

## **A coupled level set–boundary integral method for moving boundary simulations**

M. GARZON<sup>†</sup>

*Department of Applied Mathematics, University of Oviedo, Oviedo, Spain*

D. ADALSTEINSSON<sup>‡</sup>

*Department of Mathematics, University of North Carolina, Chapel Hill, NC, USA*

L. GRAY<sup>§</sup>

*Computer Science and Mathematics Division, Oak Ridge National Laboratory*

AND

J. A. SETHIAN<sup>¶</sup>

*Department of Mathematics, University of California, Berkeley, and  
Mathematics Department, Lawrence Berkeley National Laboratory, Berkeley, CA, USA*

[Received 27 September 2004 and in revised form 12 February 2005]

A numerical method for moving boundary problems based upon level set and boundary integral formulations is presented. The interface velocity is obtained from the boundary integral solution using a Galerkin technique for post-processing function gradients on the interface. We introduce a new level set technique for propagating free boundary values in time, and couple this to a narrow band level set method. Together, they allow us to both update the function values and the location of the interface. The methods are discussed in the context of the well-studied two-dimensional nonlinear potential flow model of breaking waves over a sloping beach. The numerical results show wave breaking and rollup, and the algorithm is verified by means of convergence studies and comparisons with previous techniques.

### **1. Introduction and overview**

In this paper, we develop an algorithm for solving moving boundary problems. Our approach is based on a combined level set method and boundary integral method, allowing us to handle topological changes while maintaining the highly accurate aspects of boundary integral formulations. There are many scientific and engineering areas where this capability is expected to be important, such as free surface flows [12] and electrostatically driven flows [24, 36].

For purposes of discussion, we present the techniques in the context of wave breaking over a sloping beach. This topic has been investigated extensively, both experimentally and numerically,

This work was partially supported by U.S. Department of Energy, Applied Mathematical Sciences, and the Division of Mathematical Sciences, National Science Foundation.

<sup>†</sup>E-mail:

<sup>‡</sup>E-mail:

<sup>§</sup>E-mail:

<sup>¶</sup>E-mail: sethian@Math.Berkeley.edu

due to the interest in surf-zone dynamics, sediment transport problems, and impact forces on off-shore and near-shore structures. The most commonly used mathematical models, based on various assumptions, are the nonlinear shallow water equations, the nonlinear Boussinesq models and the nonlinear fully potential models (see for example [4], [15], [17], [26], [38]). More recent models account for turbulent dissipation forces generated when the wave jet overturns. These are based on the Reynolds average Navier–Stokes equation for the mean flow and several  $k$ - $\epsilon$  models for the turbulent field ([19]). Slightly different approaches which also include turbulent effects can be found in [11]. The physical validity of these various models to accurately predict wave breaking is difficult to assess, since physical experiments (Lasser doppler velocities, particle image velocity) fail to give reproducible velocity data in the roller region of the breaking wave.

Under the assumptions that water is an incompressible, inviscid fluid, the motion is irrotational, and imposing appropriate boundary conditions on the free surface, the governing equations for the water wave motion are referred to as ‘fully nonlinear potential model’ (FNPM) and are able to model strongly nonlinear waves. This model has been extensively used by, for example, Grilli et al. ([15], [16], [18]) to predict solitary wave shoaling and wave overturning until the jet of the wave impinges against the flat water surface. They use a Lagrangian-Eulerian formulation of the model equations and a high order boundary element method (BEM) to approximate the boundary integral equation for the computations of free surface velocity.

Such an approach can provide accurate solutions to wave breakage, however the numerical issues associated with regriding to maintain accuracy, topological change, complexities in three dimensions, etc., are challenging. If one considers the more general problem of two-phase flow, in which fluids (in this case, water and air) form part of the computational domain, fully Eulerian techniques for tracking the moving interface which avoid these regriding and topological issues, are available, such as level set methods, introduced by Osher and Sethian [25]. A large collection of simulations have been performed coupling level set methods to Chorin’s projection method ([10]) to compute the solution of incompressible, viscous and inviscid two-phase flow, often in the presence of interface surface tension and considerable density variation between the two fluids (see [8, 34, 37, 32, 39]). In this approach, boundary conditions are required for both fluids. However, it can be difficult to specify appropriate numerical boundary conditions for the air region in open domains that do not adversely affect the calculation.

