INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 3, 399–424 (1983)

FREE SURFACE WAVE MOTIONS AND INTERACTIONS

E. W. MINER

Fluid Dynamics Branch, Naval Research Laboratory, Washington, D.C. 20375, U.S.A.

M. J. FRITTS

Laboratory for Computational Physics, Naval Research Laboratory, Washington, D.C. 20375, U.S.A.

O. M. GRIFFIN AND S. E. RAMBERG

Fluid Dynamics Branch, Naval Research Laboratory, Washington, D.C. 20375, U.S.A.

SUMMARY

This paper investigates the problems involved in the numerical simulation of free surface wave motions and surface wave effects on marine structures. Various approaches that might be taken in meeting these problems are discussed, and their relative advantages and disadvantages are considered. One particular approach combines a Lagrangian formulation of the governing equations, a triangular grid and a finite-difference solution procedure. Since this approach has some distinct advantages in the numerical calculation of fluid flows which include a free surface, it formed the basis for the development of one particular computer code, SPLISH. Sufficient progress has been made with the SPLISH code to demonstrate the attractiveness of numerical calculations for wave flow problems. Recent computational results demonstrate that realistic time-varying local flow fields, pressures and forces on and near structures such as a half-cyclinder on the ocean floor can be determined from numerical calculations for certain conditions. Good agreement is found in comparison of the numerical results from SPLISH, recent linear wave Green's function and fifth-order asymptotic solutions for wave motion over a bottom seated half-cylinder, and an experimental simulation in a wave channel.

KEY WORDS Free Surface Waves Wave Motions Fluid–Structure Interactions Finite Difference Methods Lagrangian Formulation Triangular Grids Numerical Methods

INTRODUCTION

The effects of surface waves on structures in the ocean are of considerable practical interest, but until recently the complexity of this flow–structure interaction has limited most theoretical studies to idealized flows and structural geometries. A primary contribution to the complexity of the wave–structure problem is, of course, the presence of the free surface. Also, the structure may be submerged in water of intermediate or shallow depth, so that the ocean bottom effect is important and must be included in the solution of the problem. Analytical and numerical solutions, limited primarily to small-amplitude linear waves, have been obtained in two and three dimensions for a number of problems relating to wave flows over cylindrical forms of various cross-sections and orientations. A review of these solutions is given by Naftzger and Chakrabarti.[1]

A research programme has been under way at the Naval Research Laboratory to develop numerical methods for predicting the effects of surface waves on submerged and partially submerged structures in the ocean in the hope of eventually removing the linearizing assumptions. One phase of this research has been concerned with computations using a

0271–2091/83/040399–26$02.60 *Received 28 May 1982*

Lagrangian finite-difference code called SPLISH which was developed at NRL.[2,3] Other continuing efforts at NRL are devoted to applications of finite element and boundary element methods to the problem of wave flow over obstacles.

The first section of this paper discusses the appropriate choice of a numerical method to simulate free surface flow problems. An overview is given of the numerical techniques available for the solution of this class of problems. The advantages and limitations of these techniques are discussed and compared in terms of the physical problem to be studied and the accuracy and efficiency required of the final computer code. This is followed by a more detailed presentation of the numerical algorithms that are implemented in the computer code SPLISH.

The next section presents numerical results which have been obtained for several benchmark cases of surface wave motion over a bottom seated half-cylinder.[3,4] These results are compared with both linear and fifth-order potential flow solutions to the problem. All of these results are then compared with experimental measurements of the wave pressure field about the half-cylinder which were obtained in a wave channel at NRL. The agreement is good between the numerical, potential flow and experimental results for conditions of low wave reflection from the cylinder. Although the numerical method presently is applicable only to a limited range of wave parameters, water depths, and structural dimensions, the results achieved thus far do show the flexibility and power of the Lagrangian formulation. The final section of this paper summarizes the results achieved thus far, the problems encountered, and the outlook for future progress.

THE NUMERICAL METHOD

Numerical methods

The accuracy of any numerical method chosen to study wave–structure interactions will be limited primarily by that method's accuracy in formulating boundary conditions. This is due to the presence of the free surface which dominates the evolution of the fluid flow, particularly for wave-breaking, surging or slamming. It is also due to the possible movement of the submerged or partially submerged structure in response to the free surface motion, an interaction which can both generate free surface waves and absorb energy from them.

The free surface boundary condition is quite complex, since it is non-linear in the near region and radiative in the far region. The waves generated by wave–structure interactions propagate both upstream and downstream, and these radiating waves must pass through the computational boundaries without affecting the transient, non-linear behaviour of the free surface in the interior of the region.

These difficulties limit the applicability of an Eulerian formulation for such complex flows. In a general Eulerian representation the position of a surface or interface is only known within one grid spacing. Fluid diffuses across the boundary and the interface cannot be correctly advanced in time. Mixed Eulerian–Lagrangian codes rectify this dilemma by tracking the motion of the surface in a Lagrangian fashion on the sub-grid scale. This generally entails special techniques to update additional Lagrangian vertices or markers which then flow with the interface. However, these specialized techniques are computationally inefficient for multiple surfaces, and generally fail in tracking the evolution of a surface to a multiply connected configuration.

The necessity of an accurate treatment of boundary motion makes a Lagrangian formulation of the fluid equations of motion particularly attractive. The strength of the Lagrangian

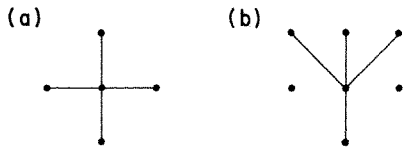

Figure 1(a). Grid connections for a regular mesh; (b). A distorted mesh formed by shear motion of the central column relative to its neighbours.

formulation resides in the fact that fluid elements are advected with the flow. Grid points which define surfaces remain at those surfaces and permit the maximum accuracy in formulation of boundary conditions. The variation of grid resolution which must occur in Eulerian schemes as surfaces pass through cells is also alleviated and the numerical diffusion across boundaries minimized. Since the grid moves with the fluid, the non-linear convective terms are not present in the Lagrangian formulation, resulting in higher accuracy and less stringent resolution requirements.

However, the strength of the Lagrangian approach is also its weakness. The advection of the mesh with the flow leads to large mesh deformations and a corresponding decrease in accuracy in both finite difference and finite element methods. To illustrate the effect of grid deformation let us examine first- or second-order accurate finite-difference methods. The mesh points commonly used to evaluate gradients and Laplacians are shown in Figure 1a for a regular grid. Figure 1b illustrates a simple grid distortion produced by a shear flow in which the center row of vertices has moved downward relative to the vertices to each side. A well-formulated Lagrangian finite-difference algorithm will properly account for the angles between grid lines and the variable mesh spacing produced by this distortion. Nevertheless, numerical approximations based on this mesh can still be grossly in error because differences no longer involve neighbouring vertices. Mesh points now closer to the central vertex do not enter in the approximation, while those further removed do.

As shown in Figure 2, higher order approximations may lead to even greater error. Figure 2(a) shows the vertices commonly used in higher-order approximations. Figure 2(b) illustrates that these approximations on a distorted mesh may include vertices which even further removed from the central vertex while neglecting other vertices which lie closer. In other words the distorted mesh cannot be used self-consistently to improve the accuracy of the approximation. The problem can be resolved only by differencing over the appropriate vertices. That is, the mesh distortion must be reduced. This same conclusion holds true for the finite element method. Whether triangular or quadrilateral elements are chosen the accuracy on the deformed grid must be diminished regardless of the basis functions, simply because the grid joins the wrong vertices.

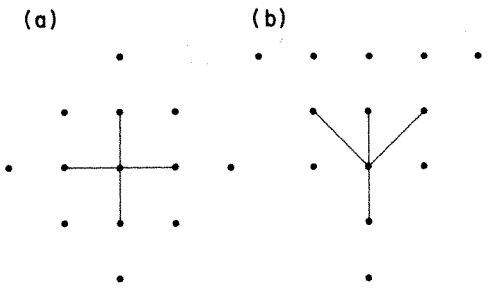

Figure 2(a). Vertices used in higher order approximations; (b). The vertex positions on the mesh distorted by shear

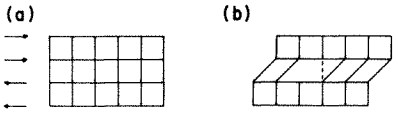

Figure 3(a). A quadrilateral mesh about a shear interface; (b). Distortions induced in the Lagrangian mesh by the shear. A reconnection has been made to join nearest neighbours for a single vertex

The traditional solution to this problem has been to perform an Eulerian rezoning phase which allows the mesh to pass through the fluid. There are several objections that can be made to this solution to mesh distortion. The first is that the result of a rezoning phase is the introduction of artificial diffusion as well as a reduction in distortion. Fluid is allowed to diffuse through the mesh as it is repositioned. Secondly, the rezoning phase phase may be less accurate than needed. In shear flows such as illustrated in Figures 1(b) and 2(b) accuracy can be maintained through rezoning only by an Eulerian motion counter to the fluid flow. If the shear persists, the Eulerian motion will eventually have to be equal and opposite to the advective, Lagrangian motion. In such cases the flow calculation is Eulerian. Unless the algorithms adopted for rezoning approach the accuracy available for finite-difference approximations of the advective terms on an Eulerian mesh, the accuracy of the calculation is necessarily degraded and even more diffusive than a purely Eulerian calculation. A third objection to rezoning is that it is useless in fluid flows which force changes from simply-connected to multiply-connected regions, as in the case of wave-breaking. In such cases no amount of rezoning can prevent mesh tangling and collapse of cells. A final objection is that it is not a definitive solution. A grid deformation *must* force a less accurate approximation. Rezone algorithms generally seek to preserve a reasonable appearance of the grid and may obscure the question of what is the best approximation possible given the current Lagrangian vertex positions. That is, under the guise of preventing inaccuracies in the approximations, they may become the very vehicle for preserving them.

The solution that rezoning offers to gird deformation is vertex motion to rectify the distortions. An alternative solution is illustrated in Figure 3. A section of a quadrilateral mesh about a shear layer is shown in Figure 3a. A Lagrangian calculation quickly leads to the mesh shown in Figure 3b, in which mesh connections about the layer no longer join neighbouring vertices. In this Figure one grid line has been reconnected to show the connection which is now appropriate. For a periodic system all stretched grid lines could be reconnected, thereby restoring the mesh to its original configuration *without* moving any vertices. In general, all such reconnections either are inappropriate or cannot be made due to boundaries, so that one triangular and one pentagonal cell remain. Therefore reconnection on a quadrilateral grid cannot by itself resolve the problem of distorted grids.

