

On Simulation of Turbulent Nonlinear Free-Surface Flows

Ben R. Hodges¹ and Robert L. Street

Environmental Fluid Mechanics Laboratory, Stanford University, Stanford, California

E-mail: street@cive.stanford.edu

Received September 18, 1997; revised September 9, 1998

A method for numerical simulation of the unsteady, three-dimensional, viscous Navier–Stokes equations for turbulent nonlinear free-surface flows is presented and applied to simulations of a laminar standing wave and turbulent open-channel flow with a finite-amplitude surface wave. The solution domain is discretized with a boundary-orthogonal curvilinear grid that moves with the free surface, allowing surface deformations to be smoothly resolved down to the numerical grid scale. The nonlinear kinematic and dynamic boundary conditions for boundary-orthogonal curvilinear coordinates are developed and discussed with a novel approach for advancing the free surface in curvilinear space. Dynamic large-eddy-simulation techniques are used to model subgrid scale turbulence effects. The method is shown to correctly produce the shape of a nonlinear free-surface wave and its decay due to viscosity. Application to finite-amplitude waves moving over a turbulent channel flow allows demonstration of the clear differences between a channel flow with and without waves, particularly the instantaneous turbulence structure. An interesting sidelight is the appearance of short-crested cross-channel surface waves caused by natural resonance. © 1999 Academic Press

Key Words: numerical simulation; free surface; open-channel flow; turbulence; nonlinear waves; dynamic boundary condition; kinematic boundary condition; Navier–Stokes equations.

1. INTRODUCTION

Free-surface flows with associated surface wave phenomena are ubiquitous in both engineering and geophysical applications. Until the seminal marker-and-cell (MAC) numerical simulations of Harlow and Welch [1], the understanding of the kinematic and dynamic

¹ Current address: Centre for Water Research, University of Western Australia, Nedlands, WA 6907, Australia.
E-mail: hodges@cw.r.uwa.edu.au.

effects of surface waves was limited to considerations of inviscid wave theory, viscous wave theory for small amplitude waves, and parameterizations of field data and laboratory experiments. In the past 30 years, numerical simulation methods (e.g., [2–4]) have proved valuable for investigating wave behavior. It is only recently that numerical methods have been applied to free surface flows with accurate resolution of viscous effects below significant surface deformations (e.g., [5–7]); however, these methods have thus far only been applied in the study of laminar vortex flows.

To date, the viscous free-surface numerical simulation methods presented in the literature have been limited by one or more of the following simplifications: (1) two space dimensions, (2) steady state, (3) small-amplitude waves, (4) laminar flow, (5) no boundary-layer resolution, (6) inability to handle steep waves, and (7) use of high-viscosity fluid (rather than water). In addition there exists a large body of literature that exploits the irrotational approximation for surface waves and solves the inviscid equation set to determine free-surface motion. The different methods and their limitations have been discussed in recent reviews of numerical methods for free-surface simulation [8, 9]. The methods used for turbulent free-surface simulation in the literature have been direct Navier–Stokes simulation (DNS) [10–12], applied to small free-surface motions, and Reynolds-averaged Navier–Stokes simulations (RANS), typically applied to steady flows (e.g., [13]) and more recently to unsteady flows (e.g., [14]). There does not appear to be any published use of large-eddy-simulation (LES) methods applied to free-surface flows prior to the present work. While DNS methods are *a priori* capable of resolving all the turbulent structure in a simulation, they have thus far been limited to simulation of small-amplitude surface deformations, generally with linearized free-surface boundary conditions. Steady-state RANS computations have been used extensively in the naval architecture community to model the waves developed by ship hull forms. The capability of unsteady RANS methods to resolve turbulent eddies near the free surface has not yet been demonstrated.

The lack of a comprehensive method with the capability of simulating unsteady, viscous, turbulent flows with finite-amplitude waves can be attributed to the problems noted by Sarpkaya [15], namely, that “...*the modeling of free-surface phenomena still poses difficulties, not only because of an insufficient understanding of the physics of the vorticity/free-surface interaction, but also because of the necessity to devise and use mathematical formulations, numerical schemes, and physical-property experiments of far greater complexity than had hitherto been used...*”

This provides the motivation for the present work: the development of a numerical method that is: (1) not limited by the simplifications used in prior free-surface simulation methods, (2) capable of simulating the physics of turbulent free-surface interactions, (3) straightforward in its implementation (using the least-complicated second-order numerical algorithms), and (4) formulated to handle steep or overturning waves. This last objective is the primary numerical challenge. This paper demonstrates the ability of the numerical method to simulate finite-amplitude surface waves in a formulation that is not limited to single-valued waves. The method can be extended to overturning waves with some caveats: (1) fine grid resolution must be applied to resolve the overturn in the curling portion of a wave—this implies the need for adaptive grid refinement; (2) the kinematic free surface boundary condition does not apply once the wave surface becomes multiply connected, and therefore, a new model is required for the physics of free-surface motion after the crest touches the underlying free surface; (3) the present grid generation method must be modified to smoothly handle a multiply connected material surface. Each of these issues is an

area requiring significant further research, for which the present work is a necessary first step.

The numerical approach in this paper is a free-surface/moving-grid adaptation of the method developed by Zang *et al.* [16] for internal-flow simulations. This finite-volume method employs second-order-accurate discretization in time and space of the primitive variables in curvilinear coordinates. Extensive simulation experience [17–21] has demonstrated the efficiency and accuracy of the code for computation of four-dimensional turbulent flow problems at the scales simulated herein. The Navier–Stokes equations are solved by a fractional-step method in conjunction with a multigrid solution of the pressure Poisson equation. The free-surface algorithm developed in this paper has been designed to work within the framework of Zang *et al.*'s method [16], but is general enough to be adapted to other methods. Our approach to simulating a viscous free-surface flow uses a boundary-fitted grid that moves with each time step to conform to the free surface [5, 6, 22]. This eliminates difficulties in treating the dynamic boundary condition at the free surface that occur in fixed-grid methods (e.g., [2, 23]). Computation of the free-surface motion is through an algorithm that decouples the grid generation and the flow solution, providing a flexible framework for modifying and adapting the method. Large-eddy-simulation techniques [18] are used to model subgrid-scale turbulence effects that cannot be adequately resolved on the computational grid. The curvilinear, boundary-fitted grid is generated with the Poisson equation method using an adaptation of the 3DGRAPE/AL code [24].

The following sections of this paper present the mathematical formulation of a viscous incompressible free-surface flow, a description of the numerical algorithm, validation of the moving grid and free-surface algorithms, and demonstration of the numerical method for the simulation of turbulent open-channel flow with finite-amplitude free-surface waves.

2. MATHEMATICAL FORMULATION

The flow of a fluid beneath a free surface is governed by the Navier–Stokes equations subject to conservation of mass in the fluid volume along with kinematic and dynamic boundary conditions at the free surface. This equation set is considerably more complex than the equation set for an internal flow due to the nonlinear effects of the boundary conditions and the temporal deformation of the domain boundary. The hyperbolic kinematic boundary condition that governs the evolution of the free surface is nonlinear in the velocity and spatial gradients of the surface. The dynamic boundary condition that enforces the free-surface stress condition has a nonlinear effect from its linkage to the nonlinear momentum equations through the surface pressure.

2.1. Navier–Stokes Equations

The spatially filtered, constant-density, incompressible Navier–Stokes equations in Cartesian (physical) space can be presented in a non-dimensional conservation-law form in terms of the Cartesian velocities u_i as

$$\frac{\partial \bar{u}_i}{\partial \tau} + \frac{\partial \bar{A}_{ij}}{\partial x_j} = \bar{B}_i \quad (1)$$

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0, \quad (2)$$

where the overbar represents a LES filter [25]. The non-dimensional momentum flux and source terms are

$$\bar{A}_{ij} = \bar{u}_j \bar{u}_i + \bar{p} \delta_{ij} - \text{Re}^{-1} \frac{\partial \bar{u}_i}{\partial x_j} + \tau_{ij} \quad (3)$$

$$\bar{B}_i = \frac{\partial \bar{\Psi}}{\partial x_i}. \quad (4)$$

All terms in Eqs. (1) through (4) are non-dimensionalized with a velocity scale (\mathcal{U}) and a length scale (\mathcal{L}). The Reynolds number is defined using the kinematic viscosity (ν) such that $\text{Re} = \mathcal{U}\mathcal{L}/\nu$. The non-dimensional modified pressure (p) is defined from the dimensional total pressure (P), density (ρ), and gravitational acceleration (g) as

$$p \equiv \frac{P}{\rho \mathcal{U}^2} + \frac{gx_3}{\mathcal{U}^2}. \quad (5)$$

The source term $\partial \Psi / \partial x_i$ represents a body force (e.g., the driving pressure gradient in an open-channel flow). The term τ_{ij} in the momentum flux is introduced to represent the additional subgrid-scale terms that arise due to the filtering of the nonlinear advection terms:

$$\tau_{ij} \equiv \overline{u_i u_j} - \bar{u}_i \bar{u}_j. \quad (6)$$

The subgrid-scale stress term contains both the interaction of subgrid scales with themselves and the interaction of the subgrid scales with the resolved scales. The turbulent flow simulations in this paper use the two-parameter dynamic model [18, 26], which was developed from the dynamic-mixed model [25]. The subgrid-scale stress is modeled by

$$\frac{\partial \tau_{ij}}{\partial x_j} = -\frac{\partial}{\partial x_j} \left(\text{Re}_T^{-1} \frac{\partial \bar{u}_i}{\partial x_j} \right) - \frac{\partial \text{Re}_T^{-1}}{\partial x_j} \frac{\partial \bar{u}_j}{\partial x_i} + C_r \frac{\partial L_{ij}^{(m)}}{\partial x_j}. \quad (7)$$

This model introduces three terms to the filtered Navier–Stokes equations: the eddy viscosity Reynolds number Re_T , the scale-similarity coefficient C_r , and the modified Leonard tensor $L_{ij}^{(m)}$. The eddy viscosity Reynolds number and scale-similarity coefficient are dynamically modeled, while the modified Leonard tensor is directly computed from the resolved flow field.

It is convenient to split the subgrid-scale stress so that part is contained in the momentum flux and part is held in the source term. This requires that the momentum equations be rewritten as

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{F}_{ij}}{\partial x_j} = \bar{S}_i, \quad (8)$$

where

$$\bar{F}_{ij} = \bar{u}_j \bar{u}_i + \bar{p} \delta_{ij} - (\text{Re}^{-1} + \text{Re}_T^{-1}) \frac{\partial \bar{u}_i}{\partial x_j} \quad (9)$$

$$\bar{S}_i = \frac{\partial \bar{\Psi}}{\partial x_i} + \frac{\partial \text{Re}_T^{-1}}{\partial x_j} \frac{\partial \bar{u}_j}{\partial x_i} - C_r \frac{\partial L_{ij}^{(m)}}{\partial x_j}. \quad (10)$$

2.1.1. *Curvilinear transformation of Navier–Stokes equations.* Solution of numerical problems in complicated domains using boundary-fitted curvilinear coordinates is now a standard technique requiring little introduction. Boundary-fitted curvilinear coordinate transformations for moving grids are derived using the chain rule for partial differential equations, resulting in [27]

$$\frac{\partial}{\partial x_j} = S_j^q \frac{\partial}{\partial \xi^q} \quad (11)$$

$$\frac{\partial}{\partial t} = \frac{\partial}{\partial \tau} - \dot{x}_j S_j^q \frac{\partial}{\partial \xi^q}, \quad (12)$$

where ξ^q with $q = 1, 2, 3$ are the computational space coordinates; $\partial/\partial t$ is a time derivative taken at a fixed point in physical space; $\partial/\partial \tau$ is a time derivative taken at a fixed point in computational space; repeated subscript/superscript combinations imply summation; and the surface metric tensor and grid velocity are defined as

$$S_j^q \equiv \frac{\partial \xi^q}{\partial x_j} \quad (13)$$

$$\dot{x} \equiv \frac{\partial x_j}{\partial \tau}. \quad (14)$$

Note that the former is only a correct tensor representation when x_j is a Cartesian coordinate system so that $x_j \equiv x^j$.