## 2. General approach

We are interested here in problems in which a moving interface interacts with an associated partial differential equation describing associated physics which both transports, and is influenced by, the position of the interface. Natural candidates for this sort of motion include Hele–Shaw cells, the evolution of air bubbles in water, and electrosprays. We are interested in problems in which topological reconnection is possible, in which geometric terms on the interface may play important roles, and in which issues of stability and regularization are important.

The approach taken in this paper is designed to take advantage of the well-studied robustness and topological properties of level set methods for tracking moving interfaces, while maintaining the accuracy, sharpness, and desirable single-fluid approach that can be obtained from a boundary integral formulation.

The central ideas are as follows. The interface is represented by the zero level set of an embedded level set function, defined throughout a narrow band about the interface in question (see [1]). Similarly, an artificial velocity potential is defined in this region, equaling the correct velocity

potential along the interface. To advance the position of the interface, first a nodal representation is extracted from the level set function, as well as the velocity potential at these nodes. Then a boundary integral method is used to compute the velocity at each of these nodes, and this velocity is then extended throughout the narrow band. The level set function and the velocity potential are updated by advancing initial value Eulerian partial differential equations for both the level set function and the velocity potential. The updating of the velocity potential is carried out using a method similar to that in [3]. This entire process is then repeated.

The principal new features incorporated in the algorithm are (a) a fast Galerkin method for computing the surface gradient of the velocity potential [14]; and (b) a new level set method for transport and diffusion of material quantities on propagating interfaces [3]. To test the algorithm, it is applied to the numerical solution of the FNPM for two-dimensional waves shoaling over flat and sloping bottoms. This approach provides a simple and direct way to solve the model equations by reformulating the problem in a complete Eulerian framework, and straightforward upwind numerical schemes give sufficiently accurate wave profiles while shoaling and breaking. Moreover, the algorithm is successful despite using simpler, lower order, approximations than those employed in previous work. The formulation is unchanged in three dimensions, offering the possibility of computing complex breaking wave motions. We note that an early reference coupling level set methods to boundary integral methods is [33].

Several comments are in order. To begin, although we extract the interface in order to provide nodes for the boundary integral formulation, our approach should not be thought of as resting on a parametrized representation, since this discrete parametrization is used only to obtain physical quantities which are then returned to the underlying Eulerian mesh; the discrete parametrization is then discarded, rather than time-advanced, as is typical in Lagrangian techniques.

Second, in the example of breaking waves considered below, the formulation of the extension potential follows a form which we believe is a prototype for many such problems.

Third, the problem we consider in fact does not contain topological change: rather than be a limitation of the algorithm, this is a limit on the validity of the model. The model equations for breaking surface waves are only valid until the plunging waves impact the surface. Once this happens, an accurate and viable model requires attention to the trapped air as well as the energy dissipation that occurs due to the impact. We are currently trying to formulate a good model which includes these effects. We are also currently at work applying these techniques to an axisymmetric problem in electrosprays.

### 3. The governing equations

We now derive our coupled level set/extension potential equations for breaking waves. Let  $\Omega(t)$  be the 2D fluid domain in the vertical plane  $(x, z)$  at time  $t$ , with  $z$  the vertical upward direction (and  $z = 0$  at the undisturbed free surface), and  $\Gamma_t(s) = (x(s, t), z(s, t))$  a parametrization of the free boundary at time  $t$ . Here,  $\Omega(t)$  is the region bounded by the air/fluid interface on top, the left wall, and the sloping floor (see Figure 1).

Under the above mentioned assumptions, the mass and momentum conservation equations are given by

$$\nabla \mathbf{u} = 0 \quad \text{in } \Omega(t), \quad (1)$$

$$\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{-\nabla p}{\rho} + \mathbf{b} \quad \text{in } \Omega(t), \quad (2)$$

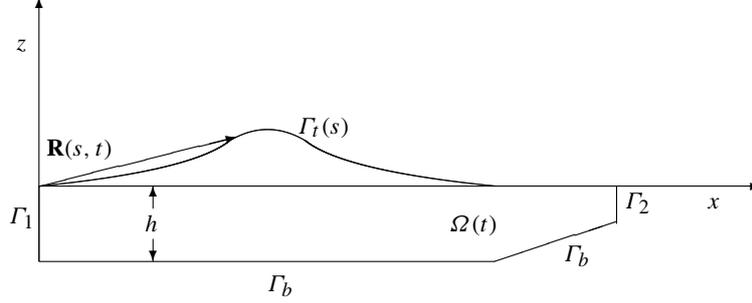


FIG. 1. The domain.

where  $\mathbf{u}(x, z, t)$  is the fluid velocity,  $p(x, z, t)$  the pressure field,  $\mathbf{b}(x, z, t)$  the body forces (per unit mass) and  $\rho$  the fluid density.