On a triangular mesh, however, there are no such complications. As shown in Figure 4, a reconnected grid line on a triangular mesh still results in two triangular cells. This technique was first used in a computer code by Crowley[5] and represents a very attractive alternative to rezoning for triangular grids. There is no Eulerian vertex motion. The full accuracy of the

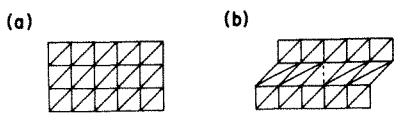

Figure 4(a). A triangular grid formed by drawing diagonals through the mesh of Figure 3(a), (b). The reconnection of Figure 3(b) on a triangular mesh

Lagrangian technique for advancing grid positions is always used, and the advection terms need never be evaluated. Although a grid line must be repositioned, fluid does not pass through the mesh in the previous sense since vertices remain stationary, and totally non-diffusive reversible algorithms are possible. Most importantly, the solution is definite. The simplicity of the technique is particularly suited to considerations of how best to construct approximations given the current grid positions. As shown below, such questions result in very attractive conservative algorithms for the reconnection operation.

The technique is similar to rezoning in two ways. First, it does nothing to enhance the accuracy of the approximations on the distorted grid before the remedial action is taken. As shown below, the decision of when to reconnect is made in a straightforward manner. But, as in a rezone method, the approximations made over the distorted mesh are, none the less, less accurate. The question of accuracy will be addressed in more detail in the following section. The solution of reconnecting is also similar to rezoning in that it cannot by itself solve the problem of fluids evolving into multiply-connected regions or of flows at stagnation points. However, it does aid in the remedy. The number of grid lines meeting at any vertex can be reduced to three by reconnections, with those three neighbouring vertices forming a triangle which surrounds only that vertex. If the fluid is accumulating vertices locally, then that central vertex can be eliminated with the three grid lines, leaving only the surrounding triangle. The result is the desired decrease in resolution and the avoidance of the formation of thin, narrow triangles near the point of converging flow. Conversely, a vertex may be added inside any triangle or along any line simply by providing the necessary grid lines to other vertices within the affected triangles. Subsequent reconnections will link the added vertices to their neighbours. In this way the transition to multiply connected regions and the flow near stagnation points can be handled smoothly merely by decreasing or increasing resolution where appropriate. The combination of grid line reconnection with vertex addition and deletion therefore provides a means of smoothly restructuring the grid without recourse to Eulerian vertex movement.

The price that is paid for this flexibility is the loss of any global ordering for the vertices. A reconnecting grid has no fixed indexing. Partly because of this difficulty general triangular grids have not received the attention given to regular grids and are not as well understood. Solution procedures for irregular grids are currently more costly and the range of available numerical techniques is much more restricted. For example, in a finite-element scheme a reassembly of the system matrix is required whenever the grid is required whenever the grid is restructured. Because of the cost of this step computationally, the finite element approach was not considered for this problem. Nevertheless, the approach finally adopted shares many features of finite elements, particularly in the use of integral methods in maintaining conservation. Since a minimization over the entire mesh is avoided in the technique described in this paper, a system matrix is not required and only local changes are necessary. Much of tne later discussion on evaluating the accuracy of a restructuring mesh is directly related to finite element meshes, and the results carry over for implementations in finite element grid initializers.

Fortunately, grid alterations can be accommodated in finite-difference schemes with relatively mild calculational repercussions. For that reason we will focus on finite-difference techniques from this point on. The relative lack of experience in differencing over general triangular grids has historically led to some mistaken impressions, two of which are important enough to address directly—their 'stiffness' and lack of accuracy.

The use of the work of 'stiff' to describe triangular grids seems to have evolved from an improper placement of physical variables on the mesh. In a quadrilateral mesh, there is a

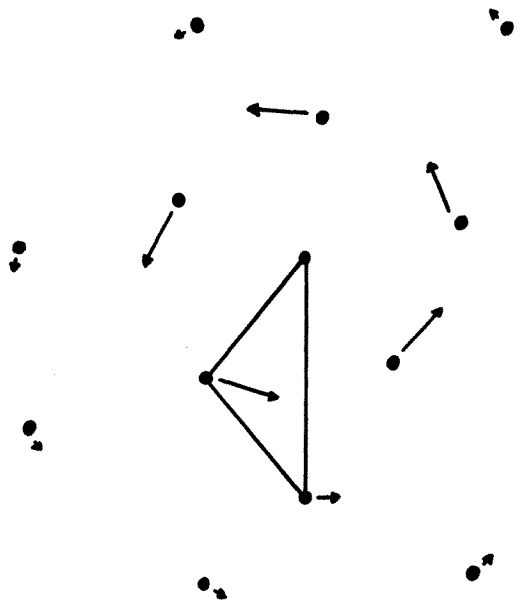

Figure 5. An isolated vortex about a central vertex. Velocities of neighbouring vertices reach a maximum at some radius from the central vertex and diminish further out

one-to-one correspondence between cells and vertices. For example, each quadrilateral cell may be associated with its upper right vertex. The vertices remaining lie on boundaries, and their variables are specified by boundary conditions. If that same quadrilateral mesh is subdivided into a triangular mesh by drawing diagonals through each cell, it is clear the correspondence between cells and vertices is destroyed. The number of vertices is the same, but there are twice the number of cells. On the quadrilateral mesh it is attractive to assign pressures as cell-centered quantities, since pressure forces are easily calculated. Such a positioning on a triangular mesh can be disastrous. Twice as many pressures must be specified than in the well-formulated, quadrilateral case. Numerically, an iterative solution for pressures would converge extremely slowly, and in this sense the system of differenced equations would be stiff. If the idea of conserving cell size for incompressible flows is also carried over to triangular grids, the algorithms are stiff in a much worse sense. A vortex is shown in Figure 5, where the arrows designate local fluid velocities. All velocities are about the vortex centre, and there exists a maximum speed at some radius from the centre. The triangular grid that is shown will therefore eventually invert, since the vertex with the maximum speed must pass between the two more slowly moving vertices. If triangle areas are conserved, a pressure must build up to resist this tendency until the speeds of all vertices are commensurate with cell area conservation. The presence of such a pressure component at any stage of the calculation is totally non-physical, since it arises from improper placement of variables and is directly opposed to the physical flow. Such behaviour is definitely 'stiff' since the solution will tend to resemble solid body rotation rather than vortex motion. Such 'stiffness' of triangular grids seems to be due entirely to such improper differencing.

The use of pressures defined on vertices will illustrate the second question, that of accuracy of difference schemes over a general triangular mesh. First, let us incorporate our rather novel placement of pressures in the one-dimensional case. There are both forward and

backward expansions of the pressure about point i;

$$p_{i+1} = p_i + \frac{\partial p}{\partial x_i}\Delta x + \frac{1}{2}\frac{\partial^2 p}{\partial x^2}\Delta x^2 + \frac{1}{6}\frac{\partial^3 p}{\partial x^3}\Delta x^3 + O(\Delta x^4) \tag{1}$$

and

$$p_{i-1} = p_i - \frac{\partial p}{\partial x_i}\Delta x + \frac{1}{2}\frac{\partial^2 p}{\partial x^2}\Delta x^2 - \frac{1}{6}\frac{\partial^3 p}{\partial x^3}\Delta x^3 + O(\Delta x^4) \tag{2}$$

Of course the forward and backward difference approximations can be obtained directly from equations (1) and (2);

$$\frac{\delta p^f}{\delta x_i} = \frac{p_{i+1} - p_i}{\Delta x} + O(\Delta x) \tag{3}$$

and

$$\frac{\delta p^b}{\delta x_i} = \frac{p_i - p_{i-1}}{\Delta x} + O(\Delta x) \tag{4}$$

Both are only first-order accurate and can be viewed as cell-centred quantities. If, instead, equations (1) and (2) are subtracted, the result is

$$p_{i+1} - p_{i-1} = 2\frac{\partial p}{\partial x_i}\Delta x + \frac{1}{3}\frac{\partial^3 p}{\partial x^3}\Delta x^3 + O(\Delta x^5). \tag{5}$$

The centred difference approximation is then

$$\frac{\delta p^c}{\delta x_i} = \frac{p_{i+1} - p_{i-1}}{2\Delta x} + O(\Delta x^2) \tag{6}$$

which is second order accurate and vertex centred. For the case of variable mesh spacing, we can replace Δx by $\Delta x'$ in equation (2), so that equation (5) becomes instead

$$p_{i+1} - p_{i-1} = \frac{\partial p}{\partial x_i}(\Delta x + \Delta x') + \frac{1}{2}\frac{\partial^2 p}{\partial x^2}(\Delta x^2 - \Delta x'^2)'(\Delta x^3) \tag{7}$$

If we define an average mesh spacing

$$\overline{\Delta x} = (\Delta x + \Delta x')/2 \tag{8}$$

and a mesh spacing change

$$\delta x = (\Delta x' - \Delta x)/2 \tag{9}$$

then

$$\begin{aligned} \Delta x &= \overline{\Delta x} - \delta x \\ \Delta x' &= \overline{\Delta x} + \delta x \end{aligned} \tag{10}$$

or δx is a measure of the deviation of the central grid point from the average position, i.e. the central position for equal grid spacing. Rewriting equation (7) we have

$$\frac{\delta p^c}{\delta x_i} = \frac{p_{i+1} - p_{i-1}}{2\overline{\Delta x}} + O(\max(\delta x, \Delta x^2, \Delta x'^2)) \tag{11}$$

which has an accuracy between first and second order depending on the relative sizes of δx, Δx^2 and $\Delta x'^2$ and their coefficients. Second order accuracy can be obtained in general only if the grid is equally spaced. First order accuracy results only when two grid points coincide.

The result is first order accuracy but not in the grid spacing, but rather in the change in grid spacing. Therefore, for gradual grid variation, nearly second order accuracy results. If we rewrite equations (3) and (4) using primes also, then equation (11) can be rewritten as

$$\frac{\delta p^c}{\delta x_i} = \frac{\dfrac{\delta p^f}{\delta x_i}\Delta x_i + \dfrac{\delta p^b}{\delta x_i}\Delta x_i'}{\Delta x_i + \Delta x_i'}$$

or, viewing the foward and backward differences as cell-centred quantities,

$$\frac{\delta p^c}{\delta x_i} = \frac{\sum_{j=1}^{2} (\delta p/\delta x_j)\,\Delta x_j}{\Sigma\,\Delta x_j}. \tag{12}$$

where the index j sums over the approximations on either side of the ith vertex. That is, the central difference can be obtained by an 'area' weighted sum of the forward and backward differences.