To simulate a flow with a free surface in boundary-fitted curvilinear coordinates, Eqs. (11) and (12) are used to transform the physical space Navier–Stokes equations into computational space. Completing the transformation requires the metric identity [27]

$$\frac{\partial}{\partial \xi^q} (J^{-1} S_i^q) \equiv 0 \quad (15)$$

along with the conservation of space [28]

$$\frac{\partial}{\partial \tau} (J^{-1}) - \frac{\partial}{\partial \xi^q} (J^{-1} S_j^q \dot{x}_j) \equiv 0. \quad (16)$$

Application of Eqs. (11) through (16) to Eqs. (2) and (8) through (10) provides the unsteady, incompressible, constant-density, filtered, non-dimensional Navier–Stokes equations in time-dependent boundary-fitted curvilinear coordinates as

$$\frac{\partial}{\partial \tau} (J^{-1} \bar{u}_i) + \frac{\partial}{\partial \xi^q} (J^{-1} \bar{\mathcal{F}}_i^q) = \bar{S}_i \quad (17)$$

$$\frac{\partial}{\partial \xi^q} (J^{-1} \bar{U}^q) = 0, \quad (18)$$

where the curvilinear momentum tensor $\bar{\mathcal{F}}_i^q$ and the curvilinear source vector \bar{S}_i are

$$\bar{\mathcal{F}}_i^q = (\bar{U}^q - \dot{X}^q) \bar{u}_i + S_i^q \bar{p} - (\text{Re}^{-1} + \text{Re}_T^{-1}) G^{qr} \frac{\partial \bar{u}_i}{\partial \xi^r} \quad (19)$$

$$\bar{S}_i = \frac{\partial}{\partial \xi^q} (J^{-1} S_i^q \bar{\Psi}) + J^{-1} S_j^q S_i^r \frac{\partial \text{Re}_T^{-1}}{\partial \xi^q} \frac{\partial \bar{u}_j}{\partial \xi^r} - C_r \frac{\partial}{\partial \xi^q} (J^{-1} S_j^q L_{ij}^{(m)}). \quad (20)$$

All physical space variables are non-dimensional, and curvilinear-space variables are defined as

$$\text{inverse Jacobian} \quad J^{-1} = \det \left| \frac{\partial x_i}{\partial \xi^s} \right| \quad (21)$$

$$\text{contravariant velocity} \quad \bar{U}^q = S_j^q \bar{u}_j \quad (22)$$

$$\text{contravariant grid velocity} \quad \dot{X}^q = S_j^q \dot{x}_j \quad (23)$$

$$\text{contravariant volume metrics} \quad G^{qr} = S_j^q S_j^r \quad (24)$$

$$\text{contravariant surface metrics} \quad S_i^q = \frac{\partial \xi^q}{\partial x_i}. \quad (25)$$

2.2. Kinematic Boundary Condition

2.2.1. Cartesian space. The kinematic boundary condition is the Lagrangian statement of a material surface which requires that a particle on the surface must remain on the surface. If $F = 0$ is a function that describes the location of the surface, then the kinematic condition requires that

$$\frac{\partial F}{\partial t} + \mathbf{u} \cdot \nabla F = 0. \quad (26)$$

It is possible to use directly the Lagrangian condition ($DF/Dt = 0$ or $Dx_i/Dt = u_i$) by moving unconstrained marker particles at the free surface; however, this method is unstable for long simulations where an explicit advance is used to integrate the free-surface position [29]. While the instability of the Lagrangian condition can be resolved by filtering [3], the present authors deemed it more valuable to develop an Eulerian free-surface boundary condition that is not limited to single-valued waves. If the kinematic boundary condition, grid generation, and Navier–Stokes equations are solved as an implicit, fully coupled set of equations, such instability should not occur with the Lagrangian boundary condition. Fully coupled methods have been used for laminar simulation in two space dimensions [5]; however, the complexities of the coupled approach seem to preclude its use with three space dimensions.

A physical-space Eulerian form of the kinematic boundary condition can be obtained through a Taylor-series expansion of the Lagrangian condition [30], resulting in

$$\frac{\partial H}{\partial t} = u_3 - u_1 \frac{\partial H}{\partial x} - u_2 \frac{\partial H}{\partial y}, \quad (27)$$

where H is the height (in the x_3 direction) of the free surface measured from some baseline in physical space. The advantage of this approach is that it can be numerically decoupled from the Navier–Stokes solution and grid generation without long-term instabilities arising in the simulation [29]. Unfortunately, this boundary condition is enforced on surface particles that are restricted to vertical motion in physical space and is therefore unsuited for overturning waves. Our objective is the development of a simulation method that is suitable for waves that are steep or overturning, so the physical-space Eulerian form of the kinematic boundary condition is not useful in the present context.

2.2.2. *Curvilinear-space kinematic boundary condition.* Curvilinear coordinate transformations, Eqs. (11) and (12), can be applied to the physical-space Eulerian kinematic boundary condition, Eq. (27), for use in numerical simulations with boundary-fitted curvilinear coordinates [22]. This approach retains the underlying vertical motion restriction on surface particles, making the method unsuitable for waves which do not remain single-valued. In some RANS simulations, a hybrid Cartesian/curvilinear approach has been applied [13, 14]. This approach uses curvilinear velocities for surface-tangential terms, while the Cartesian velocity is used for the curvilinear coordinates that varies across the surface:

$$\frac{\partial H}{\partial t} = u_3 - U^1 \frac{\partial H}{\partial \xi^1} - U^2 \frac{\partial H}{\partial \xi^2}. \quad (28)$$

Application is restricted to single-valued waves and is only a consistent boundary condition when the curvilinear coordinate that varies across the free surface is aligned with the vertical Cartesian axis. There remains a question as to the ability of the method to handle steep waves where u_3 is not a reasonable approximation of the surface-normal velocity. A more general approach was used by Hino [31]; it does not have a single-valuedness restriction in physical space, but requires deriving the Eulerian kinematic boundary condition directly in curvilinear coordinates. However, it was applied in a fixed curvilinear system rather than in a moving-grid system and appears to have been abandoned in the author's later work. The derivation of a fully curvilinear kinematic boundary condition is an extension of the derivation for the physical-space kinematic boundary condition. A brief derivation is presented here because it does not appear elsewhere in the literature.

To obtain directly a curvilinear Eulerian kinematic boundary condition, consider a *fixed* curvilinear space (ξ^1, ξ^2, ξ^3) such that the free surface is single-valued in ξ^3 . For the purposes of derivation with a fixed grid, the boundary-fitted restriction (used in numerical discretization, Section 3.1) is superfluous. Define F as a scalar function for the free surface such that

$$F(\boldsymbol{\xi}, t) = \xi^3 - \mathcal{H}(\xi^1, \xi^2, t) = 0, \quad (29)$$

where $\boldsymbol{\xi}$ is a vector representing the curvilinear coordinates of a surface position at time t , and \mathcal{H} is the height of the free surface measured from $\xi^3 = 0$ along a line of constant ξ^1 and ξ^2 in fixed curvilinear space. After some small time Δt , the free surface has moved, while the curvilinear coordinate system remains fixed. We require that Δt is small, so the free surface remains single-valued in ξ^3 . A Taylor-series expansion gives

$$F(\boldsymbol{\xi} + \mathbf{U}\Delta t, t + \Delta t) = F(\boldsymbol{\xi}, t) + \left(\frac{\partial F}{\partial t} + \mathbf{U} \cdot \nabla F \right) \Delta t + O(\Delta t)^2, \quad (30)$$

where \mathbf{U} is the contravariant velocity vector of a point on the surface. It follows that

$$\frac{\partial F}{\partial t} + \mathbf{U} \cdot \nabla F = 0. \quad (31)$$

Substitution of Eq. (29) into Eq. (31) provides the curvilinear kinematic boundary condition in *fixed* curvilinear coordinates as

$$\frac{\partial \mathcal{H}}{\partial t} = U^3 - U^1 \frac{\partial \mathcal{H}}{\partial \xi^1} - U^2 \frac{\partial \mathcal{H}}{\partial \xi^2}. \quad (32)$$

This allows computation of the evolution of the free surface with reference to any fixed curvilinear system in which the free surface is single-valued in one coordinate. As long as the kinematic free-surface condition is valid (i.e., singularities may not exist), a series of appropriate curvilinear systems can be defined which will be suitable for the evolution of an overturning wave.

2.2.3. Filtered curvilinear kinematic boundary condition. Because the kinematic boundary condition is inherently a nonlinear condition, the spatial filtering of the equation results in subgrid-scale terms. The filtered kinematic boundary condition can be presented as

$$\frac{\partial \bar{\mathcal{H}}}{\partial t} = \bar{U}^3 - \bar{U}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha} + \chi_\alpha^\alpha; \quad \alpha = 1, 2, \quad (33)$$

where the subgrid-scale terms (χ) are defined as

$$\chi_\alpha^\alpha \equiv \overline{U^\alpha \frac{\partial \mathcal{H}}{\partial \xi^\alpha}} - \bar{U}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha}. \quad (34)$$

The presence of nonlinear terms in the kinematic boundary condition provides the subgrid-scale term under either spatial or temporal filtering. The requirement for consistent averaging of the boundary conditions has been neglected in RANS simulations of free-surface flows in the literature and provides an additional challenge for turbulence closure schemes.

For LES closure, the velocity and surface height in the kinematic boundary condition are decomposed into resolved and unresolved parts so that

$$U^\alpha = \bar{U}^\alpha + u^\alpha \quad (35)$$

$$\mathcal{H} = \bar{\mathcal{H}} + h, \quad (36)$$

where the overbars indicate resolved terms, and the lowercase letters represent subgrid-scale terms (and should not be confused with Cartesian variables in this instance). It follows that

$$\chi_\alpha^\alpha \equiv \overline{(\bar{U}^\alpha + u^\alpha) \left(\frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha} + \frac{\partial h}{\partial \xi^\alpha} \right)} - \overline{(\bar{U}^\alpha + u^\alpha)} \left(\overline{\frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha}} + \overline{\frac{\partial h}{\partial \xi^\alpha}} \right). \quad (37)$$

Borrowing the modeling nomenclature for subgrid-scale velocities, we define the modified ‘‘Leonard,’’ ‘‘cross,’’ and ‘‘Reynolds’’ terms of the filtered kinematic boundary condition as

$$L_\alpha^\alpha \equiv \overline{\bar{U}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha}} - \bar{U}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha} \quad (38)$$

$$C_\alpha^\alpha \equiv \overline{\bar{U}^\alpha \frac{\partial h}{\partial \xi^\alpha} + u^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha}} - \left(\overline{\bar{U}^\alpha} \frac{\partial \bar{h}}{\partial \xi^\alpha} + \bar{u}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha} \right) \quad (39)$$

$$R_\alpha^\alpha \equiv \overline{u^\alpha \frac{\partial h}{\partial \xi^\alpha}} - \bar{u}^\alpha \frac{\partial \bar{h}}{\partial \xi^\alpha}, \quad (40)$$

where $\alpha = 1, 2$ and repeated subscript/superscript combinations imply summation.

For the kinematic boundary condition, the modified Leonard term L_α^α is made up of resolved quantities and can be computed explicitly using a method developed for subgrid-scale density effects in LES of stratified flows [25]. For the cross term and the Reynolds

term, new models are required. In the development of the dynamic mixed model for density variations, the subgrid-scale density terms were assumed to respond to the strain rates in a fashion similar to the subgrid-scale velocity terms. This allowed the development of a dynamic model where the subgrid-scale density variations are modeled with an eddy diffusivity term (similar to an eddy viscosity term) and a scale-similarity term. Both effects are assumed to be proportional to the magnitude of the resolved strain rate and are computed dynamically on the basis of a test-filter scheme and least-squares fit [25]. However, the free surface will not support a shear stress, and the basis of a kinematic boundary condition model solely on the irrotational strain rates may be questionable (especially in the near-surface region, where viscosity may be important). One can certainly make a scale-similarity argument that the cross terms should be proportional to the Leonard term, but there is an open question as to the appropriate constant of proportionality. Certainly when dealing with the small-scale kinematics of the free surface it would be wise to consider the dynamics of the flow and the dynamic boundary condition. For the small scales of motion near the surface, the effects of pressure, viscosity, and capillarity in the dynamic boundary condition may all be of similar orders of magnitude and are more likely to drive the kinematics of the subgrid-scale flow than is the resolved velocity field.

A simple approach suitable for initial investigations into LES modeling of the kinematic boundary condition is to compute directly the modified Leonard term using a test-filter scheme [25]. If the cross terms and Reynolds terms are neglected the kinematic boundary condition can be written as

$$\frac{\partial \bar{\mathcal{H}}}{\partial t} = \bar{U}^3 - \bar{U}^1 \frac{\partial \bar{\mathcal{H}}}{\partial \xi^1} - \bar{U}^2 \frac{\partial \bar{\mathcal{H}}}{\partial \xi^2} + \mathcal{L}_1(\bar{U}^1, \bar{\mathcal{H}}) + \mathcal{L}_2(\bar{U}^2, \bar{\mathcal{H}}), \quad (41)$$

where \mathcal{L} is defined as a Leonard stress operator

$$\mathcal{L}_\alpha(\bar{U}^\alpha, \bar{\mathcal{H}}) = \overline{\bar{U}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha}} - \bar{U}^\alpha \frac{\partial \bar{\mathcal{H}}}{\partial \xi^\alpha}, \quad (42)$$

which can be computed from grid-scale-resolved quantities.