If irrotationality is also assumed, fluid particles do not rotate and vorticity vanishes everywhere in the field of flow. In this case, this means that the velocity field can be represented as the gradient of a scalar function referred to as the *velocity potential*  $\phi(x, y, t)$ . If  $\mathbf{u} = \nabla\phi$ , and  $\mathbf{b} = -g\mathbf{z}$ ,  $\mathbf{z}$  being a unit vector in the vertical direction ( $\mathbf{z} = \nabla z$ ), the momentum equation (2) reduces to the so-called *Bernoulli equation*

$$\phi_t + \frac{1}{2}(\mathbf{u} \cdot \mathbf{u}) + \frac{p}{\rho} + gz = C(t).$$

We can take  $C(t) = 0$  using the transformation  $\phi \mapsto \phi + \int_0^s C(s) ds$ , which does not affect the velocity field, and, together with the transformation  $p \mapsto p - p_a$ , which does not affect the basic Euler equations (they only depend upon  $\nabla p$ ), we have

$$\phi_t + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) + \frac{p - p_a}{\rho} + gz = 0. \quad (3)$$

This gives the pressure field once  $\phi$  is known (here  $p_a$  denotes the atmospheric pressure).

On the free boundary, the following boundary conditions are imposed:

1. Continuity of stress tensor between water and air leads to  $p = p_a$ , and thus we have

$$\phi_t + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) + gz = 0 \quad \text{on } \Gamma_i(s).$$

2. If  $\mathbf{R}(s, t) = (X(s, t), Z(s, t))$  is the position vector of a fluid particle on the free surface, we have the kinematic boundary condition

$$\mathbf{R}_t(s, t) = \mathbf{u}(\mathbf{R}(s, t), t) \quad \text{on } \Gamma_i(s)$$

where  $s$  identifies the fluid particle that is at  $x = X(s, t)$ ,  $z = Z(s, t)$  at time  $t$ .

Therefore, the model equations are:

$$\mathbf{u} = \nabla\phi \quad \text{in } \Omega(t), \quad (4)$$

$$\Delta\phi = 0 \quad \text{in } \Omega(t), \quad (5)$$

$$\mathbf{R}_t = \mathbf{u} \quad \text{on } \Gamma_i(s), \quad (6)$$

$$\frac{D\phi}{Dt} = -gz + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) \quad \text{on } \Gamma_t(s), \quad (7)$$

$$\phi_n = 0 \quad \text{on } \Gamma_b \cup \Gamma_1 \cup \Gamma_2, \quad (8)$$

with the material derivative defined in the standard way as

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla.$$

#### 4. Embedding the equations of motion in a level set framework

Level set [25] methods embed a propagating interface as the zero level set of a time-dependent, implicit function, and then solve the resulting equations of motion in a fixed grid Eulerian setting. They rely in part on the theory of curve and surface evolution given in [27, 28] and on the link between front propagation and hyperbolic conservation laws discussed in [29]. Physically appropriate viscosity solutions are obtained by exploiting schemes from the numerical solution of hyperbolic conservation laws. Level set methods are designed for problems involving topological change, curvature dependence and singularities, and complex three-dimensional problems.

Briefly, the main idea is to embed the initial position of the front as the zero level set of a higher-dimensional function  $\Psi(x, z, t)$ . One then links the evolution of this function  $\Psi$  to the propagation of the front itself through a time dependent initial value problem. At any time, the front is given by the zero level set of the time-dependent level set function  $\Psi$ . An equation for the motion of this level set function  $\Psi$  which matches the zero level set of  $\Psi$  with the evolving front comes from observing that the level set value of a particle on the front with path  $\mathbf{R}(s, t)$  must always be zero:

$$\Psi(\mathbf{R}(s, t), t) = 0.$$

Hence, by the chain rule, we have

$$\Psi_t + \nabla\Psi(\mathbf{R}(s, t), t) \cdot \mathbf{u} = 0. \quad (9)$$

For our wave problem, let  $\Omega_1$  be a fictitious fixed squared domain that contains the free boundary at any time  $t$ . Equation (6), which states that the front moves with velocity  $\mathbf{u}$ , can be replaced by the level set equation (9) posed on  $\Omega_1$ .

