global representation of the mesh in general, although for some specific geometries, such a global representation may be possible. Connection paths between vertices can be determined only by searching sequentially through neighboring vertices. The lack of a global representation in which the vertex numbering denotes a corresponding relative spatial position greatly complicates implicit calculations. Poisson's Equation, for example, must be solved by iteration, and explicit time and space derivatives are the best that one has any right to hope for. Nevertheless, using the incompressible formulation ensures that timestep limitations due to acoustic transit times are not a problem.

We will use the following notations and conventions. The subscript i will generally be used to label vertices and the

subscript j will generally be used to label triangles. Thus $\sum_i$ denotes a sum over all vertices and $\sum_j$ is a sum over triangles. The sum over all three vertices of triangle j is denoted by $\sum_{i\textcircled{j}}$ where the symbol is read "the sum over vertices i around triangle j". Similarly a sum over triangles around a central vertex c would be denoted by $\sum_{j\textcircled{c}}$. The notation $\sum_{i\textcircled{c}}$ is the sum over vertices i around a central vertex c. In such sums the sequence of vertices is assumed to be counter-clockwise around the central vertex. Thus the quantity $A_{i+\frac{1}{2}}$ can be used to represent the triangle area of the triangle with vertices (c, i, i+1). Similarly $L_{j+\frac{1}{2}}$ appearing in $\sum_{j\textcircled{c}}$ would be the length of the side radiating from vertex c and separating triangle j from triangle j+1.

In Figure 2 the area of triangle j is given by

$$2A_j = (\underset{\sim}{r}_3 - \underset{\sim}{r}_2) \times (\underset{\sim}{r}_1 - \underset{\sim}{r}_3) \cdot \hat{z} \tag{3}$$

and thus h_1, the height of vertex 1 above the opposite side (S_2), is simply

$$h_1^{-1} = |(\underset{\sim}{r}_3 - \underset{\sim}{r}_2)|/(2A_j). \tag{4}$$

Notice that the area is a signed quantity and will be positive when the vertices are sequenced in counter-clockwise order around the triangle. In fact, it is often convenient to use the sign of the area to test whether or not a triangle has inverted during the flow, i.e., whether vertex 1, for instance, has passed through side 2.

As illustrations of these formulae and notations, the area of the cell centered on vertex i is

$$A_i = \sum_{j\textcircled{i}} \tfrac{1}{3} A_j \tag{5}$$

and the area-weighted average cell velocity $\underset{\sim}{V}_i$ is given by

$$\underset{\sim}{V}_i = \Big(\sum_{j(i)} \tfrac{1}{3} A_j \, \underset{\sim}{V}_j \Big) / A_i . \tag{6}$$

Of course, all the sums have to be modified near a wall, an interface, or a free surface.

If a scalar function f is specified at each vertex and is assumed to be piecewise linear everywhere between the vertices, the vector gradient of f (constant throughout the triangle and discontinuous at the triangle sides) is given by

$$\begin{aligned} (\underset{\sim}{\nabla} f)_j &= f_1 \frac{\hat{z} \times (\underset{\sim}{r}_3 - \underset{\sim}{r}_2)}{2A_j} + f_2 \frac{\hat{z} \times (\underset{\sim}{r}_1 - \underset{\sim}{r}_3)}{2A_j} + f_3 \frac{\hat{z} \times (\underset{\sim}{r}_2 - \underset{\sim}{r}_1)}{2A_j} \\ &= \sum_{i(j)} f_i \frac{\hat{z} \times (\underset{\sim}{r}_{i-1} - \underset{\sim}{r}_{i+1})}{2A_j} . \end{aligned} \tag{7}$$

For a vector field $\{\underset{\sim}{V}_i\}$ defined at all the vertices, the z component of the curl and the divergence of the vector are also defined for the triangles as follows:

$$A_j \zeta_j = \int_j (\underset{\sim}{\nabla} \times \underset{\sim}{V})_j^z \cdot d\underset{\sim}{A} = \tfrac{1}{2} \sum_{i(j)} (\underset{\sim}{V}_{i+1} + \underset{\sim}{V}_i) \cdot (\underset{\sim}{r}_{i+1} - \underset{\sim}{r}_i), \tag{8}$$

$$A_j \langle \underset{\sim}{\nabla} \cdot \underset{\sim}{V} \rangle_j = \int_j (\underset{\sim}{\nabla} \cdot \underset{\sim}{V})_j = dA = \tfrac{1}{2} \sum_{i(j)} (\underset{\sim}{V}_{i+1} + \underset{\sim}{V}_i) \times (\underset{\sim}{r}_{i+1} - \underset{\sim}{r}_i) \cdot \hat{z}. \tag{9}$$

Equations (7-9) define various quantities at triangle centroids. The corresponding quantities can usually be defined at the vertices of the finite-difference mesh by area weighting the triangle quantities as in Eq. (6).

The converse definition is also valid. A vector field may be defined on triangles, where the $\underset{\sim}{V}_j$ are constant throughout triangles and discontinuous at the triangle sides. For example, the triangle velocity $\underset{\sim}{V}_j$ is given by

$$\underset{\sim}{V}_j \equiv (\underset{\sim}{\nabla}\times\psi)_j = \sum_{i\,\mathcal{D}} \psi_i \frac{(\underset{\sim}{r}_{i-1} - \underset{\sim}{r}_{i+1})}{2A_j} \tag{10}$$

where $\{\psi_i\}$ is the vertex-defined velocity stream function (only a single z component in 2D).

Such a representation is appropriately both unique and orthogonal. By uniqueness we merely mean that the triangle vector equation

$$\underset{\sim}{V} = \underset{\sim}{\nabla}\phi + \underset{\sim}{\nabla} \times \psi_z. \tag{11}$$

uniquely relates $\underset{\sim}{V}$ and (ϕ, ψ_z)(assuming well-posed conditions on ϕ and ψ_z at the system boundaries where the derivatives are not defined). Given ϕ and ψ_z on vertices, Eqs. (7) and (10) allow an unambiguous determination of $\underset{\sim}{V}$ at the triangle centroids.

By orthogonal we mean that the numerical finite difference operators satisfy the continuum orthogonality conditions

$$\underset{\sim}{\nabla} \times \underset{\sim}{\nabla}\phi = 0, \qquad \underset{\sim}{\nabla} \cdot (\underset{\sim}{\nabla} \times \psi_z) = 0. \tag{12}$$

The curl and divergence operators in Eqs. (12) of course act on triangle quantities since ϕ and ψ are assumed to be vertex quantities as above (but are otherwise arbitrary scalar fields).

The second of Eqs. (12), for example, can be written in our finite-difference notation as

$$A_c\langle \underset{\sim}{\nabla} \cdot (\underset{\sim}{\nabla} \times \psi_z)\rangle_c = \int_c \underset{\sim}{\nabla} \cdot (\underset{\sim}{\nabla} \times \psi_z)$$
$$= \sum_{i\textcircled{c}} \frac{(\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i)}{4A_{i+\frac{1}{2}}} \times \left[\psi_i(\underset{\sim}{r}_c-\underset{\sim}{r}_{i+1}) + \psi_{i+1}(\underset{\sim}{r}_i-\underset{\sim}{r}_c) + \psi_c(\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i)\right] \tag{13}$$

This is just the flux of $\underset{\sim}{\nabla} \times \psi$ (constant over triangles) out of cell c. Clearly the ψ_c coefficient is zero term by term because of the cross product. The ψ_i and ψ_{i+1} terms have as coefficients $+\frac{1}{2}$ and $-\frac{1}{2}$, respectively, because $(\underset{\sim}{r}_i-\underset{\sim}{r}_c) \times (\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i) = 2A_{i+\frac{1}{2}}$. Thus Eq. (13) is identically zero when summed around an interior vertex. By a virtually identical argument the first of Eqs. (12) can be demonstrated to be identically zero.

Therefore, taking the divergence of $\underset{\sim}{V}$ (Eq. (11)) expunges the ψ_z term. Thus

$$A_c\langle \underset{\sim}{\nabla} \cdot \underset{\sim}{V}\rangle_c = \sum_{i\textcircled{c}} \underset{\sim}{V}_{i+\frac{1}{2}} \times \frac{(\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i)}{2} = \underset{\sim}{\nabla} \cdot \underset{\sim}{\nabla}\phi =$$
$$= \sum_{i\textcircled{c}} \left[\phi_i \frac{\hat{z}\times(\underset{\sim}{r}_c-\underset{\sim}{r}_{i+1})}{2A_{i+\frac{1}{2}}} + \phi_{i+1}\frac{\hat{z}\times(\underset{\sim}{r}_i-\underset{\sim}{r}_c)}{2A_{i+\frac{1}{2}}} + \phi_c\frac{\hat{z}\times(\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i)}{2A_{i+\frac{1}{2}}}\right]$$
$$\times \frac{(\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i)}{2} \cdot \hat{z}. \tag{14}$$

The second line of Eq. (14) is the finite-difference form of the ∇^2 operator and Eq. (14) is a triangular-grid Poisson Equation. Notice that the coefficient of the ϕ_c term is $-\sum_{i\textcircled{c}} |\underset{\sim}{r}_{i+1}-\underset{\sim}{r}_i|^2/4A_{i+\frac{1}{2}}$ and is always negative. The matrix is diagonally dominant and hence normal

iterative procedures for finding ϕ by inverting Eq. (14) work well.

An equation similar to (14) can be constructed, except for sign, by taking the curl of Eq. (11). Then

$$A_c(\underset{\sim}{\nabla} \times \underset{\sim}{V}) = \underset{\sim}{\nabla} \times (\underset{\sim}{\nabla} \times \psi_z) \tag{15}$$

allows the determination of ψ_z from $\underset{\sim}{V}$.

III. Finite-Difference Approaches.

Given the mesh and all its interconnections, we would like to associate various physical quantities of interest with the vertices, the sides or the triangles. Since it is the mesh of vertices which determines the structure of the grid, it is natural to assume that an array of positions is recorded for the distinct vertices in the problem. To update these positions, it is necessary to know the Lagrangian velocity of the vertices. Therefore, we will also assume that the velocities of the vertices can be found in such a way that integration of the equations of motion for the vertex position is possible using these velocities. Since the velocities are assumed known at the vertices, a velocity field could be constructed by linear interpolation within a triangle which is then piecewise linear throughout the entire mesh. This velocity field is continuous, single-valued and easily determined at any point in space. Using linear interpolation, the velocities at the center of the sides are simply the average of the velocities at the starting and finishing vertex of that side.

Unfortunately, the first derivatives of this representation are discontinuous and the linear representation ensures that the incompressible equations of motion cannot be satisfied identically within a single triangle except for a few simple cases such as solid body translation. In fact, a real fluid element which is initially triangular would be very shortly transported in a real fluid to a figure with non-straight sides. These deformations of the triangle sides are a measure of the error in the numerical methods we are proposing and are fully equivalent to the errors made in neglecting the deformation of rectangle sides in a topologically rectangular representation.

The incompressibility inherent in Eq. (2) cannot be applied to the triangles themselves for another, more basic, reason. The x and y values of the vertex positions, the variables available to ensure conservation of triangle area, are fewer in number than the constraints. This is the previously mentioned "counting problem". For rectangular grids a one-to-one correspondence can be made between quadrilateral areas and the vertices; for example, associating a quadrilateral with its lower right vertex. For a triangular mesh, particularly one that allows reconnections, no such correspondence exists.

In a quadrilateral mesh, the one-to-one correspondence fails only at the boundaries. The extra vertex variables at the boundaries are available to satisfy the boundary conditions. For a triangular mesh

$$N_t \lesssim 2N_v \tag{16}$$

or there are less than two triangles per vertex. A one-to-one correspondence is approached only for the very restrictive, and not too useful, case for which all vertices lie on the boundaries, when $N_t = N_v - 2$. For most cases of interest the number of boundary vertices is small compared to the number of interior vertices, and the ratio is much worse, $N_t \sim 2N_v$. (See the Appendix for details.)

Although there are always fewer than two triangles per vertex, the number of variables left free after all of the constraints have been satisfied are extremely few in number. In this case, after all triangle areas are conserved, the x and y positions available could not fulfill boundary conditions or be used to represent the flow vorticity with acceptable accuracy.

Once the idea of conserving triangle areas has been abandoned, the next logical attempt is to conserve vertex-centered cell areas. This is attractive since the cell area $(\nabla \cdot \underset{\sim}{V})$ and the cell rotation $(\nabla \times \underset{\sim}{V})$ are both known for each vertex and there are correspondingly two variables for each vertex, V_x and V_y, which are free to satisfy these constraints. A great deal of effort has been invested in this approach with mixed results. The cell area must at all times have the value it started with initially. The cell vorticity integral is likewise known. When the curl of Eq. (1) is integrated over a cell c we obtain

$$\frac{d}{dt} \int_c \underset{\sim}{\nabla} \times \underset{\sim}{V} \cdot d\underset{\sim}{A} = - \int_c \underset{\sim}{\nabla} \times \left(\frac{\underset{\sim}{\nabla} P}{\rho}\right) \cdot d\underset{\sim}{A} \qquad (17)$$

where the term on the right hand side can be evaluated explicitly. The left hand side, the time derivative of the average vorticity times the cell area, can thus be used to advance the cell vorticity in time. Knowing the desired cell vorticity and cell area, the new vertex velocities can be iterated (ensuring momentum conservation via the equations of motion) until the actual cell areas and vorticities are correct.

Unfortunately the effective Poisson-like equations which result are non-local and convergence of the iterations is correspondingly slow. While there is no counting problem with this approach, an allied topological constraint crops up when more than six triangles meet at a vertex. We have been able to show that a purely local rotation conserving all cell areas is not generally possible where seven or more triangles meet at a vertex (see the Appendix). In other words, more than just the nearest neighbor vertices must have non-zero velocities to conserve all cell areas; purely local vortices are not generally possible and this gets reflected in the convergence rate of the iterated solution. Similarly, a purely vorticity-free local expansion is not generally possible when more than six triangles meet at a vertex. This constraint is not entirely prohibitive, but the cost of exact area and vorticity conservation <u>with</u> momentum conservation is high in terms of computer time so faster, less

demanding algorithms were required.

There is no strict requirement to maintain physically conserved quantities any more accurately than the truncation error levels implicit in the finite-difference algorithms--even though such strict conservation is esthetically attractive. With this in mind we considered algorithms which minimized a positive definite error functional over the whole mesh. Since the desired curl $\{C_i^*\}$ and divergence $\{D_i^*\}$ are known, and since Eqs. (6), (8), and (9) allow us to define at any instant the actual values $\{C_i\}$ and $\{D_i\}$, we chose as our error functional

$$E = \sum_i \left[\omega_i (C_i - C_i^*)^2 + W_i (D_i - D_i^*)^2 \right] \tag{18}$$

The weights $\{\omega_i\}$ and $\{W_i\}$ are free to emphasize small or large triangles, fast or slow flow regions, etc. Taking the partial derivatives $\partial E / \partial \underset{\sim}{V}_i = 0$ gives as many equations as one needs to advance the fluid velocities. We never coded up this last method, however, because the programming complication is prohibitive and the application of boundary conditions and momentum conservation is obscure.

The superficially attractive approaches have been mentioned briefly only to prevent (hopefully) as many false starts on the part of our readers as we have undergone. We will now indicate two distinct approaches which were successful; a ψ-ζ and a P-$\underset{\sim}{V}$ formalism. Both define velocities as triangle-based quantities,

and derive vertex velocities from Eq. (6). This obviates the counting and mesh problems mentioned above through the introduction of more variables.

The first method relies on the orthogonality of the curl and gradient operators discussed in the previous section. We force the velocity, now defined on the triangles, to arise from a stream function according to Eq. (10). Such a flow is divergence-free by construction as described earlier; the flux into or out of each vertex cell is identically zero. In fact, such a flow, as shown in Figure 4 for one vertex and its nearest neighbors, is also divergence-free on each triangle. Since $\underset{\sim}{V}_j$ is constant throughout a triangle and since the normal component of $\underset{\sim}{V}$ is continuous across all triangle sides, the flow field described by $\underset{\sim}{V} = \underset{\sim}{\nabla} \times \psi_z$ allows no accumulation of fluid in a triangle or loss of material from a triangle. Figure 3 shows how this can be by displaying actual stream lines from such a flow. These stream lines are closer together where the flow is faster and are continuous crossing triangle sides even though their direction changes.

The introduction of triangle velocities therefore makes it possible to define a flow which is divergence-free on triangles. However, the solution is not ideal, since the triangle velocities cannot be used to advance vertices, the area-weighted vertex velocities must be used. Therefore the triangle areas must necessarily change when the vertices are advanced, leaving only the cell areas conserved. What remains is still very attractive physically,

however. If the grid is held fixed, the density of marker particles in the instantaneous flow field remains locally constant, while advancing the vertices leaves cell areas unchanged. This is the physically meaningful expression of incompressibility which is sustained where triangle velocities are the basic physical quantities.

In either formulation the incompressibility assumption forces a physically nonlocal character on the flow and this gets reflected in the numerical necessity of iterating a Poisson-like equation with appropriate boundary conditions. In the ψ-ζ form, the vorticity ζ is advanced at each cell during each timestep and then $\underset{\sim}{\nabla} \times (\underset{\sim}{\nabla} \times \psi) = \zeta$ must be solved for the velocity stream function ψ. The velocity divergence is zero by construction. In the primitive P-$\underset{\sim}{V}$ formulation, the divergence of velocity is forced to zero by choosing P to satisfy $\underset{\sim}{\nabla} \cdot (\underset{\sim}{V} \cdot \underset{\sim}{\nabla}\underset{\sim}{V}) = - \underset{\sim}{\nabla} \cdot \frac{1}{\rho} \nabla P$, and changes in vorticity are zero by construction. In either case the complicated grid connections virtually preclude a fast direct solution or even a globally implicit iteration. In solving these equations we have been using a simple point-relaxation technique with a home-grown acceleration formula. We have found, however, that simple extrapolation using the two previous values of the quantity being iterated (P or ψ) makes the biggest impact. Extrapolation reduces the residual by one to two orders of magnitude before iteration even begins. What is even more important, however, is the fact that extrapolation is most accurate for the long wavelength components of the solution desired. This is extremely helpful because the long

wavelength components of the residual are the slowest to converge under iteration. Therefore, using extrapolation not only reduces the initial residual appreciably, it effectively increases the rate of convergence of the iteration itself.

The actual temporal integration scheme chosen for either formulation is reversible, because the physical equations being solved are reversible.[7] Reversibility usually means that artificial numerical damping is absent and that higher Reynold's number flows can be treated accurately. Conversely, the total absence of numerical damping is often accompanied by an even-odd or computational mode of numerical instability in which decoupling of the finite-difference mesh variables occurs in some obnoxious manner or another. Previously we have argued that our triangular grid algorithms minimize the virulence of this problem and in Sections 5 and 6 we will show several undamped reversible calculations. For now we concentrate on defining reversible algorithms since damping can also be added where and when needed.

There are two easy ways to achieve the desired reversibility--using a leapfrog $\underset{\sim}{x}$-$\underset{\sim}{v}$ integration in time and using a centered implicit $\underset{\sim}{x}$-$\underset{\sim}{v}$ integration. Either approach can be applied with either the ψ-ζ formulation or the P-$\underset{\sim}{V}$ formulation of the basic equations. The choice appears to be a matter of convenience although we have used centered implicit in the final versions of both formulations.