The development, implementation, and testing of subgrid-scale models (such as that proposed above) is not the focus of this paper and remains an area of ongoing research. If we non-dimensionalize the kinematic boundary condition by the same length and velocity scales used in the Navier–Stokes equations, and neglect the subgrid-scale terms, the filtered, non-dimensional kinematic boundary is

$$\frac{\partial \bar{\mathcal{H}}}{\partial t} = \bar{U}^3 - \bar{U}^1 \frac{\partial \bar{\mathcal{H}}}{\partial \xi^1} - \bar{U}^2 \frac{\partial \bar{\mathcal{H}}}{\partial \xi^2}. \quad (43)$$

The development and testing of subgrid-scale models requires well-resolved DNS experiments of nonlinear free-surface flows to provide a sound basis for examining the physics near the surface. Such DNS simulations do not yet exist. Laboratory experiments could provide a basis for some testing of LES models at the free surface, but have two major drawbacks: (1) the laboratory data collection techniques need to be extended to three dimensions to obtain sufficient data to validate the three-dimensional terms in the LES models, and (2) the laboratory experiments necessarily have surface-tension effects which complicate model development and validation [32]. For simplicity, it is preferable to first develop

a model without surface-tension effects that is directly comparable to a DNS simulation that neglects surface tension.

2.3. Dynamic Boundary Condition

The dynamic boundary condition is generally obtained by assuming that (1) a free surface will only support the normal stress of a constant surface tension and (2) tangential stresses must disappear. The result is the dynamic boundary condition for an incompressible fluid in its classic form,

$$P_{s+} - P_{s-} = -2\mu e_{ij} n_i n_j + \gamma (R_1^{-1} + R_2^{-1}) \quad (44)$$

$$e_{ij} t_i n_j = 0, \quad (45)$$

where P is the total pressure, the subscripts $s+$ and $s-$ indicate the pressure on the upper and lower sides of the free surface, n_i and t_i are the unit normal and tangent vectors, γ is the surface tension coefficient, R_1 and R_2 are the principal radii of curvature of the surface, μ is the dynamic viscosity, and e_{ij} is the rate-of-strain tensor defined as

$$e_{ij} \equiv \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \quad (46)$$

For most purposes, this form of the dynamic boundary condition is adequate and is often approximated as simply $P = 0$. Equations (44) and (45) do not provide for straightforward implementation in a boundary-fitted curvilinear coordinate numerical method; therefore, our approach will begin with the tensor form of the dynamic boundary condition in general curvilinear coordinates [33]. By applying the assumptions used to get Eqs. (44) and (45) along with the requirement that the curvilinear coordinate system be boundary orthogonal, the *dimensional* dynamic boundary condition can be presented as

$$(P_{s+} - P_{s-}) = -2\mu U_{,3}^3 + 2M\gamma \quad (47)$$

$$U_{,3}^1 = -G_{33} \{ G^{11} U_{,1}^3 + G^{12} U_{,2}^3 \} \quad (48)$$

$$U_{,3}^2 = -G_{33} \{ G^{22} U_{,2}^3 + G^{12} U_{,1}^3 \}, \quad (49)$$

where G_{33} is the covariant metric

$$G_{33} = \frac{\partial x_j}{\partial \xi^3} \frac{\partial x_j}{\partial \xi^3} \quad (50)$$

and M is the mean curvature, defined for a ξ^3 surface in a boundary-orthogonal coordinate system as

$$M = \frac{1}{2} \frac{G^{\alpha\beta}}{G^{33}} \left(S_j^3 \frac{\partial^2 x_j}{\partial \xi^\alpha \partial \xi^\beta} \right), \quad (51)$$

with α and β summed over (1, 2) and j summed over (1, 2, 3). The curvilinear set of equations is considerably simplified through the assumption of boundary orthogonality. If this assumption is not applied, terms with G^{13} and G^{23} metric coefficients will occur in the dynamic boundary condition.

If we let the (dimensional) outside pressure (P_{s+}) equal zero and apply Eq. (5) for the non-dimensional reduced pressure and non-dimensionalize all other terms by the length scale \mathcal{L} and the velocity scale \mathcal{U} , the normal component of the dynamic boundary condition can be written as

$$p_{s-} = \frac{z_{s-}}{(\text{Fr})^2} + \frac{2}{\text{Re}} U_{s-}^3 - \frac{2}{\text{We}} M, \quad (52)$$

where z_{s-} is the (non-dimensional) value of the vertical Cartesian coordinate (x_3) at the surface and definitions for the Froude and Weber numbers are

$$\text{Fr} \equiv \frac{\mathcal{U}}{\sqrt{g\mathcal{L}}} \quad (53)$$

$$\text{We} \equiv \frac{\rho\mathcal{U}^2\mathcal{L}}{\gamma}. \quad (54)$$

Note that the differentiation in Eqs. (47) through (52) is covariant tensor differentiation and requires the application of Christoffel symbols for deriving a discrete implementation. Applying some algebra and tensor manipulation, we reduce the dynamic boundary condition of Eqs. (48), (49), and (52) to a form that can be more readily implemented in a numerical method. For the present second-order method, metrics can be assumed to pass through the filter operation [34] so the full dynamic boundary condition can be written as

$$\bar{p}_{s-} = \frac{\bar{z}_{s-}}{(\text{Fr})^2} - \frac{2M}{\text{We}} + \frac{1}{\text{Re}} \left\{ 2 \frac{\partial \bar{U}^3}{\partial \xi^3} + G^{33} \left[\bar{U}^1 \frac{\partial}{\partial \xi^1} (G_{33}) + \bar{U}^2 \frac{\partial}{\partial \xi^2} (G_{33}) \right] \right\} \quad (55)$$

$$\frac{\partial \bar{U}^1}{\partial \xi^3} = -G_{33} \left\{ G^{11} \frac{\partial \bar{U}^3}{\partial \xi^1} + G^{12} \frac{\partial \bar{U}^3}{\partial \xi^2} \right\} - \bar{U}^3 \left\{ G^{11} \frac{\partial G_{13}}{\partial \xi^3} + G^{12} \frac{\partial G_{23}}{\partial \xi^3} \right\} \quad (56)$$

$$\frac{\partial \bar{U}^2}{\partial \xi^3} = -G_{33} \left\{ G^{22} \frac{\partial \bar{U}^3}{\partial \xi^2} + G^{12} \frac{\partial \bar{U}^3}{\partial \xi^1} \right\} - \bar{U}^3 \left\{ G^{22} \frac{\partial G_{23}}{\partial \xi^3} + G^{12} \frac{\partial G_{13}}{\partial \xi^3} \right\}. \quad (57)$$

If our grid is sufficiently fine and the surface is sufficiently smooth, then the tangential and normal derivatives of the metrics can be neglected in the above equation set. However, if the free surface has steep surface oscillations on the order of the grid scale or slightly larger, then the metric gradient terms cannot be neglected *a priori*.

3. NUMERICAL METHOD

The numerical simulation method is a time integration of Eqs. (17) through (20) subject to the free-surface boundary conditions, Eqs. (43), (55), (56), (57), and the subgrid-scale closure model, Eq. (7). The approach uses a finite-volume, fractional-step, pressure-Poisson integration of the unsteady Navier–Stokes equations with curvilinear grid generation using a Poisson equation method.

3.1. Free-Surface Advance

The first step of the numerical method is to advance the free-surface position from time (n) to time ($n + 1$). The method used is an uncoupled-grid approach where the kinematic

boundary condition is integrated forward in time without implicit reference to the time $(n + 1)$ velocities. This requires that at time (n) there is a fixed curvilinear grid that is boundary-fitted in the ξ^3 coordinate. All three components of the contravariant velocity must be known at each grid point on the free surface. To find the change in the ξ^3 coordinates of the free surface at some small time later with reference to the same fixed curvilinear grid (see Eq. (58) ff.), a Runge–Kutta fourth-order (RK4) method is applied with a fifth-order upwind discretization for spatial derivatives [12]. After computing the free-surface advance in curvilinear space, the new curvilinear coordinates are converted back to physical space coordinates for use in calculating the new computational mesh for the $(n + 1)$ time step. With this approach, the coordinates have a tendency to drift in the streamwise direction unless restrained or redistributed. Rather than an artificial restraint being used, the grid points are allowed to move in accordance with the kinematic boundary condition, then redistributed on the free surface using a two-dimensional cubic spline. This maintains a smooth and even distribution of points on the surface in physical space.

The advantage of our free-surface method is that the kinematic boundary condition is enforced upon points which move along lines of constant ξ^1 and ξ^2 curvilinear coordinates rather than lines of constant x and y physical coordinates. Thus, the free surface motion is computed along lines that are locally orthogonal to the time n grid rather than at an angle that depends upon the steepness of the wave. As a result, the Eulerian formulation requirement of a single-valued grid in physical space is replaced by a single-valuedness requirement in curvilinear space. This is a less restrictive condition for a boundary-fitted coordinate system, requiring only that the slope of the free surface be continuous. A discontinuous surface slope implies wave breaking and a violation of the material condition of the free surface, so it can be said that the curvilinear form is generally valid and can be implemented numerically as long as the kinematic boundary condition itself is valid. The movement of the points along surface normal lines provides a simpler implementation than the curvilinear transformations of the physical space kinematic boundary condition used in the literature [22, 35].

The ability of present method to simulate near-breaking and overturning waves is primarily a matter of the availability of computational power. The number of grid points required for accurate simulation of a wave shape increases as a wave steepens and overturns. Increasing the number of grid points along the surface affects the time step required to avoid numerical instabilities at the surface. The study of near-breaking phenomena and demonstration of the numerical method for this type of problem remain subjects for future research.

To advance the free surface from time (n) to time $(n + 1)$, we consider the time (n) curvilinear grid to be *fixed* with respect to time and require that it be boundary-fitted to the time (n) free surface. Thus, at the surface,

$$\mathcal{H}^{[n]} = \xi^{3[n]} \Big|_{\text{surface}} = \text{constant}. \quad (58)$$

The gradients of the time (n) free-surface height relative to the ξ^1 and ξ^2 curvilinear coordinates will disappear, since

$$\frac{\delta \mathcal{H}^{[n]}}{\delta \xi^q} = \frac{\delta \xi^{3[n]}}{\delta \xi^q} = 0: \quad q = 1, 2. \quad (59)$$

The resulting RK4 discrete system for the kinematic boundary condition is

$$k = \Delta t U^3 \quad (60)$$

$$\bar{k} = \Delta t \left\{ U^3 - \frac{U^1}{2} \frac{\delta k}{\delta \xi^1} - \frac{U^2}{2} \frac{\delta k}{\delta \xi^2} \right\} \quad (61)$$

$$\tilde{k} = \Delta t \left\{ U^3 - \frac{U^1}{2} \frac{\delta \bar{k}}{\delta \xi^1} - \frac{U^2}{2} \frac{\delta \bar{k}}{\delta \xi^2} \right\} \quad (62)$$

$$\hat{k} = \Delta t \left\{ U^3 - U^1 \frac{\delta \tilde{k}}{\delta \xi^1} - U^2 \frac{\delta \tilde{k}}{\delta \xi^2} \right\} \quad (63)$$

$$\mathcal{H}^{n+1} = \mathcal{H}^n + \frac{1}{6}(k + 2\bar{k} + 2\tilde{k} + \hat{k}), \quad (64)$$

where a spatial derivative at location i is discretized as

$$\left(U \frac{\delta k}{\delta \xi} \right)_i = \frac{U_i}{60} \{k_{i+3} - 9k_{i+2} + 45(k_{i+1} - k_{i-1}) + 9k_{i-2} - k_{i-3}\} \\ + \frac{|U_i|}{60} \{k_{i+3} - 6k_{i+2} + 15k_{i+1} - 20k_i + 15k_{i-1} - 6k_{i-2} + k_{i-3}\}. \quad (65)$$

To obtain the time $(n+1)$ physical space position of a particle on the surface after numerical solution of Eqs. (60) through (64), we note that

$$\Delta x_i = \Delta \xi^3 \frac{\delta x_i}{\partial \xi^3}: \quad i = 1, 3, \quad (66)$$

which can be discretized as

$$x_i^{n+1} = x_i^n + (\mathcal{H}^{n+1} - \mathcal{H}^n) \left(\frac{\delta x_i}{\delta \xi^3} \right)^n. \quad (67)$$

This system provides a method for explicitly updating the physical-space free-surface position from time (n) to $(n+1)$.

3.2. Grid Generation

The numerical method requires the computation of a new curvilinear grid at each time step. To accurately resolve the near-surface boundary layer under steep or overturning waves requires a grid-generation method that (1) generates boundary-orthogonal grids, (2) minimizes grid skewness, (3) allows control of grid stretching, (4) is computationally efficient in vector or parallel implementation, (5) uses a modest amount of computer memory, (6) does not require disk access during computation, and (7) does not require user input during the grid solution process. Grid generation is a sophisticated discipline with a variety of available solution techniques that satisfy the above criteria to some extent [36]. For our purposes, the most suitable technique is the Poisson grid generation method [27] using the 3DGRAPE/AL code [24]. This uses standard techniques of iterative control function adjustment to obtain a boundary-orthogonal grid and user-specified grid distributions.