To embed equation (7) in the level set framework we do the following: the curve that represents the initial position of the front is parametrized by its arclength,  $s \mapsto \Gamma_0(s)$ . Let  $\mathbf{u}(x, z, t)$  be the velocity field the trajectory of a fluid particle with initial position  $s$  is given by the solution of

$$\mathbf{R}_t(s, t) = \mathbf{u}(\mathbf{R}(s, t), t) \quad (10)$$

$$\mathbf{R}(s, 0) = (X(s, 0), Z(s, 0)). \quad (11)$$

For any  $t > 0$  the free boundary curves are parametrized with the same parameter  $s$ ,  $s \mapsto \Gamma_t(s)$ , in order to have the identity  $\Gamma_t(s) := \mathbf{R}(s, t)$ .

On the free boundary  $\Gamma_t(s)$  we define

$$\Phi(s, t) = \phi(x, z, t)|_{\Gamma_t(s)} = \phi(\mathbf{R}(s, t), t),$$

and thus by fixing  $s$  and moving  $t$ , we are constrained to a fluid particle, which means that  $\Phi_t(s, t)$  is a total derivative and hence

$$\Phi_t = \phi_t + \mathbf{u} \cdot \nabla\phi = \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - gz.$$

Next, let  $G(x, z, t)$  be a function defined on  $\Omega_1$  with the following property:

$$G(X(s, t), Z(s, t), t) = \Phi(s, t) \quad \text{on } \Gamma_t(s). \quad (12)$$

It is important to remark here that  $G(x, z, t)$  is an auxiliary function which can be chosen arbitrarily, with the only restriction to be equal to  $\phi(x, z, t)$  on  $\Gamma_t(s)$ . Applying the chain rule in the identity (12) we obtain

$$G_t + \mathbf{u} \cdot \nabla G = \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - gz, \quad (13)$$

which holds on  $\Gamma_t(s)$ . Note that  $\mathbf{u}$  and the right hand side of (13) are only defined on  $\Gamma_t(s)$ . In order to be able to solve (13) over the whole domain  $\Omega_1$ , we need to extend these variables off the front; this strategy is discussed below.

The model equations, written in a complete Eulerian framework, are

$$\mathbf{u} = \nabla\phi \quad \text{in } \Omega(t), \quad (14)$$

$$\Delta\phi = 0 \quad \text{in } \Omega(t), \quad (15)$$

$$\Psi_t + \mathbf{u}_{\text{ext}} \cdot \nabla\Psi = 0 \quad \text{in } \Omega_1, \quad (16)$$

$$G_t + \mathbf{u}_{\text{ext}} \cdot \nabla G = f_{\text{ext}} \quad \text{in } \Omega_1, \quad (17)$$

$$\phi_n = 0 \quad \text{on } \Gamma_b \cup \Gamma_1 \cup \Gamma_2, \quad (18)$$

where  $f = \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - gz$  and  $f_{\text{ext}}$  and  $\mathbf{u}_{\text{ext}}$  are the extensions of  $f$  and  $\mathbf{u}$  onto  $\Omega_1$ .

## 5. Numerical approximations and algorithms

In this section, we provide overviews of the various components. More detailed discussions of level set methods, boundary element methods, fast extension velocities, potential initializations, may be found in the cited references.

### 5.1 Initialization

The initial front position  $\Gamma_0(s) = (x(s, 0), z(s, 0))$  and initial velocity potential  $\phi(x, z, 0)|_{\Gamma_0(s)}$  are needed to solve equations (16) and (17) respectively. Given an initial solitary wave amplitude ( $H_0$ ) and the physical length of the domain ( $L$ ), Tanaka's method gives a way of calculating these quantities (for this aim we have used the Fortran code kindly provided by S. T. Grilli). Here, we briefly discuss the theoretical basis of this method.