This result carries over directly to general triangular grids in two dimensions. We can define the pressure gradient there as

$$\nabla p_j = \sum_{i=1}^{3} p_i \frac{\hat{z}x(\mathbf{r}_{i-1} - \mathbf{r}_{i+1})}{2A_j} \tag{13}$$

where ∇p_j is the first order accurate, finite difference approximation to the gradient which is evaluated at the jth triangle centroid, A_j is the triangle area, $\hat{z}$ is a unit vector in the direction of the neglected co-ordinate and the sum extends over the three triangle vertices. The analogue of equations (12) is

$$\nabla p_j = \frac{\sum_{j=1}^{n} \nabla p_j A_j}{\Sigma A_j} \tag{14}$$

where the index j indicates a sum over all triangles around the vertex i. For special geometries this centred difference approximation is second order accurate or higher. In general it is less than second order accurate, but performs reasonably closely to second order accuracy for a general mesh provided that the triangle areas are roughly equal. In that case the error is determined by a formula similar to equation (7), an algebraic sum of squared triangle altitudes. The worst that can be achieved is first order accuracy, which is obtained only in the degenerate case of a zero area triangle at the vertex.

This implies that care must be taken in evaluating boundary conditions, but this task is alleviated by having the pressures specified at the boundaries. In a cell centred scheme, the pressures are located half a cell away, and boundary conditions, particularly at free surfaces, are more difficult to implement. Accuracy in the interior of the fluid is therefore diminished primarily by narrow triangles. As shown below, this restriction is not too serious for a reconnecting grid, since the grid can be made to reconnect to preserve regular triangles. In cases where this is not possible (near interfaces, for example), the addition or deletion of vertices can be used to regularize the mesh.

The conclusion is that the 'stiffness' and 'diminished' accuracy of approximation on the triangular grid are both in large measure due to improper placement of physical variables on the mesh. Pressures as cell-centred quantities force not only a stiff solution, but make second-order accurate approximations difficult to achieve. Two other comments should be made about equation (14). The first is that it can be used to recover all the usual gradient approximations for regular grids. An extension of the definition by taking the divergence of

both sides of equation (14) also yields all the usual Laplacian approximations (see below). The second feature of the equation is that the vertex pressure gradients can be viewed either as a sum over triangular gradients or in the more conventional way of a vertex sum. The former offers the possibility of vectorizing the equations, even though the mesh has no inherent order, a fact which greatly enhances the utility of the method for vector computers.

NUMERICAL ALGORITHMS

The computer code SPLISH is a two-dimensional Lagrangian fluid dynamics code for incompressible fluids which was developed in accordance with the philosophy of the previous section.[6,7] The basic equations for incompressible fluid flow are

$$\rho \frac{d\mathbf{V}}{dt} = -\nabla p - pg\hat{y} \tag{15}$$

and

$$\nabla \cdot \mathbf{V} = 0 \tag{16}$$

where the fluid density ρ, pressure p and velocity $\mathbf{V}$ are assumed to vary only with x and y. To ensure accuracy, equations (15) and (16) are supplemented by integral equations for conservation of mass, energy and momentum. With pressures specified at the vertices, ∇p is evaluated over triangles, and equation (15) can easily be updated implicitly or explicitly if velocities are considered to be triangle-centred. This placement of velocities as cell quantities and pressures at vertices is apparently unique to SPLISH and is the direct opposite of the usual placement. In what follows the subscript i will denote a vertex–centred quantity and j a triangle-centred quantity. In SPLISH the integration of velocities uses a split step algorithm whereby the velocities are advanced one half timestep (equation (17)), the grid is advanced a full timestep (equation (19)), and then the velocities advanced forward the other half timestep (equation (21)).

$$\mathbf{V}_j^{1/2} = \mathbf{V}_j^0 - \frac{\delta t}{2\rho_j}(\nabla p)_j^0 - \frac{\delta t}{2} g\hat{y} \tag{17}$$

$$\mathbf{V}_i^{1/2} = \tfrac{1}{2}(\mathbf{V}_i^0 + \mathbf{V}_i^n) \tag{18}$$

$$\mathbf{X}_i^n = \mathbf{X}_i^0 + \delta t \mathbf{V}_i^{1/2} \tag{19}$$

$$\tilde{\mathbf{V}}_j^{1/2} = R(\{\mathbf{X}_i^0\}, \{\mathbf{X}_i^n\}) \cdot \mathbf{V}_j^{1/2} \tag{20}$$

$$\mathbf{V}_j^n = \tilde{\mathbf{V}}_j^{1/2} - \frac{\delta t}{2\rho_j}(\nabla p)_j^n - \frac{\delta t}{2} g\hat{y} \tag{21}$$

The vertex velocity $\mathbf{V}_i^n$ appearing in equation (18) is obtained from the area-weighted $\mathbf{V}_j^n$ from the previous iteration,

$$\mathbf{V}_i^n = \frac{\sum_j \mathbf{V}_j^n A_j}{\sum A_j} \tag{22}$$

The advantage of using triangle centred velocities is the ease in conceptualizing and expressing conservation laws. Because of the paucity of experience in formulating algorithms over a general triangular grid, we employed the control volume approach, which uses an integral formulation to derive the difference algorithms. The control volume is defined by a cell, centred about each vertex. The boundaries of the vertex cell can be defined by apportioning each triangle equally to each of its vertices. One way of achieving this

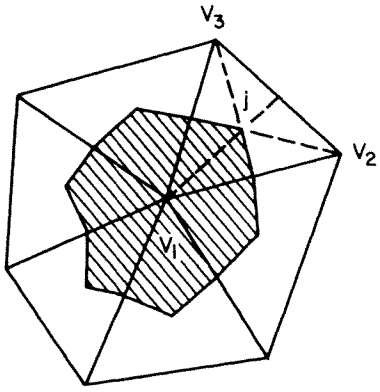

Figure 6. A definition of a vertex cell for a general triangular mesh

partitioning is to draw bisectors from the mid point of each side to the opposite vertex, as shown in Figure 6. The side bisectors meet at the centroid of the triangle and divide its area into six smaller equal area triangles. The areas of the two small triangles adjacent to each vertex are assigned to that vertex. The vertex cell of Figure 6 is constructed by summing over all the surrounding triangles. Therefore the area of a vertex cell may be defined as

$$A_c = \sum_j \tfrac{1}{3} A_j \tag{23}$$

where the sum extends over all adjacent triangles, exactly as in equation (14). With this definition equations (14) and (22) become

$$\nabla p_i = \frac{\frac{1}{3}\sum_j \nabla p_j A_j}{A_{c_i}} \tag{24}$$

and

$$\mathbf{V}_i = \frac{\frac{1}{3}\sum_j \mathbf{V}_j A_j}{A_{c_i}}. \tag{25}$$

Since triangle velocities are constant over triangles, conservation integrals about each vertex cell are easily evaluated. These conservation integrals are used consistently to ensure conservation during all phases of the calculation. For example, it is easy to show that the ∇p and gravity terms cannot alter the vorticity since numerically $\nabla \times \nabla p \equiv 0$ and gravity is a constant. Only the $(\nabla p_j)/\rho_j$ term can change vorticity, exactly as dictated by the physics. Therefore equations (17) and (21) advance velocities in keeping with the physically correct changes in vorticity under this scheme and placement of variables.

However, for a Lagrangian code, this is not enough. The price that is paid for the increased accuray in not differencing the advective terms is a change in the value of the conservation integrals during the vertex position update in equation (19). For any Lagrangian scheme, if the velocities are correctly advanced over the old grid, updating the grid may alter again the integrals about each cell. In SPLISH, conservation is ensured at this step by the transformation R in equation (20). The transformation is derived by considering the circulation about each vertex cell and calculating the changes in triangle velocities which are necessary to conserve vorticity and divergence about each cell. In this way vorticity is identically conserved in equations (17)–(21), i.e. for the full advancement of physical variables. Yet for Lagrangian calculations which distort the grid, there is still needed a

further check on conservation during the grid restructuring step. As will be described below, exactly the same integrals about each cell are used in the grid reconnection and vertex addition and deletion steps to ensure conservation. The end result is exact conservation for a full timestep always using the same conservation integrals for each stage of the calculation.

The pressures p_i^n in equation (21) are derived from the condition that the new velocities V_j^n should be divergence free at the new timestep, satisfying equation (16). The pressure Poisson equation is derived from equation (21) by setting $(\nabla \cdot \mathbf{V}_j^n)_i \equiv 0$, to obtain pressure p_i^n such that

$$\frac{\delta t}{2}\left(\nabla \cdot \frac{1}{\rho_j}(\nabla p)_j^n\right)_i = (\nabla \cdot \tilde{\mathbf{V}}_j^{1/2})_i. \tag{26}$$

The right hand side of equation (26) is the numerical analogue of the $\nabla \cdot (\mathbf{V} \cdot \nabla \mathbf{V})\, \delta t$ term which arises when the divergence of equation (15) is taken. Both terms in equation (26) are simple to evaluate, since the divergence is taken over triangle centred quantities. The paths are the 'surfaces' bounding the vertex volume of Figure 6, where the normal is directed outward from the vertex. The Poisson equation (equation (26)) that resulted from this integration has two advantages. First it is derived from $\nabla^2 \phi = \nabla \cdot \nabla \phi$ as in the continuum case. Secondly the left-hand side results in the more familiar second order accurate templates (such as the five-point formula) for the Laplacians for homogeneous fluids and regular mesh geometries.

The free surface boundary conditions are particularly easy to employ in this formulation. The pressure at free surface vertices is taken to be the ambient pressure. Since the surface vertices are advanced with the local fluid velocities, the non-linear terms in the boundary conditions which are forced in the Eulerian representation are simply not present here. The boundary condition at the bottom of the region, the surface of the obstacle and any rigid walls are also simple to implement. Vertices at a wall are constrained to stay on the wall but may move parallel to it. Therefore normal vertex velocities cannot arise, and pressures at the wall correctly evolve from the divergence equation.

The difficult boundary condition proves to be the radiative condition at the sides of the mesh. As was mentioned earlier, the formulation of an accurate radiative free surface boundary condition remains a very difficult problem. It is particularly difficult for the case of a general restructuring Lagrangian mesh because of the vertices from which extrapolations must be made are themselves changing in both position and grid connectivity. Physical complexity presents an additional problem of separate, distinct flow regions which may advect through the boundary. For these reasons periodic boundary conditions are imposed instead at the sides of the mesh in this paper. Such a restriction necessarily alters the problem itself to one of flow over an infinite array of obstacles. Therefore, the computational domain must be enlarged to reduce the effect of reentrant reflected and transmitted waves. What is gained is an exact boundary condition, which is particularly useful in calculations designed to test the method. The effect of approximations at the outflow and inflow boundaries are avoided, and the numerical technique can be tested separately from a test of the boundary conditions themselves.