3.3. Navier–Stokes Solution

Our numerical approach to solving the Navier–Stokes equations follows the method developed by Zang *et al.* [16] that descends from the methods of Kim and Moin [37] and

Harlow and Welch [1]. To discretize the momentum equation we apply the explicit second-order Adams–Bashforth (AB2) algorithm to the convective terms and the off-diagonal viscous terms, with the implicit Crank–Nicolson second-order (CN2) scheme for the diagonal viscous terms. The use of a moving grid requires a convective grid-flux term to account for the convective motion of the grid. This term is discretized with a second-order approximation using the volume flux of the grid between the time (n) and ($n + 1$) physical space positions for each cell face and the time (n) velocity. The pressure is removed from the momentum equation in the predictor stage of the fractional-step method and a numerical pressure variable (ϕ) is defined and computed in the solution of a Poisson equation. Second-order-accurate approximate factorization is used on the left-hand side of the discretized momentum equation for increased computational efficiency. For clarity, we drop the overbar notation for filtered variables in discrete equations and use δ instead of ∂ to indicate discrete derivatives; the resulting system can be presented as

1. predictor step:

$$(I - D_1^{n+1})(I - D_2^{n+1})(I - D_3^{n+1})(u_i^* - u_i^n) = S_i; \quad (68)$$

2. pressure-Poisson equation:

$$\frac{\delta}{\delta\xi^q} \left(J^{-1} G^{qr} \frac{\delta\phi}{\delta\xi^r} \right)^{n+1} = \frac{1}{\Delta t} \frac{\delta}{\delta\xi^q} (J^{-1} U^{*q}); \quad (69)$$

3. corrector steps:

(a) for the Cartesian velocity (on cell centers):

$$u_i^{n+1} = u_i^* + \left[\frac{\Delta t}{J^{-1}} B_i(\phi) \right]^{n+1}; \quad (70)$$

(b) for the normal component of contravariant velocity (on cell faces):

$$(J^{-1} U^q)^{n+1} = J^{-1} U^{*q} - \Delta t \left(J^{-1} G^{qr} \frac{\delta\phi}{\delta\xi^r} \right)^{n+1}. \quad (71)$$

The pressure variable (ϕ) is related to the reduced pressure (p) by

$$B_i(p) = \left[J^{-1} - \frac{\Delta t}{2} D_I \right] \frac{B_i(\phi)}{J^{-1}} \quad (72)$$

and the source term of the predictor (Eq. [68]) is

$$S_i = \frac{\Delta t}{(J^{-1})^{n+1}} \left\{ \frac{3}{2} (C_i^n + D_E^n [u_i^n]) - \frac{1}{2} (C_i^{n-1} + D_E^{n-1} [u_i^{n-1}]) \right. \\ \left. + \frac{1}{2} (D_I^n [u_i^n] + D_I^{n+1} [u_i^n]) + Q_i \right\} + \left\{ \frac{(J^{-1})^n}{(J^{-1})^{n+1}} - 1 \right\} u_i^n. \quad (73)$$

The use of time ($n + 1$) metric terms in the source of the predictor, Eq. (73), is allowable as our numerical method integrates the kinematic boundary condition for the time ($n + 1$)

free-surface position and computes the curvilinear grid prior to the solution of the predictor step. The last term in Eq. (73) is applied so that the velocity u_i^n on the left-hand side of the predictor, Eq. (68), is multiplied by the time $(n + 1)$ inverse Jacobian during the derivation of the discrete equations. This prevents the appearance of the time (n) Jacobian in the corrector step and the pressure-Poisson equation, and provides for a simpler implementation.

Discrete operators from Eq. (68) through (73) are defined as

$$D_\alpha(\cdot) = \frac{\Delta t}{2J^{-1}} \frac{\delta}{\delta \xi^\alpha} \left\{ v J^{-1} G^{\alpha\alpha} \frac{\delta}{\delta \xi^\alpha} (\cdot) \right\}, \quad (74)$$

where $\alpha = 1, 2, 3$ with no summation:

$$D_E(\cdot) = \frac{\delta}{\delta \xi^q} \left\{ v J^{-1} G^{qr} \frac{\delta}{\delta \xi^r} (\cdot) \right\}_{q \neq r} \quad (75)$$

$$D_I(\cdot) = \frac{\delta}{\delta \xi^q} \left\{ v J^{-1} G^{qr} \frac{\delta}{\delta \xi^r} (\cdot) \right\}_{q=r} \quad (76)$$

$$C_i = -\frac{\delta}{\delta \xi^q} \{ J^{-1} U^q u_i \} \quad (77)$$

$$B_i(\cdot) = -\frac{\delta}{\delta \xi^q} \{ J^{-1} S_i^q(\cdot) \} \quad (78)$$

$$Q_i = \frac{\delta}{\delta \xi^q} \{ (J^{-1} \dot{X}^q)^{n+1/2} u_i^n \}. \quad (79)$$

In the operator Q_i , we compute $(J^{-1} \dot{X}^q)^{n+1/2}$ as the volume swept out by the q side of a cell as the grid moves from the time (n) to the time $(n + 1)$ positions.

The conservation of space, Eq. (16) can be written in a discrete form as

$$(J^{-1})^{n+1} = (J^{-1})^{n+1} + \frac{\delta}{\delta \xi^q} (J^{-1} \dot{X}^q)^{n+1/2}. \quad (80)$$

The conservation of space must be used to compute the new inverse Jacobian at each time step [28]. To prevent numerical inconsistency, the term $J^{-1} \dot{X}^q$ must be numerically identical in both the implementation of the conservation of space, Eq. (80), and in the discrete grid motion term Q_i , in Eq. (79).

3.4. Dynamic Boundary Conditions at the Free Surface

3.4.1. Tangential components. The tangential components of the dynamic boundary condition are used to obtain the tangential velocities on (1) the free surface (for use in integrating the kinematic boundary condition), and (2) the numerical ghost points outside the free surface (for use in the boundary conditions on the predictor step of the solution method). We experimented with linear, quadratic, and cubic implementations of the boundary conditions and found the simple linear approach worked best when the boundary layer was well resolved. The linear approach can be presented as

$$U_{\text{surf}}^\alpha = U_{\text{surf}-1/2}^\alpha + \frac{1}{2} \frac{\delta U^\alpha}{\delta \xi^3} \Big|_{\text{surf}} : \quad \alpha = 1, 2, \quad (81)$$

where the subscript notation (surf) indicates the value at the free surface and the notation (surf - 1/2) indicates the value at the center of a cell face for the first cell inside the surface. The gradient of U^α across the boundary is found from a discrete implementation (using central differences) of the tangential dynamic boundary condition, Eqs. (56) and (57). The U^3 component at the center of the cell face on the free surface is computed directly in the corrector step, Eq. (71), from the computed pressure field and the U^{*3} value at the surface.

Computation of contravariant velocities on the faces and centers of the ghost cells outside the free surface is accomplished in a similar fashion. Once the contravariant velocities are computed, the three components of the Cartesian velocity must be calculated at the centers of the ghost cells outside the free surface. These values provide part of the boundary condition for the u^* estimated velocity computation [37]. The Cartesian velocities for each ghost point are obtained by inverting a 3×3 matrix, Eq. (22), which relates the curvilinear and Cartesian velocities.

3.4.2. Normal component. The normal component of the dynamic boundary condition, Eq. (55), is discretized using central difference operators. This provides a Dirichlet condition on the modified physical pressure (p). A subtle point that is overlooked in some of the literature is that the numerical pressure variable (ϕ) is an approximation of the physical pressure that may not be of the same order of accuracy as the solution method. The relationship between the numerical and physical pressure variables is a function of the discretization method, and is given by Eq. (72) for the present approach. In general, it is not mathematically rigorous to simply substitute ϕ for p in the dynamic boundary condition (or any other pressure computation). However, our experience has been that it is in keeping with the order of accuracy of the simulation method for the flows investigated. In test simulations we computed the difference between the right-hand and left-hand sides of the p/ϕ relation, Eq. (72), with the result that the difference was always of the order Δt^2 or smaller. This conclusion was also reached by R. Calhoun (Personal communication, 1996), who conducted a more detailed analysis by numerically integrating the p/ϕ relation in a simulation of flow over a wavy boundary with turbulent separation.

The normal component of the dynamic boundary condition is used as a Dirichlet boundary condition on the pressure in the solution of the Poisson pressure equation. To be numerically consistent in the discretization, all the terms in the normal component of the dynamic boundary condition, Eq. (55), should be time $(n + 1)$ values. However, unlike a no-slip boundary, the velocity on the free surface at time $(n + 1)$ is not known *a priori*. A precise discretization would involve substituting the contravariant corrector step, Eq. (71), into the dynamic boundary condition to change the U^{n+1} values into U^* values plus ϕ^{n+1} gradients. This boundary condition includes both ϕ and second derivatives of ϕ , changing the form of the boundary condition and making implementation significantly more complicated (especially for use with a multigrid solver).

Two approaches can be used to obtain second-order accurate discretizations of the dynamic boundary condition, Eq. (55), in terms of time (n) variables while retaining a simple Dirichlet pressure boundary form. The contravariant corrector step, Eq. (71), and a Taylor series expansion for the pressure can be used to write

$$(U^q)^{n+1} = U^{*q} - \Delta t \left(\frac{G^{qr}}{J^{-1}} \frac{\delta \phi}{\delta \xi^r} \right)^n + O(\Delta t)^2. \quad (82)$$

Substituting Eq. (82) into a discretization of Eq. (55) provides a Dirichlet boundary condition on the pressure in terms of the U^* velocities and second derivatives of the time (n) pressure variable. This has the disadvantage of feeding numerical errors in the time (n) pressure computation back into the computation of the time ($n + 1$) pressure boundary condition. Such feedback can induce undesirable numerical oscillations in the pressure field.

The approach used in the present work does not have feedback of the time (n) pressures into the boundary condition. This requires a Taylor-series expansion for each velocity term on the right-hand side of Eq. (55), so that the resulting dynamic boundary condition is

$$\begin{aligned} \phi_{S-}^{n+1} &= \frac{z_{s-}}{(\text{Fr})^2} + 2M^{[n+1]}\gamma \\ &+ \frac{1}{\text{Re}} \left\{ 2 \frac{\delta U^3^{[n]}}{\delta \xi^3} + \frac{G^{33}}{J^{-1}} \left[U^1 \frac{\delta}{\delta \xi^1} (J^{-1} G_{33}) + U^2 \frac{\delta}{\delta \xi^2} (J^{-1} G_{33}) \right]^{[n]} \right\} \\ &+ \frac{\Delta t}{\text{Re}} \left\{ 2 \frac{\delta}{\delta \xi^3} (U^3^{[n]} - U^3^{[n-1]}) + \frac{G^{33}}{J^{-1}} (U^1^{[n]} - U^1^{[n-1]}) \frac{\delta}{\delta \xi^1} (J^{-1} G_{33})^{[n]} \right. \\ &\left. + \frac{G^{33}}{J^{-1}} (U^2^{[n]} - U^2^{[n-1]}) \frac{\delta}{\delta \xi^2} (J^{-1} G_{33})^{[n]} \right\} + O(\Delta t)^2. \end{aligned} \quad (83)$$

In our simulations, $1/\text{Re} \leq O(\Delta t)$; so without loss of accuracy, we can neglect terms of order $(\Delta t/\text{Re})$ in our second-order method. This is an improvement over the $P = 0$ boundary condition (used in many free-surface simulations) that generally is $O(1/\text{Re})$ or $O(\Delta t)$ accurate. Using our approach, the discrete, surface-normal, dynamic boundary condition becomes

$$\begin{aligned} \phi_{S-}^{n+1} &= \frac{z_{s-}}{(\text{Fr})^2} + 2M^{[n+1]}\gamma \\ &+ \frac{1}{\text{Re}} \left\{ 2 \frac{\delta U^3^{[n]}}{\delta \xi^3} + \frac{G^{33}}{J^{-1}} \left[U^1 \frac{\delta}{\delta \xi^1} (J^{-1} G_{33}) + U^2 \frac{\delta}{\delta \xi^2} (J^{-1} G_{33}) \right]^{[n]} \right\}. \end{aligned} \quad (84)$$

Implementation of the Dirichlet pressure boundary condition in the multigrid solver is accomplished using a linear approach that is consistent with the linear prolongation/restriction operators of the multigrid method [17]. The velocities on boundaries other than the free surface are known *a priori* so pressure boundary conditions are only required when a grid is not boundary-orthogonal. To implement the pressure boundary condition on the free surface, we first obtain the estimated contravariant velocity normal to the free surface (U^{*3}) using linear interpolation from interior and ghost point u_i^* velocities and the computed boundary metrics. This provides the U^* on the boundary needed for the pressure-Poisson equation source term (see Eq. [69]). The primary difficulty in implementing the pressure boundary condition is that the boundary condition is defined on the edges of computational cells while the pressures in the interior are defined at the centers of computational cells. Our implementation computes the ghost point pressure using linear extrapolation from the center of the first cell inside the boundary and the boundary pressure. The ghost point pressure appears in the discrete stencil for the left-hand-side of the pressure-Poisson equation (69) and thus provides for efficient implementation in the multigrid solver.

3.5. Summary of the Numerical Method

1. Use RK4 and fifth-order upwind discretization of the kinematic boundary condition, Eqs. (60) through (67), to advance the free surface from time (n) to time ($n + 1$).