Since the flow is Lagrangian, we can use the following leapfrog time-integration template

$$\underset{\sim}{V}_i(t+\delta t/2) = \underset{\sim}{V}_i(t-\delta t/2) + \delta t\ \underset{\sim}{a}_i(t,\ \{\underset{\sim}{X}\},\ \text{etc.}),$$

$$\underset{\sim}{X}_i(t+\delta t) = \underset{\sim}{X}_i(t) + \delta t\ \underset{\sim}{V}_i(t+\delta t/2). \tag{19}$$

The subscript i labels the vertices (or the triangle centroids. The important points are that the acceleration terms $\{\underset{\sim}{a}_i\}$ can be calculated at time t using positions and other quantities whose values are already known at time t and that this "leapfrog" integration is explicit as expressed above. Alternately the centered implicit formulation can be used.

$$\underset{\sim}{V}_i(t+\delta t) = \underset{\sim}{V}_i(t) + \delta t\ \underset{\sim}{a}_i(t+\delta t/2),$$

$$\underset{\sim}{X}_i(t+\delta t) = \underset{\sim}{X}_i(t) + \frac{\delta t}{2}\left[\underset{\sim}{V}_i(t) + \underset{\sim}{V}_i(t+\delta t)\right] \tag{20}$$

where the accelerations $\underset{\sim}{a}_i(t+\delta t/2)$ are calculated either as the outer average $\frac{1}{2}[\underset{\sim}{a}_i(t) + \underset{\sim}{a}_i(t+\delta t)]$ or the inner average $\underset{\sim}{a}_i(t+\delta t/2, \frac{1}{2}\{\underset{\sim}{X}_i(t) + \underset{\sim}{X}_i(t+\delta t)\}$, etc.). The formulation of Eqs. (20) is clearly reversible and, insofar as the velocities $\underset{\sim}{V}_i(t+\delta t)$ are not known a priori, the formulation is implicit and must be iterated. Since we are iterating the acoustic waves (incompressibility) anyway, this extra iteration imposes no great hardship. Furthermore, the implicit nature sometimes means that longer timesteps can be taken stably.

We will illustrate the ψ-ζ method using the easier algorithm, the leapfrog method. We start knowing the positions of the vertices $\underset{\sim}{X}_i$ at time t and the vorticity at these same vertices at time $t-\delta t/2$. We also know the parallel velocity of any free surface vertices $\underset{\sim}{V}_i$ at time $t-\delta t/2$. Since the pressure is constant at the free surface, the free surface boundary condition can be deduced from the component of Eq. (1) resolved parallel to the free surface. If $\hat{\tau}$ denotes a unit vector tangent to the free surface, then

$$\frac{dV_\tau}{dt} = - g\hat{y} \cdot \hat{\tau} \tag{21}$$

Therefore the new free surface velocities have a parallel component which is

$$V_{\tau i}(t+\delta t/2) = V_{\tau i}(t-\delta t/2) - \delta t\, g\, \hat{y} \cdot \hat{\tau}(t). \tag{22}$$

This is reversible and the vorticity can likewise be advanced to $\zeta_i(t+\delta t/2)$ reversibly using Eq. (17). Of course the vorticity is conserved in the Lagrangian frame unless $\underset{\sim}{\nabla} p \neq 0$. When $\underset{\sim}{\nabla} p \neq 0$, the pressure field has to be determined, a point we return to shortly.

The next step is to calculate ψ from the usual equation $\underset{\sim}{\nabla} \times (\underset{\sim}{\nabla} \times \psi) = \zeta$. Here we encounter a snag; a spatial grid at time $t+\delta t/2$ is required if $\psi_i(t+\delta t/2)$ is to be found. We do this in SPLISH, our Cartesian code, by calculating $\underset{\sim}{X}_i(t+\delta t/2)$ as the average of the new and the old grid positions. Of course this is reversible provided we iterate <u>exactly</u> as in the implicit formulation.

Once $\{\psi_i\}$ has been found the triangle velocities at time $t+\delta t/2$ can be calculated using Eq. (10). Forming the vertex area-weighted averages of $\underset{\sim}{V}_i$ from Eq. (6) gives $\{\underset{\sim}{V}_i(t+\delta t/2)\}$ which can be used to advance the vertex positions $\underset{\sim}{X}_i$ from t to $t+\delta t$.

The boundary conditions in the Poisson Equation are simple on a solid wall; ψ = constant. On the bottom of a fluid layer periodic in X, we set $\psi = 0$. If a lid were placed on the top of the fluid, the constant value of ψ at the top could be different from the value on the bottom. The difference in the two constants would be the rate of flow across a vertical surface from the top to the bottom. This rate is independent of the X position of the imaginary surface because the fluid is incompressible.

The boundary condition at the free surface is merely that the component of velocity parallel to the surface is known for a free-surface vertex. The normal gradient of ψ, area-weighted to the free-surface vertices, must give the known values. Since the point-by-point iteration of the interior ψ points can be performed, the normal gradient can be adjusted by appropriately choosing the free-surface ψ values. An explicit evaluation of these values is possible, just as with the interior points, but we have chosen an implicit formulation of this boundary condition which couples all of the surface ψ values in a tridiagonal system. In addition to taking the interaction with neighboring surface vertices into account, this method speeds convergence over the whole mesh because two widely separated vertices can "communicate numerically" very quickly

if they are reasonably close to the surface.

When an internal cavity is present in the fluid due to the existence of a bubble or to cavitation, the ψ-ζ boundary conditions are greatly complicated. For example, the formation of a bubble has to move the free surface upward to conserve area. Since $\partial\psi/\partial\tau$ along the surface is the velocity normal to the surface, the net upward velocity, $\int_0^L \frac{\partial\psi}{\partial\tau}\, d\ell$, must be non-zero. In a periodic system the formation of a bubble or cavity thus requires ψ to be non-periodic even though the derivative must be. In other words, ψ must have a branch cut whose jump varies with the rate of expansion or contraction of the bubble. While this nasty complication is not insurmountable, it certainly requires the calculation of a self-consistent pressure field. Since the fully nonlinear source term for internal waves, as mentioned earlier, also requires this pressure, a primitive solution method using P and $\underset{\sim}{V}$ has definite advantages over the ψ-ζ formulation, even though it is intrinsically a little more complicated in 2D. A further argument in favor of the P-$\underset{\sim}{V}$ formulation arises when we consider the eventual generalization to three dimensions. Then $\underset{\sim}{\psi}$ is a true vector having three components, and three Poisson equations must be solved. In the P-$\underset{\sim}{V}$ formulation P would still be a scalar so only one Poisson equation needs to be solved even in three dimensions.

Because of these advantages the P-$\underset{\sim}{V}$ formulation has been the major focus of our work. All the tests and examples cited below will use this method, and all further discussion will refer

exclusively to the primitive formulation. However, it should be stressed that the ψ-ζ formalism has also been coded and tested, and it has exhibited very good accuracy and convergence. Although its utility is more restricted, on certain flows it is the more preferable of the two formalisms because of its more simple structure and faster speed of execution.

Our P-$\underset{\sim}{V}$ formulation has P_i, $\underset{\sim}{V}_j$, and $\underset{\sim}{X}_i$, all specified at times t, t+δt, t+2δt, ····. Using a split step algorithm[7], we integrate the velocities forward half a timestep, advance the grid one full timestep, and then advance the velocities forward the other half timestep.

$$\underset{\sim}{V}_j^{\frac{1}{2}} = \underset{\sim}{V}_j^{o} - \frac{\delta t}{2\rho_j} (\underset{\sim}{\nabla} P)_j^{o} - \frac{\delta t}{2} g\hat{y}, \tag{23}$$

$$\underset{\sim}{V}_i^{\frac{1}{2}} = \tfrac{1}{2}(\underset{\sim}{V}_i^{o} + \underset{\sim}{V}_i^{n}), \tag{24}$$

$$\underset{\sim}{X}_i^{n} = \underset{\sim}{X}_i^{o} + \delta t \, \underset{\sim}{V}_i^{\frac{1}{2}}, \tag{25}$$

$$\underset{\sim}{\tilde{V}}_j^{\frac{1}{2}} = \underset{\approx}{R} (\{\underset{\sim}{X}_i^{o}\}, \{\underset{\sim}{X}_i^{n}\}) \cdot \underset{\sim}{V}_j^{\frac{1}{2}}, \tag{26}$$

$$\underset{\sim}{V}_j^{n} = \underset{\sim}{\tilde{V}}_j^{\frac{1}{2}} - \frac{\delta t}{2\rho_j} (\underset{\sim}{\nabla} P)_j^{n} - \frac{\delta t}{2} g\hat{y} \tag{27}$$

The velocity $\underset{\sim}{V}_i^{n}$ appearing in Eq. (24) is area-weighted from $\{\underset{\sim}{V}_j^{n}\}$ from Eq. (27) as calculated at the previous iteration. Eqs. (23) and (27) contain the evolution of $\{\underset{\sim}{V}_j\}$ according to the Lagrangian

equation of motion. Eq. (26) is the numerical reflection of the fact that triangle velocities must rotate and stretch as the grid rotates and stretches. The transformation $\underset{\approx}{R}$ is linear and given by the following three scalar equations:

$$\begin{aligned}
\underset{\sim}{V}_j^{\frac{1}{2}} \cdot (\underset{\sim}{X}_2^n - \underset{\sim}{X}_1^n) &= \underset{\sim}{V}_j^{\frac{1}{2}} \cdot (\underset{\sim}{X}_2^o - \underset{\sim}{X}_1^o), \\
\underset{\sim}{V}_j^{\frac{1}{2}} \cdot (\underset{\sim}{X}_3^n - \underset{\sim}{X}_2^n) &= \underset{\sim}{V}_j^{\frac{1}{2}} \cdot (\underset{\sim}{X}_3^o - \underset{\sim}{X}_2^o), \qquad (28) \\
\underset{\sim}{V}_j^{\frac{1}{2}} \cdot (\underset{\sim}{X}_1^n - \underset{\sim}{X}_3^n) &= \underset{\sim}{V}_j^{\frac{1}{2}} \cdot (\underset{\sim}{X}_1^o - \underset{\sim}{X}_3^o).
\end{aligned}$$

This transformation ensures that the vorticity integral, calculated about <u>any</u> interior vertex, is invariant during the displacement of the grid. Clearly Eqs. (28), and hence Eq. (26), are fully reversible. It is gratifying, furthermore, that the three equations are not linearly independent since only two components of $\underset{\sim}{V}_j^{\frac{1}{2}}$ need to be determined.

As has been established previously, the $\underset{\sim}{\nabla}P$ and gravity terms cannot alter the vorticity either, since $\underset{\sim}{\nabla} \times \underset{\sim}{\nabla}P \equiv 0$ and gravity is a constant. Only the physically correct variations of $\frac{1}{\rho_j}$ can cause changes in vorticity, exactly as they should. Thus the entire algorithm advances the positions of the vertices and the triangle velocities reversibly while evolving the correct vorticity about every interior vertex. The pressures $\{P_i^n\}$ at $t+\delta t$ are derived from the condition that the new velocities $\{\underset{\sim}{V}_j^n\}$ should be divergence-free

at time t+δt. Thus, from Eq. (27) we derive the pressure Poisson Equation by setting $(\underset{\sim}{\nabla} \cdot \underset{\sim}{V}_j^n)_i = 0$. We obtain the pressures $\{P_n\}$ such that

$$\frac{\delta t}{2}(\underset{\sim}{\nabla} \cdot \frac{1}{\rho_j} \underset{\sim}{\nabla} P_i^n)_i = (\underset{\sim}{\nabla} \cdot \underset{\sim}{\tilde{V}}_j^{\frac{1}{2}})_i. \tag{29}$$

The right hand side, which can be evaluated at each iteration explicitly, is the <u>exact</u> numerical analogue of the $\underset{\sim}{\nabla} \cdot (\underset{\sim}{V} \cdot \underset{\sim}{\nabla} V)$ term which arises when the divergence of Eq. (1) is taken.

Solving Eq. (29) iteratively for $\{P_i^n\}$ completes the calculation of the timestep. The boundary conditions on P for Eq. (29) are very similar to those described for the ψ-ζ formulation earlier. At a free-surface P = constant (P = 0 at the top and a constant value within a bubble, determined by the bubble volume, surface tension, etc.). At the bottom or on a wall ∂P/∂n must be chosen so that the velocity normal to the wall is zero. Thus an implicit tridiagonal system can be developed linking all pressures on the wall (or bottom) to the nearest neighbor interior values.

Since pressures at the boundaries are all specified through the boundary conditions, the velocities around the partial cells at these boundaries cannot be kept divergence-free by iterations over the boundary pressures. The boundary cell areas can be apportioned uniquely to the cell areas beneath the surfaces, however, as shown in Figure 4. This construct permits the pressures near boundaries to adjust for divergence-free flow in the expanded cell areas,

insuring that the divergence can be iterated to zero throughout the entire fluid.

IV. Adjusting and Restructuring the Mesh

The major advantage of using a general connectivity triangular mesh is the flexibility which it permits to follow complicated flows with a Lagrangian grid over long times. To make full use of this flexibility requires that we provide for several types of adjustment and local mesh restructuring to maintain the uniformity and accuracy of the discrete mesh representation. A mesh adjustment is a nonphysical movement or adjustment of the position of one or more vertices which is accomplished without changing the connectivity of the mesh vertices. These adjustments are designed to simplify and regularize the mesh and result in the effective transfer of fluid across triangle boundaries.

A restructuring of the mesh, on the other hand, does not generally involve movement of any of the vertices but does include many sorts of triangle and side additions, reconnections, and subtractions. In a sense, adjustment and restructuring are orthogonal procedures--one leaving the vertex positions unchanged, the other leaving the mesh connectivity or topology unchanged. Since restructuring involves the changed position of a side, it also can involve the nonphysical flow of fluid across triangle boundaries.

Both adjustment and restructuring represent departures from a purely Lagrangian description and hence threaten to introduce unwanted numerical diffusion into the system. To minimize the

diffusive and other errors, it is necessary to pay strict attention to mass and momentum conservation and to leave undisturbed the vertices and triangle sides lying along boundaries, surfaces, and interfaces. There are a number of different types of mesh adjustment and restructuring which are possible, too many to cover in depth here. Therefore we will fully develop here only the most useful of the restructuring techniques, that of reconnection. We will merely list with a brief description the other techniques which we have considered. We will conclude this section with the illustration of a generalized mesh initialization which incorporates many of the adjustment and restructuring procedures.

The triangular mesh can quickly become distorted through the migration of formerly neighboring vertices in the fluid flow. Without restructuring, the distorted mesh forces the computation of derivatives using non-neighboring vertices, and quickly leads to both computational instabilities and non-physical behavior. This situation is typified by regions of long, narrow triangles bordering larger ones. This disparity in size also causes problems in that timestep errors become severe because of the short triangle sides. For extremely distorted triangles, triangle inversion becomes increasingly likely. Clearly this disparity cannot be allowed to increase indefinitely, and some restructuring is needed.

On a triangular grid, every non-boundary line uniquely specifies its two bordering triangles. These triangles form a quadrilateral for which the included line is drawn as one of two possible diagonals.

Figure 5a illustrates a configuration in which the present diagonal should obviously be reconnected. In the algorithms employed her, the shortest diagonal for each quadrilateral is always chosen, unless reconnection would produce too large a disparity in triangle areas. This is primarily a safeguard against reconnecting across inverted quadrilaterals, which would produce negative triangle areas, as in Figure 5b. It also hinders regions of sparse gridding due to an accumulation of area associated with a particular vertex.

In order to conserve momentum locally, triangle velocities after a reconnection must be constrained such that the momentum of the quadrilateral is unchanged. Furthermore, to keep the vorticity conserved, choices of triangle velocities are further restricted to those which leave the vorticity about each vertex unchanged. These requirements are sufficient to uniquely specify post-reconnection velocities for the two new triangles. The algorithm resulting from these constraints is reversible. Replacing the reconnected diagonal with the original one returns the triangle velocities to their initial values.

Further complications to the algorithms arise for reconnections affecting boundary vertex cells and by the alteration of a correction term which is carried to correct for residual errors in the pressure iteration. Of course, general bookkeeping of grid interconnections must also be updated.

This reconnection procedure allows one row of triangles to slip smoothly by another row without having to adjust the Lagrangian

vertex positions or velocities. In the corresponding fixed grid system, the triangles or rectangles bordering either side of the shear interface would soon become stretched unacceptably and a diffusive rezone procedure would have to be applied. The mesh reconnection described here is a much less drastic change since the vertex positions can be left alone. The final vertex velocities, the area-weighted averages of triangle velocities, will be consistent with conservation of vorticity and momentum, and any changes in divergence will be resolved through future pressure changes. For the special case of an unperturbed shear interface bordered by two symmetric layers of triangles, the vertex velocities would be left unchanged.

In many cases, the Lagrangian motion of the vertices naturally leads to changes in vertex density, such as the concurrent accumulation and separation of vertices at a separatrix. In such cases, although the reconnection procedures keep the mesh as uniform as possible, the grid develops regions of small and large triangles, and it may become necessary to move vertices non-physically. In the course of such a local relocation in the fluid the total mass and momentum of the triangles or cells affected should of course be conserved. Many possible formulae for the new position have been tried in LINUS2. One of the simplest has also proven to be one of the most useful--the new vertex position is taken to be the average of the surrounding positions.

In many cases the non-physical moving of vertices can still be avoided by adding and subtracting vertices where needed. For example, it may be necessary to increase resolution in some region where the flow is not naturally accumulating vertices. Such a situation occurs in cyclindrical coordinates when the fluid flow is converging on the axis somewhere. Then the triangular zones become larger and larger in cross-sectional area, but the radius of curvature becomes smaller and smaller. Better resolution is clearly required.

There are at least two ways of adding triangles to improve the resolution, triangle trisection and side bisection. In side bisection a new vertex is inserted somewhere on a side which has become too long (for want of another position we can assume the side is bisected). Lines from the opposing vertices to the new vertex are added, resulting finally in two new triangles, three new sides, and a new vertex.

The values of physical variables at the new vertex must be determined by interpolation, and hence some numerical smoothing is implied. The improvement in resolution arises because the bisected side becomes two sides and is no longer required to be straight. Thus increasing radius of curvature can be met by increased resolution. Furthermore, there is no restriction implied by material interfaces. If any of the sides is an interface, the bisection can still be performed.

A new vertex can also be added within a triangle--not on any

triangle side--in a restructuring we call trisection. Trisection alone is irrelevant because the three new smaller triangles are surrounded by a single large triangle whose behavior is constrained just as if the new vertex were not there. To be effective, trisection must be followed immediately by at least a single side reconnection. The result then is two new triangles, three new sides, and a new vertex which does not lie on the original quadrilateral diagonal. Physical variables at the new vertex are determined by interpolation just as in the case of side bisection.

Subtracting vertices can be accomplished simply as the inverse of these two processes. For "inverse trisection", a side is reconnected so that the resulting configuration has a single triangle which surrounds a vertex and three subtriangles. The interior vertex is then erased and two triangles and three sides disappear.

Similarly, a vertex can be relocated until two of the sides emanating from it form a straight line. If these two sides belong to triangles which share common sides, the two interior sides which do not form a straight line can be erased and the two straight lines combined into a single line. Here again the starting configuration for "inverse side bisection" must be a quadrilateral with four triangles inside. If such a configuration does not exist, it can be forced by local reconnections.

In subtracting triangles care should again be taken to ensure conservation of the appropriate quantities on the new reduced mesh; this means careful monitoring of the interpolation formulae.

In some cases the accumulation or separation of vertices may be occurring far from the region of physical interest. It may therefore become more desirable to let the flow slip through the grid locally. This local Eulerian behavior is similar to the rezone procedure employed by Chan[6] in a topologically rectangular mesh code. Of course, one would like to eliminate these numerically diffusive Eulerian adjustments within the fluid region so the numerical model of the fluid mirrors the usual reversibility characteristic of an inviscid fluid as nearly as possible. When Eulerian slippage is necessary, however, the recently developed Flux-Corrected Transport algorithms[8] can be adapted to minimize the diffusion and dispersive phase errors associated with the flow across the cell boundaries.