Assuming constant depth, the flow field can be reduced to steady state by using a coordinate system that moves horizontally with speed equal to the wave celerity  $c$ . The stream function  $\psi(x, z)$  is also harmonic and takes constant values at the bottom and at the free surface of the domain. From the definition of stream function and velocity potential we have

$$\phi_x = \psi_y, \quad \phi_y = -\psi_x.$$

Under sensible assumptions about the smoothness of  $\phi$  and  $\psi$ , these are just the Cauchy–Riemann equations which are satisfied by the real and imaginary parts of the function  $W = \phi + i\psi$ , called the complex potential, which is an analytic function of the complex variable  $Z = x + iz$  in the domain

occupied by the fluid. By interchanging the roles of the variables  $Z$  and  $W$ , we can take  $\phi$  and  $\psi$  as independent variables, since  $W = \phi + i\psi$  provides a one-to-one correspondence between the physical and complex potential planes. With this transformation, the fluid region is mapped into the strip  $0 < \psi < 1$ ,  $-\infty < \phi < \infty$  in the  $W$  plane with  $\psi = 1$  on the free surface,  $\psi = 0$  on the bottom and  $\phi = 0$  at the wave crest. Denote by  $u, v$  the horizontal and vertical components of the velocity  $\mathbf{u}$ ,  $q = |\mathbf{u}|$  and  $\theta$  the angle between the velocity and the  $x$ -axis. The complex velocity is defined by

$$\frac{dW}{dZ} = \phi_x + i\phi_y = u - iv = qe^{i\theta}$$

and it is also analytic in the flow domain. Therefore, the quantity

$$\omega = \ln\left(\frac{dW}{dZ}\right) = \ln q - i\theta$$

is an analytic function of  $W$ , so  $\tau = \ln q$  must be harmonic in the strip  $0 < \psi < 1$ ,  $-\infty < \phi < \infty$ . The Bernoulli condition at the free surface and the bottom condition can be expressed in terms of  $q$  and  $\theta$  as

$$\frac{dq^3}{d\phi} = -\frac{3}{F^2} \sin \theta \quad \text{on } \psi = 1, \quad (19)$$

$$\theta = 0 \quad \text{on } \psi = 0, \quad (20)$$

where  $F$  is the Froude number defined by  $F = c/\sqrt{gh}$ .

The problem of finding a solitary wave solution can thus be transformed into the problem of finding a complex function  $\omega$  that is analytic with respect to  $W$  within the unit strip  $0 < \psi < 1$ , decays at infinity, and satisfies the boundary conditions (19) and (20). Tanaka's method provides a way to solve the previously outlined equations in terms of the new variables  $\tau$ ,  $\theta$ , and a full description of the algorithm can be found in [35].

## 5.2 Level set methods

We use the standard narrow band level method, introduced by Adalsteinsson and Sethian [2], which limits computation to a thin band around the front of interest. Following the algorithm discussed in [25], we use second order in space upwind differences to approximate the gradient in the level set equation, and a first order in time scheme to update the solution. For boundary conditions, homogeneous flux boundary conditions are usually chosen, which are implemented by creating an extra layer of ghost cells around the domain whose values are simply direct copies of the  $\psi$  values along the actual boundary. The level set function is built from the initial position of the front by computing the signed distance function. This is done by using the fast marching method [30], which is a Dijkstra-like finite difference method for computing the solution to the eikonal equation in  $O(N \log N)$ , where  $N$  is the total number of points in the computational domain.

The velocity and the velocity potential are both initially defined only on the interface. In order to create values throughout the narrow band, which are required to update the fixed grid Eulerian partial differential equations, we use the extension methodology developed by Adalsteinsson and Sethian in [2] to construct appropriate extensions. The idea of building extension velocities was first introduced in [20]; in that approach, the extension velocity at any grid point in the domain was taken

as equal to the velocity at the closest point on the front itself. As shown in [7], this is equivalent to solving the equation  $\nabla V_i \cdot \nabla \Psi = 0$  ( $i = 1, 2$ ) for the velocity components, and in that paper, the equation was solved using a finite difference iteration. In [2], Adalsteinsson and Sethian present a technique for computing this extension velocity using the very efficient fast marching methodology. Finally, in [3], this approach was developed to build extension values for arbitrary material quantities whose evolution affects the underlying interface dynamics.

### 5.3 The boundary integral equation and the BEM approximation

A first order boundary element method is used to approximate equation (15). Boundary integral equations are well suited to moving boundary problems for two principal reasons. First, determining the surface velocity generally requires computing function derivatives on this boundary, which are accurately evaluated within this formulation. Second, remeshing the moving boundary is clearly simpler than remeshing the entire domain.

The Laplace equation for the velocity potential (15) is solved by approximating the corresponding boundary integral equation. Boundary conditions are given by (18) and, on the free boundary, at each time step, by the updated potential velocity given by equation (17). Again, on the free surface,  $\phi$  is known (or more accurately, is computed by the level set method), and the boundary element method is used to compute  $\partial\phi/\partial n$ . However, we do not need the potential on the side walls in order to advance the wave, and thus this part of the solution is ignored. The approximation of the integral equation is done by the BEM, which calculates the potential and the potential gradient on the free surface, that is, its velocity  $\mathbf{u}$ .