2. Compute a boundary-orthogonal grid and metrics for time ($n + 1$) grid using the Poisson equation method.

3. Use conservation of space to compute new Jacobians of grid cells.

4. Solve for u_i^* at center of cells using AB2 with quadratic upwind interpolation [38] discretization for convective terms, CN2 for diagonal viscous terms, and approximate factorization for the implicit solution. This applies a vectorized tridiagonal solver to Eq. (68).

5. Use linear interpolation to obtain the normal component of U^* on each cell face.

6. Solve the Poisson pressure equation for the pressure variable ϕ using a 3D vectorized multigrid solver [16]. The normal component of the dynamic boundary condition is used to provide a Dirichlet pressure boundary condition on the free surface. A pressure boundary on the bottom (Dirichlet boundary) is not required since the grid is boundary orthogonal. As demonstrated by Zang *et al.* [16], under this condition only a zero contravariant velocity normal to the boundary is required to make the Poisson equation for the pressure well posed.

7. Using the Cartesian corrector step, Eq. (70), compute the time ($n + 1$) Cartesian velocity (u_i) at cell centers.

8. Using the contravariant corrector step, Eq. (71), compute the time ($n + 1$) contravariant velocity components normal to cell surfaces, U^q .

9. Using the tangential components of the dynamic boundary condition, Eqs. (56) and (57) along with Eq. (81), compute the tangential components of contravariant velocity U^q on free surface and ghost points. This provides the velocities needed to advance the free surface in the next time step. Transform the contravariant velocities into Cartesian velocities for use in the u^* boundary condition in the next time step.

4. NUMERICAL EXPERIMENTS FOR CODE VALIDATION

4.1. Decaying Vortex with a Moving Grid

The use of second-order-accurate discretizations does not guarantee second-order accuracy in a numerical simulation. This is especially true with boundary-fitted curvilinear coordinates and moving grids. Because of the complexity of computational code required for a curvilinear, moving-grid simulation of the Navier–Stokes equations, there is always a chance of error either in derivation of the transformation or in implementation of the discretized forms. The fractional-step method used in this work was demonstrated to provide second-order spatial accuracy for Cartesian grids in the work of Kim and Moin [37] through grid-refinement tests of a decaying vortex. The decaying vortex is an analytical solution of the two-dimensional Navier–Stokes equations over the domain of ($0 \leq x_1, x_2 \leq \pi$) that can be written as

$$u_1 = -\cos(x_1) \sin(x_2) e^{-2t} \quad (85)$$

$$u_2 = \sin(x_1) \cos(x_2) e^{-2t} \quad (86)$$

$$p = -0.25\{\cos(2x_1) + \cos(2x_2)\} e^{-4t}. \quad (87)$$

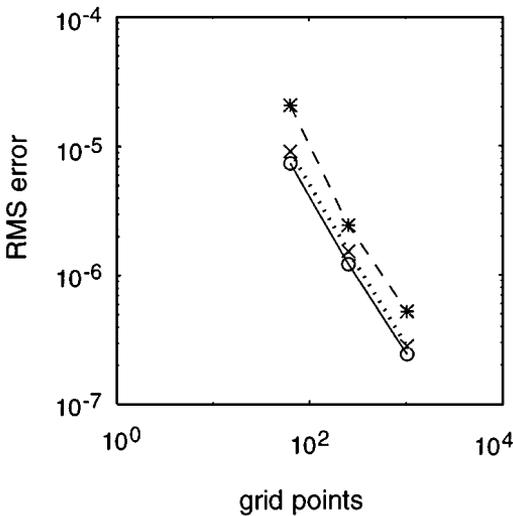


FIG. 1. Decaying vortex accuracy: (—○—) fixed grid; (· · × · ·) translating grid; (- * -) stretching grid.

In Zang *et al.* [16] the curvilinear coordinate adaptation of Kim and Moin's fractional-step method was shown to provide second-order spatial accuracy using the same set of tests. To provide continuity with the previous works, we have conducted similar tests to demonstrate that the moving-grid algorithm and its implementation in the code result in second-order spatial accuracy for both fixed and moving grids.

Figure 1 provides simulation results showing the reduction of the RMS velocity error caused by increasing the number of grid points. The time step is reduced as the grid is refined to maintain a constant maximum CFL condition. Three different error lines are shown. The first error line represents the results for a fixed grid, the second is for a grid that is fixed in shape but translates through the decaying vortex domain, and the third is for a grid that has boundaries which remain fixed but whose interior grid lines are stretched with each time step. It can be seen that accuracy is approximately second-order in all these cases.

4.2. Monochromatic Standing Waves

Unfortunately, there does not appear to be a simple analytical solution of the Navier–Stokes equations with a free surface that could be used to compute the accuracy of the implementation of the full free-surface algorithm. However, there exist approximate solutions for laminar, monochromatic standing waves in an irrotational flow field that can be used to validate the kinematics of the free surface. The rotational effects due to the free surface motion are small and confined to a thin free-surface boundary layer; thus, we should be able obtain a viscous solution of the Navier–Stokes equations that results in the wave form attributable to the irrotational flow field. As a check on the dynamics of the free-surface solution, the viscous damping of the wave caused by the free-surface boundary layer can be approximated (to the first order) from energy arguments [39].

We performed simulations of standing waves in a two-dimensional rectangular basin with free-ship boundary conditions on the sides and bottom of a 32×32 cell domain. The domain length and still-water depth were one-half the wavelength of the primary standing wave. The initial wave slope (ϵ) in the simulation is defined from the wave amplitude (a)

and the wave number (k) as

$$\epsilon \equiv ak. \quad (88)$$

Simulations were conducted for small-amplitude waves with $\epsilon = 0.03$ and finite-amplitude waves with $\epsilon = 0.3$. A standing wave begins to break [40] at $\epsilon = 0.32$, so the latter test is a severe test of the ability of the free-surface algorithm to correctly maintain proper kinematics. Simulations were conducted with wave Reynolds numbers ranging from 50 to 5000. The wave Reynolds number is defined as

$$\text{Re}_w = \sigma / \nu k^2, \quad (89)$$

where σ is the radian frequency. The wave Froude number

$$\text{Fr}_w = \sigma / \sqrt{gk} = \tanh kD, \quad (90)$$

where D is the depth, was fixed at unity (i.e., deep-water waves). Uniform grid distributions were used in the horizontal and vertical directions to allow the largest possible time step. For the finite-amplitude wave, simulations up to $\text{Re}_w = 500$ were conducted with the uniform grid. This was the largest Reynolds number possible while maintaining five uniformly distributed grid cells in the free-surface boundary layer. Simulations were successfully conducted at higher Reynolds numbers using grid stretching to obtain resolution of the boundary layer. However, the stretched grid tests are not necessarily good indicators of the system performance since the fine grid resolution requires a small time step, allowing accurate results to be obtained without adequately testing the robustness of the method.

According to linear theory for small amplitude waves, the wave shape should be a sinusoid, where the surface height (η) above the still-water level is

$$\eta(x, t) = a \sin(kx) \sin(\sigma t). \quad (91)$$

Nonlinear, second-order theory for finite-amplitude standing waves predicts a wave shape given by [40]

$$\begin{aligned} \eta(x, t) = & a \sin(kx) \sin(\sigma t) - \frac{1}{2} a \epsilon \coth(kD) \cos(2kx) \\ & \times \left\{ \sin^2(\sigma t) - \frac{3 \cos(2\sigma t) + \tanh^2(kD)}{4 \sinh^2(kD)} \right\}. \end{aligned} \quad (92)$$

A comparison of the free-surface wave shape against linear and nonlinear theory for a finite-amplitude standing wave with an ϵ of 0.3 at a Reynolds number of 500 is shown in Fig. 2. It can be seen that the agreement of the surface shape with nonlinear theory is quite good. This correspondence was achieved regardless of whether the initial wave shape of the simulation was based on linear or nonlinear theory.

The damping of a free wave due to viscosity as a function of time can be approximated from [39]

$$a(t) = a(0) e^{-2\nu k^2 t}. \quad (93)$$

This is based upon an energy dissipation argument for linear waves in deep water, so we can only expect this to provide a rough guide to damping since the simulation waves are

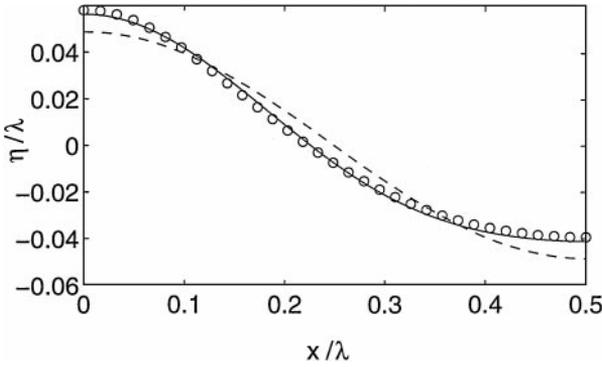


FIG. 2. Free-surface wave shape: η is surface displacement, λ is wavelength, x is domain length; (o) simulation; (—) nonlinear theory; (---) linear theory.

nonlinear and at intermediate depth. For a wave ϵ of 0.3 at a Reynolds number of 50, Fig. 3 shows the evolution of the wave wall height over time. The wave damping is in reasonable agreement with the theory. A comprehensive review of the results of monochromatic wave simulations are found in Hodges *et al.* [41] and Hodges [42]. Validation of the numerical method for 3D standing waves can also be found in Hodges and Street [43].

5. SIMULATION OF FINITE-AMPLITUDE WAVES ON 3D TURBULENT CHANNEL FLOW—AN APPLICATION OF THE METHOD

As an application, we chose to examine the interaction of nonlinear surface waves and a turbulent current. The simplest means of generating and maintaining a turbulent current in a numerical simulation is to drive the flow with a mean pressure gradient and apply a Dirichlet bottom boundary condition. The flow domain for these simulations is a rectangular three-dimensional channel with a wavy free surface on the upper boundary. The turbulent open-channel flow is driven by a constant body force over the length of the domain, computed from the relation

$$\frac{d\Psi}{dx_1} = -\frac{u_\tau^2}{D}, \quad (94)$$

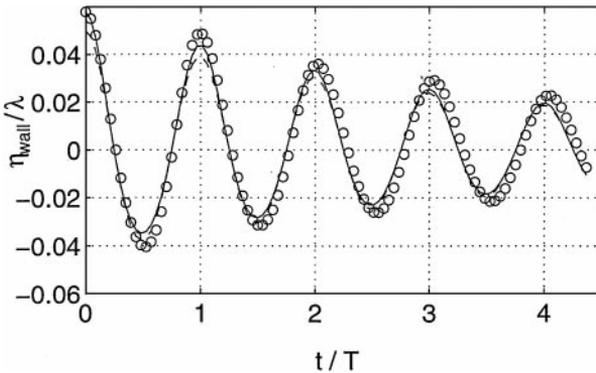


FIG. 3. Free surface wall height: η is surface displacement, λ is wavelength, t is simulation time, T is theoretical wave period; (o) simulation; (—) nonlinear theory; (---) linear theory.

where u_τ is the shear velocity (or friction velocity) at the bottom boundary, D is the depth of the domain, and $d\Psi/dx_1$ is the body force from Eq. (4). The boundary conditions on the flow are (1) periodic in the streamwise and spanwise directions, (2) Dirichlet on the bottom boundary, and (3) the nonlinear kinematic and dynamic boundary conditions on the free surface. The dimensions of the domain are $2\pi \times \pi \times 1$ when non-dimensionalized by the channel depth. Initial conditions for the simulations were developed from DNS data of an open-channel flow with a rigid free-slip lid [44] and the velocity field for a Stokes second-order progressive wave.

5.1. Simulation Setup

These simulations use a wave-following reference frame so that grid motion is limited to perturbation of the wave shape rather than advective motion of the wave. This allows a larger time step than a fixed frame of reference without violating the CFL condition on explicit motion of the free surface [42]. In a wave-following frame, the wave celerity is not known *a priori*, so the wave celerity is computed from the wave crest motion during enforcement of the kinematic boundary condition. The wave celerity is applied as a Dirichlet velocity boundary condition on the bottom boundary, in a direction opposite to wave propagation. There is an adjustment period at the start of the simulation during which the computed wave celerity oscillates. Under these conditions the reference frame is not inertial, so results in this period cannot be considered valid. However, within three to five wave periods, the computed celerity is constant and the reference frame is inertial.

The use of periodic boundary conditions in the streamwise directions simplifies the computations by eliminating requirements for inflow/outflow boundary conditions, which would otherwise require multiple wavelengths of the surface wave to obtain a reasonable simulation. To ensure the suitability of periodic boundary conditions for the turbulent flow, the dimensions of the simulation domain and friction velocity Reynolds number (Re_τ) were identical to those previously used in DNS simulations of a turbulent channel flow with periodic streamwise boundary conditions [44] and LES of turbulent decay in a channel [18]. The present code is based on the LES code used in Salvetti *et al.* [18], which quite accurately reproduced the DNS results to which it was compared. The results of these previous works show that the temporal evolution of the turbulent flow field in this domain can be adequately captured with periodic boundary conditions. Our objective was to examine the interactions between the velocity field generated by surface wave and turbulence for the case where the wavelength is longer than the turbulent length scale. Wave-wave interactions over longer length scales than the wavelength are removed from consideration by limiting the domain to a single wavelength of a Stokes wave. Streamwise periodic boundary conditions in a wave-following reference frame correspond to following the temporal evolution of one wave from an infinite train of periodically identical surface waves. Since the boundary conditions are exactly periodic, the surface wave does not have the spatial decay seen in mechanically generated waves in a laboratory flume, but can be considered a model of regular surface swell at a long distance from the region of generation.