Finally, there may arise other, less general, mesh restructuring which is required by a particular problem. For example, periodic boundary conditions are often imposed to restrict a computational region. However, flow out of the periodic region will carry the grid with it. It is necessary, therefore, to construct some means of restoring triangles which exit at one boundary as triangles entering the flow at the opposing boundary. Since all physical variables remain the same for the "fictitious" triangle transfer, the procedure is mere bookkeeping, albeit complicated, and does not alter the dynamics of the flow in any way.

We conclude this section with a brief description of a generalized mesh initialization procedure. Clearly one could specify by hand all the initial vertex positions, all the sides, all the

triangles, etc. This is both laborious and apt to be repeated time and again as the physical problem is changed. The process can be automated using general utility routines to connect vertices (and store the information in the appropriate mesh index lists), to bisect sides, to reconnect quadrilateral diagonals, and to adjust interior vertex locations.

This routine requires as input an ordered set of boundary points surrounding the desired region as in Figure 6a, using cuts to reach an arbitrary number of interior subregions which are to be excluded. The initialization program can then proceed sequentially around the periphery of the region forming triangles from the specified boundary points, until the region is completely tessellated. As shown in Figure 6b, the interior subregions are automatically excluded through the inclusion of the cuts in the set of boundary points.

All quadrilaterals are then repeatedly scanned, reconnecting diagonals until the shorter diagonal is connected for each quadrilateral in the mesh, as in Figure 6c. To add interior vertices the longest remaining interior side is bisected repeatedly until either the required number of vertices has been added or all interior sides are smaller than a specified length. By reconnecting diagonals after each bisection, a fairly regular grid can be obtained, as in Figure 6d. The reconnection and vertex adjustment routines are then used iteratively until a final relaxed mesh is obtained as in Figure 6e.

This procedure can be repeated until each separate region has been filled with an appropriate number of vertices and the grid smoothed. It should be obvious that if a more symmetric grid is desired (for example, for use in debugging the dynamics of a program) this same procedure can be used to force the grid to follow any desired configuration with a minimum of effort.

Since the initialization begins with an ordered set of boundary vertices and ends with a tessellated region corresponding to a physically identifiable region, the application of boundary conditions and the storing of triangle and vertex physical variables can be handled smoothly. For boundaries which fall along straight line segments, utility programs can be employed so that only corner positions, vertex spacing and boundary conditions need be specified as input.

V. Free-Surface Waves

Stationary free-surface waves of varying amplitudes in a finite depth incompressible fluid were chosen as a test problem for the code SPLISH. The primitive equation formulation was used, and no reconnections were allowed to restrict the test to the fundamental algorithms. The linear theory is well understood and provides copious checks on the numerical results. Nonlinear theory, though less developed, also gives useful analytical results. Finally, the long-time solutions are well defined in the linear case, thus yielding a convenient test of the code's stability.

For these tests a grid was constructed to represent a homogeneous incompressible fluid of finite depth. Periodic boundary conditions were used at the sides of the region, permitting the examination of an infinite wave train while restricting the computational region to one wavelength. At the rigid bottom normal velocities must be identically zero. In this code this boundary condition is met by initializing normal velocities to zero and then choosing values of P at the lower boundary which cause normal accelerations to vanish. The pressure at the free surface is taken to be zero.

The waves are generated by specifying an impulsive sinusoidal pressure distribution at the free surface at t = 0. For all later times the free surface pressure remains zero. The accuracy of the numerical solutions is tested through the shape, amplitude and period of the generated waves.

A. Accuracy

The linear theory yields an expression for the period of a standing wave in a fluid of finite depth as

$$\tau = 2\pi \left[gk \tanh (kh)\right]^{-\frac{1}{2}} \qquad (30)$$

where $k = 2\pi/\lambda$, g is gravity and h is the undisturbed fluid depth. For our tests $\lambda = 2.5$ cm and $h = 1.0$ cm with $g = 980$ cm^2/sec, which yields $\tau = 0.12744$ sec.

In the linear theory the wave period is independent of the wave amplitude; but the theory is valid only when the ratio of wave amplitude to wavelength and the ratio of wave amplitude to

depth are both small. For our calculations of wave period, these ratios were $A/\lambda = .0269$ and $A/h = .0672$, since the amplitude had to be large enough to visually determine the period. For these values we might expect some slight nonlinear effects in the wave shape. However, as shown below, at this value of a/λ the nonlinear change in period should be negligible.

Figure 7 presents the numerically determined wave periods for different grid resolutions. The different mesh configurations are shown in the inserts to the right of the figure. To obtain a value for the period which had at least four significant figures for each of these cases, without resorting to computer simulations lasting thousands of timesteps, we interpolated between timesteps for our time measurements. For a standing wave, times can be determined most precisely as the amplitude passes through zero, when the free surface velocities are greatest. We defined the numerical value of the wave period by a least squares fit to such time determinations over computations lasting many wave periods. The error bars indicate the aggregate error in the period due to the uncertainty in the time localization. It includes an estimate of systematic error due to a possible bias in selecting the precise time when the amplitude was exactly zero.

We found the numerical period in the limiting case of an infinitesimally small mesh spacing by extrapolation, using the function

$$\tau = a(\Delta S)^2 + \tau_c.$$

The values of the parameters a and τ_c are determined by a least squares fit to the calculated wave periods, τ. The value for the period is $\tau_c = .12726 \pm .00046$, a number within 0.15 per cent of the theoretical linear value.

Contributions to error in τ from timestep size errors have been minimized by reducing the timestep when the mesh size is reduced. However, the timestep was not increased beyond .004 sec, or roughly eight timesteps per quarter-period. In basically second-order systems the Courant condition guarantees that timestep error terms are smaller than spatial derivative error terms because the non-dimensional timestep must be smaller than the space step for stability. This is demonstrated by the parabolic nature of the error curve, verified at six widely different mesh sizes.

For waves of finite amplitude, the wave profile should have the form[12]

$$y = c(ka) - \left(a + \frac{1}{32} k^2a^3 - \frac{47}{1344} k^4a^5\right) \cos kx$$

$$+ \left(\frac{1}{2} ka^2 - \frac{79}{672} k^3a^4\right) \cos 2kx - \left(\frac{3}{8} k^2a^3 - \frac{12563}{59136} k^4a^5\right) \cos 3kx$$

$$+ \frac{1}{3} k^3a^4 \cos 4kx - \frac{295}{768} k^4a^5 \cos 5kx + \cdots \qquad (31)$$

In the limit of small ka, this expression reverts to $y = h - a \cos kx$, exactly the linear result. The theoretical results then predict a smooth transition from a sinusoidal to a

nearly trochoidal form with increasing amplitude. Our results also corroborate these predictions, as shown in Figure 8. The curves in this figure have been normalized through the amplitudes of the initial perturbing pressure distributions (A_p). Inserts at the right of the figure illustrate the actual wave heights achieved for each case.

A more detailed comparison is shown in Figure 9 which includes a superposition of the theoretical predictions (Eq. (31)) and the numerical results for the $A_p = 8$ case of Figure 8. Although there are slight differences in the wave shape, they are not surprising since Eq. (31) is derived in the limit of infinite depth. As the inserts in Figure 8 show, the wave trough in this case is a significant fraction of the depth.

The waves shown in Figure 8 were all generated by an initial sine wave pressure distribution with only the amplitude of the impulse varying from case to case. The waves are labeled by the impulse amplitude (A_p) in each case. The wave amplitudes achieved, a_n, defined as $\frac{1}{2}(y_{max} - y_{min})$ are given in Table 1.

The linear theory predicts an amplitude

$$a = A_p \frac{2\pi}{\rho g} . \tag{32}$$

Using the theoretical value for the period we have

$$a_t = .01761\ A_p \tag{33}$$

where we have included an additional factor of 0.35, to be explained below. As can be seen from the table, the numerically predicted amplitudes deviate from the linear theory by 4.6 percent at small amplitudes to 7.0 percent at large amplitudes. However, the size of

the mesh introduces a finite-difference truncation error in the period, as determined by Figure 7. If we use this value a_c for the period, instead of the value for an infinitely small mesh, we obtain

$$a_c = .01681\ A_p \tag{34}$$

as shown in the table. The errors are seen to be negligibly small for the first three cases and deviate from the linear prediction only for the obviously nonlinear cases. These numbers, coupled with the observation that there is no measurable increase in the period up to the $A_p = 8$ case and an increase in period for $A_p = 8$ and 16, indicate that the $A_p = 4$ case, used in constructing Figure 7, is indeed a good approximation to the linear period. A further check that the percentage deviation for the $A_p = 8$ and 16 cases is indeed due to nonlinear effects can be made from theoretical calculations of the period for the nonlinear case. Tadjbakhsh and Keller[13] determine to third order in a/λ that the period should increase if the initial fluid depth h is larger than 0.17λ. In our case the ratio $h/\lambda = 0.4$, and so the increase in period which is observed is indeed reasonable. Penney and Price[12] also predict an increase in their expansion to fifth order for the case of infinite depth. The change in period predicted by each of these theories is included in Table 1, as $\delta\tau_T$ and $\delta\tau_P$, respectively, for comparison with the amplitude results. The percentage change should be equivalent since the amplitude achieved in Eq. (32) is inversely proportional to the period. As can be seen from the table, the general trend of the agreement is good, with the numerical

results lying intermediate between the two theories.

B. Stability

In numerical integrations convergence to the correct solutions is always incomplete and leaves a residual error. To ensure that this residual error does not accumulate significantly, affecting both accuracy and stability of the code, it has been included explicitly as "negative feedback". For the pressure iteration in which the velocity divergence around a mesh point is iterated to zero, $(V_c - V)/\Delta t$ is used as a residual correction, where V_c is the initial ($t = 0$) volume of the cell and V is its current volume. For pressures at the bottom of the fluid, a $\underset{\sim}{v}_\perp$ source term is added to compensate for accelerations through pressure gradient forces perpendicular to the rigid boundary which have accumulated due to incomplete convergence.

While the feedback mechanisms affect stability indirectly, there exists a need for more direct control of numerical instability. As mentioned above, the divergence is zero only at full timesteps and, therefore, follows a time history as shown in Figure 10. The discontinuous change in divergence results from the linear transformation at $t-\frac{1}{2}\delta t$ while the linear change reflects the pressure iteration forcing the divergence to zero at t and $t-\delta t$. As seen from the figure, this scheme is marginally stable with the absolute value of the divergence at $t+\frac{1}{2}\delta t$ equal to that at $t-\frac{1}{2}\delta t$. However, moving the "half" timestep slightly forward of center ensures that the divergence at $t+\delta t$ will be smaller and, hence, cannot accumulate

secularly. However, this is done at the sacrifice of perfect time reversibility of the algorithm. It is the latter scheme which was used throughout this report, since many periods were needed for accurate statistics. For example, 12 complete wave periods were taken for the four coarser grids in Figure 7. The forward shift ($\alpha = 0.35$) is also the source of the 0.35 factor alluded to previously. Since the impulse pressure is applied only for the first "half" timestep, shortening the time also reduced the amplitude of the impulse.

As was mentioned above, long time integrations are possible with these algorithms, with runs typically 500 timesteps. For the best cases, no discernible changes of shapes, amplitudes or periods of the waves occur until $t \approx 15\tau$. Figure 11 illustrates the type of instability presently found, here evident at $t = 12\tau$. At $t = 10\tau$ and $t = 11\tau$ there is a slight flattening of the crests of the waves presaging the more violent onset of instability at $t = 12\tau$. This behavior reflects an accumulation of truncation error which initially only perturbs the grid near the surface, but which grows in time until the grid is sufficiently displaced locally to finally become Courant-unstable. It is not surprising that the instability appears. Rather, it is surprising that its appearance is so retarded despite ample encouragement through the suppression of numerical averaging and artificial viscosity. It should also be noted that there is no evidence of nonlinear alternating error even after the instability is well developed, as was suggested[9] for the triangular grid.

In summary, the free-surface wave test has shown the code to

have both excellent accuracy and stability. The detailed examination of its convergence has further shown the code to be effectively second-order. Finally, it has shown that good stability can be achieved without artificial viscosity or spatial numerical averaging.

VI. The Kelvin-Helmholtz Instability

The dynamics of a perturbed shear layer near a free surface for an inviscid, incompressible fluid was chosen as a test of the numerical techniques devised for the reconnections. The Kelvin-Helmholtz instability is an appropriate vehicle for this test since it has received a great deal of attention both in homogeneous and stratified flows. It remains a focus for research at present, partially because of the complexity involved in following its development even numerically, particularly at high Reynolds numbers, and also because areas such as interactions with nearby free surfaces have remained almost totally unexplored.

Although the purpose of this study is to test new numerical methods and is not an exhaustive study of the evolution of shear layer instability, a good deal of insight into the physical processes involved can be gained from even coarsely gridded simulations such as these. In particular, the techniques presented here enable us to trace the evolution of the layer from the initial perturbations, through their linear stages of growth, into the manifestly nonlinear growth of Kelvin-Helmholtz billows and beyond, into the formation of a turbulent layer.

A. Shear Layer Dynamics

The complete evolution of a shear layer near the free surface of a fluid must be calculated numerically since no analytic, closed form solutions exist. However, the early growth should parallel that predicted by linear theory until the free surface becomes sufficiently deformed to effect perturbations in the shear layer. The linear theory as developed by Holmboe[14] is quite amenable to computer calculations and has the further advantage of offering additional physical insight into the problem. The geometry he employs is that of a finite shear layer bounded by equal outer layers of irrotational fluid. He examines the evolution of waves symmetrically imposed on each of the shear layer boundaries.

In Figure 12 we have formed a symmetric disturbance of the layer by giving a small sinusoidal perturbation to each boundary of the layer. If the boundaries are deformed in phase, the symmetric wave is in the "a-state". For deformations which are out of phase, the wave is in the "b-state". Physically, each deformation has transferred vorticity across the undeformed layer boundary, forming a vorticity wave. This wave in turn impresses a velocity field on the surrounding fluid. The subsequent motion of the layer is derived from the mutual forcing of each of the symmetric boundary waves through these fields, which augment each other in the b-state and oppose each other in the a-state.

For perturbations of the shear layer with wavelengths less than 4.9 times the thickness of the layer, the upper and lower

deformations travel with the flow (to the right and left, respectively) with oscillating phase speed and amplitude. For wavelengths greater than 4.9 times the layer thickness, the deformations travel with the flow from the a-state to a stationary state of exponential growth. Deformations reach the same stationary state from the b-state by propagation against the flow. The phase of the stationary state is given by

$$\tan^2\sigma_s = - \frac{(1-n_b)}{(1-n_a)} ,$$

where $$n_a = (1 - e^{-kd})/kd, \tag{35}$$

and $$n_b = (1 + e^{-kd})/kd.$$

The growth rate, n, for this state is

$$\left(\frac{n}{kU}\right)^2 = (1-n_a)(n_b-1), \tag{36}$$

where 2U is the change in velocity over the layer of width d, and $k = 2\pi/\lambda$. For $\lambda \gg d$, $kd \ll 1$ and $n \to Uk$; i.e., the shear layer behaves like a vortex sheet. The wavelength of maximum growth $\left(\frac{dn}{dk} = 0\right)$ is given by $\lambda_{max} = 7.9d$.

For a homogeneous shear layer, the main difference between the behavior of a shear layer and a vortex sheet is therefore a short-wavelength cutoff for growth of an initial perturbation. For

wavelengths longer than the cutoff, the instability grows in a mode in which the upper and lower boundaries of the layer remain phase-locked and the growth proceeds at a rate generally smaller than that for a vortex sheet. While the vorticity associated with the shear layer remains constant, its distribution changes, and becomes more clustered as the amplitude of the stationary state increases (Figure 12).

If a statically stable density distribution exists within the shear layer, however, spatial fluctuations of particles within the region of density gradients will alter the vorticity baroclinically, and the changing vorticity field will produce a concomitant velocity field in the surrounding fluid. This field can also oppose or augment the symmetric wave at the boundaries of the layer. Yet the forcing is not mutual, since the baroclinically generated wave is produced locally and is not directly affected by the deformations of the shear layer boundaries.

The behavior of a stratified shear layer depends heavily on the geometry of the layer and the surrounding irrotational layers and on the functional form of the density within the layer. For a constant density gradient over the layer, Goldstein[15] found that if the outer layers were of infinite depth, the wavelengths accessible to the phase-locked growth mode were restricted to a spectral band which shifted to shorter wavelengths with increasing Richardson number. Holmboe found similar restrictions for this mode when the depth of the symmetric outer layers is h;

$$\coth kh + \coth \frac{kd}{2} > (1 + J)/(\frac{1}{2} kd) > \coth kh + \tanh \frac{kd}{2}, \qquad (37)$$

where J is the Richardson number,

$$J = \frac{g\Delta\rho d}{\rho(\Delta U)^2} .$$

Linear analysis of perturbations of shear layers in general predicts a phase-locked growth for unstable waves, with the growth rates and allowed wave numbers determined by the details of the shear layer and its stratification. Although the linear theory is inapplicable for later growth of the layer, the principal mechanism for the nonlinear growth has already been determined.

As shown in Figure 13, at the end of the linear stage of growth the vorticity has been concentrated in a series of tilted bands and depleted from the sheared regions between them. We will call these concentrations of vorticity "cores" and the regions of strong shear connecting them the "braids". As the vorticity increases in the cores, they rotate more strongly and further stretch the braids, transferring even more vorticity to the cores. As the roll-up continues, the portions of the braids nearest the cores are wound onto the cores also, accelerating the vorticity transfer. For a homogeneous fluid, this process could continue until the braids are totally depleted of vorticity, leaving the core, in the absence of other instabilities, rotating about an axis at the center

of the layer.

For a stratified fluid, the vorticity in the braids increases with tilt, and stretching intensifies the local density gradients there. Vorticity is also generated in the cores in such a way as to keep the total vorticity per wavelength constant.

In either case, the Kelvin-Helmholtz billows which are formed increase in size due to the entrainment of surrounding fluid. Turbulence begins in the center of the core and gradually increases as the entrainment progresses. For stratified fluids, the braids and the edges of the cores become regions of intense density gradients along which smaller scale perturbations may appear. Under the combined influence of shear, strong density gradients, increasing billow size due to entrainment, and a growing central turbulence, the billows eventually coalesce into a layer which becomes nearly isotropically turbulent.

Surprisingly, at times long in comparison with the initial growth rate times, this turbulent layer is not without structure. Micro-layers appear which, by their inclination to the original shear layer, can be shown to be associated with the original billows[16], despite their subsequent shearing and coalescence. The mechanism for the retention of the identity of the original billows long after the formation of the turbulent layer has remained obscure.

B. Numerical Solutions

Figure 14 illustrates the linear development of a shear layer near a free surface as calculated by SPLISH. The inserts in the figure are taken from the computer calculations and show the location of marker particles. The initial flow is specified by the triangle velocities which are chosen to be constant in each layer of fluid. We have given a velocity U = 5 cm/sec (to the right) to all triangles above the central discontinuity and U = - 5 cm/sec for all triangles below it. In terms of triangle velocities, the shear layer can then be localized to exactly the discontinuity. However, the layer will behave as if it were one cell wide, simply because the equations governing its motion are differenced over a grid.

A sinusoidal perturbation is given to the central discontinuity at t = 0 with an amplitude equal to 2% of the fluid depth or 20% of the shear layer. The perturbation was exaggerated to help visualize the flow in its early stages, but because of the rather large amplitude,nonlinear effects will be evident early in the simulation. No effort has been made to impose velocity fields consistent with these initial deformations, since the velocities will adjust themselves self-consistently after a short period of time. For small perturbations, these fludtuations are not large, and do not significantly affect the dynamics. It should be emphasized, though, that the initial conditions are not eigen-solutions for a given layer perturbation, but merely represent an instantaneous deformation at the center of the layer.