The boundary integral equation for the potential  $\phi(P)$ , in a domain  $\Omega(t)$  having boundary  $\Sigma = \partial\Omega(t)$ , can be written as

$$\mathcal{P}(P) = \phi(P) + \lim_{P_I \rightarrow P} \int_{\Sigma} \left[ \phi(Q) \frac{\partial \mathcal{G}}{\partial \mathbf{n}}(P_I, Q) - \mathcal{G}(P_I, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) \right] dQ = 0, \quad (21)$$

where  $\mathbf{n} = \mathbf{n}(Q)$  denotes the unit outward normal on the boundary surface and  $\{P_I\}$  are interior points converging to the boundary point  $P$ . The Green's function or fundamental solution (in two dimensions) is

$$\mathcal{G}(P, Q) = -\frac{1}{4\pi} \log(r). \quad (22)$$

The integral equation is usually written with the  $\partial\mathcal{G}/\partial\mathbf{n}$  singular integral evaluated as a Cauchy principal value (CPV), resulting in an 'interior angle' coefficient  $c(P)$  multiplying the leading  $\phi(P)$  term [5, 6]. The reason for employing the seemingly more complicated limit process will become clear in the discussion of gradient evaluation. The exterior limit equation

$$\lim_{P_E \rightarrow P} \int_{\Sigma} \left[ \phi(Q) \frac{\partial \mathcal{G}}{\partial \mathbf{n}}(P_E, Q) - \mathcal{G}(P_E, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) \right] dQ = 0 \quad (23)$$

yields precisely the same equation: the jump in the CPV integral as one crosses the boundary accounts for the 'free term' difference.

In this work, a Galerkin (weak form) approximation of (21) has been employed, and the boundary and boundary functions are interpolated using the simplest approximation, linear shape functions. Thus, the equations that are solved are of the form

$$\int_{\Sigma} \hat{\psi}_k(P) \mathcal{P}(P) dP = 0, \quad (24)$$

where the weight functions  $\hat{\psi}_k(P)$  are comprised of all shape functions which are nonzero at a particular node  $P_k$  (cf. [5]). The calculations reported herein employed the simplest approximation, linear shape functions. These approximations reduce the integral equation to a finite system of linear equations, and invoking the boundary conditions allows the solution of the unknown values of potential and flux on the boundary. Details concerning the limit evaluation of the singular integrals can be found in [13].

As noted above, for the wave problem, and moving boundary problems in general, knowledge of the normal flux is not sufficient: we may need both velocity components with respect to the Cartesian coordinates. Note that in (16) the separate Cartesian components of the velocity are what appears, not the flux. What we are required to do is to compute a post-boundary-element solve in order to find these individual components.

The remainder of this section will present the algorithm for computing this gradient.

From (21) a gradient component can be expressed as

$$\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \lim_{P_I \rightarrow P} \int_{\Sigma} \left[ \frac{\partial \mathcal{G}}{\partial \mathbf{E}_k}(P_I, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q) \frac{\partial^2 \mathcal{G}}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_I, Q) \right] dQ. \quad (25)$$

Once the boundary value problem has been solved, all quantities on the right hand side are known: a direct evaluation of nodal derivatives would therefore be easy were it not for well known difficulties with the hypersingular (two derivatives of the Green's function) integral [22, 23, 21]. As described in [14], a Galerkin approximation of this equation,

$$\begin{aligned} & \int_{\Sigma} \hat{\psi}_k(P) \frac{\partial \phi(P)}{\partial \mathbf{E}_k} dP \\ &= \lim_{P_I \rightarrow P} \int_{\Sigma} \hat{\psi}_k(P) \int_{\Sigma} \left[ \frac{\partial \mathcal{G}}{\partial \mathbf{E}_k}(P_I, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q) \frac{\partial^2 \mathcal{G}}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_I, Q) \right] dQ dP, \end{aligned} \quad (26)$$

allows a treatment of the hypersingular integral using standard continuous elements.

Interpolating  $\partial \phi(P)/\partial \mathbf{E}_k$  as a linear combination of the shape functions results in a simple system of linear equations for nodal values of the derivative everywhere on  $\Sigma$ ; the coefficient matrix is obtained by simply integrating products of two shape functions. However, the complete boundary integrations required to compute the right hand side are quite expensive.