Three types of open-channel simulations have been conducted: (1) a flow that begins with a flat free surface and a turbulent current (the “current-only” simulation), (2) a laminar progressive wave without an imposed streamwise current (the “wave-only” simulation), and (3) a flow that begins with a finite-amplitude progressive wave superposed over a turbulent current (the “wave/current” simulation). The current-only simulation was run until a

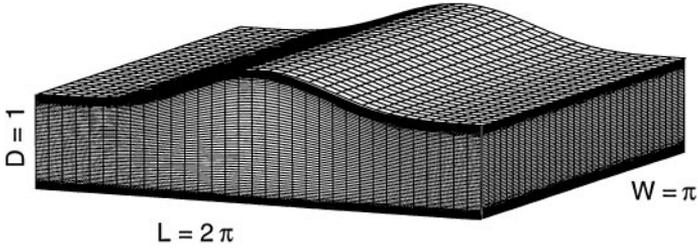


FIG. 4. Initial computational domain for turbulent channel flow.

statistically steady state was reached to obtain a baseline turbulent velocity field. This was used as part of the initial conditions in simulations with a turbulent current. The wave-only initial conditions are the two-dimensional irrotational velocity field of a Stokes wave projected across the three-dimensional domain. The wave/current initial conditions superpose the irrotational velocity field and surface deformation of a Stokes wave onto the turbulent velocity field and surface deformation of the current-only open-channel simulation. The system is then allowed to evolve in accordance with the solution of the Navier–Stokes equations and the boundary conditions.

The simulation domain was discretized with $32 \times 32 \times 64$ grid cells, as shown in Fig. 4. The simulations were conducted at $Re_\tau = 171$, which Pan and Banerjee [44] demonstrated could be resolved to DNS accuracy with 64^3 grid cells using a pseudo-spectral method. Salvetti *et al.* [18] further demonstrated that the fundamental characteristics of decaying turbulence in the DNS simulation of Pan and Banerjee could be captured with the present numerical method and LES model with a 32^3 grid. With a different second-order finite-difference method, Komori *et al.* [12] demonstrated good results in a DNS simulation at $Re_\tau = 160$ using $60 \times 60 \times 40$ grid cells. Given the experience in the literature, the present work provides reasonable resolution for LES of turbulence in the open-channel flow. The primary effect of coarser-than-DNS resolution for the present simulations appears to be an increase in the spanwise spacing of slow speed streaks in the lower boundary layer. The slow-speed streaks have been observed to be spaced at approximately $100z_+$ units in laboratory experiments and well-resolved DNS simulations [45], while the present simulations obtain streak spacing of $170z_+$ units (similar to previous simulation results at coarser resolution [45]).

This setup allows analysis of the wave–turbulence interaction that has heretofore not been presented in the literature. Figure 5 shows spanwise/vertical planes of instantaneous data from current-only and wave–current simulations. The vectors are x_2 velocity and the x_3 velocity fluctuation, while the color scale represents the streamwise x_1 velocity fluctuation. The black areas represent fluid that is slow relative to a spanwise average, and the white areas represent flow that is fast relative to a spanwise average. In these figures the classic “mushroom cap” shape of the hairpin vortices is readily apparent. Of particular interest in Fig. 5 is the difference in the velocity vectors near the surface. In the current-only flow, as the vortex core approaches the free surface, the fluid is accelerated through the effective “nozzle” created between the vortex core and the free surface. In contrast, for the wave–current flow, the vortex core appears to be interacting with the surface in the formation of three-dimensional short-crested waves (Section 5.2).

The primary difficulty of conducting turbulent free-surface numerical simulations is obtaining sufficient grid resolution in the free-surface boundary layer. The length scale

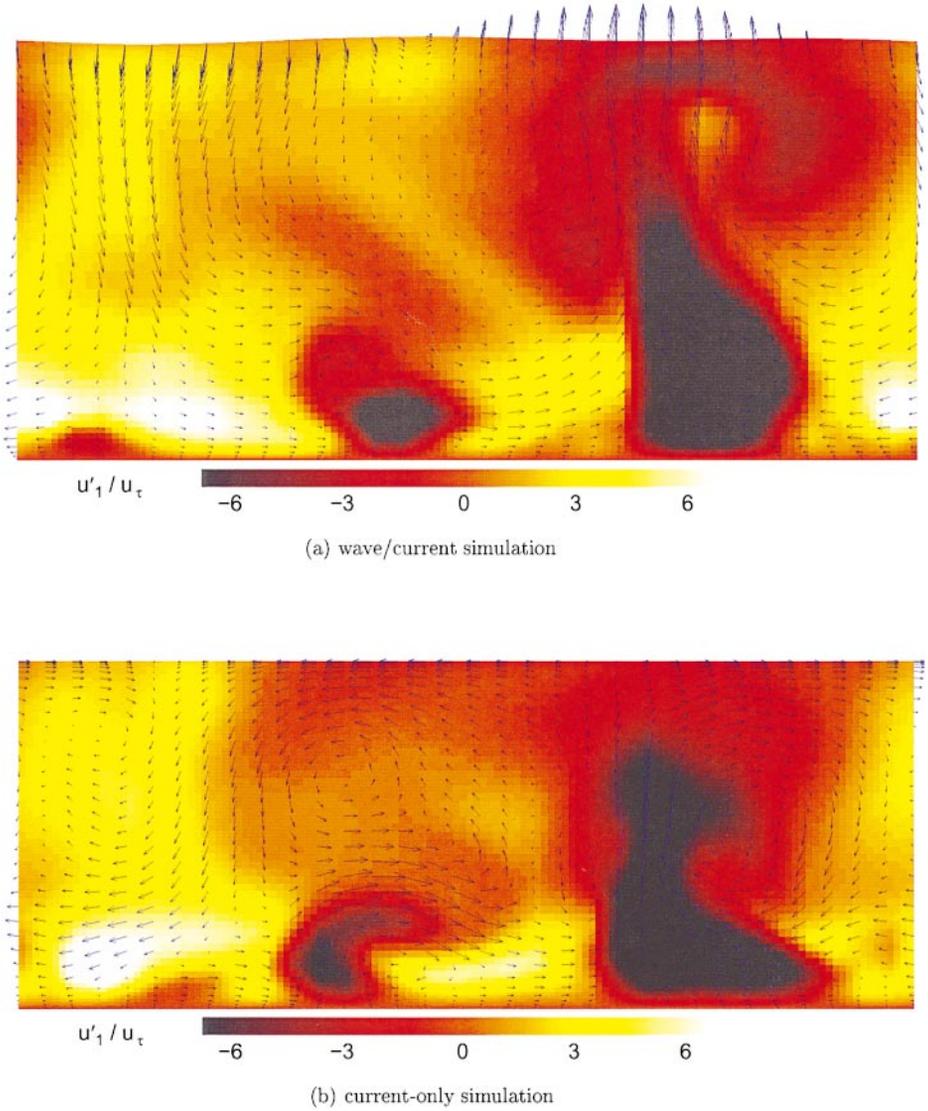


FIG. 5. Instantaneous velocity fluctuations (normalized by u_τ) on spanwise, vertical plane beneath wave crest (color scale shows streamwise component; arrows are vertical and spanwise components).

(β^{-1}) of this layer can be approximated as [39]

$$\beta^{-1} = \sqrt{2\nu/\sigma}. \quad (95)$$

For gravity waves on water, this value is usually $O(0.1)$ mm. For accurate numerical simulation of near-surface viscous effects there should be on the order of five grid cells within the boundary layer. Note that if this minimum resolution is not attained, then the free-surface dynamic boundary condition should be modeled rather than enforced. The present simulations have at least 10 grid cells within the free-surface boundary layer to ensure the near-surface physics are correctly represented.

Another test of the performance of the simulation in the near-surface region is its ability to produce the correct vorticity generation in the free-surface boundary layer. Longuet-Higgins

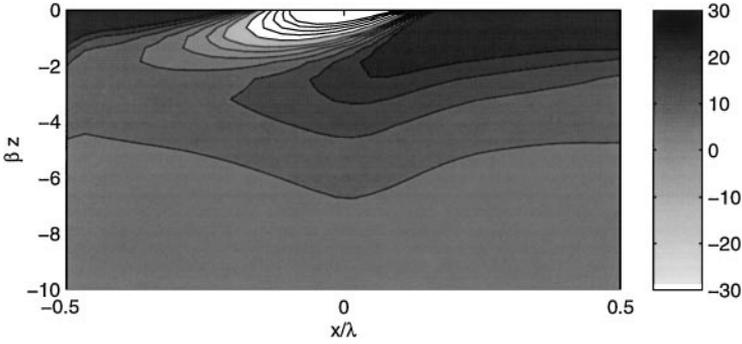


FIG. 6. Contours of near-surface spanwise vorticity in wave/current flow normalized by u_τ/D . Vertical axis is distance below free surface normalized by free surface viscous length scale.

[46] demonstrated that the spanwise vorticity generated near the crest must necessarily be negative (in an $x - z$ system), while the spanwise vorticity near the trough must be positive. Figure 6 shows the mean spanwise vorticity generated near the free surface where the vertical scale is normalized by the viscous length scale β^{-1} and is measured down from the free surface. The wave crest is at $x/\lambda = 0$ with the trough at $x/\lambda = 0.5$. In this figure we can clearly see the wave-induced negative vorticity generated at the crest and the positive vorticity generated in the trough. We can also see that the vorticity generated at the free surface is confined to two or three times the viscous free-surface boundary layer thickness (β^{-1} defined in Eq. [95]). This is in agreement with the arguments of Longuet-Higgins [47].

5.2. Phase-Averaged Spatial Structure

As a precursor to examination of the turbulence beneath the waves, it is useful to compare the instantaneous surface deformations in the wave-only, current-only, and wave/current flows shown in Fig. 7. In the wave-only case, the flow beneath the wave is laminar, with no significant perturbations of the waveform. In the current-only case, small dimples on the surface develop as the surface response to the turbulence below. Tsai [11] showed similar results in a DNS of free surface flow with a sheared current using linearized free-surface boundary conditions for small-amplitude surface motions. In the present wave/current case illustrated in Fig. 7c, the wave and current are interacting to produce a surface signature that is significantly different than either the wave-only or the current-only case. To provide further analysis, an instantaneous phase-averaged monochromatic wave can be defined as the average surface deformation in the spanwise direction. When the phase-averaged wave is subtracted from the surface of the wave/current case, the remaining surface deformation ($\Delta\eta$) appears as shown in Fig. 8a.

The waveform of the deformation shown in Fig. 8a is similar to linear theory for short-crested waves [48] which has a surface deformation (η_{sc}) of

$$\eta_{sc} = a_{sc} \sin(mx) \cos(ny), \quad (96)$$

where a_{sc} is the amplitude of the short-crested wave, m is the wave number in the streamwise direction, and n is the wave number in the spanwise direction. A plot of this waveform is shown in Fig. 8b for $m = \pi$, $n = \pi$, and $a_{sc} = 0.2a$. It can be seen that the agreement between the two waveforms is quite good. This waveform does not appear in the wave-only or current-only simulations, which indicates that the appearance of this wave is due to the

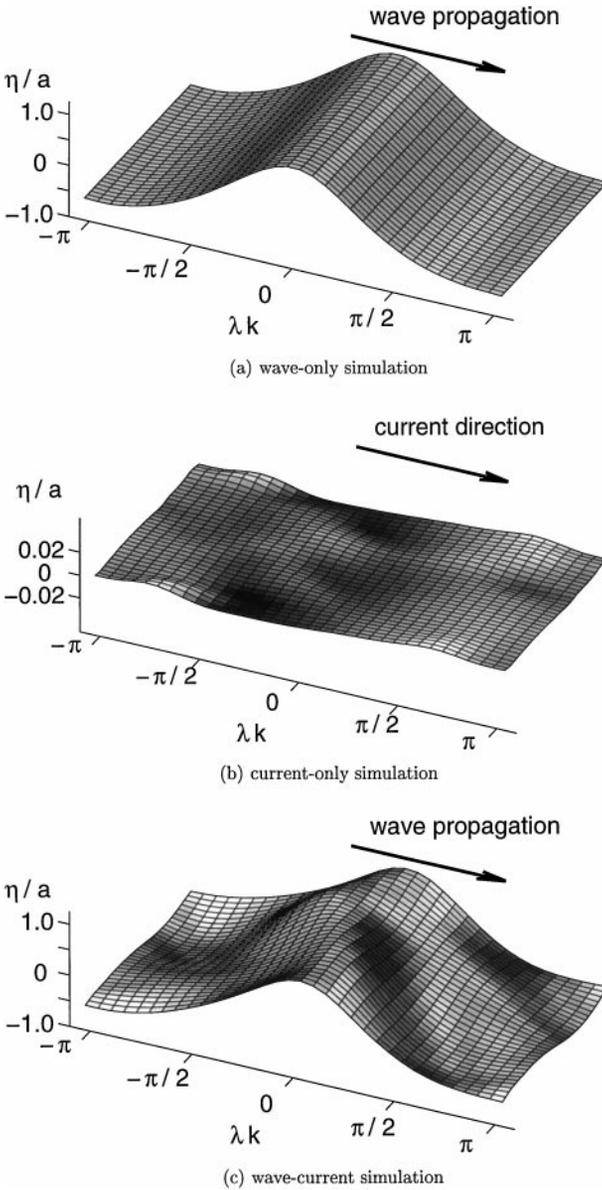


FIG. 7. Free surface shape ($t/T = 10$): coordinate non-dimensionalized by wave amplitude; ordinate non-dimensionalized by wave number k .

interactions of the monochromatic wave and the current rather than to instability of the monochromatic wave or numerical method. The apparent lack of higher order modes in Fig. 8a should not be taken as a failure of the numerical method to resolve such modes. The appropriate amplitude of the higher order modes is shown in Fig. 7b and is an order of magnitude smaller than the short-crested wave mode.