As shown in Figure 14, a symmetric wave near the b-state is impressed on the shear layer by the central perturbation. Initially the symmetric wave is small compared to the central deformation. The phase speed of the symmetric wave is reversed by the velocity fields associated with the central wave as it passes through the b-state toward the a-state. The symmetric wave then enters the predicted phase-locked mode of growth as it nears the stationary state. As shown in the graph, the numerical calculation determines the stationary state to be at $\sigma_s = -70.5^\circ$ and gives a growth rate of $n = 19.6$. From Eqs. (35) and (36) for $d = 0.1$ cm, $\lambda = 1.0$ cm and $U = 5$ cm/sec, the linear theory predicts $\sigma_s = -67.5^\circ$ and $n = 19.13$, in quite good agreement with the numerical result. Roughly half of the small discrepancy is due to nonlinear effects, evident by the peaking in the last insert, and the rest is due to the finite resolution of the grid.

To check whether this agreement was merely fortuitous, the simulation was rerun with half the initial perturbation. The numerical results were $\sigma_s = -69.5^\circ$ and $n = 19.2$, even closer to the predictions of the linear theory, as should be expected.

We have also performed this test in the limiting case of letting the perturbation equal zero. In this case the shear layer is in an unstable equilibrium, and any error in the assignment of post-reconnection velocities should be mirrored in a similarly exponentially growing disturbance. This test has been run for 500 timesteps with no changes in the grid aside from the initial

shear (vertical grid positions indicate errors $\leq 10^{-5}$ for a single precision calculation). The final grid looks exactly as it did initially, yet each vertex in the irrotational fluid above the shear layer has traversed the entire grid ten times.

Figure 15 presents another check on the accuracy of the computer calculations in the linear regime. Here the wavelength of the computational region is increased to $\lambda = 1.5$ cm instead of $\lambda = 1.0$ cm as in the previous simulations. Although the initial perturbation has this same wavelength, the half-wavelength wave, with $\lambda = 0.75$ cm, has a substantially larger growth rate. This is because $\lambda = 7.5d$ and, as shown above, this is quite near the wavelength for maximum growth for the layer, $\lambda = 7.9d$. Physically, we would expect that such a layer would develop through the half-wavelength mode due to small random perturbations. Numerically, we expect the same behavior, since the initial perturbation is not an eigensolution for the layer.

As shown in Figure 15, the symmetric wave that initially develops is in the b-state ($t = 0.02$), but here it continues to grow in the b-state at the expense of the central wave. By $t = 0.08$, the central perturbation is reduced to nearly one-tenth of its original amplitude and has departed from its sinusoidal form, whereas the symmetric wave is well developed and is predominantly in a full-wavelength ($\lambda = 1.5$) b-state. As the symmetric wave moves further off the b-state, the fastest growing perturbations of the central wave are not the full-wavelength modes but the half-

wavelength (t = 0.12). By t = 0.16, the growing central perturbation has induced a half-wavelength mode on the symmetric wave, and at t = 0.22 the layer is clearly predominantly perturbed by a half-wavelength disturbance. The symmetric wave at this time is not far from the stationary state for the half-wavelength mode, and its rapid growth clearly carries the layer's development into the nonlinear regime in just 0.08 sec more (t = 0.30).

Figure 16 presents the results of calculations for the nonlinear growth of a shear layer into a Kelvin-Helmholtz billow. This simulation is a continuation of the linear stages shown in Figure 14. The lower four inserts present vorticity contours at the same times as the marker particles displayed above. Since the initial shear layer for this problem is one cell wide with constant velocities above and below the layer, those cells defining the layer will have a uniform vorticity. In the inserts the vorticity is contoured over the mesh triangles. The total width of the layer therefore appears twice as large since the contours extend to neighboring vertices. Because the algorithms identically conserve vorticity, those cells initially in the layer always have the same vorticity, but the distribution of these cells will change.

In the first insert the vorticity has begun to cluster into bands with an accompanying thinning and stretching of vorticity outside the bands. Due to our finite resolution, the thinning in this plot is evidenced more by the density of vorticity-carrying vertices than by an actual decrease in width of the band. The

enlarged band tilts and breaks, entraining the surrounding fluid. The roll-up and gathering-in of vorticity is shown by the core rotation as well as by the movement of two vorticity-carrying cells left in the braids. In the last frame the Kelvin-Helmholtz billow has matured and continues to wind in the remaining vorticity.

The marker particles carry complementary information. The nonlinear peaking at the end of linear growth rapidly leads to breaking as the band tilts. In finer resolution this breaking at the band edges would be represented by small vortices. In simulations using smaller initial perturbations these vortices are replaced by a much more uniform roll-up about the central core. Entrainment in those cases is always directed toward the center of the core. The smaller scale motion shown here is more effective in entraining fluid. The billow grows more rapidly through the increased transfer of fluid to the core by the additional entrainment of the small vortices.

In these plots the stretching and thinning of the braids is much more evident, particularly as the billow matures and continues to wind the braid around the core. The kinks in the braids at the locations of the remaining vorticity are not physical, but are due to finite resolution. However, they do not represent a numerical instability, since these perturbations do not increase with time.

The details of these simulations, particularly those with smaller initial perturbations, are reminiscent of the results obtained by Patnaik et al.[17] in their beautifully resolved simulations

for which at higher Reynolds numbers the core rotation becomes rapid. The primary differences appear to be the result of our nearly four times coarser gridding. For example, none of their simulations has a well developed braid deformation, even at their highest Reynolds numbers, which confirms that their origin here is due to finite resolution.

Of course the main difference between these calculations and previous ones is the presence of a free surface near the shear layer. In Figure 17, the evolution of the free surface near a K-H billow is presented as a function of time for two different amplitudes of the initial perturbation. Time begins at roughly the end of the linear growth stage and ends at the formation of a mature billow in each case. The difference in the times is accounted for by the larger initial amplitude in the second case and by the more rapid development of its billow due to the small-scale motion.

The general shape of the free surface is the same in both cases. The free surface is lowest directly above the billow and highest over the braids. Wave steepening is driven by the fluid entrainment into the billow below. The steepest portion of the surface wave remains above the strongest vertical velocities of the entraining fluid, moving to the right as the billow grows.

Details of the shape of the surface wave are strongly coupled to the size of the billow and its distance below the free surface. In the figure we see that the amplitude is 50% greater for the $A_0 = .02$ case, although the total time elapsed is significantly shorter than

the $A_0 = 0.1$ case. This occurs because the billow builds more rapidly for the larger perturbation, and this more rapid growth brings the billow closer to the free surface.

Throughout this section all examples have used homogeneous shear layers, and connotations such as "braids" have been used on the basis of path lines for Lagrangian particles alone. With the addition of stratification, such entities become loci for changing density gradients, and may in themselves now baroclinically generate vorticity. Figure 18 presents vorticity contours for a stratified shear layer using an even coarser resolution. At $t = 0.16$, vorticity generation in the braids is already evident, as well as the more obvious vorticity depletion in the region which will soon develop into a core. As the layer rolls up at $t = 0.24$, this trend is quite striking, and the small extent of the core is emphasized. By $t = 0.28$, the core's vorticity is severely depleted and the core is winding in the large concentrations of vorticity in the braids. At $t = 0.32$, the regeneration of vorticity in the thinned braids is already well underway.

It is clear that despite the addition of density stratification, the shear layer is still unstable to phase-locked roll-up. This is exactly what is predicted by Eq. (37); the limits are 1.49 and 3.09, and the value of $(1 + J)/(kd/2)$ is 2.06.

We checked to see if stabilization could be achieved with a constant gradient by increasing the depth of the shear layer to $d = \frac{1}{2}$ and $\Delta\rho$ to 0.2. Here Eq. (37) predicts a strongly stabilized

layer for $\lambda = 1.0$ disturbances. The numerical results conformed to this expectation. In the previous case the growth rate for the perturbation had merely been attenuated by the stratification. Here the deformation begins collapsing immediately, and at $t = 0.4$ is nearly gone. The layer continues its progress through the mean position and undergoes a standing oscillation.

The computer simulations therefore seem to agree in all respects with predictions of both linear and nonlinear theory. However, there is the added advantage in that the simulations can be extended economically to longer times. In fact, we have performed calculations at the coarser resolutions which have followed the shearing of billows into their coalescence with neighboring billows. A sample calculation is illustrated in Figure 19. At $t = 0.356$ the entire outer edge of the billow is ringed by former braid wound onto the billow, and in a stratified fluid would be a locus for high density gradients. Prior to this time, the billow has already sheared and irrotational fluid has been trapped between the spreading billow and the braid, accentuating the stretching of the braid even more. As shown at $t = 0.396$ and $t = 0.416$, the sheared billow continues to overlap the braid in this region, and by $t = 0.436$, yet another layer of irrotational fluid is about to be pressed against this already twice overlaid portion of the braid. Although the marker density is quite small at later times, it is possible to see that at $t = 0.456$ and $t = 0.476$ this portion of braid is still intact and further stretched, being merely overlaid by the shearing billow in successive layers of turbulent and

irrotational fluid.

This development is mirrored in the vorticity contours. It is immediately obvious that the vorticity-carrying vertices perform localized rotations during the shear, and do not migrate horizontally. By transposing an image of the braid locus onto the contours and tracing the movement of individual vertices, it can be seen that the braid effectively divides the fluid into separate regions, despite the increasingly isotropic turbulence throughout other regions of the sheared billow.

The long-time simulations therefore indicate that portions of the braid survive well into the turbulent regime, despite the fragmentation of the billows which created them. These extremely thinned and overlaid braids inhibit motion across themselves, and serve as an excellent locus for the building of the micro-layers seen in late-time turbulent regions as noted by Thorpe.[16]

VII. Conclusions

In this paper we have described an entire series of techniques and algorithms for using a Lagrangian 2D mesh of triangles to represent and solve free surface problems in incompressible hydrodynamics. The results of our research in this area have been incorporated into the free-surface hydrodynamics code SPLISH, which was used in calculations to test the accuracy and stability of the algorithms. These results on the problem of free-surface waves have shown that the code is an extremely accurate, second-order code and has good convergence. It also has the ability to calculate

long-time solutions accurately without the benefit of artificial viscosity or numerical averaging.

The simulations of the Kelvin-Helmholtz instability have shown that it is indeed possible to devise accurate mesh reconnection algorithms which permit long-time solutions even in the presence of strong shear. Here too, the numerical solutions have been found to be accurate, agreeing with both linear and nonlinear theory for both homogeneous and stratified flow. It has been encouraging that even coarsely gridded problems such as these not only preserve the details of previous higher resolution, Eulerian calculations, but even elucidate current problems such as the behavior of free surfaces near a shear layer and the development of microlayers in turbulent regions formed by coalescing Kelvin-Helmholtz billows.

Our experience with this code is by no means extensive, however. Prior development through LINUS2, a triangular code for electromagnetically imploded liquid metal rings, and subsequent extensions, such as TORUS2, an r-z quasi-static MHD code for calculating plasma equilibria in Tokamaks, have led to further experience in how to difference properly equations over a triangular mesh. Yet in all cases we have proceeded only with those schemes which seemed most productive for the immediate future. In this paper we have tried to illuminate some of the pitfalls which we did not avoid in the hope that others will. It is only through cumulative experience such as this that an understanding of hydrodynamics using Lagrangian triangular meshes will begin to be complete.

Acknowledgment

This work was sponsored by the Office of Naval Research under Contract No. RR0110941.

Appendix

Topological Constraints

Since little is known about the convergence of Lagrangian algorithms differenced over triangular meshes, it is important to obtain a feel for the restrictions that the grid itself imposes on the physical solutions to a problem. Perhaps the most fundamental restriction is the relation between the number of vertices and the number of triangles, the "counting problem" mentioned in the text and derived in detail below. Other constraints may be more subtle, such as the case presented here in which local grid structures can force non-local behavior onto the physical equations which are differenced over the mesh.

To define the"counting problem" we can consider first the case of a rectangular computational region as shown in Fig. 20a. The derivation was pointed out to us by Dr. Glynn Roberts of Science Applications, Inc. We want to find the sum of the angles subtended by each vertex. For an interior vertex, we have simply

$$\sum_{\text{i}} \measuredangle = 2\pi \qquad \text{(A-1)}$$

For vertices on the boundary, but not at a corner

$$\sum_{\text{b}} \measuredangle = \pi, \qquad \text{(A-2)}$$

while, for corner vertices,

$$\sum_{\text{c}} \measuredangle = \pi/2 \ . \tag{A-3}$$

Therefore, the total is the sum of Eqs.(A-1) to (A-3)

$$\sum \measuredangle = (N_v - N_{v_b})2\pi + (N_{v_b} - 4)\pi + 4(\pi/2) , \tag{A-4}$$

where N_v is the total number of vertices, and N_{v_b} the number of boundary vertices. Simplifying

$$\sum \measuredangle = 2\pi N_v - \pi N_{v_b} - 2\pi \ . \tag{A-5}$$

This sum can be calculated another way, through the triangles in the mesh,

$$\sum \measuredangle = \pi N_t \ , \tag{A-6}$$

where N_t is the total number of triangles. Equating Eqs. (A-5) and (A-6) gives the result,

$$N_t = 2N_v - N_{v_b} - 2 \ . \tag{A-7}$$

The equations in the text follow immediately. If there are no interior vertices, $N_v = N_{v_b}$, and $N_t = N_v - 2$. If the mesh is finely

tesselated, $N_v \gg N_{v_b}$, and $N_t \sim 2N_v$.

For an arbitrarily shaped boundary, the proof is essentially the same. Equation (A-4) is replaced by

$$\sum \measuredangle = (N_v - N_{v_b})2\pi + (N_{v_b} - n)\pi + (n-2)\pi, \qquad \text{(A-8)}$$

where n is now the number of corner vertices, and $(n-2)\pi$ is just the sum of interior angles of a polygon of order n.

To illustrate the influence of the topology of the grid on the physical solution, we will use the example of an isolated vortex in an incompressible fluid. We would like to determine if the mesh itself allows such a vortex to persist while the fluid remains divergence-free.

Fig. 20b illustrates the problem. We define a point vortex at the central vertex c, and further assume that the peripheral vertices d → o are all fixed. We would like to know if it is possible, under these restrictions, to move vertices 1 → 6 in a manner consistent with divergence-free flow about a central vortex. Clearly, since the periphery is fixed, the total area is conserved, and so at least in this sense the flow is divergence-free. We would like to be more restrictive, however, and see if the area about each vertex can be conserved. We can use the usual definition of the vertex cell as 1/3 of the area of all triangles including the vertex.

To illustrate, assume vertex 2 is moved first. If it moves parallel to side gf, the area of triangle Δ2gf is conserved. To

conserve the area of vertex cell f, vertex 1 must be moved in the same sense as vertex 2 and in such a way as to conserve the combined areas of triangles Δ12f and Δ1ef. Once the velocity of 2 is chosen, there exists just one possibility for the velocity of vertex 1; it must be parallel to the line df and of a magnitude such that the area of Δ12f is conserved. It must be parallel to df to conserve the area of quadrilateral 1def, which assures the constancy of cell e. Motion parallel to df assures the area of triangle Δ1df is conserved. Triangle Δdef is conserved trivially since all its vertices are fixed.

Once the velocity of vertex 1 has been so chosen, the velocity of vertex 6 is found in a similar manner, and so on around the central vertex c. In this way vertices 1 through 6 could be moved conserving the area of all peripheral cells. Unfortunately, the scheme fails at vertex 5. The velocity of 5 is constrained to be parallel to ℓn to conserve the area of vertex cell m, and parallel to km to conserve cell ℓ's area. Therefore, it is possible to define a velocity field consistent with the desired motion only if all the vertices have 6 or fewer connections. It is not possible if any of the vertices $1 \rightarrow 6$ has seven or more connections.

The local topology of the grid has forced a non-local character onto our solution. If a system of algorithms is used in which the vertex velocities are found iteratively such that cell areas are conserved, vertex ℓ or m must move also. If, further, the vorticity is advanced in a similar manner, the conservation of vorticity about c is in conflict with the divergence

iteration, and extremely slow convergence will result. Therefore, as discussed in the text, improved convergence can be achieved only by conserving one of these quantities by construction. In the ψ-ζ formalism, the flow is divergence-free by construction, whereas for the P-$\underset{\sim}{V}$ formalism, vorticity is exactly conserved by construction. Iterations of vertex variables to conserve the remaining quantity will still be non-local, but will converge much faster than schemes in which conservation of both quantities is attempted by iteration.

The implications of this proof are not restricted to triangular meshes, since any quadrilateral mesh can be reduced to a triangular mesh by drawing diagonals. Viewed in this light, the advantage of the triangular mesh is to explicitly indicate which diagonal should be used for more accurate computation. This improved accuracy can be obtained in a quadrilateral mesh by a non-linear correction term, but it would fail for exactly the same reasons as given here; it would emphasize the connection which forces non-locality. In other words, in the triangular mesh this problem could presumably be avoided by forcing only six connections or less per vertex; i.e., reconnect diagonal km vs. 05. This would remove the non-local character, but at the expense of a less accurate solution.

References

1. W. P. Crowley, in Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics, (Springer-Verlag, New York, 1971).
2. J. Boris and K. Hain, Bull. Am. Phys. Soc., 18, 1299 (1973).
3. P. C. Liewer and J. P. Boris, Bull. Am. Phys. Soc., 19, 483 (1974).
4. A. Barcilon, D. Book, J. Boris, A. Cooper, K. Hain, P. Liewer, A. Robson, R. Shanny, P. Turchi, and N. Winsor, Plasma Physics and Controlled Nuclear Fusion Research, 1974, IAEA-CN-33, p. 567.
5. C. Brennen and A. K. Whitney, in 8th Symposium on Naval Hydrodynamics, Pasadena, August 1970.
6. R. K-C. Chan, Science Applications, Inc. Report 73-575-LJ, November 1973; J. U. Brackbill, Methods in Comp. Phys. (Academic Press, New York, 1976), Vol. 16, p. 1.
7. J. P. Boris, in Proceedings of the Fourth Conference on the Numerical Simulation of Plasmas, Naval Research Laboratory, Washington, D. C., 2-3 November 1970.
8. J. P. Boris, in Proceedings of the Second European Conference on Computational Physics, North Holland, 1976.
9. J. P. Boris, K. L. Hain, and M. J. Fritts, in Proc. First Int'l Conference on Numerical Ship Hydrodynamics, NBS, Gaithersburg, Md., Oct. 1975.
10. M. J. Fritts, Science Applications, Inc. Report SAI-76-528-WA, 1976.

11. M. J. Fritts, Science Applications, Inc., Report SAI-76-632-WA, 1976.

12. W. G. Penney and A. T. Price, Phil. Trans. A. 244, 254 (1952).

13. I. Tadjbakhsh and J. B. Keller, J. Fluid Mech. 8, 442 (1960).

14. J. Holmboe, Geofys. Publ., 24, 67 (1962).

15. S. Goldstein, Proc. Roy. Soc., A132, 524 (1931).

16. S. A. Thorpe, Boundary-Layer Meteorol., 5, 95 (1973).

17. J. C. Patnaik, F. S. Sherman, and G. M. Corcos, J. Fluid Mech., 73, 215 (1976).

Table I. Comparison of numerical results with linear and nonlinear theory.