The computational cost of this procedure can be significantly reduced by exploiting the exterior limit equation, (23). It appears to be useless for computing tangential derivatives for, lacking the free term, the corresponding derivative equation takes the form

$$0 = \lim_{P_E \rightarrow P} \int_{\Sigma} \left[ \frac{\partial \mathcal{G}}{\partial \mathbf{E}_k}(P_E, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q) \frac{\partial^2 \mathcal{G}}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_E, Q) \right] dQ, \quad (27)$$

and the derivatives obviously do not appear. However, subtracting this equation from (25) yields (with shorthand notation)

$$\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \left\{ \lim_{P_I \rightarrow P} - \lim_{P_E \rightarrow P} \right\} \int_{\Sigma} \left[ \frac{\partial \mathcal{G}}{\partial \mathbf{E}_k} \frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q) \frac{\partial^2 \mathcal{G}}{\partial \mathbf{E}_k \partial \mathbf{n}} \right] dQ. \quad (28)$$

The advantage of this formulation is that now *only the terms that are discontinuous crossing boundary* contribute to the integral. In particular, all nonsingular integrations, by far the most time

consuming, drop out. The calculation of the right hand side in (28) reduces to a few ‘local’ singular integrations, and as these integrations are carried out partially analytically, this produces an accurate algorithm. Further details about the evaluation of (28) can be found in [14].

#### 5.4 The velocity potential updating

The potential equation (17) is a convection equation with a strong nonlinear source term, and homogeneous Neumann boundary conditions are imposed on the boundary of  $\Omega_1$ . To update in time this equation, note that it is similar to (16) except that it has a nonlinear source term, and therefore we use similar schemes. For example a straightforward first order scheme is

$$G_{i,j}^{n+1} = G_{i,j}^n - \Delta t (\max(u_{i,j}^n, 0) D_{i,j}^{-x} + \min(u_{i,j}^n, 0) D_{i,j}^{+x} \\ + \max(v_{i,j}^n, 0) D_{i,j}^{-z} + \min(v_{i,j}^n, 0) D_{i,j}^{+z}) + \Delta t f_{i,j}^n$$

where

$$D_{i,j}^{-x} = D_{i,j}^{-x} G_{i,j}^n = \frac{G_{i,j}^n - G_{i-1,j}^n}{\Delta x}, \quad D_{i,j}^{+x} = D_{i,j}^{+x} G_{i,j}^n = \frac{G_{i+1,j}^n - G_{i,j}^n}{\Delta x}$$

are the backward and forward finite approximations for the derivative in the  $x$  direction (we have the same expressions for  $D_{i,j}^{-z}$  and  $D_{i,j}^{+z}$ ). Note that for simplicity we have written  $u, v, f$  instead of  $u_{\text{ext}}, v_{\text{ext}}, f_{\text{ext}}$ , and we describe a first order explicit scheme with a centered source term. Initial values of  $G_{i,j}^0$  are obtained by extending  $\phi(x, z, 0)|_{\Gamma_0(s)}$  as previously discussed. However, at any time step  $n$  it is always possible to perform a new extension of  $\Phi^n(s, n\Delta t)$  to obtain a better value of  $G_{i,j}^n$ .

A key issue is how one obtains  $f_{\text{ext}}$  at the grid points of  $\Omega_1$ . There are several ways of doing so. Here we calculate  $f = \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - gz$  on free surface nodes, and use these values together with the condition  $\nabla f \cdot \nabla\Psi = 0$  to obtain  $f_{\text{ext}}$ . This algorithm for extending off the front quantities defined on the front works very well for the velocity field in the case of equation (16), as it maintains the signed distance function for the level sets of  $\Psi$ . However, regarding equation (17) for this particular wave problem, the previous method creates strong  $G$  and  $f$  gradients in  $\Omega_1$ . This is due to the high variations of  $f$  along the front together with its topological structure when overturning. This fact limits the grid spacing in  $\Omega_1$  and the time step needed to maintain accuracy (see the section on numerical experiments).