In summary the finite difference formulas for SPLISH are derived using a control volume approach. Specifications of pressure at vertices leads naturally to the choice of positioning velocities at triangles. Although pressure gradients are constant over triangles, the resultant algorithms are expected to be nearly second order accurate since vertex velocities are derived from pressure gradient forces through sums about vertices, which in effect takes the central differences. The code has been tested extensively on finite amplitude standing waves

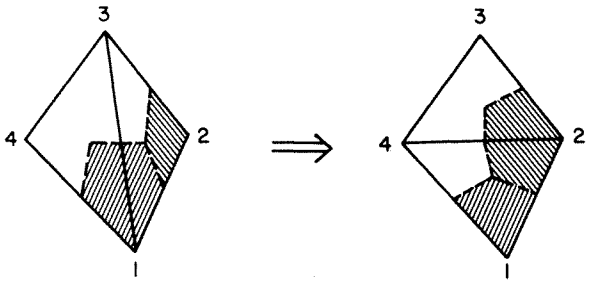

Figure 7. The vertex cells associated with two vertices before and after a reconnection

and has been shown to be basically second order accurate by the variation in period with mesh size.[8,9]

RESTRUCTURING THE GRID

The derivation of the reconnection and vertex addition and deletion algorithms is accomplished in the same manner through the control volume approach and the use of triangle velocities. The accuracy of a computer code which uses a reconnecting grid is determined by two aspects of the algorithms—how accurately post-reconnection physical variables are chosen and when the reconnections occur. As mentioned above, a reconnection offers the possibility of much less diffusion, since fluid passes through only one grid line for reconnections, whereas four grid lines are involved in every vertex rezone movement. Every mesh line can be viewed as one diagonal of the quadrilateral formed by the triangles to either side. A reconnection merely chooses the other diagonal. During a reconnection the smallest definable cell is the quadrilateral, not the two triangles, and it is necessary to ensure that quadrilateral properties are unchanged during a reconnection. That is, the quadrilateral is a control volume over which certain physical variables are conserved. As shown in Figure 7, a reconnection alters the vertex cells for each of the four quadrilateral vertices. To keep vorticity and divergence conserved the portions of the ingegrals $\int \nabla \cdot \mathbf{V}\, dV$ and $\int \nabla \times \mathbf{V} \cdot d\mathbf{A}$ about each vertex which lie within the quadrilateral must be the same before and after the reconnection. Since there are four velocity components to be specified after the reconnection, it would seem possible only to keep either the divergence or the vorticity about each vertex constant, but not both. However, the divergence and vorticity equations are not all independent and it is possible exactly to conserve both vorticity and divergence about each vertex with a unique set of triangle velocities. This same solution also conserves the quadrilateral velocity and has the added feature of time reversibility. Re-reconnecting a line yields the identical grid and physical variables both on vertices and triangles. This is extremely desirable since the basic finite-difference equations were also time reversible, as are the physical equations themselves.

The question of when to reconnect remains unresolved. As mentioned above, the accuracy of a general triangular mesh is diminished by narrow triangles. With reconnections, accuracy can be recovered by ensuring that small angles are preferentially eliminated. There are many ways of quantifying such an algorithm. The one we have chosen arises naturally from the pressure Poisson equation. Since the equation is solved in SPLISH by iteration, it is desirable that the convergence of the iteration be as rapid as possible. Mathematically, convergence is assured if the equation exhibits diagonal dominance. For a general triangle mesh diagonal

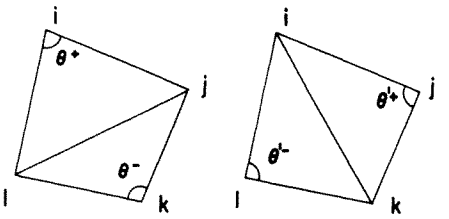

Figure 8(a). Definition of the angles θ^+ and θ^- for the diagonal line drawn from j to l; (b). The angles θ^+ and θ^- formed by connecting the other quadrilateral diagonal

dominance is obtained only if all coefficients

$$a = 1/2(\cot\theta^- + \cot\theta^+) \tag{27}$$

are positive, where θ^+ and θ^- are defined in Figure 8a. For postive area triangles θ^+ and θ^- are both between 0° and 180°, so that each term is negative only when $\theta^+ + \theta^- > 180°$, since

$$a = \frac{\sin(\theta^+ + \theta^-)}{2\sin\theta^+ \sin\theta^-} \tag{28}$$

In SPLISH the reconnection algorithm is chosen to be exactly the algorithm needed to preserve diagonal dominance. If $\theta^+ + \theta^-$ is greater than 180°, the grid line is reconnected as shown in Figure 8b. The new angles θ'^+ and θ'^- must seem to less than 180° since $\theta^+ + \theta^- + \theta'^+ + \theta'^-$ is the sum of the interior quadrilateral angles, which must be 360°. Therefore the reconnection algorithm is unique as well as straightforward. It also preferentially eliminates small angles for triangles, since the diagonal is chosen which divides the largest opposing angles.

Because the reconnection algorithms are specified to ensure diagonal dominance and eliminate small angle triangles, the second-order accuracy of SPLISH may be expected to be preserved. As yet, no test has been made of the accuracy of the reconnection algorithms for the complete range of gridding situations. This is mainly because reconnections cannot themselves ensure second-order accuracy, since flows near stagnation points *must* force narrow triangles. The reconnection algorithms have been tested extensively, however, through the Kelvin–Helmholtz instability. In the linear, non-linear and turbulent regimes, the algorithms have been shown to provide accurate calculations by comparisons with both theory and experiment.[9,10]

The reconnection algorithm given in equation (28) can also be interpreted as a means of defining a triangular grid which connects vertices which are nearest neighbors. For a collection of vertices on a plane, a Voronoi mesh can always be constructed which consists of general polyhedra about each vertex. The sides of each polyhedron enclose the area closer to its central vertex than to any other vertex. A triangular mesh can be derived from the Voronoi mesh by connecting vertices that share a polyhedron side. This triangular mesh is a dual of the Voronoi mesh. The reconnection algorithm given in equation (28) enforces the condition that the triangular mesh used by SPLISH is always that dual mesh.[11] Therefore the mesh always reflects the appropriate connections to define nearest neighbours.

The derivation of the remaining grid restructuring algorithms proceeds in exactly the same manner as above. For vertex addition, satisfaction of conservation integrals is particularly simple. A vertex added at the centroid of a triangle subdivides that triangle into three smaller triangles. If the new triangle velocities are all the same as the velocity of the subdivided triangle, all conservation laws are automatically satisfied. Since the reconnection

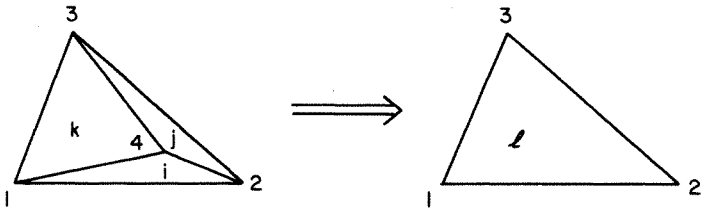

Figure 9(a). Vertex 4 isolated within a larger triangle before its removal; (b). The larger triangle remaining after deletion of vertex 4 and three associated sides and triangles

algorithm is also conservative, subsequent reconnections to other vertices ensure that the only effect of the addition is an increase in resolution.

The case is not as obvious for vertex deletion. Reconnections can be used to surround any interior vertex within a triangle. It can be shown, however, that once the vertex is surrounded by a triangle, the motion of that triangle is not altered if the vertex is removed and the new larger triangle given a velocity which is the area-weighted sum of the old velocities,

$$A_4\mathbf{V}_4 = A_1\mathbf{V}_1 + A_2\mathbf{V}_2 + A_3\mathbf{V}_3. \tag{29}$$

Such a substitution also conserves vorticity exactly, and effects a redistribution in accordance with area co-ordinates. Figure 9 illustrates the triangles before and after vertex removal. If ξ_4 is the vorticity about vertex 4 before removal, then the vorticity about each of the other three vertices is increased by an amount ξ_i' where

$$\begin{aligned} \xi_1' &= A_j\xi_4/A_l \\ \xi_2' &= A_k\xi_4/A_l \\ \xi_3' &= A_i\xi_4/A_l \end{aligned} \tag{30}$$

Here $\xi_1' + \xi_2' + \xi_3' = \xi_4$ since $A_i + A_j + A_k \equiv A_l$. Therefore total vorticity is conserved and redistributed in a reasonable and natural manner. Since the behaviour of the divergence is governed by a similar set of equations and conservation of momentum is ensured by equations (29), the conservation of the flow variables is guaranteed and a loss in resolution is the primary effect of deletion.[12]

The vertex addition and deletion rountines, together with the reconnection algorithms, are incorporated in SPLISH through driving routines which test for large or small grid lines and large, small or skewed triangles, as well as special tests at boundaries and interfaces. The resultant code automatically restructures the grid under constraints in the form of a maximum allowable skewness and maximum and minimum areas and line lengths. The complete routing has been tested using the Rayleigh–Taylor instability and in flows about hydrofoils. Preliminary results of these tests are extremely satisfying, providing the first totally Lagrangain calculations of these phenomena at long times.[7,11]

The tests performed using this code for the wave–structure interaction program are specifically aimed at the non-linear free surface effects to answer the question of whether the code does indeed still behave as a basically seond-order code and if the boundary conditions used thus far are realistic. In the following sections we will describe one way of initializing progressive waves which is simple and efficient, and subsequent tests of the code's behaviour by comparison with theory.

NUMERICAL RESULTS AND COMPARISON WITH THEORY

The use of periodic boundary conditions restricts the calculational domain to an integral number of wavelengths, so that the domain represents one element of an array of identical domains. For many conditions of wave motion this yields a quite realistic situation, and there is no concern for numerical damping at radiative boundaries. This restriction on the problem has in fact permitted considerable progress to be made in the numerical calculation of free-surface, progressive waves, even for cases with obstacles such as bottom seated half-cylinders in the flow. The present findings show that when wave reflection from an obstacle is small, the periodicity of domains does not prevent good agreement between numerical results, experimental data and theoretical results.

Calculations with SPLISH for progressive surface waves in a uniform depth ocean have given good agreement with classical wave theory for the wave period. The calculation is started by a sinusoidal pressure pulse on the free surface which initiates a standing wave. At the quarter period of the standing wave ($t = T/4$ s) a second pulse (phase shifted a quarter wave length) is applied to form the progressive wave. This initialization procedure may not be the most appropriate, especially when an obstacle is in the flow field, but it is simple, easy to apply, and exhibits good numerical stability for standing waves[7] and for progressive waves in a straight channel. Also, it works well with the periodic boundary conditions and introduces minimal grid distortion and variation in the triangle areas. Previous calculations with SPLISH[13] had given wave period values for a standing wave which agreed very well with theory. The period converged to the theoretical value as the grid step size was reduced.