The dominant scale of the short-crested wave mode is a natural resonance phenomena. The short-crested waves produced in the present simulation move at almost the same wave speed as the long-crested waves despite their shorter wavelength. This result is a serendipitous confluence of the chosen geometry and wave characteristics. For other bounded domains,

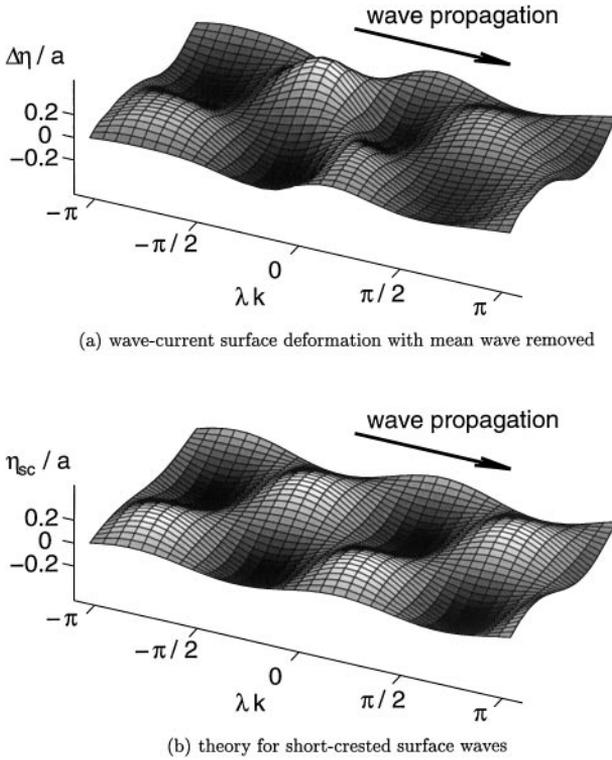


FIG. 8. Comparison of parasitic gravity waves in wave-current simulation to theory for short-crested waves: coordinate non-dimensionalized by wave amplitude from wave-current case; ordinate non-dimensionalized by wave number k .

the resonant short-crested waves may not be supported modes. However, from theory [48], one can demonstrate the existence of short-crested waves which move at speeds identical to those of longer monochromatic waves for general unbounded domains. Thus, the present results with short-crested waves provide a more correct picture of the interactions between a turbulent current and long-crested waves than would be obtained in a simulation where the resonant short-crested modes were not supported by the domain dimensions. Since the short-crested waves did not occur in the wave-only simulation or the current-only simulation, the short-crested wave field may require the existence of turbulent structures and a preexisting monochromatic wave to initiate and maintain the short-crested wave motion. A comparison of the instantaneous cross sections in the current-only and wave/current flows, Fig. 5, shows that the approach of a vortex core near to the free surface may provide an initial impetus for the short-crested waves. Since the current-only flow cannot maintain the short-crested wave, it can be argued that energy transfer from the long-crested wave and/or the turbulence is necessary to maintain the short-crested wave motion.

The results of the numerical simulation can be used to examine the effects of wave-turbulence interaction. The flow is homogeneous in the spanwise direction (except for short-crested waves, as discussed below), so we can compute phase-averaged turbulence quantities as fluctuations from the spanwise mean to obtain two-dimensional instantaneous pictures of the turbulence. This collapses the four-dimensional data set into a three-dimensional data set that can be used to examine the evolution of the instantaneous turbulence field. To fully examine the processes, comparison of the evolution of turbulence quantities for

wave/current simulations and the current-only simulations is presently being undertaken. To provide a more compact (but perhaps less accurate) analysis, we can further reduce the data set by computing time-averaged mean velocity fields using detrending techniques [49]. It is clear that the presence of short-crested waves must be taken into account when interpreting the near-surface structure of the wave/current flows. Unlike long-crested (monochromatic) waves, the short-crested waves are inherently three-dimensional, with velocity fluctuations in the spanwise direction as well as the direction of wave propagation. Wave modes that are not monochromatic provide velocity fluctuations that appear as “turbulence” when using spanwise averaging to compute turbulence quantities. In Fig. 9 the spatial structure

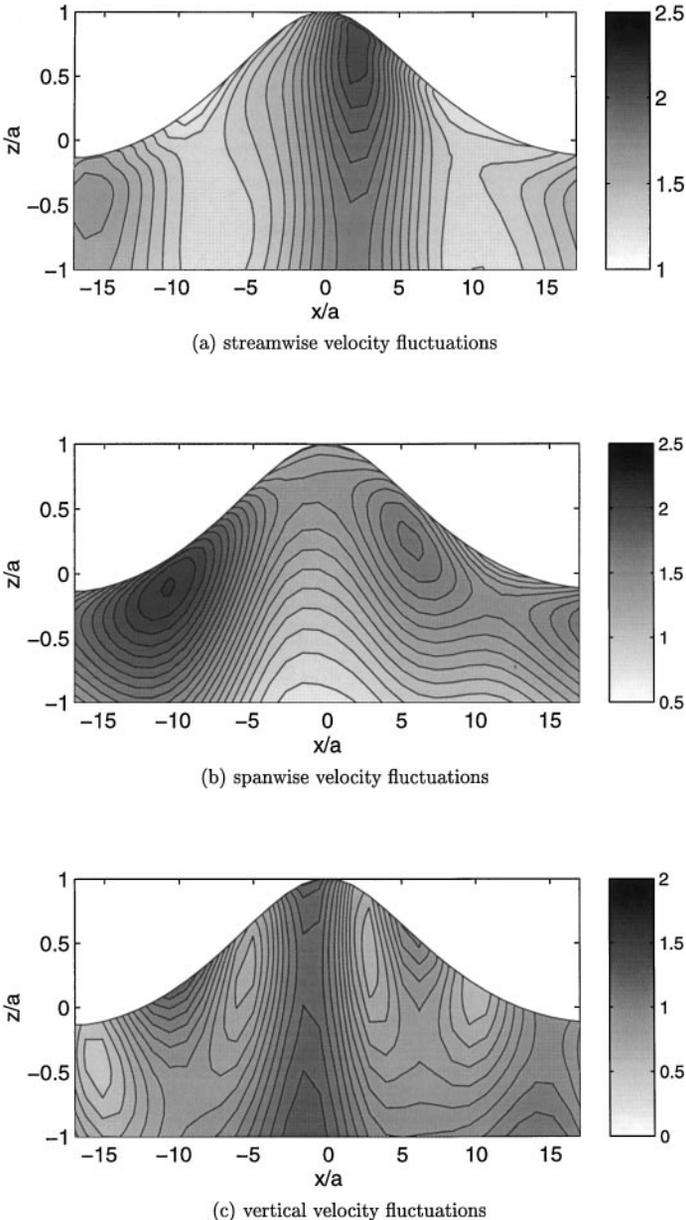


FIG. 9. Mean phase-averaged velocity fluctuations normalized by u_τ for wave/current simulation.

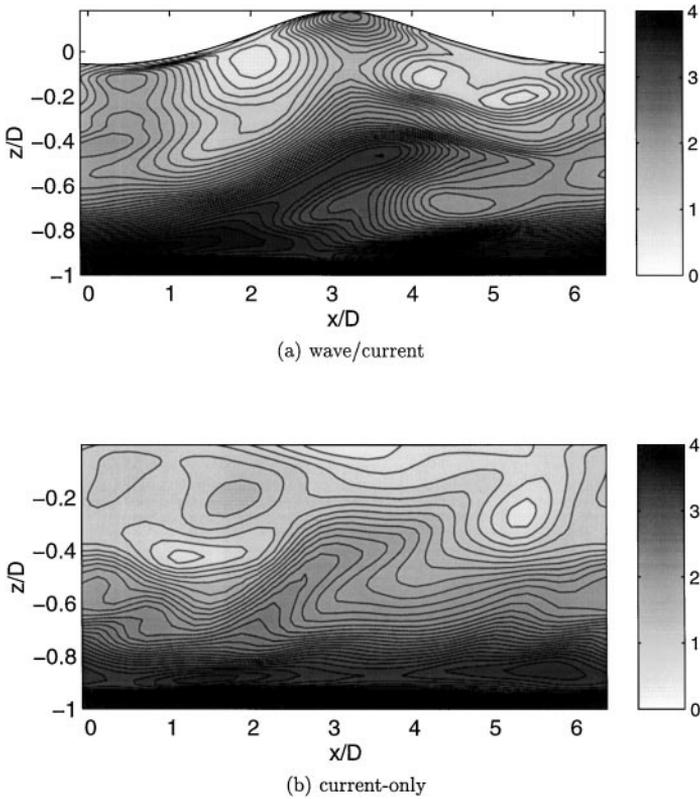


FIG. 10. Instantaneous phase-averaged streamwise velocity fluctuations normalized by u_τ .

of the time-averaged mean of the streamwise, spanwise, and vertical velocity fluctuations is shown in the near-surface region for a wave/current simulation (case W3 in [42]). The periodicity of the structure in the region less than the wave amplitude from the surface is readily apparent and appears to be primarily a function of the short-crested waves (i.e., wave motion rather than turbulent motion). The intensification of the spanwise velocity fluctuations on the trailing edge of the wave appears to be caused by the interaction of bursting coherent vortices with the short-crested wave velocity field.

As an example of the instantaneous turbulence structure, Figs. 10 through 12 show the instantaneous velocity fluctuation fields for a wave/current simulation (case W3 in [42]) and the current-only simulation. These flows were started with the same initial conditions and run for the same period of time. In the wave/current flow, a bursting event in the streamwise velocity fluctuation is intensified beneath the crest of the wave. Examination of the evolution of the turbulence fields shows that the bursting phenomenon tends to resonate with the passage of the wave. In the spanwise velocity fluctuation, Fig. 11, the most noticeable difference is in the free-surface region, where an intense region of spanwise fluctuation is seen along the trailing edge of the wave. This appears to be an interaction between the bursting structure and short-crested parasitic waves that occur on the surface of the simulation. In the vertical velocity fluctuations, the instantaneous near-surface structure is similar to the time-averaged mean shown in Fig. 9c. In the flow core, the wave has two effects: intensification of the vertical fluctuations and vertical oscillation of the flow structure as the wave passes over.

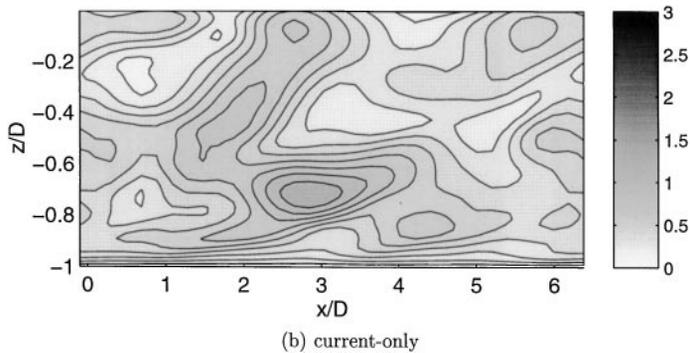
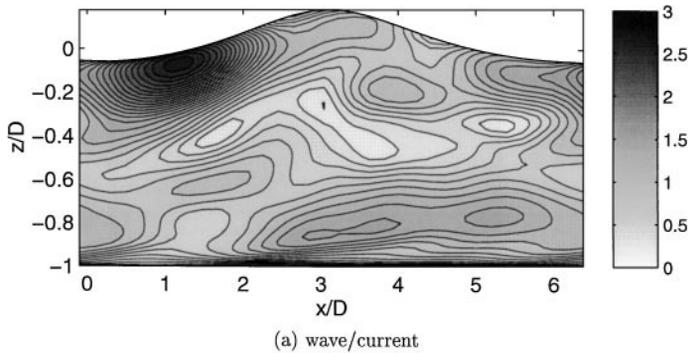


FIG. 11. Instantaneous phase-averaged spanwise velocity fluctuations normalized by u_τ .