A_p	a_n	a_t	δa_t	a_c	δa_c	$\delta\tau_T$	$\delta\tau_P$
1	.0168	.01761	-4.6%	.01681	0%	.02%	.004%
2	.0336	.03522	-4.6%	.03362	0%	.09%	.016%
4	.0672	.07044	-4.6%	.06724	0%	.38%	.062%
8	.1338	.14088	-5.0%	.13447	-0.5%	1.6 %	.25 %
16	.2620	.28176	-7.0%	.26894	-3.6%	6.6 %	1.01 %

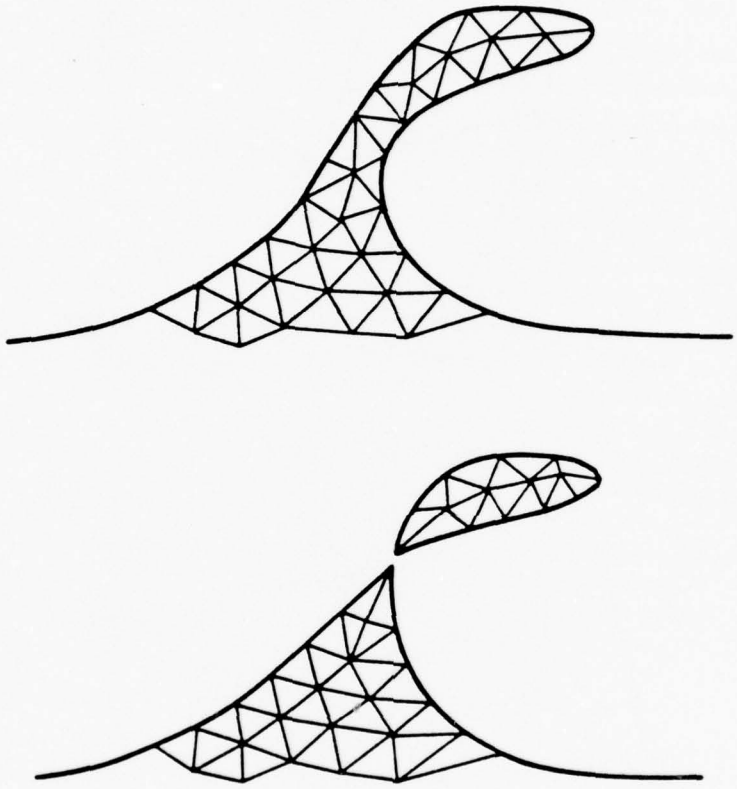


Fig. 1 — Schematic representation of the use of a Lagrangian triangular mesh to facilitate the nonlinear hydrodynamics of separating free-surface flows

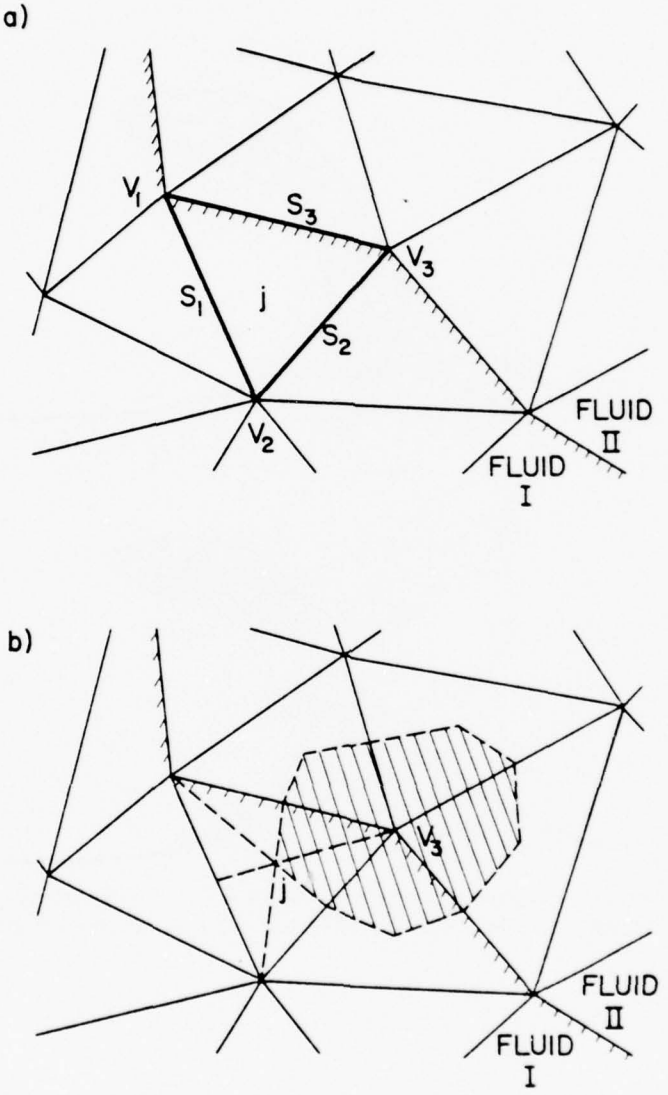


Fig. 2 — A section of a triangular mesh including an interface and showing a vertex cell

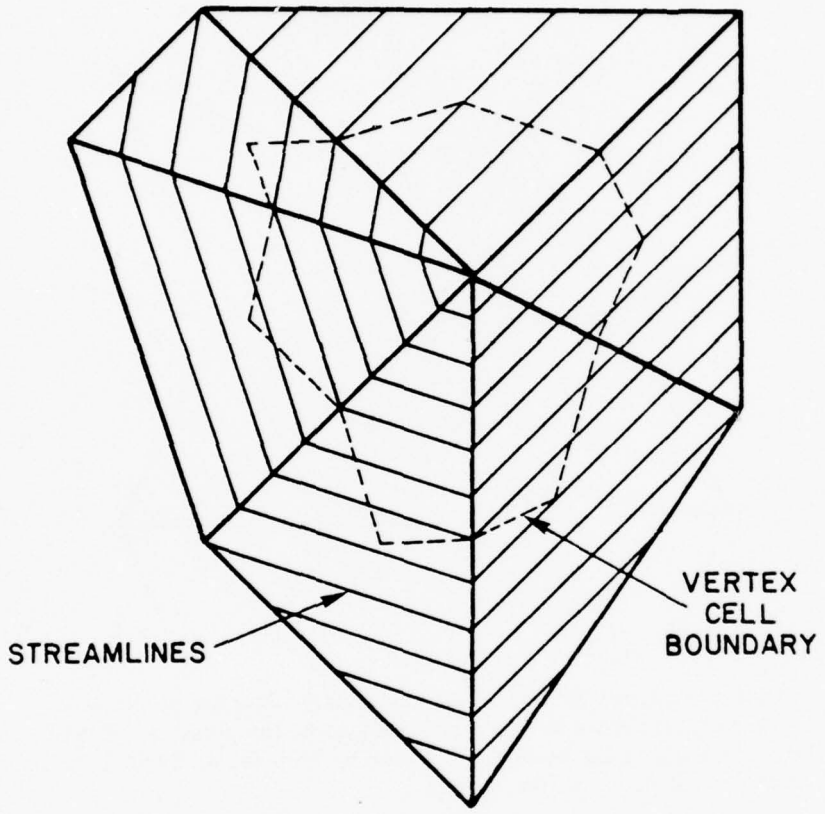


Fig. 3 — Streamlines of a flow field calculated from a vertex-centered stream function. $\underline{V}$ is constant throughout each triangle.

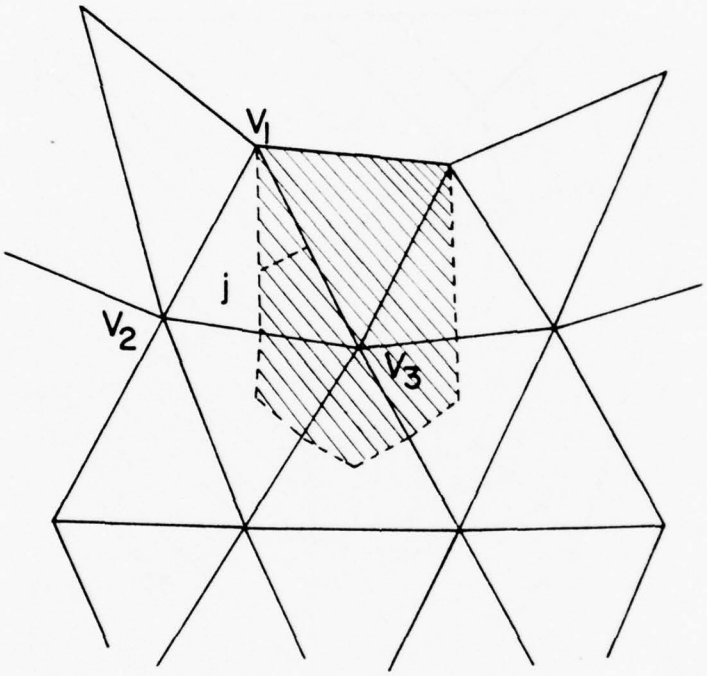


Fig. 4 — Enlarged cell areas near a boundary. Because pressures at boundaries are fixed by boundary conditions, the pressures of vertices adjacent to the boundary are used to keep the additional surface areas divergence-free as well.

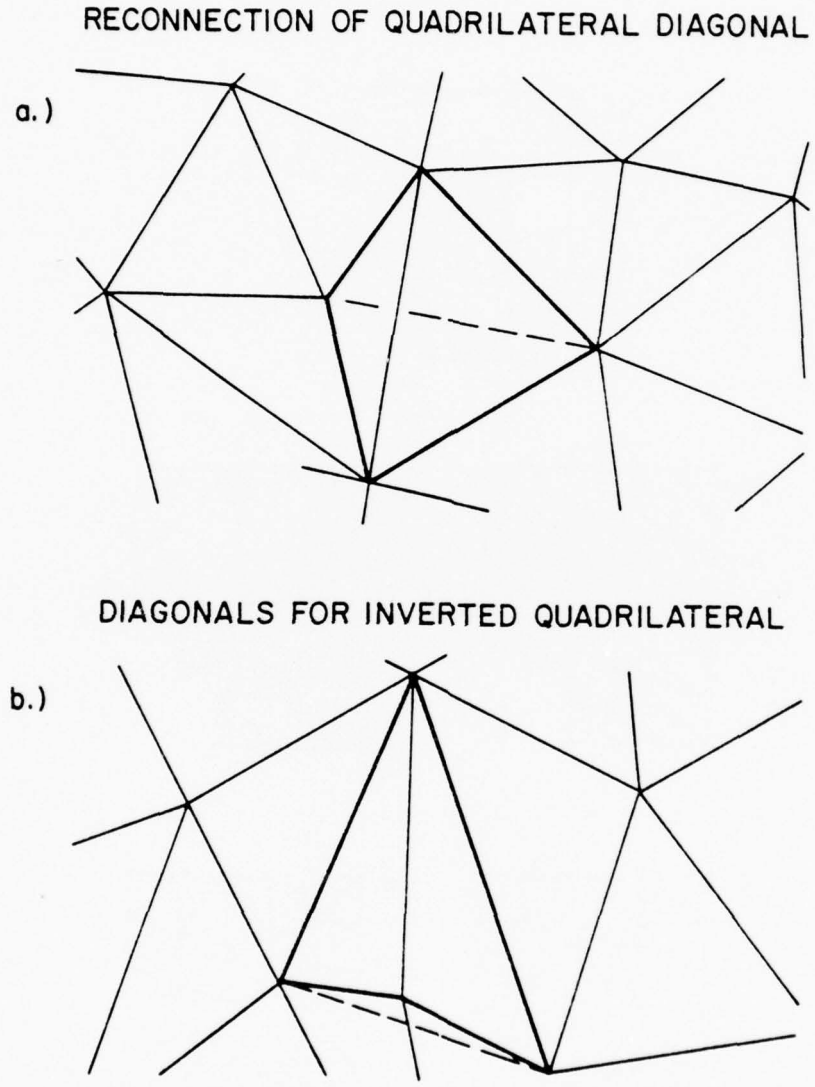


Fig. 5 — Portions of a grid illustrating possible reconnections. In part (a) the dashed diagonal will be chosen for the shaded quadrilateral rather than the present, longer, diagonal. In part (b) the diagonal cannot be reconnected since the alternative diagonal, though shorter, lies outside the "inverted" quadrilateral.

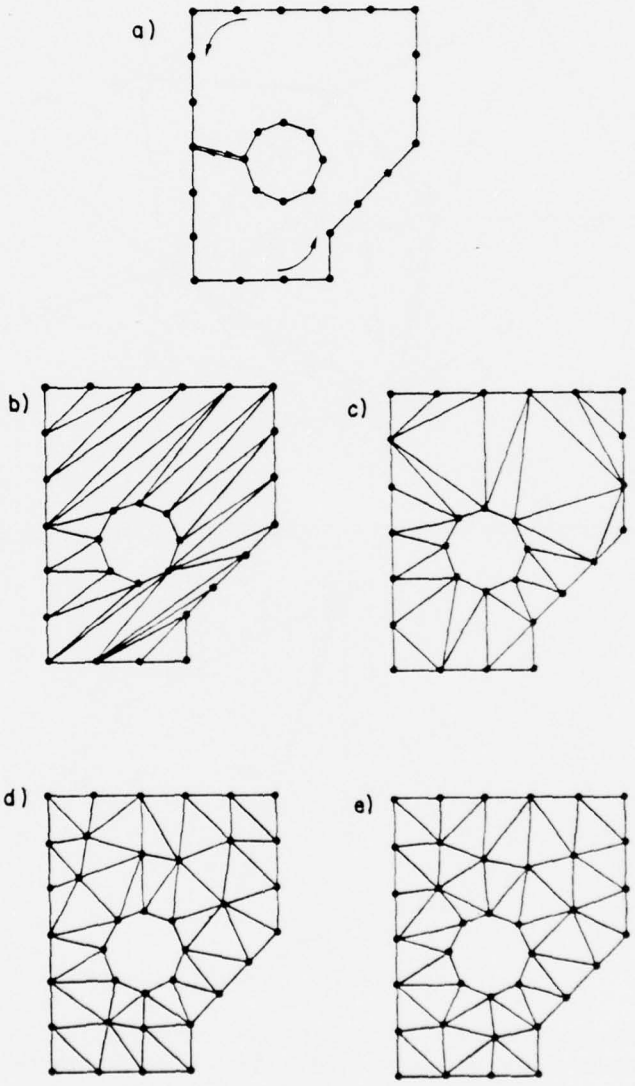


Fig. 6 — The five stages of automatic initialization for a triangular mesh region with irregular boundaries

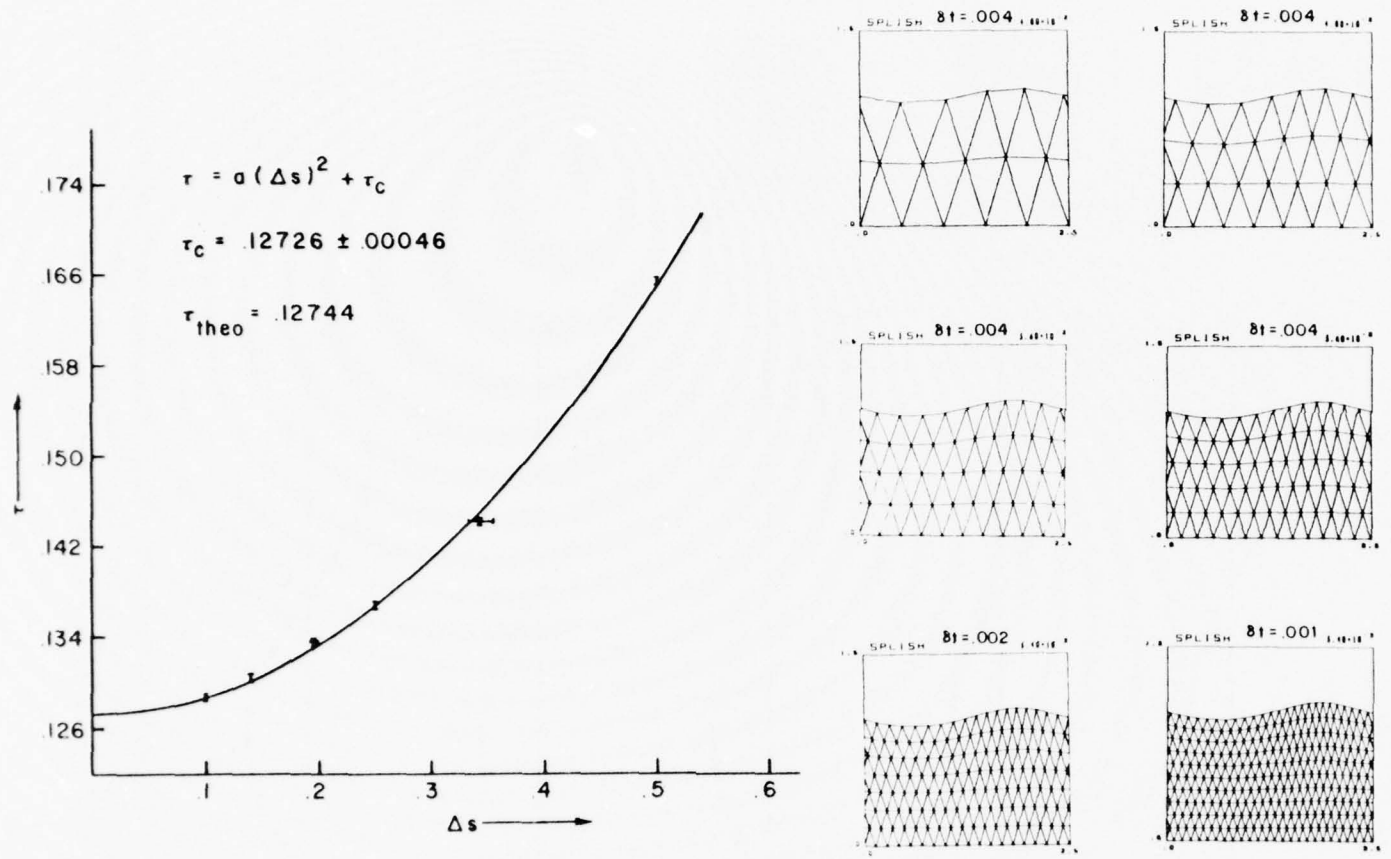


Fig. 7 — Richardson extrapolation of the wave period as a function of the grid spacing. Inserts illustrate the variation in mesh sizes used.

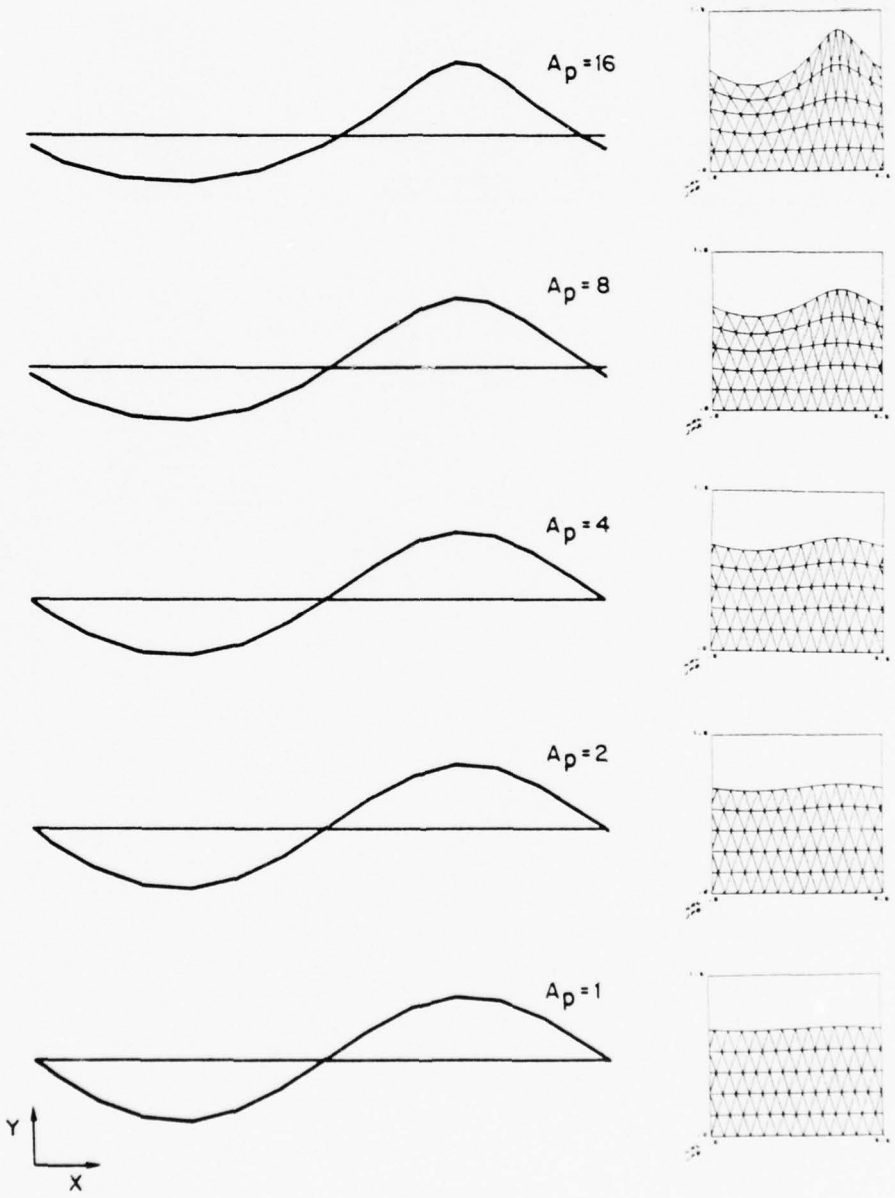


Fig. 8 — Transition of the wave profile from the linear to the nonlinear regime. The wave profiles are normalized by the amplitudes of the initial pressure perturbations $P = A_p \sin kx$. Note the compressed scale in the x-direction for the inserts.