The triangular grid for the SPLISH code is illustrated in Figure 10 for a case of wave motion (from left to right) in a uniform depth channel. Some triangle reconnections have taken place as may be seen by the departures from the initial regular grid. Also, under the periodic boundary conditions, some triangles have been moved from the right side of the domain to the left side in order to keep the domain compact. For wave flow over a bottom seated half-cylinder, an initial automatic grid adjustment makes room for the half-cylinder by shifting upward the triangle vertices above it.

Low reflection wave flow over a half-cylinder

The configuration which we consider is waves moving over a bottom seated half-cylinder, which might correspond to a half buried pipe or a storage tank on the ocean floor. However,

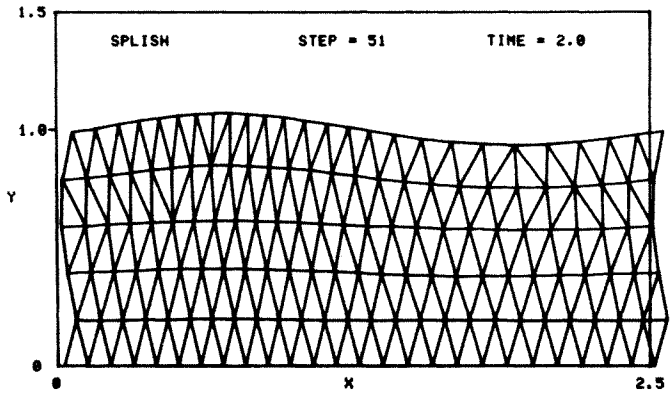

Figure 10. An example of a SPLISH-generated computational grid

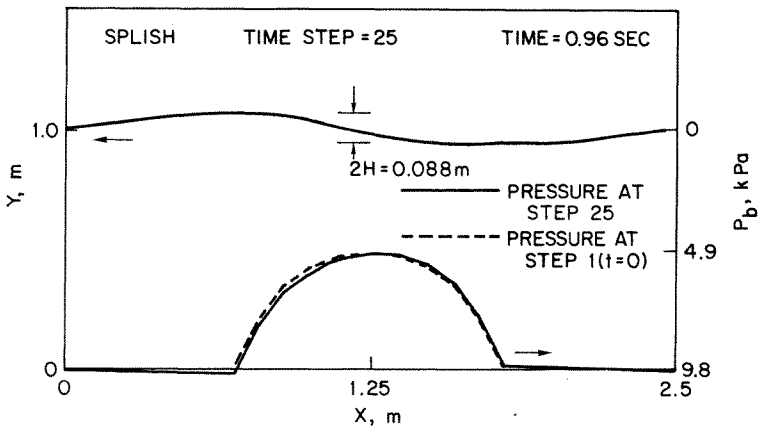

Figure 11. The surface contour and the bottom pressure from a numerical calculation of the wave flow over a bottom seated half-cylinder

with periodic boundary conditions, the numerical simulation actually corresponds to an array of pipes which are spaced one or more wavelengths apart. There are combinations of wavelength, depth, and cylinder radius for which only a small amout of the incident wave is reflected by the half-cylinder. Conditions were chosen so that the reflection coefficient $R = 0{\cdot}03$ based upon a linear wave Green's function formulation.[1] Under these conditions, neighbouring half-cylinders should have negligible effect on the flow. A series of calculations was made for these low reflection conditions.

Figure 11 shows the free surface contour and the resultant bottom pressure P_b at one instant in the passage of a wave (from left to right) over a half-cylinder (radius $a = 0{\cdot}5$ m) in a channel (depth $d = 1$ m) for a wave length $\lambda = 2{\cdot}5$ m. The finite amplitude of the wave ($H = 0{\cdot}044$ m, from the undisturbed free surface) causes only a very small deviation from a sinusoidal surface shape in spite of the presence of the bottom seated half-cylinder. Note that the pressure scale increases downward. Since the calculation was started from still water, the pressures at step 1 correspond to the hydrostatic pressure on the channel bottom and on the surface of the half-cylinder. At step 25 the wave gives an increased pressure on the left side of the half-cylinder and a reduced pressure on the right side. This is responsible for a net force (at this instant) from left to right. At other steps in the calculation, the pressure fluctuation accurately follows the passage of peaks and troughs of the wave as will be shown.

There is also a depth dependence of the pressure fluctuation. As the crest and the trough of the wave pass, the pressure fluctuation on top of the half cylinder (at a depth of 0·5 m) is approximately twice that on the channel bottom (at a depth of 1·0 m). At intermediate depths (i.e. along the sides of the cylinder) the magnitude of the pressure fluctuation is between that on the channel bottom and that on the top of the cylinder.

In Figures 12 and 13 fluctuation in P_b, i.e. ΔP, is normalized by its maximum value (ΔP_{REF}) which occurs at the top of the cylinder as the wave crest passes by. Figure 12 shows the pressure distribution around the half-cylinder at times $t = 1{\cdot}28$ and $t = 1{\cdot}92$ s which are the instants when the wave crest and trough pass over the half-cylinder. Figure 13 shows the pressure distributions around the half-cylinder at $t = 0{\cdot}96$ s and at $t = 1{\cdot}64$ s which correspond to times when the wave crest is first to the left of the cylinder and then to the right of the cylinder. In these two figures the values of ΔP from the SPLISH numerical calculations are shown by the open symbols. The solid and dashed curves shown in these figure are the

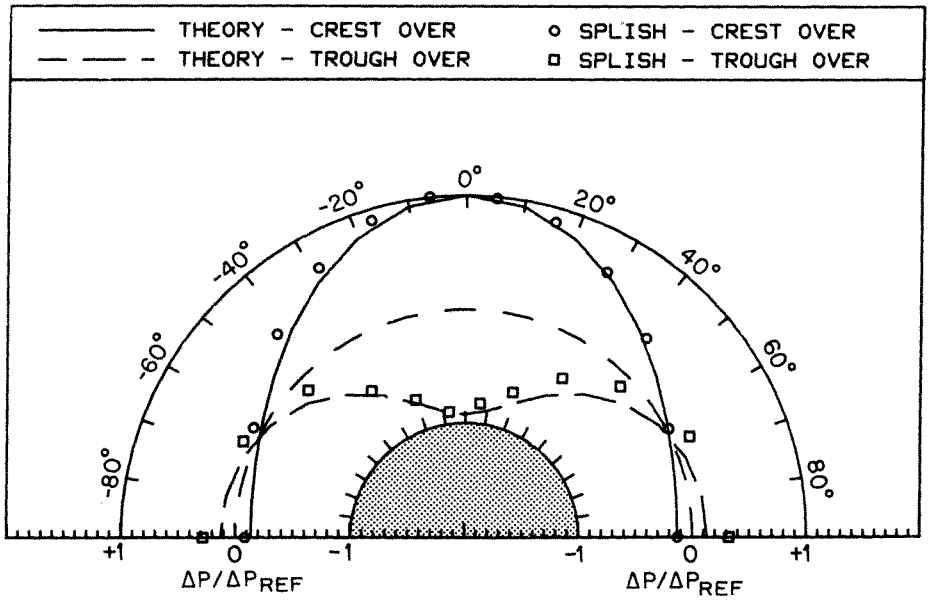

Figure 12. A comparison of the distribution of pressure fluctuations around the half-cylinder from an asymptotic fifth-order wave theory[14] and from a SPLISH calculation, for the wave crest and trough passage (wavelength $\lambda = 2{\cdot}5$ m, depth $d = 1$ m, radius $a = 0{\cdot}5$ m)

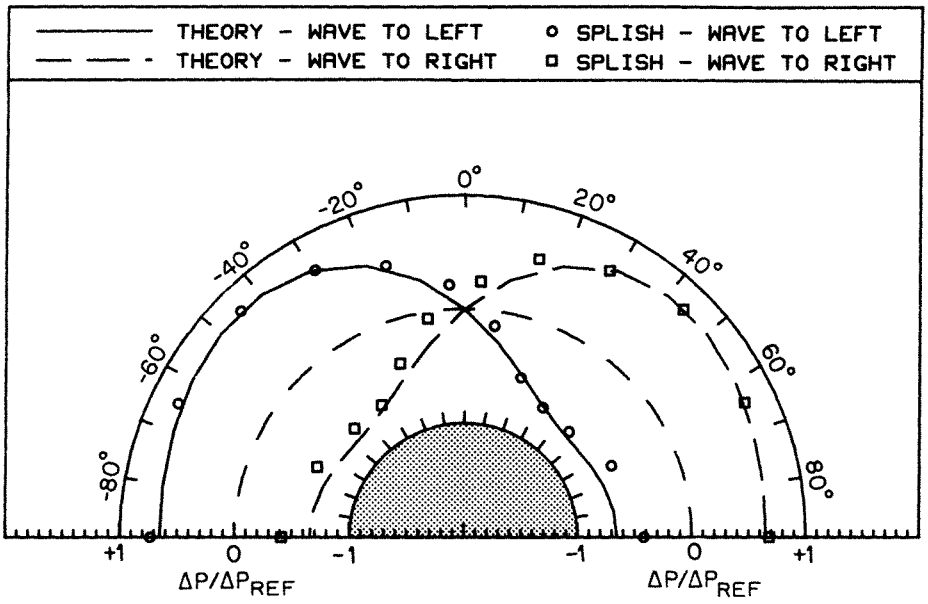

Figure 13. A comparison of the distribution of pressure fluctuations around the half-cylinder from fifth-order wave theory and from a SPLISH calculation, for the wave crest to the left and right of the half-cylinder (wavelength $\lambda = 2{\cdot}5$ m, depth $d = 1$ m, radius $a = 0{\cdot}5$ m)

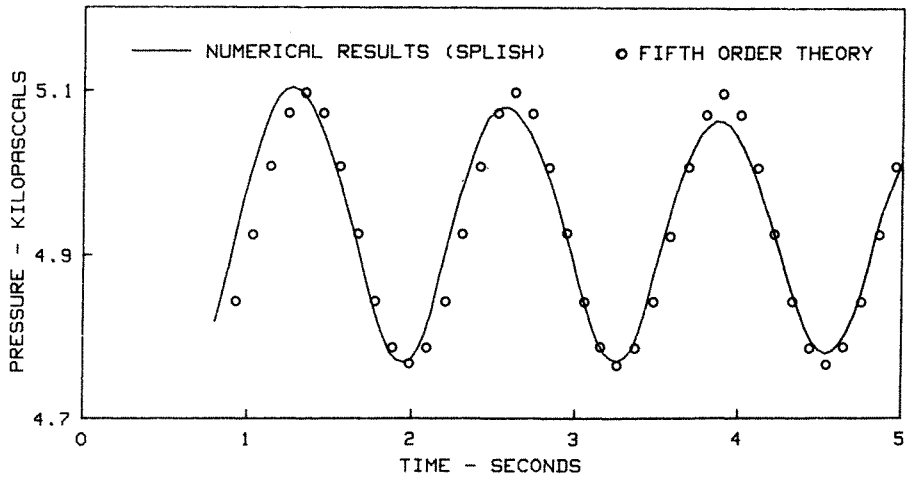

Figure 14. A comparison of the pressure time history at the top of a bottom seated half-cylinder from a SPLISH calculation and from fifth-order wave theory (wavelength $\lambda = 2{\cdot}5$ m, depth $d = 1$ m, radius $a = 0{\cdot}5$ m, wave amplitude $H = 0{\cdot}038$ m)

values of ΔP obtained for these conditions using the approach of Chakrabarti and Naftzger[14] which was based on Stokes' fifth-order wave theory and the assumption that the effect of the free-surface on the reflected wave potential could be neglected. For the present low-reflection conditions (with a reflection coefficient of 3 per cent) such an assumption is reasonable. For these conditions the SPLISH results are in good agreement with the fifth-order theory results.