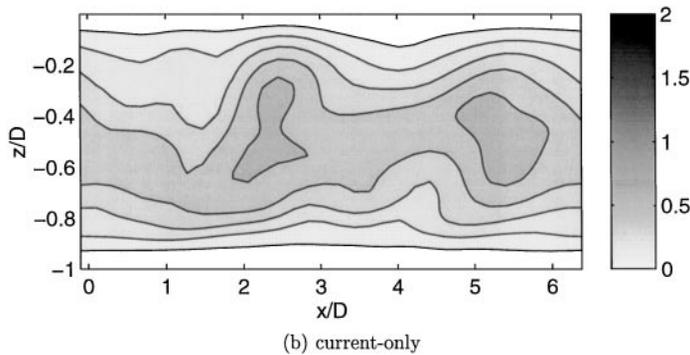
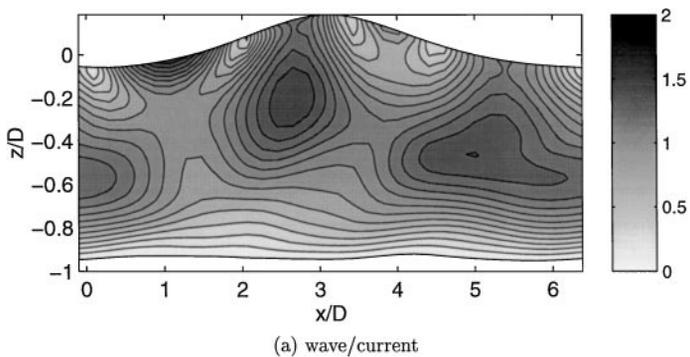


FIG. 12. Instantaneous phase-averaged vertical velocity fluctuations normalized by u_τ .

6. SUMMARY

This paper presented a simulation method for free-surface flows using a boundary-orthogonal, moving-grid, curvilinear coordinate system to solve the time-dependent, viscous, incompressible Navier–Stokes equations. The method is shown to be effective in simulating decaying vortices, laminar standing waves, and turbulent flow in an open channel with a finite-amplitude surface wave.

The numerical method has been developed for large-eddy-simulation techniques with dynamic subgrid-scale modeling. For the first time in the literature, the subgrid-scale filtering and modeling of nonlinear terms in the kinematic boundary condition are derived and discussed. A novel numerical approach for integrating the kinematic boundary condition in curvilinear space is developed. This new approach allows the kinematic boundary condition to be integrated numerically for a free surface that may be multiple-valued in physical space. Numerical discretization of the dynamic boundary condition to serve as a boundary condition on the pressure-Poisson equation in a free-surface flow is presented and discussed in detail.

The capabilities of the method are demonstrated in the simulation of a turbulent open-channel flow with a nonlinear, nonbreaking, progressive, surface wave. The method is designed for overturning waves, although further work on the physics and modeling of wave breaking is necessary to demonstrate this capability. The use of this method for small-amplitude waves is not recommended since the boundary-orthogonal nature of the curvilinear grid is a significant complication that is unnecessary where waves are not steep. Furthermore, the use of vertical grid lines (rather than the boundary-orthogonal method used herein) can speed up the convergence of the pressure-Poisson equation if the hydrostatic pressure is computed as a separate source term in the predictor step of the Navier–Stokes solution [50]. This approach does not appear to be practical for boundary-orthogonal grids due to the introduction of curvilinear interpolation errors into the computation of hydrostatic pressure gradients. For these reasons, the use of the boundary-orthogonal approach outlined in this paper should be limited to simulations of steep waves where the errors associated with vertical grid lines would distort the dynamics of the free-surface boundary layer.

ACKNOWLEDGMENTS

This research was supported by the Fluid Dynamics Program, Mechanics and Energy Conversion Division of the Office of Naval Research, Grant N00014-94-1-0190, scientific officer: Dr. E. P. Rood. The first author also thanks Professor J. Imberger at the Center for Water Research for support during analysis of simulation results and in preparation of this paper.

REFERENCES

1. F. H. Harlow and J. E. Welch, Numerical calculation of time-dependent viscous incompressible flow of a fluid with a free surface, *Phys. Fluids* **8**, 2182 (1965).
2. R. K.-C. Chan and R. L. Street, A computer study of finite-amplitude water waves, *J. Comput. Phys.* **6**, 68 (1970).
3. M. S. Longuet-Higgins and E. D. Cokelet, The deformation of steep surface waves on water. 1. Mass transport in water waves; a numerical method of computation, *Proc. Roy. Soc. London A* **350**, 1 (1976).
4. Y. Yao, P. Wang, and M. Tulin, Wave groups, wave–wake interaction, and wave breaking: Results of numerical experiments, in *Twentieth Symposium on Naval Hydrodynamics* (National Academy Press, 1996), p. 551.

5. H. J. Lugt and S. Ohring, The oblique ascent of a viscous vortex pair toward a free surface, *J. Fluid Mech.* **236**, 461 (1992).
6. D. G. Dommermuth, The laminar interaction of a pair of vortex tubes with a free surface, *J. Fluid Mech.* **246**, 91 (1993).
7. P. Ananthakrishnan and R. W. Yeung, Nonlinear interaction of a vortex pair with clean and surfactant-covered free surfaces, *Wave Motion* **19**, 343 (1994).
8. J. M. Floryan and H. Rasmussen, Numerical methods for viscous flows with moving boundaries, *Appl. Mech. Rev.* **42**, 323 (1989).
9. W.-T. Tsai and D. K. P. Yue, Computation of nonlinear free-surface flows, *Annu. Rev. Fluid Mech.* **28**, 249 (1996).
10. V. Borue, S. A. Orszag, and I. Staroselsky, Interaction of surface waves with turbulence: Direct numerical simulations of turbulent open-channel flow, *J. Fluid Mech.* **286**, 1 (1995).
11. W.-T. Tsai, A numerical study of the evolution and structure of a turbulent shear layer under a free surface, *J. Fluid Mech.* **354**, 239 (1998).
12. S. Komori, R. Nagaosa, Y. Murakami, S. Chiba, K. Ishii, and K. Kuwahara, Direct numerical simulation of three-dimensional open-channel flow with zero-shear gas-liquid interface, *Phys. Fluids A* **5**, 115 (1993).
13. J. Farmer, L. Martinelli, and A. Jameson, Fast multigrid method for solving incompressible hydrodynamic problems with free surfaces, *AIAA J.* **32**, 1175 (1994).
14. Y. Tahara and F. Stern, A large-domain approach for calculating ship boundary layers and wakes and wave fields for nonzero Froude number, *J. Comput. Phys.* **127**, 398 (1996).
15. T. Sarpkaya, Vorticity, free surface, and surfactants, *Annu. Rev. Fluid Mech.* **28**, 83 (1996).
16. Y. Zang, R. L. Street, and J. R. Koseff, A non-staggered grid, fractional step method for time-dependent incompressible Navier-Stokes equations in curvilinear coordinates, *J. Comput. Phys.* **114**, 18 (1994).
17. Y. Zang, *On the Development of Tools for the Simulation of Geophysical Flows*, Ph.D. thesis, Department of Mechanical Engineering, Stanford University (1993).
18. M. V. Salvetti, Y. Zang, R. L. Street, and S. Banerjee, Large-eddy simulation of free-surface decaying turbulence with dynamic subgrid-scale models, *Phys. Fluids* **9**, 2405 (1997).
19. L. L. Yuan, R. L. Street, and J. H. Ferziger, Large eddy simulations of a round jet in a crossflow, *J. Fluid Mech.* **379**, 71 (1999).
20. L. L. Yuan and R. L. Street, Trajectory and entrainment of a round jet in crossflow, *Phys. Fluids* **10**(9), 2323 (1998).
21. R. Calhoun, *Numerical Investigations of Turbulent Flow over Complex Terrain*, Ph.D. thesis, Department of Civil Engineering, Stanford University (1998).
22. M. Hinatsu, Numerical simulation of unsteady viscous nonlinear waves using moving grid system fitted on a free surface, *J. Kansai Soc. Naval Architects Jpn.* **217**, 1 (1992).
23. P. E. Raad, S. Chen, and D. B. Johnson, The introduction of micro cells to treat pressure in free surface fluid flow problems, *ASME J. Fluids Engrg.* **117**, 683 (1995).
24. R. L. Sorenson and S. J. Alter, 3DGRAPE/AL: The Ames/Langley technology upgrade, in *5th International Conference on Numerical Grid Generation in Computational Field Simulations*, edited by B. K. Soni, J. F. Thompson, J. Häuser, and P. Eiseman (NSF Engineering Research Center for Computational Field Simulations, Mississippi State University, 1996), p. 343.
25. Y. Zang, R. L. Street, and J. R. Koseff, A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows, *Phys. Fluids A* **5**, 3186 (1993).
26. M. V. Salvetti and S. Banerjee, A priori tests of a new dynamic subgrid-scale model for finite-difference large-eddy simulations, *Phys. Fluids* **7**, 2831 (1995).
27. J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin, *Numerical Grid Generation, Foundations and Applications* (Elsevier, Amsterdam/New York, 1985).
28. I. Demirdžić and M. Perić, Space conservation law in finite volume calculations of fluid flow, *Int. J. Numer. Methods Fluids* **8**, 1037 (1988).
29. R. L. Street, R. K.-C. Chan, and J. E. Fromm, Two methods for the computation of the motion of long water waves—A review and applications, in *8th Symposium on Naval Hydrodynamics, ARC-179* (Office of Naval Research, Department of the Navy, 1970), p. 147.

30. C. C. Mei, *The Applied Dynamics of Ocean Surface Waves* (Wiley-Interscience, New York, 1983).
31. T. Hino, Computation of a free surface flow around an advancing ship by the Navier–Stokes equations, in *Proceedings of the Fifth International Conference on Numerical Ship Hydrodynamics, Hiroshima, Japan, 1989*, p. 103.
32. E. A. Cowen, *An Experimental Investigation of the Near-Surface Effects of Waves Traveling on a Turbulent Current*, Ph.D. thesis, Department of Civil Engineering, Stanford University (1996).
33. L. E. Scriven, Dynamics of a fluid interface, *Chem. Engrg. Sci.* **12**, 98 (1960).
34. S. Ghosal and P. Moin, The basic equations for the large eddy simulation of turbulent flows in complex geometry, *J. Comput. Phys.* **118**, 24 (1995).
35. J.-C. Park, M. Zhu, and H. Miyata, On the accuracy of numerical wave making techniques, *J. Soc. Naval Architects Jpn.* **173**, 35 (1993).
36. B. K. Soni, J. F. Thompson, J. Häuser, and P. Eiseman, in *5th International Conference on Numerical Grid Generation in Computational Field Simulations* (NSF Engineering Research Center for Computational Field Simulations, Mississippi State University, 1996).
37. J. Kim and P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, *J. Comput. Phys.* **59**, 308 (1985).
38. B. P. Leonard, A stable and accurate convective modelling procedure based on quadratic upstream interpolation, *Comput. Methods Appl. Mech. Engrg.* **19**, 59 (1979).
39. H. L. Lamb, *Hydrodynamics*, 6th ed. (Dover, New York, 1945).
40. R. L. Wiegell, *Oceanographical Engineering* (Prentice Hall, New York, 1964).
41. B. R. Hodges, R. L. Street, and Y. Zang, A method for simulation of viscous, nonlinear, free-surface flows, in *Twentieth Symposium on Naval Hydrodynamics* (National Academy Press, 1996), p. 791.
42. B. R. Hodges, *Numerical Simulation of Nonlinear Free-Surface Waves on a Turbulent Open-Channel Flow*, Ph.D. thesis, Department of Civil Engineering, Stanford University (1997).
43. B. R. Hodges and R. L. Street, Three-dimensional, nonlinear, viscous wave interactions in a sloshing tank, in *Proceedings of the Fluids Engineering Summer Meeting 1996, Vol. 3 FED-Vol. 238* (ASME, 1996), p. 361.
44. Y. Pan and S. Banerjee, A numerical study of free-surface turbulence in channel flow, *Phys. Fluids* **7**, 1649 (1995).
45. I. Nezu and H. Nakagawa, *Turbulence in Open-Channel Flows* (Balkema, Rotterdam, 1993).
46. M. S. Longuet-Higgins, Capillary rollers and bores, *J. Fluid Mech.* **240**, 659 (1992).
47. M. S. Longuet-Higgins, Mass transport in water waves, *Philos. Trans. Roy. Soc. London A* **245**, 535 (1953).
48. R. A. Fuchs, On the theory of short-crested oscillatory waves, in *Gravity Waves*, Circular 521 (National Bureau of Standards, Department of Commerce, 1952), p. 187.
49. J. S. Bendat and A. G. Piersol, *Random Data: Analysis and Measurement Procedures* (Wiley–Interscience, New York, 1986).
50. A. Mahadevan, J. Oliger, and R. Street, A nonhydrostatic ocean model. I. Well-posedness and scaling, *J. Phys. Oceanography* **26**, 1868 (1996).