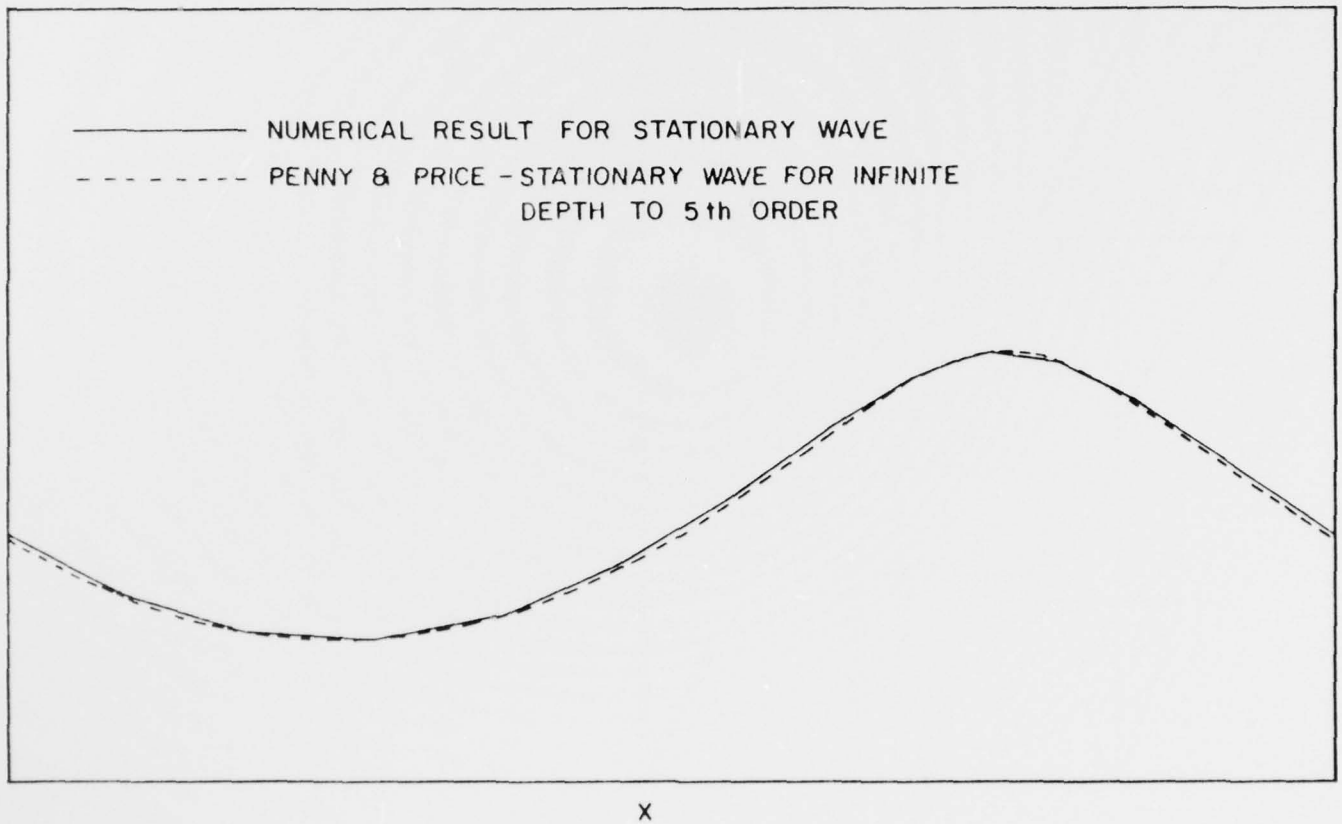
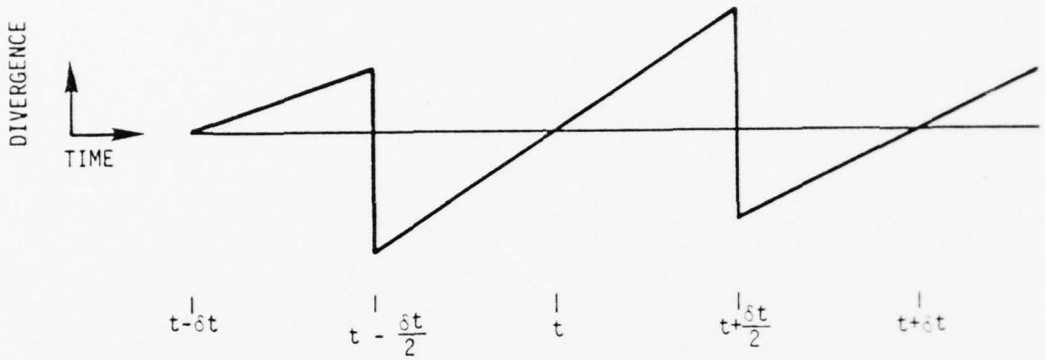
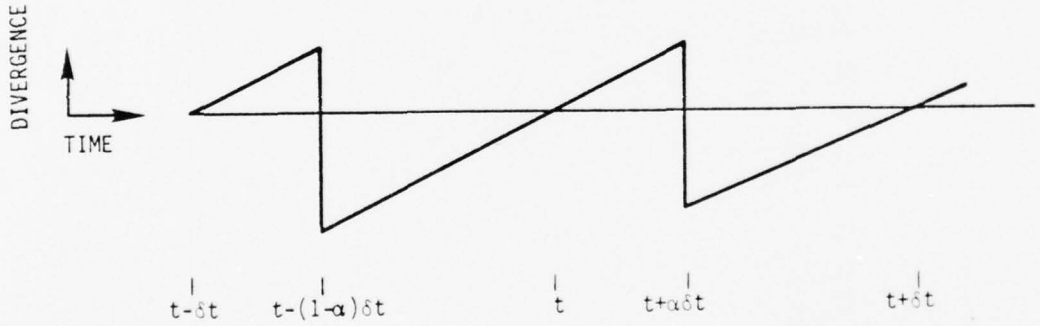


Fig. 9 — Comparison of the numerical results with the fifth-order theoretical prediction in the limit of infinite depth



A) CENTERED TIME SPLITTING



B) FORWARD TIME SPLITTING

Fig. 10 — The time development of the divergence for two differencing schemes

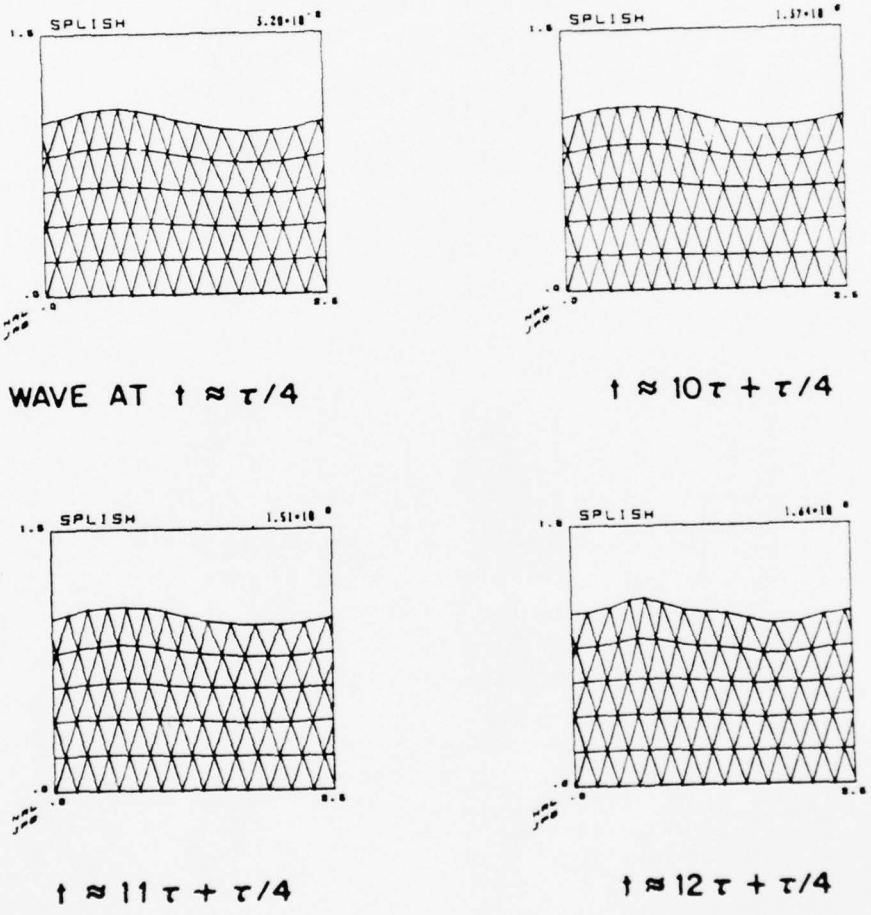


Fig. 11 — An example of the growth of numerical instability

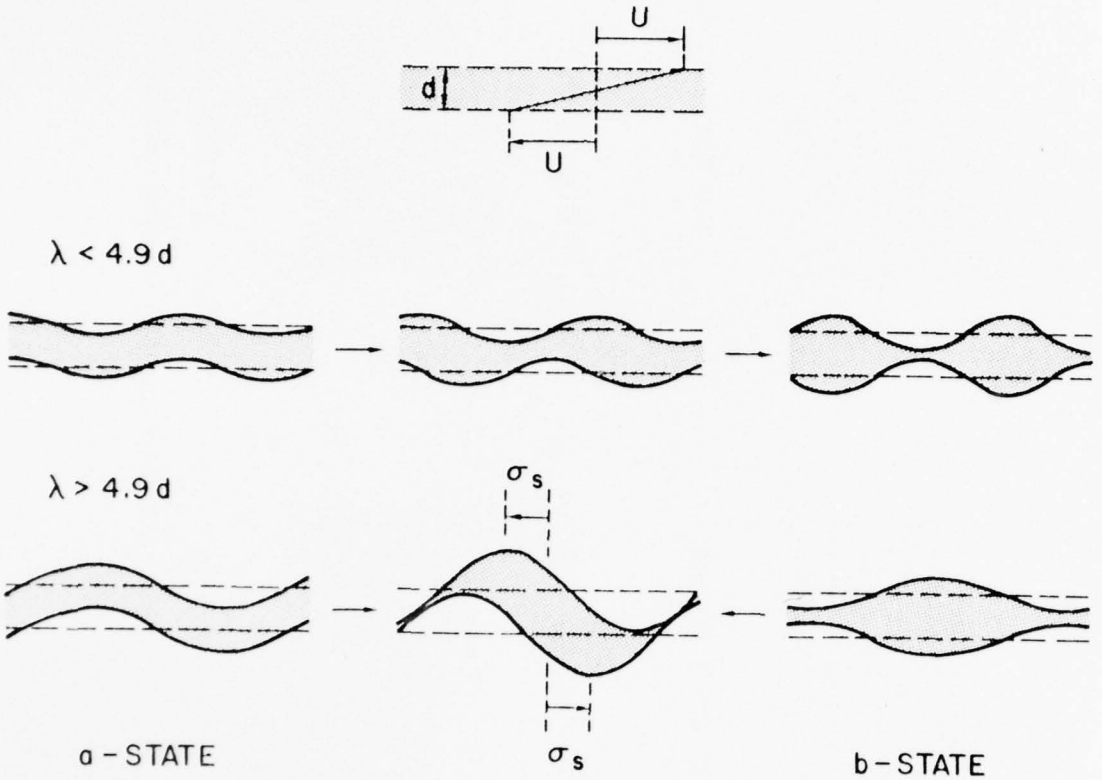


Fig. 12 — The development of a symmetric wave in a homogeneous shear layer

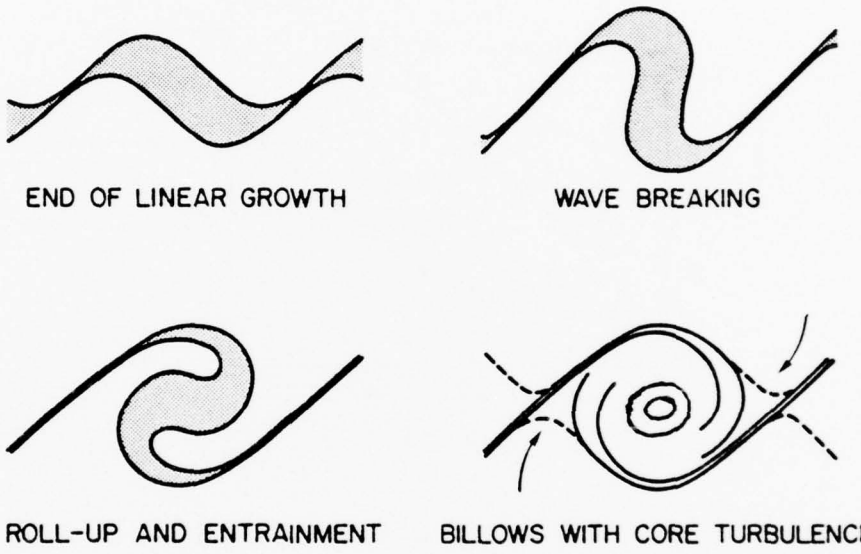


Fig. 13 — The formation of a Kelvin-Helmholtz billow from a perturbed shear layer. In the homogeneous case the vorticity in the shaded area remains constant throughout the different stages of growth. For stably stratified layers, vorticity is generated and depleted baroclinically along the deformed layer according to its tilt.

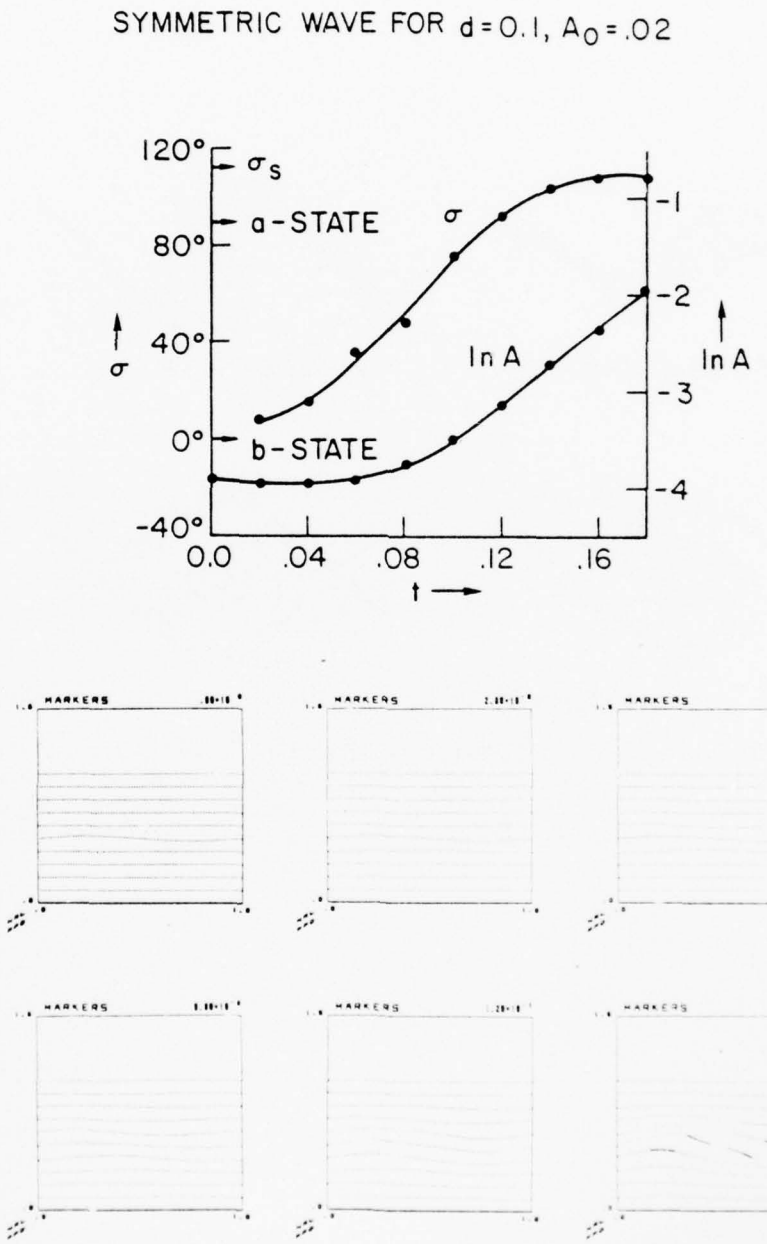


Fig. 14 — Growth rate and phase of the stationary state for d = 0.1 and A_0 = 0.02. The mesh is 11 × 11 and the timestep δt = 0.001 sec, or the transit time between vertices is 20 timesteps in the irrotational layers.