The time history for the pressure (in kiloPascals or kN/m^2) at a vertex on the top of the half-cylinder is shown in Figure 14. The solid curve shows the SPLISH data and the open symbols are values obtained using a full fifth-order solution obtained by Chakrabarti and Naftzger[14]. The pressure here represents the actual gauge pressure that might be measured. The wave amplitude was taken to be $H = 0{\cdot}038$ m for the fifth-order calculations, and the agreement between the numerical calculations and the fifth-order calculation is quite good. With the finite grid size employed here the period given by the numerical results is about 4 per cent greater than that from the theoretical dispersion relation, so that the theoretical and numerical results shift slightly relative to each other with time. From fifth-order theory $T = 1{\cdot}269$ s and from linear theory $T = 1{\cdot}274$, whereas SPLISH gives a value of $T = 1{\cdot}30$ s based on the intervals between maxima or between minima in the pressure-time history.

The forces and pressures on the half-cylinder are the principal results of the numerical calculations, but at times the quality of the numerical solution is not shown by the pressure and force data. Plots of the Lagrangian particle paths have proven to be a very useful diagnostic aid for determining the quality of the solution and for examining the physical mechanism in some wave flow situations. Figure 15 shows the Lagrangian paths of three surface vertices, in Figure 15(a) the vertex initially at $x = 0$ m, in 15(b) the vertex at $x = 0{\cdot}6$ m, and in 15(c) the vertex at $x = 1{\cdot}2$ m. The particle paths begin at step 11 after the standing wave is fully formed and the second pressure pulse has been applied. The presence of the half-cylinder in the flow seems to affect the path of the surface particle above it only slightly. A larger effect would be seen for a larger wave amplitude or for vertices nearer to the cylinder. The particle paths in Figure 15 also exhibit a slight decrease in amplitude with time. This decrease in wave amplitude (and thus wave energy) is not due to viscous effects in

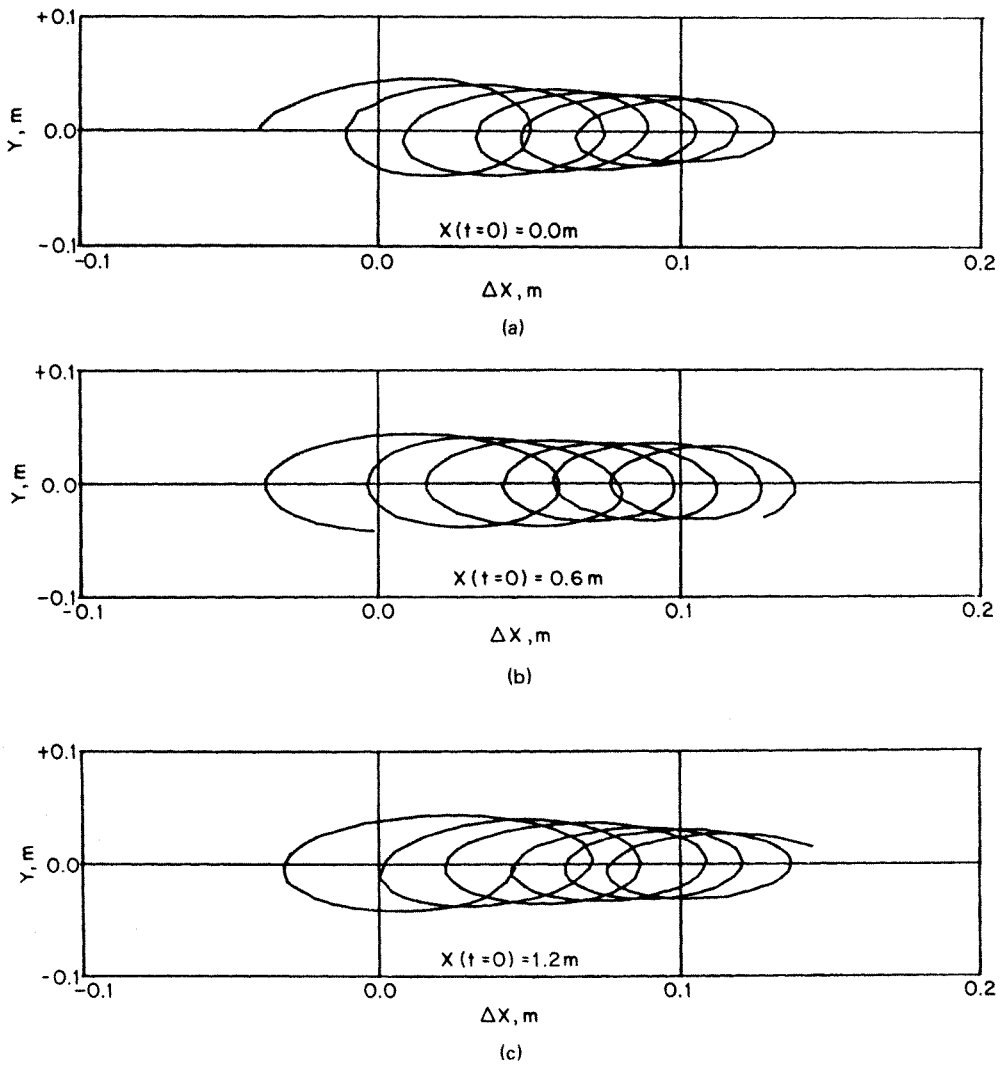

Figure 15. Particle paths on the free surface from a numerical calculation of wave flow over a bottom seated half-cylinder. Vertices were initially at $x = 0{\cdot}0$ m, $x = 0{\cdot}6$ m and $x = 1{\cdot}2$ m (wavelength $\lambda = 2{\cdot}5$ m, depth $d = 1$ m, radius $a = 0{\cdot}5$ m)

the governing equations, but it is caused by an effective dissipation of energy due to the incomplete convergence of the successive-over-relaxation (SOR) method used for the solution of the Poisson equation for the pressure.

Even though there is some loss of energy in the numerical solution (due to the incomplete convergence of the SOR method), the Lagrangian particle paths clearly show that, for these low wave reflection conditions, SPLISH is generating a stable, well-behaved solution for the surface wave motion over the half-cylinder. The particle paths seem to indicate that the wave reflection from the half-cylinder is sufficiently low that it neither adversely affects the quality of the solution nor significantly changes the physical situation which is being modelled numerically. We consider in the next section a case in which the amount of wave reflection is significant.

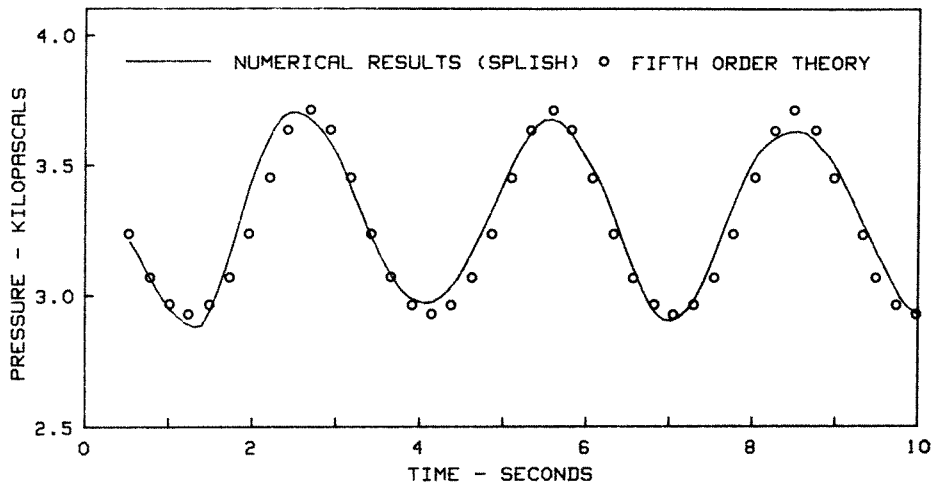

Figure 16. A comparison of the pressure-time history at the top of a bottom seated half-cylinder from a SPLISH calculation and fifth-order wave theory calculation for high reflection wave flows (wave length $\lambda = 8{\cdot}4$ m, depth $d = 1$ m, radius $a = 0{\cdot}667$ m wave amplidute $H = 0{\cdot}038$ m)

High reflection wave flow over a half-cylinder

Numerical calculations also have been performed with SPLISH for a case with significant wave reflection. For this case the depth $d = 1{\cdot}0$ m, the radius $a = 0{\cdot}667$ m and the wavelength $\lambda = 8{\cdot}4$ m. From the results of Naftzger and Chakrabarti[1] the wave reflection coefficient $R = 0{\cdot}4$ and the reflected wave has 16 per cent of the energy of the incident wave.

Surprisingly, some of the results from the SPLISH calculation show little adverse effect from the wave reflection. For example, the time history of the pressure at a point on top of the half-cylinder, shown in Figure 16, agrees quite well with an earlier asymptotic fifth order wave theory model.[14] The method is asymptotic in the sense that the water depth is sufficiently large that the effect of the free surface boundary on the diffracted wave can be neglected. This approximate method was selected because of its simplicity and relative accuracy for the conditions tested. Other numerical results for these conditions (not shown here) agree rather poorly with the theoretical and experimental results. This disagreement appears to arise from two factors. The first is due to a mismatch in timing for this case of the second pressure pulse initializing the calculation. The second is that boundary conditions other than the periodic ones used are needed for cases with significant wave reflection. The periodic boundary conditions allow the reflected wave to reenter the calculation instead of radiating outward from the obstacle. For high wave reflection, the disturbance to the flow field is significant and the intended physical situation is not properly modelled numerically.

In the above discussion we have only compared the results obtained with SPLISH with results from classical linear and fifth-order wave theories. Even though most of these comparisons showed good to very good agreement, it is also appropriate to compare the calculations with experimental data. This is done in the next section.

COMPARISONS WITH EXPERIMENTAL DATA

Several wave motion experiments were performed in the NRL wave channel with a bottom seated half-cylinder so that actual pressure measurements could be compared both to the

present numerical results and to the previous work of Chakrabarti and Naftzger[1,14] A 1·07 m ($3\frac{1}{2}$ ft) diameter half-cylinder, which spanned the entire width of the channel, was placed about one-half of the channel's length from the mechanical wavemaker. At the other end of the wave channel a sloping, porous beach with a rubberized horsehair blanket served to absorb nearly all of the incident wave energy. Nineteen equally-spaced ($\Delta\theta = 10°$) pressure taps were located around the circumference of the half-cylinder at its midsection. The individual taps were connected to a differential pressure transducer by a rotary pressure switch. The wave height along the channel was obtained from several traversing capacitance-type wave gauges. Calibrations were performed on all sensors before and after each test series to ensure that the overall accuracy of the measurement system remained well within ±5 per cent. The pressure and waveheight signals were digitized and processed by means of a Hewlett-Packard 5420A Digital Signal Analyzer. The experimental systems and methods are discussed further in a related NRL report.[4]

The cylinder reflection coefficient R, defined as the ratio of reflected to incident wave amplitudes, was obtained for a range of wavelengths and water depths. Reflection coefficients were obtained using a modified form of the method, introduced by Ursell, Dean and Yu[15] based upon modulations of the waveheight along the channel. The results of these measurements are plotted in Figure 17 against the wavenumber $ka = 2\pi a/\lambda$ and for several values of the relative water depth d/a. Also shown in the Figure (as solid lines) are theoretical values based on linear theory which were calculated by Naftzger and Chakrabarti.[1] There is general agreement with the observed cylinder reflections.

It should be noted, however, that some care was taken to avoid finite amplitude effects through the use of small wave steepness ratios ($H/\lambda \lesssim 0{\cdot}05$). The wave steepness values during the $d/a = 1{\cdot}25$ tests were reduced further, typically to less than 0·02, in order to avoid second harmonic wave generation at the cylinder. This non-linear effect is associated with the finite waveheight being a significant fraction of the finite water depth at the top of the

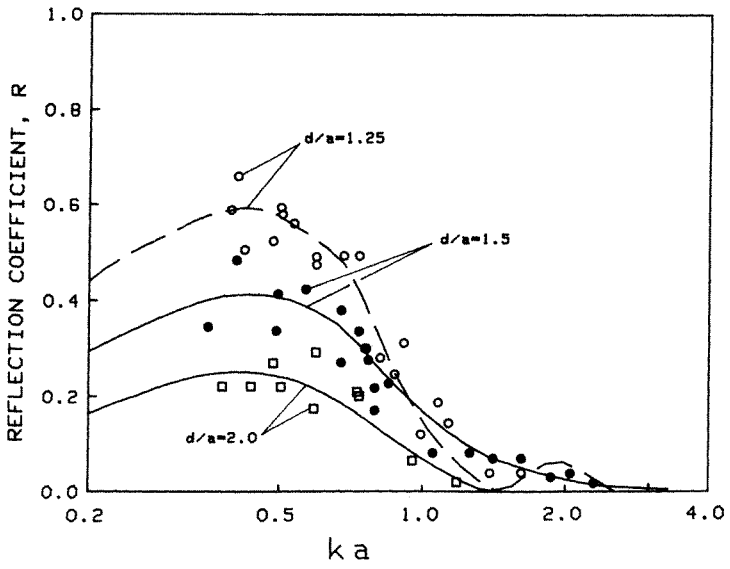

Figure 17. Experimental (○, ●, □) and computed (—) values for the wave reflection coefficient of a bottom-seated half cylinder. The computed values were obtained using linear wave Green's function approach by Naftzger and Chakrabarti[1]

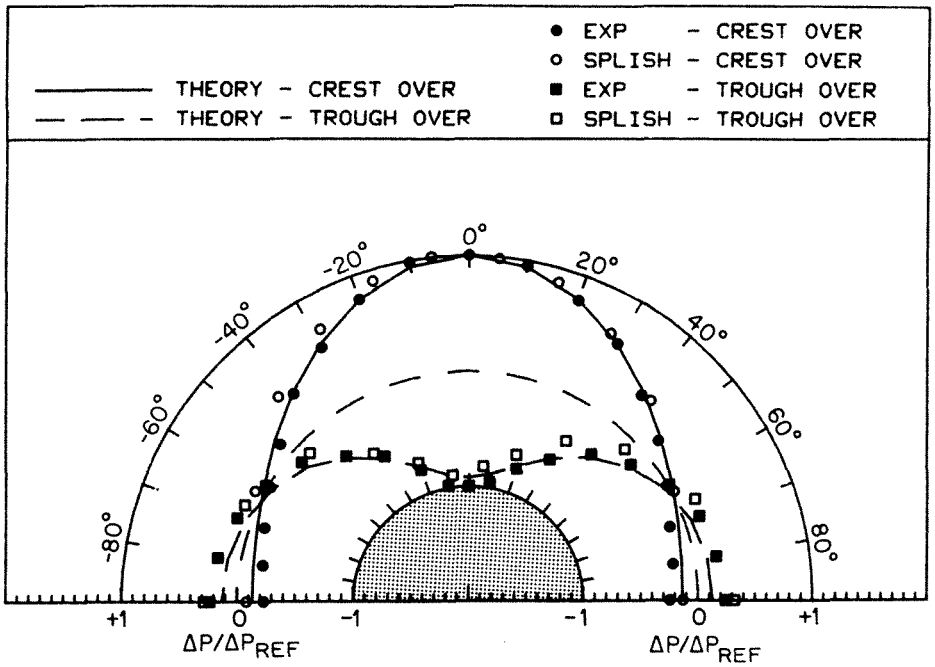

Figure 18. A comparison of theoretical,[14] numerical, and experimental results for selected instantaneous pressure distributions on a bottom seated half-cylinder in waves, for wave crest and trough directly over the cyclinder (low reflection wave flow, $d/a = 2{\cdot}0$, $ka = 1{\cdot}25$, $R \lesssim 0{\cdot}05$)

cylinder. The second harmonic wave could be seen as the formation of a secondary crest when a trough was over the cylinder.

A rule of thumb emerged from this observation and from several wave gauge spectral records wherein second harmonic amplification did not occur for waveheights less than about $\frac{1}{4}$ of the water depth over the cylinder. This criterion for the validity of the analytical method is more stringent than the one originally proposed by Naftzger and Chakrabarti. The onset and form of the non-linearity is the subject of a recent investigation.[16]

From the range of water depths and wavelengths shown in Figure 18, two cases were selected for detailed pressure studies and for comparisons with the theoretical and numerical results. The first is a relatively low reflection case ($d/a = 2{\cdot}0$ and $ka = 1{\cdot}25$). so that $R \lesssim 0{\cdot}05$. This represents a situation where good agreement between the three sets of results was expected. The second case is a relatively high reflection condition ($d/a = 1{\cdot}5$, $ka = 0{\cdot}5$, $T \approx 0{\cdot}4$) which was selected as a significant test for SPLISH with regard to the use of periodic boundary conditions in the code.

Low reflection case

A comparison between the experimental results and a linear approximation to the fifth order model[14] was obtained. This approximation was selected for the comparison because of its simplicity and relative accuracy. The theoretical and experimental results agreed well except near the intersection of the half-cylinder sides and the wave channel floor. These differences could be a consequence of a small gap which existed between the bottom of the half-cylinder and the wave channel.

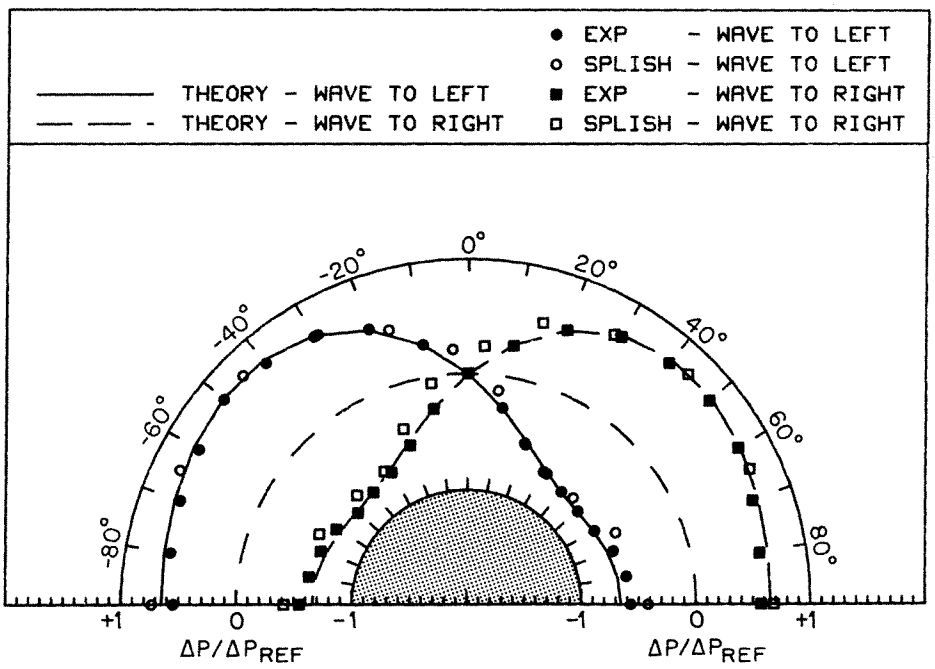

Figure 19. A comparison of linear wave theory,[14] numerical, and experimental results for selected instantaneous pressure distributions on a bottom seated half-cylinder in waves, for crest to the left and the right of the half-cylinder (low reflection wave flow, $d/a = 2{\cdot}0$, $ka = 1{\cdot}25$, $R \leq 0{\cdot}05$)

Since the numerical results are Lagrangian whereas the theoretical and experimental results are Eulerian, the simplest format for comparing all three is the pressure distribution around the cylinder at selected times in the wave cycle. Figures 18 and 19 present several such comparisons for the experimental conditions cited above. The three sets of results compare well except near the bottom of the cylinder. Although the discrepancy appears to be small in such a plot the effect on a measured horizontal component of the wave force can be significant.