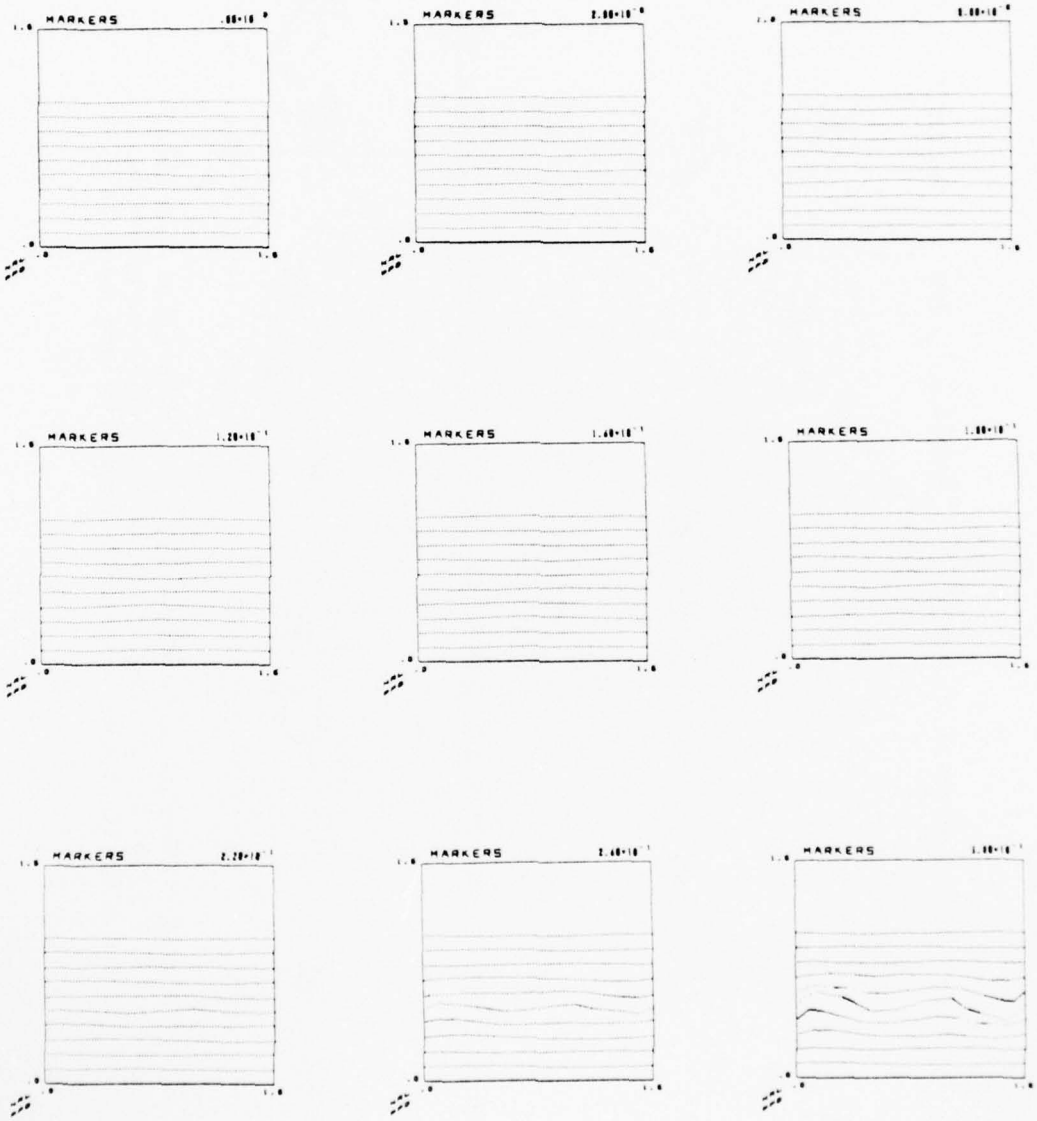


Fig. 15 — Growth of a favored harmonic of the initial perturbation. Here $\hat{a} = 0.1$, $A_o = 0.01$, $\delta t = 0.001$, and the mesh is 11×11. The initial perturbation of wavelength $\lambda = 1.5$ has a theoretically predicted growth rate of $n = 15.4$. Its first harmonic has a more favorable growth rate, $n = 20.1$.

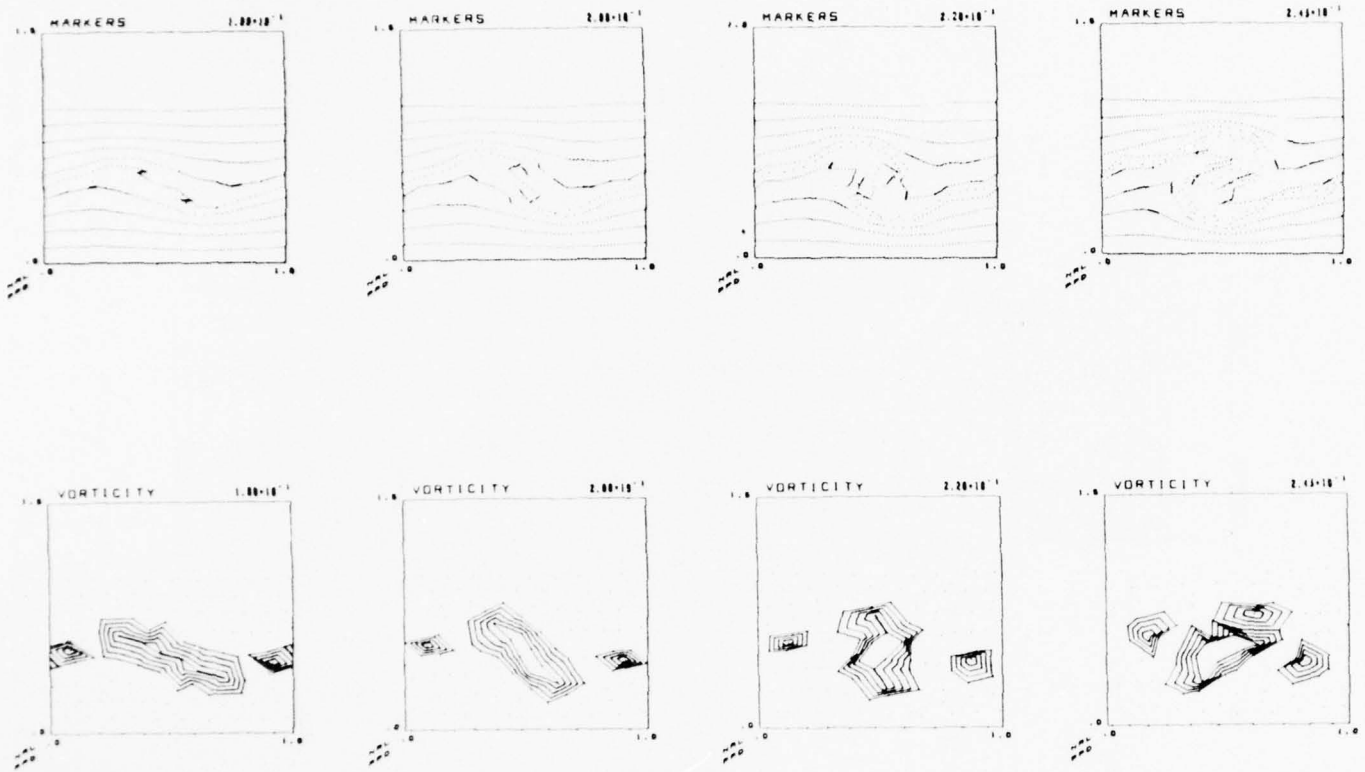


Fig. 16 — Billow growth for the d = 0.1, A_o = 0.02 simulation. Mesh size is 11 × 11 and δt = 0.001 sec. Marker particles continue the growth illustrated in Fig. 14. Vorticity contours are shown at the same times.

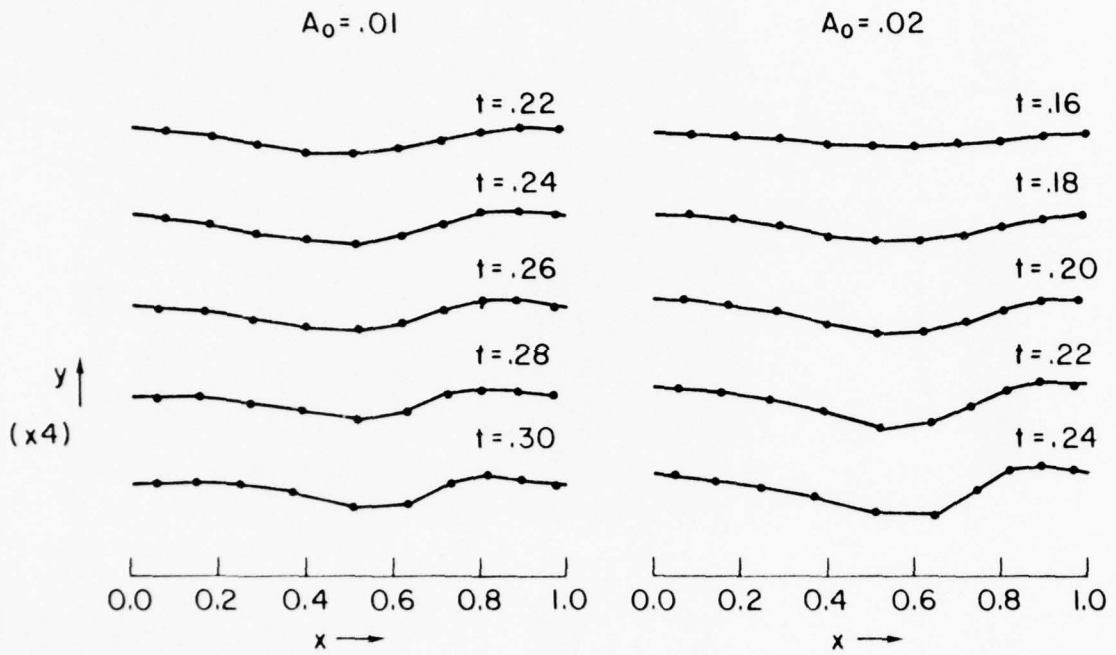


Fig. 17 — Evolution of the free surface as a function of perturbation amplitude. The free surface develops more strongly when driven by the faster developing billow with $A_0 = 0.02$.

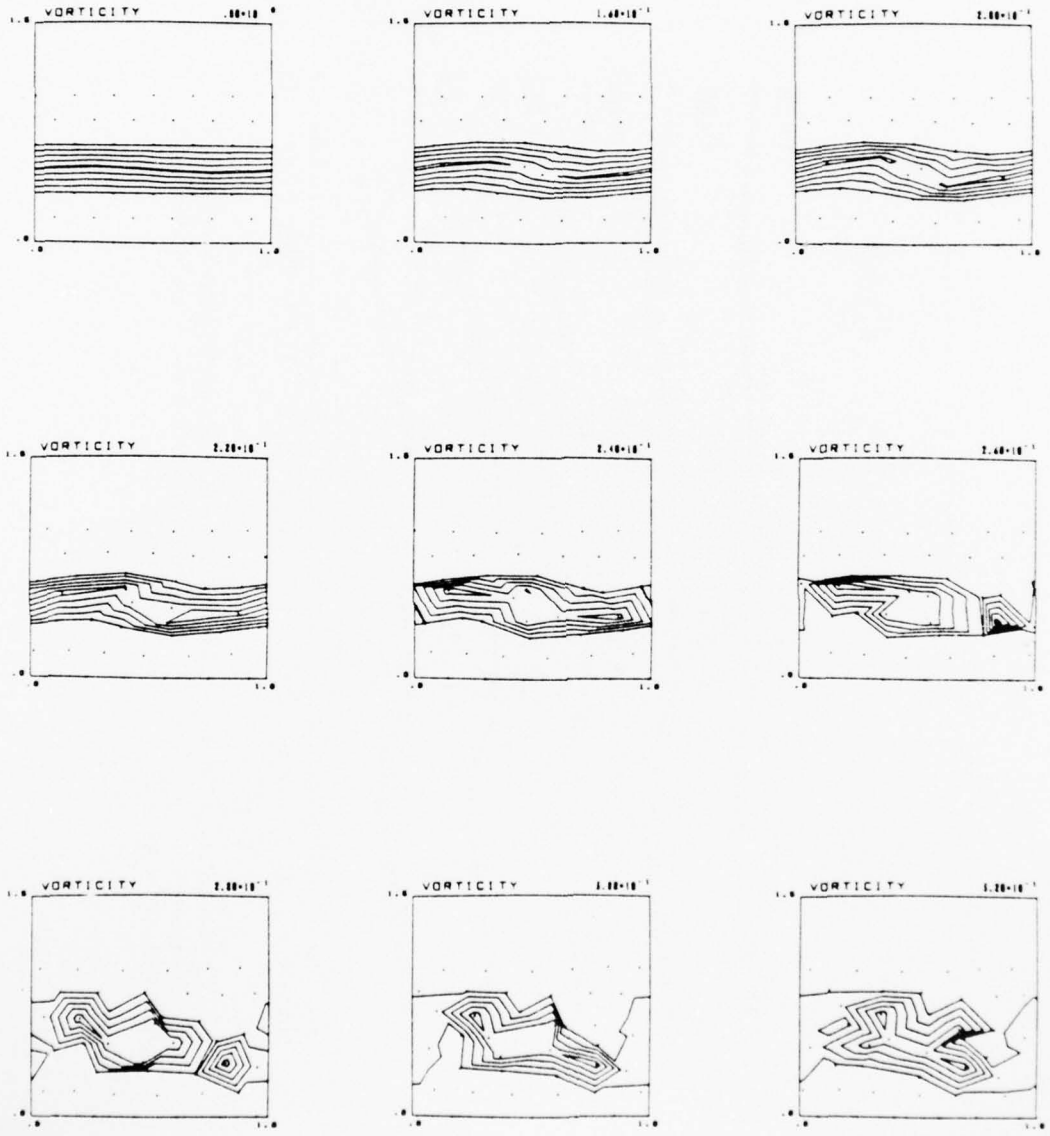


Fig. 18 — Vorticity contours for a stratified layer. A stratification of $\Delta\rho$ = 0.05 is added across the shear layer of depth d = 0.167. The initial perturbation A_o = 0.02, δt = 0.004 sec and the mesh is 7 × 7.

LAYER DEVELOPMENT FOR $d = 1/6$, $A_0 = .02$

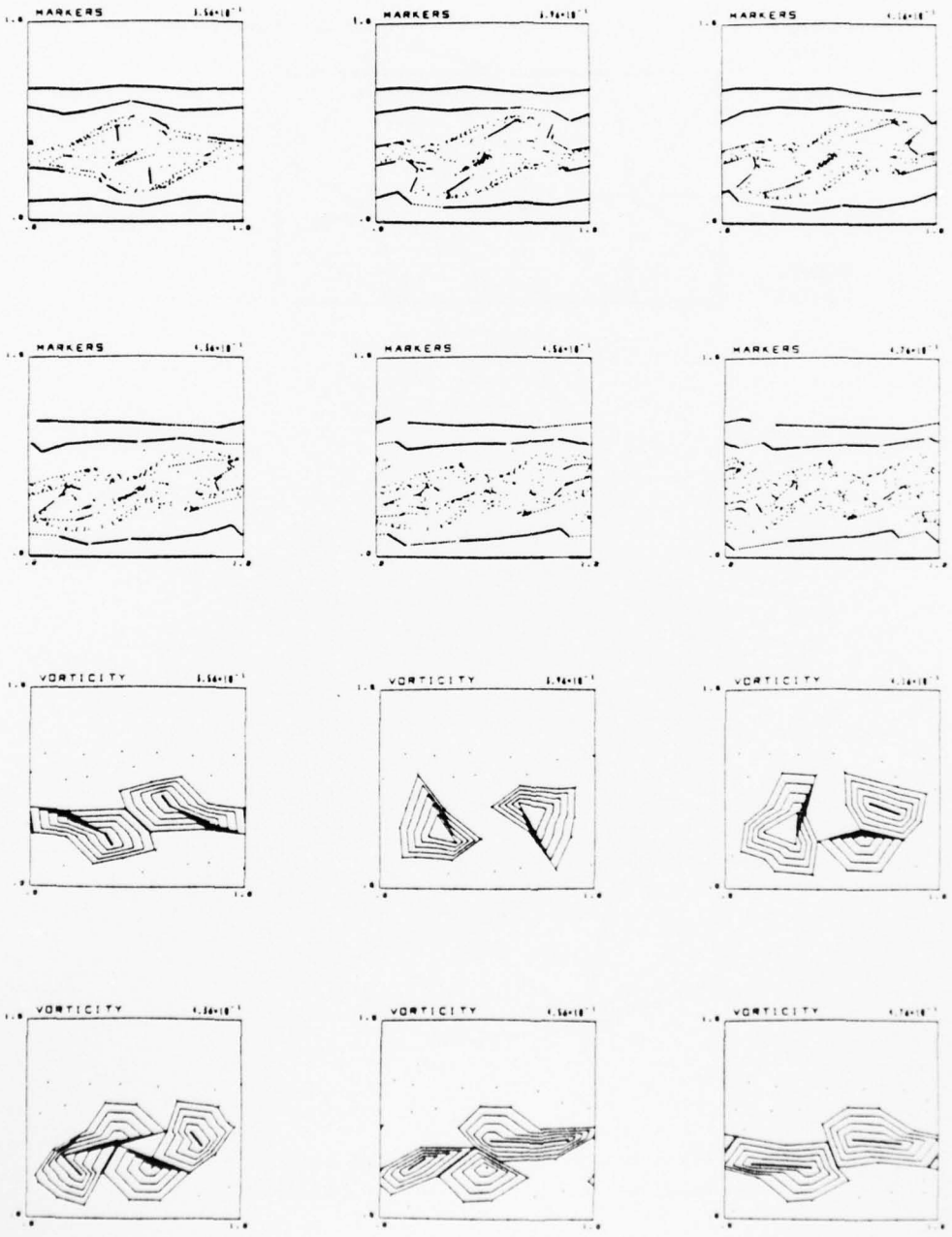
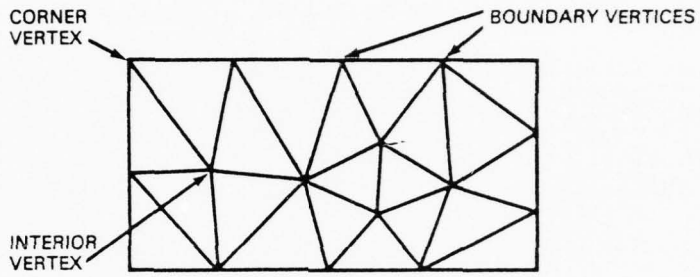
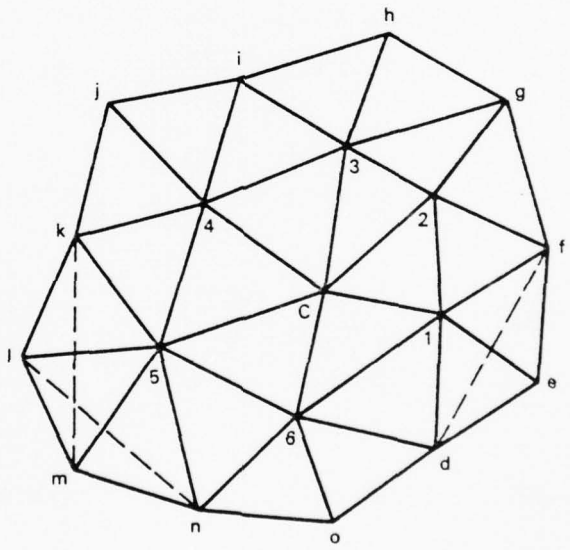


Fig. 19 — Late development of the shear layer. For this calculation d = 0.167, A_0 = 0.02, with δt = 0.004 sec for a 7 × 7 mesh. Gaps in the marker particles, particularly at the free surface and rigid bottom, are artificial and are due to round-off in the low-precision marker routines.



(a)



(b)

Fig. 20 — (a) A rectangular computational region tesselated by triangles. (b) Section of a triangular mesh.

DISTRIBUTION LIST

Defense Documentation Center
Cameron Station
Alexandria, VA 22314 12 copies

Technical Library
David W. Taylor Naval Ship Research
and Development Center
Annapolis Laboratory
Annapolis, MD 21402

Professor Bruce Johnson
Engineering Department
Naval Academy
Annapolis MD 21402

Library
Naval Academy
Annapolis, MD 21402

Professor T. Francis Ogilvie
Department of Naval Architecture
and Marine Engineering
University of Michigan
Ann Arbor, MI 48105

Professor C. S. Yih
Department of Engineering Mechanics
University of Michigan
Ann Arbor, MI 48105

Dr. Code Pan
Shaker Research Corporation
Northway 10 Executive Park
Ballston Lake, NY 12019

NASA Scientific and Technical Information
Facility
P. O. Box 8757
Baltimore/Washington International Airport
Maryland 21240

Librarian
Department of Naval Architecture
University of California
Berkeley, CA 94720

Professor P. Lieber
Department of Mechanical Engineering
University of California
Berkeley, CA 94720

Professor P. Naghdi
College of Mechanical Engineering
University of California
Berkeley, CA 94720

Professor W. C. Webster
Department of Naval Architecture
University of California
Berkeley, CA 94720

Professor J. V. Wehausen
Department of Naval Architecture
University of California
Berkeley, CA 94720

Director
Office of Naval Research Branch Office
495 Summer Street
Boston, MA 02210

Commander
Puget Sound Naval Shipyard
Bremerton, WA 98314

Dr. Alfred Ritter
CALSPAN Corporation
P. O. Box 235
Buffalo, NY 14221

Professor G. Birkhoff
Department of Mathematics
Harvard University
Cambridge, MA 02138

Professor G. F. Carrier
Division of Engineering and
Applied Physics
Pierce Hall
Harvard University
Cambridge, MA 02138

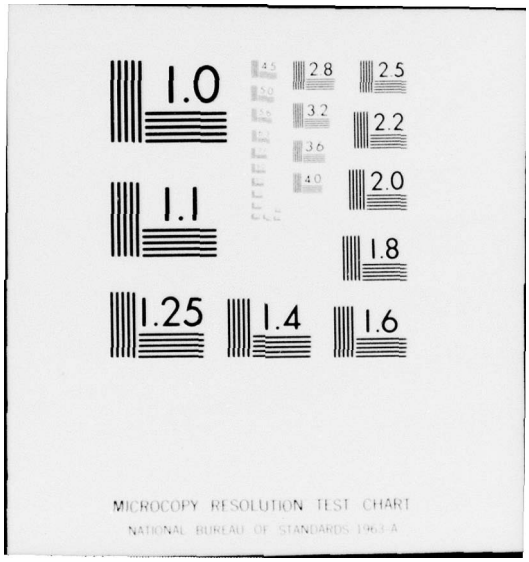
AD-A037 717 NAVAL RESEARCH LAB WASHINGTON D C F/G 20/4
SOLUTION OF TRANSIENT PROBLEMS IN FREE SURFACE HYDRODYNAMICS.(U)
FEB 77 J P BORIS, M J FRITTS

UNCLASSIFIED NRL-MR-3446 NL

2 OF 2
ADA037717

END

DATE
FILMED
4-77


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

Professor M. A. Abkowitz
Department of Ocean Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Commanding Officer
NROTC Naval Administrative Unit
Massachusetts Institute of Technology
Cambridge, MA 02139

Professor L. N. Howard
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, MA 02139

Professor Phillip Mandel
Department of Ocean Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Professor C. C. Mei
Department of Civil Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Professor E. Mollo-Christensen
Department of Meteorology
Room 54-1722
Massachusetts Institute of Technology
Cambridge, MA 02139

Professor J. Nicholas Newman
Department of Ocean Engineering
Room 5-324A
Massachusetts Institute of Technology
Cambridge, MA 02139

Professor R. F. Probstein
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Library
C. S. Draper Laboratory
68 Albany Street
Cambridge, MA 02139

Dr. S. Orszag
Flow Research, Inc.
1 Broadway
Cambridge, MA 02142

Dr. Edgar J. Gunter, Jr.
University of Virginia
School of Engineering and Applied
Science
Charlottesville, VA 22903

Director
Office of Naval Research Branch Office
536 South Clark Street
Chicago, ILL. 60605

Library
Naval Weapons Center
China Lake, CA 93555

Professor E. Reshotko
Division of Chemical Engineering Science
Case Western Reserve University
Cleveland, OH 44106

Commander
Charleston Naval Shipyard
Naval Base
Charleston, SC 29408

Professor J. M. Burgers
Institute of Fluid Dynamics and
Applied Mathematics
University of Maryland
College Park, MD 20742

Professor Pai
Institute for Fluid Dynamics and
Applied Mathematics
University of Maryland
College Park, MD 20740

Technical Library
Naval Weapons Surface Center
Dahlgren Laboratory
Dahlgren, VA 22418

Computation and Analyses Laboratory
Naval Weapons Surface Center
Dahlgren Laboratory
Dahlgren, VA 22418

Dr. R. Chan
JAYCOR
1401 Camino Del Mar
Del Mar, CA 92014

Dr. J. A. Young
JAYCOR
1401 Camino Del Mar
Del Mar, CA 92014

Dr. R. H. Kraichnan
Dublin, NH 03444

Dr. Martin H. Bloom
Director of Gas Dynamics Research
Polytechnic Institute of New York
Long Island Center
Farmingdale, NY 11735

Research and Technology Division
Army Engineering Reactors Group
Fort Belvoir, VA 22060

Technical Documents Center
Building 315
Army Mobility Equipment Research Center
Fort Belvoir, VA 22060

Technical Library
Webb Institute of Naval Architecture
Glen Cove, NY 11542

Professor E. V. Lewis
Webb Institute of Naval Architecture
Glen Cove, NY 11542

Dr. M. Poreh
Technion-Israel Institute of Technology
Department of Civil Engineering
Haifa, Israel

Dr. J. P. Breslin
Davidson Laboratory
Stevens Institute of Technology
Castle Point Station
Hoboken, NJ 07030

Dr. J. P. Craven
University of Hawaii
1801 University Avenue
Honolulu, HI 96822

Professor F. Hussain
Department of Mechanical Engineering
Cullen College of Engineering
University of Houston
Houston, TX 77004

Professor J. F. Kennedy, Director
Institute of Hydraulic Research
University of Iowa
Iowa City, IA 52242

Professor L. Landweber
Institute of Hydraulic Research
University of Iowa
Iowa City, IA 52242

Professor E. L. Resler
Graduate School of Aerospace Engineering
Cornell University
Ithaca, NY 14851

Dr. D. R. S. Ko
Flow Research Inc.
1819 S. Central Avenue
Kent, WA 98031

Professor V. W. Goldschmidt
School of Mechanical Engineering
Purdue University
Lafayette, IN 47907

Professor J. W. Miles
Institute of Geophysics and Planetary
Physics, A-025
University of California, San Diego
La Jolla, CA 92093

Flow Research, Inc.
Los Angeles Division
9841 Airport Blvd., Suite 1004
Los Angeles, CA 90045

Director
Scripps Institute of Oceanography
University of California
La Jolla, CA 92037

Mr. Virgil Johnson, President
Hydronautics, Incorporated
7210 Pindell School Road
Laurel, MD 20810

Mr. M. P. Tulin
Hydronautics, Incorporated
7210 Pindell School Road
Laurel, MD 20810

Commander
Long Beach Naval Shipyard
Long Beach, CA 90801

Dr. C. W. Hirt
University of California
Los Alamos Scientific Laboratory
P. O. Box 1663
Los Alamos, NM 87544

Professor John Laufer
Department of Aerospace Engineering
University of Southern California
University Park
Los Angeles, CA 90007

Lorenz G. Straub Library
St. Anthony Falls Hydraulic Laboratory
University of Minnesota
Minneapolis, MN 55414

Dr. E. Silberman
St. Anthony Falls Hydraulic Laboratory
University of Minnesota
Minneapolis, MN 55414

Library
Naval Postgraduate School
Monterey, CA 93940

Professor J. Wu
College of Marine Studies
University of Delaware
Newark, DE 19711

Technical Library
Naval Underwater Systems Center
Newport, RI 02840

Office of Naval Research
New York Area Office
715 Broadway - Fifth Floor
New York, NY 10003

Professor J. J. Stoker
Courant Institute of Mathematical
Sciences
New York University
251 Mercer Street
New York, NY 10003

Professor V. Castelli
Department of Mechanical Engineering
Columbia University
New York, NY 10027

Professor H. G. Elrod
Department of Mechanical Engineering
Columbia University
New York, NY 10027

Engineering Societies Library
345 East 47th Street
New York, NY 10017

Society of Naval Architects and
Marine Engineers
74 Trinity Place
New York, NY 10006

Librarian, Aeronautical Laboratory
National Research Council
Montreal Road
Ottawa 7, Canada

Technical Library
Naval Coastal System Laboratory
Panama City, FL 32401

Professor H. W. Liepmann
Graduate Aeronautical Laboratories
California Institute of Technology
Pasadena, CA 91109

Professor M. S. Plesset
Engineering Science Department
California Institute of Technology
Pasadena, CA 91109

Professor A. Roshko
Graduate Aeronautical Laboratories
California Institute of Technology
Pasadena, CA 91109

Professor T. Y. Wu
Engineering Science Department
California Institute of Technology
Pasadena, CA 91109

Director
Office of Naval Research Branch Office
1030 E. Green Street
Pasadena, CA 91101

Mr. R. Wade
Tetra Tech., Inc.
Marine and Environmental Engineering Division
630 N. Rosemead Blvd.
Pasadena, CA 91107

Professor K. M. Agrawal
Virginia State College
Department of Mathematics
Petersburg, VA 23802

Technical Library
Naval Ship Engineering Center
Philadelphia Division
Philadelphia, PA 19112

Technical Library
Philadelphia Naval Shipyard
Philadelphia, PA 19112

Professor R. C. MacCamy
Department of Mathematics
Carnegie Institute of Technology
Pittsburgh, PA 15213

Dr. Paul Kaplan
Oceanics, Inc.
Technical Industrial Park
Plainview, NY 11803

Technical Library
Naval Missile Center
Point Mugu, CA 93041

Commander
Portsmouth Naval Shipyard
Portsmouth, NH 03801

Commander
Norfolk Naval Shipyard
Portsmouth, VA 23709

Mr. C. duP Donaldson Associates
Aeronautical Research of Princeton, Inc.
50 Washington Road
Princeton, N. J. 08540

Professor F. Hama
Department of Aerospace and
Mechanical Science
Princeton University
Princeton, NJ 08540

Dr. J. Clarke
Division of Engineering
Brown University
Providence, RI 02912

Professor J. Liu
Division of Engineering
Brown University
Providence, RI 02912

Dr. E. Baum
TRW Systems Group
One Space Park
Redondo Beach, CA 90278

Chief, Document Section
Redstone Scientific Information Center
Army Missile Command
Redstone Arsenal, AL 35809

Dr. A. Eshel
Ampex Corporation
401 Broadway
Redwood City, CA 94063

Army Research Office
P. O. Box 12211
Research Triangle Park, NC 27709

Dr. H. Norman Abramson
Southwest Research Institute
8500 Culebra Road
San Antonio, TX 78228

ONR Scientific Liaison Group
American Embassy - Room A-407
APO San Francisco 96503

Editor
Applied Mechanics Review
Southwest Research Institute
8500 Culebra Road
San Antonio, TX 78228

Dr. J. W. Hoyt
Code 2501
Naval Undersea Center
San Diego, CA 92132

Technical Library
Naval Undersea Center
San Diego, CA 92132

Dr. Andrew Fabula
Code 4007
Naval Undersea Center
San Diego, CA 92132

Office of Naval Research
San Francisco Area Office
760 Market Street, Room 447
San Francisco, CA 94102

Library
Peral Harbor Naval Shipyard
Box 400
FPO San Francisco 96610

Technical Library
Hunters Point Naval Shipyard
San Francisco, CA 94135

Professor A. Hertzberg
Director, Aerospace Research Laboratory
University of Washington
Seattle, WA 98105

Professor C. E. Pearson
Aerospace Research Laboratory
University of Washington
Seattle, WA 98105

Dr. Steven Crow, President
Poseidon Research
11777 San Vicente Blvd., Suite 641
Los Angeles, CA 90049

Mr. J. Enig (Room 3-252)
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, MD 20910

Librarian
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, MD 20910

Mr. J. Rogers
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, MD 20910

Fenton Kennedy Document Library
The Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Rd.
Laurel, Md. 20180

Professor Milton van Dyke
Department of Aeronautical Engineering
Stanford University
Stanford, CA 94305

Professor J. Thompson
Department of Aerophysics and Aerospace
Engineering
Mississippi State University
State College, MS 39762

Professor R. DiPrima
Department of Mathematics
Rensselaer Polytechnic Institute
Troy, NY 12181

Dr. L. A. Segel
Department of Mathematics
Rensselaer Polytechnic Institute
Troy, NY 12181

Professor J. L. Lumley
Department of Aerospace Engineering
Pennsylvania State University
University Park, PA 16802

Dr. J. M. Robertson
Department of Theoretical and Applied
Mechanics
University of Illinois
Urbana, IL 61803

Technical Library
Mare Island Naval Shipyard
Vallenc, CA 94592

Office of Naval Research
Code 438
800 N. Quincy Street
Arlington, VA 22217 3 copies

Office of Naval Research
Code 200
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 210
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 211
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 212
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 221
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 473
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 480
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 481
800 N. Quincy Street
Arlington, VA 22217

Office of Naval Research
Code 1021P (ONRL)
800 N. Quincy Street
Arlington, VA 22217 6 copies

Naval Research Laboratory
Washington, D. C. 20375

Attn: Code 2627 - 6 copies
Code 4000
Code 7000 - 25 copies
Code 7750 - 75 copies
Code 8841 (R. J. Hansen)

Mr. L. Benen (Code 0322)
Naval Sea Systems Command
Washington, DC 20362

Mr. J. Schuler (Code 032)
Naval Sea Systems Command
Washington, DC 20362

Code 03B
Naval Sea Systems Command
Washington, DC 20362

Mr. T. Peirce (Code 03512)
Naval Sea Systems Command
Washington, DC 20362

Library (Code 09GS)
Naval Sea Systems Command
Washington, DC 20362

Code 6034
Naval Ship Engineering Center
Center Building
Prince Georges Center
Hyattsville, MD 20782

Code 6101E
Naval Ship Engineering Center
Center Building
Prince Georges Center
Hyattsville, MD 20782

Code 6110
Naval Ship Engineering Center
Center Building
Prince Georges Center
Hyattsville, MD 20782

Code 6114
Naval Ship Engineering Center
Center Building
Prince Georges Center
Hyattsville, MD 20782

Code 6136
Naval Ship Engineering Center
Center Building
Prince Georges Center
Hyattsville MD 20782

Code 6140
Naval Ship Enginnering Center
Center Building
Prince Georges Center
Hyattsville, MD 20782

Dr. A. Powell (Code 01)
David W. Taylor Naval Ship
Research and Development Center
Bethesda, MD 20084

Dr. W. E. Cummins (Code 15)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Mr. G. H. Gleissner (Code 18)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Dr. P. C. Pien (Code 1520)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Mr. Paul S. Granville (Code 1541)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Mr. J. H. McCarthy, Jr. (Code 1552)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Dr. Nils Salvesen (Code 1552)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Mrs. Joanna Schot (Code 1843)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Library (Code 5641)
David W. Taylor Naval Ship Research
and Development Center
Bethesda, MD 20084

Code 03
Naval Air Systems Command
Washington, DC 20361

Code 03B
Naval Air Systems Command
Washington, DC 20361

Code 310
Naval Air Systems Command
Washington, DC 20361

Code 5301
Naval Air Systems Command
Washington, DC 20361

Strategic Systems Projects Office
Department of the Navy
Washington, DC 20376

Mr. Norman Nilsen (SP 2022)
Strategic Systems Projects Office
Department of the Navy
Washington, DC 20376

Oceanographer of the Navy
200 Stovall Street
Alexandria, VA 22332

Commander
Naval Oceanographic Office
Washington, DC 20373

Dr. A. L. Slafkosky
Scientific Advisor
Commandant of the Marine Corps (Code AX)
Washington, DC 20380

Librarian Station 5-2
Coast Guard Headquarters
NASSIF Building
400 Seventh Street, SW
Washington, DC 20591

Office of Research and Development
Maritime Administration
411 G Street, NW
Washington, DC 20235

Division of Ship Design
Maritime Administration
441 G Street, NW
Washington, DC 20235

Air Force Office of Scientific
Research/NA
Building 410
Bolling AFB
Washington, DC 20332

AFDRD-AS/M
U. S. Air Force
The Pentagon
Washington, DC 20330

Chief of Research and Development
Office of Chief of Staff
Department of the Army
Washington, DC 20310

Dr. G. Kulin
Fluid Mechanics Section
National Bureau of Standards
Washington, DC 20234

National Science Foundation
Engineering Division
1800 G Street, NW
Washington, DC 20550

Science and Technology Division
Library of Congress
Washington, DC 20540

Defense Research and Development Attache
Australian Embassy
1601 Massachusetts Avenue, NW
Washington, DC 20036

Dr. A. S. Iberall, President
General Technical Services, Inc.
451 Penn Street
Yeadon, PA 19050

Dr. Ricardo A. Bastianon
Instituto de Tecnologia Naval
E.E.V.V. 25 - Capital Federal
Buenos Aires, Argentina

Professor James C. Wu
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Professor A. J. Chorin
Department of Mathematics
University of California
Berkeley, CA 94720

Dr. S. A. Berger
Department of Mechanical Engineering
University of California
Berkeley, CA 94720

Dr. G. R. Inger
Department of Aerospace Engineering
Virginia Polytechnic Institute
Blacksburg, VA 24061

Professor A. H. Nayfeh
Department of Engineering Mechanics
Virginia Polytechnic Institute
Blacksburg, VA 24061

Dr. John D. Anderson, Jr.
Chairman, Department of Aerospace
Engineering
College of Engineering
University of Maryland
College Park, MD 20742

Professor R. T. Davis
Department of Aerospace Engineering
and Applied Mechanics
University of Cincinnati
Cincinnati, OH 45221

Professor O. Burggraf
Department of Aeronautical and
Astronautical Engineering
Chio State University
Columbus, OH 43220

Dr. W. R. Briley
United Aircraft Corporation Research
Laboratory
East Hartford, CT 06108

Dr. S. Nadir
Northrop Corporation
Aircraft Division
3901 West Broadway
Hawthorne, CA 90250

Technical Library
Naval Ordnance Station
Indian Head, MD 20640

Professor S. F. Shen
Graduate School of Aerospace
Engineering
Cornell University
Ithaca, NY 14850

Mr. John L. Hess
Douglas Aircraft Company
3855 Lakewood Boulevard
Long Beach, CA 90801

Dr. H. K. Cheng
Department of Aerospace Engineering
University of Southern California
University Park
Los Angeles, CA 90007

Dr. J. Trulio
Applied Theory, Inc.
1010 Westwood Blvd.
Los Angeles, CA 90024

Dr. T. D. Taylor
The Aerospace Corporation
Post Office Box 95085
Los Angeles, CA 90045

Dr. J. R. Spreiter
Nielsen Engineering and Research, Inc.
850 Maude Avenue
Mountain View, CA 94040

Professor S. I. Cheng
Gas Dynamics Laboratory
Forrestal Campus
Princeton University
Princeton, NJ 08540

Dr. H. Yoshihara
Mail Zone 630-00
General Dynamics-CONVAIR
P. O. Box 1128
San Diego, CA 92112

Dr. R. J. Hakkinen
Department 222
McDonnell Douglas Corporation
P. O. Box 516
St. Louis, MO 63166

Mr. R. Siewert (AIR 320D)
Naval Air Systems Command
Washington, D. C. 20361

Dr. Harvey R. Chaplin
Code 16
Naval Ship Research and Development Center
Bethesda, MD 20034

Mr. Robert Moore
Deputy Director, Tactical
Technology Office
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

A. B. Langdon
Lawrence Livermore Laboratory
Livermore, California 94550

W. P. Crowley
Lawrence Livermore Laboratory
Livermore, Calif. 94550

Dr. Robert Voight
ICASE
Mail Stop 132C
NASA Langley Research Center
Hampton, VA 23665

Dr. James Ortega
ICASE
Mail Stop 132C
NASA Langley Research Center
Hampton, VA 23665

Dr. William Buzbee
P. O. Box 1663
Los Alamos Scientific Laboratory
Los Alamos, NM 87344

Dr. James Welch
Princeton Geophysical Laboratory
Princeton, NJ 08540

Dr. Francis J. Balint
Naval Oceanic and Atmospheric Administration
Bldg. 5 - AD3
6010 Executive Blvd.
Rockville, MD 20852

A. H. Makomasi
Mechanical Engineering Division
National Research Council of Canada
Ottawa, Canada
KIA OR6

S. A. Thorpe
National Institute of Oceanography
Wormley, Godalming, Surrey
England

P. C. Patnaik
Science Applications, Inc.
1205 Prospect Street
La Jolla, CA 92037

F. S. Sherman
Department of Mechanical Engineering
University of California
Berkeley, CA 94720

Fred Fajen
Mission Research Corporation
P. O. Drawer 719
Santa Barbara, Calif. 93102