High reflection case

Similar results and comparisons for a relatively high reflection case ($d/a = 1{\cdot}5$, $ka = 0{\cdot}5$, $R \approx 0{\cdot}4$) are shown in Figures 20 and 21. As before, the pressures are normalized by the magnitude at the top of the cylinder. In this case, however, the maximum pressure fluctuation occurs on the upstream side of the cylinder where both the incident and reflected waves are present. The experimental pressures shown in the Figures were obtained from the data after a correction for the additional standing waves in the laboratory channel. This correction, often large, neglected secondary reflections and was based on linear wave theory which may be responsible for all or part of the discrepancies between results. The computed results shown for the method developed by Naftzger and Chakrabarti were provided by them (R. A. Naftzger, private communication, 1980).

In spite of the difficulties implied, above, the agreement between the linear theory and the experiments is relatively good. The previously mentioned restriction on wave amplitude ought to be reiterated. The SPLISH results show the effect of the periodic boundary

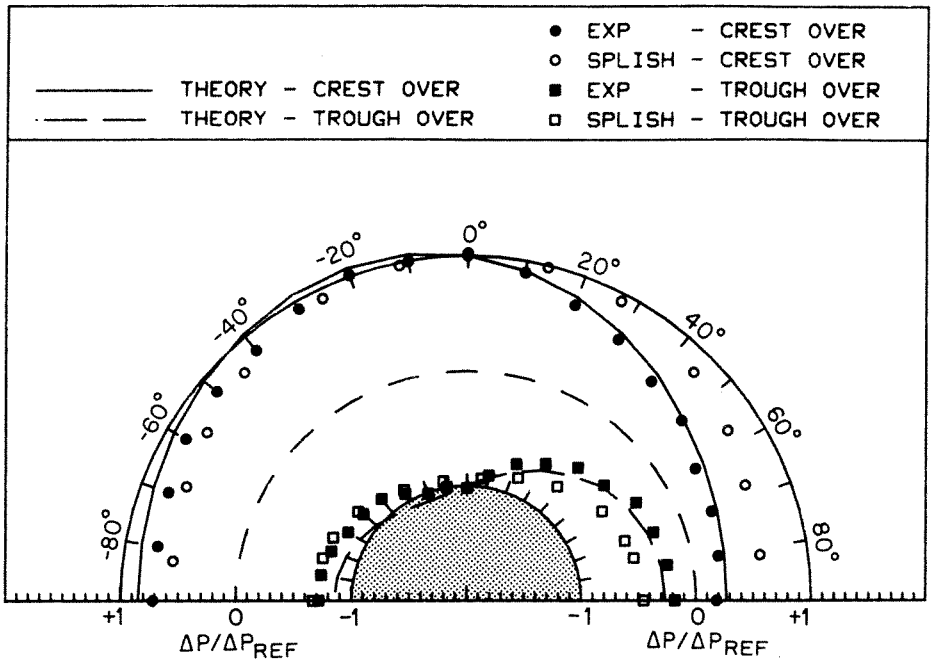

Figure 20. A comparison of linear wave theory,[1] numerical, and experimental results for selected instantaneous pressure distributions on a bottom seated half-cylinder in waves, for wave crest and trough directly over the cylinder (high reflection wave flow, $d/a = 1{\cdot}5$, $ka = 0{\cdot}5$, $R \approx 0{\cdot}4$)

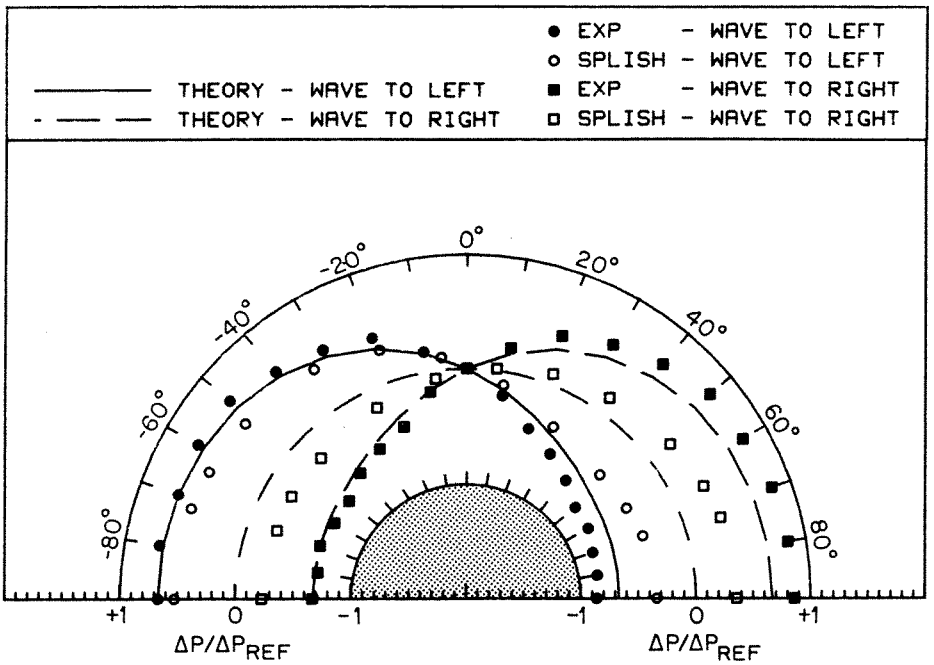

Figure 21. A comparison of linear wave theory,[1] numerical, and experimental results for selected instantaneous pressure distributions on a bottom seated half-cylinder in waves, for wave crest to the left and the right of the half-cylinder ($d/a = 1{\cdot}5$, $ka = 0{\cdot}5$, $R \approx 0{\cdot}4$)

conditions and perhaps also of the improper wave initialization. There is less satisfactory agreement between the numerical calculations, the experimental data and the theoretical results for conditions of high wave reflection.

SUMMARY AND CONCLUSIONS

A finite-difference numerical method for solving the governing equations of motion for inviscid, irrotational flow with a free surface using a Lagrangian triangular grid has been used and shown to yield reasonable results. Calculations for progressive surface wave motions have given results for the wave period, the drift velocity and the surface particle movements which are in good agreement with results obtained from classical wave theories.

Calculations for the passage of waves over a submerged obstacle are encouraging and show some promise of providing practical results over a relatively wide range of wave conditions. These calculations demonstrate the adaptability which the triangular grid provides. The advantages of the Lagrangian formulation are shown in that the grid conforms to the fluid area and that no interpolation is needed to locate the free surface or the surface of the submerged obstacle.

Two cases of wave motion over a submerged, half-cylindrical obstacle have been considered. The results for the low wave reflection case indicate that, even with the periodic boundary conditions, a code such as SPLISH can be employed with reasonable confidence to calculate the motion of waves over obstacles (thus wave–structure interactions). However, for cases with significant wave reflection from an obstacle, continuative or radiative boundary conditions are necessary, since the periodic boundary conditions lead to the simulation of a flow quite different from that desired. Also, alternative techniques for initiating the traveling surface wave seem to be required.

ACKNOWLEDGMENTS

This paper has been prepared as part of the research programme in hydrodynamics of the Naval Research Laboratory. The authors wish to thank S. K. Chakrabarti and R. A. Naftzger of the Chicago Bridge and Iron Company for providing detailed computational results for several of the test cases discussed in this paper.

REFERENCES

1. R. A. Naftzger and S. K. Chakrabarti, 'Scattering of waves by two-dimensional obstacles', *Journal of Ship Research*, **23**, 32–42 (1979).
2. E. W. Miner, M. J. Fritts and O. M. Griffin, 'A finite-difference method for calculating free surface waves', *Proceedings of the First International Conference on Numerical Methods in Laminar and Turbulent Flow*, Swansea, U. K., 597–608 (1978).
3. E. W. Miner, O. M. Griffin, S. E. Ramberg and M. J. Fritts, 'A numerical and experimental study of free surface wave flow over a half-cyclinder', *Proceedings of the Second International Conference on Numerical Methods in Laminar and Turbulent Flow*, Pineridge Press, Swansea, U.K., 705–718 (1981).
4. E. W. Miner, O. M. Griffin, S. E. Ramberg and M. J. Fritts, 'Numerical calculation of surface wave effects on marine structures', *Naval Research Laboratory Memorandum Report* 4395, 1980.
5. W. P. Crowley, *Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics*, Springer-Verlag, 1971.
6. J. P. Boris, K. L. Hain and M. J. Fritts, 'Free surface hydrodynamics using a Lagrangian triangular grid', *Proceedings of the First International Conference on Numerical Ship Hydrodynamics*, NBS, Gaithersburg MD., 683–716 (1975).
7. M. J. Fritts and J. P. Boris, 'Solution of transient problems in free-surface hydrodynamics', *Naval Research Laboratory Memorandum Report* 3446, 1977.

8. M. J. Fritts, 'A numerical study of free-surface waves', Science Applications, Inc. *Report SAI-76-528-WA*, 1976.
9. M. J. Fritts and J. P. Boris, 'The Lagrangian solution of transient problems in hydrodynamics using a triangular mesh, *J. Comp. Phys.*, **31,** 173–215 (1979).
10. M. J. Fritts, 'Lagrangian simulations of the Kelvin–Helmholtz instability', Science Applications, Inc. *Report SAI-76-632-WA*, 1976.
11. M. J. Fritts, 'Numerical approximations on distorted Lagrangian grids', *Advances in Computer Methods for Partial Differential Equations—III, Proceedings of the Third IMACS International Symposium of Computer Methods for Partial Differential Equations*, Bethlehem, Pa., 20–22 June 1979.
12. M. J. Fritts, 'Transient free-surface hydrodynamics', *Naval Research Laboratory Memorandum Report 3651*, 1977.
13. M. J. Fritts and J. P. Boris, 'Transient free surface hydrodynamics', *Proceedings of the Second International Conference on Numerical Ship Hydrodynamics*, Berkeley, CA., 319–328 (1977).
14. S. K. Chakrabarti and R. A. Naftzger, 'Nonlinear wave forces on half cylinder and hemisphere', *Proceedings of the ASCE, Journal of the Waterways, Harbors and Coastal Engineering Division*, **100,** 189–204 (1974).
15. F. Ursell, R. G. Dean and Y. S. Yu, 'Forced small-amplitude water waves: a comparison of theory and experiment', *Journal of Fluid Mechanics*, **7,** 33–52 (1959).
16. S. E. Ramberg and C. L. Bartholomew, 'Computer-based measurements of incipient wave breaking', G. A. Keramidas and C. A. Brebbia (eds.), *Computational Methods and Experimental Measurements*, Springer-Verlag, 1982, pp. 102–115.