#### 5.5 Regridding of the free surface

In a level set formulation the position of the front is only known implicitly through the node values of the level set function  $\Psi$ . In order to extract the front, it is possible to construct first order and second order approximations of the interface using local data of  $\Psi$  on the mesh (see [9] for example). Here we use a first order linear approximation of the free surface, which yields a polygonal interface formed by unevenly distributed nodes, which we call LS nodes. As a result of this extraction technique, we can sometimes get front nodes which are very close together, and this can cause difficulties and instabilities for boundary element calculations. To overcome this problem, and also to achieve more front resolution when needed, we employed a front node regridding technique. An initialization point on the front is selected according to a particular criterion, such as maximum value of height, velocity modulus, or front curvature. This point divides the front into two halves

and new nodes are chosen so that, lying in the same polygon, they are redistributed by arclength according to the formula

$$s_{i+1} - s_i = d_0(1 + s_i(f_0 - 1))$$

where  $s_i$  denotes the arclength distance from node  $i$  to the initialization point ( $i = 0$ ), and  $d_0, f_0$  are user selected parameters. These regridded nodes on the front are used to create the input file for the BEM calculations and are denoted by BEM nodes.

### 5.6 The algorithm

To initialize the position of the front and the velocity potential on the front, we use Tanaka's method for computing numerical exact solitary waves.

The basic algorithm can be summarized as follows:

1. Compute the initial front position and velocity potential  $\Phi(s, 0)$  on  $\Gamma_0(s)$ .
2. Extend  $\Phi(s, 0)$  onto the grid points of  $\Omega_1$  to initialize  $G$ .
3. Generate  $\Omega(t)$  and solve (15), using the boundary element method. This yields the velocity  $\mathbf{u}$  and source term  $f$  at the front nodes.
4. Extend  $\mathbf{u}$  and  $f$  off the front onto  $\Omega_1$ .
5. Update  $G$  using (17) in  $\Omega_1$ .
6. Move the front with velocity  $\mathbf{u}$  using (16) in  $\Omega_1$ .
7. Interpolate (bicubic interpolation)  $G$  from grid points of  $\Omega_1$  to the front nodes to obtain new boundary conditions for (15). Go back to step 3 and repeat forward in time.

A more detailed algorithm including regridding is:

Initialization: Given  $\Gamma^0 = \Gamma_0(s)$ ,  $\Phi^0 = \Phi(s, 0)$

1. Calculate  $\Psi^0$  and LS nodes.
2. Extend  $\Phi^0$  to obtain  $G^0$ .
3. Redistribute LS nodes to obtain BEM nodes.
4. Calculate  $\mathbf{u}^0$  at BEM nodes.
5. Find  $\mathbf{u}^0$  and  $f^0$  at LS nodes and extend onto  $\Omega_1$ .

Steps: Given  $\Psi^n, \Phi^n, \mathbf{u}^n$

1. Calculate  $\Psi^{n+1}$  and LS nodes.
2. Calculate  $G^{n+1}$  at  $\Omega_1$  grid points.
3. Redistribute LS nodes to obtain BEM nodes.
4. Interpolate  $G$  on BEM nodes to find  $\Phi^{n+1}$ .
5. Calculate  $\mathbf{u}^{n+1}$  at BEM nodes.
6. Find  $\mathbf{u}^{n+1}$  and  $f^{n+1}$  at LS nodes and extend onto  $\Omega_1$ . Go to step 1 and repeat.
7. If reinitialization
  - (a) Take LS nodes and reinitialize  $\Psi^{n+1}$ .
  - (b) Take BEM nodes and extend  $\Phi^{n+1}$ .

### 5.7 Numerical accuracy

The model equations imply that the wave mass and its total energy should be conserved as the wave evolves in time. One way to check the numerical accuracy of the discretized equations is to compute

these quantities at each time step (here, we are checking the conservation of energy to see how it depends on grid resolution). The wave mass above  $z = 0$  is given by

$$m(t) = \int_{\Omega(t)} d\Omega = \int_{\partial\Omega(t)} zn_z ds = \int_{\Gamma_t(s)} zn_z ds$$

and the total energy is  $E(t) = E_p(t) + E_k(t)$ , where  $E_p(t)$ ,  $E_k(t)$  denote the potential and kinetic wave energy respectively. They can be calculated using the expressions

$$E_p(t) = \frac{1}{2}\rho g \int_{\Omega(t)} z d\Omega = \frac{1}{2}\rho g \int_{\Gamma_t(s)} z^2 n_z ds,$$

which is the potential energy with respect to  $z = 0$ , and

$$E_k(t) = \frac{1}{2}\rho \int_{\Omega(t)} \nabla\phi \cdot \nabla\phi d\Omega = \frac{1}{2}\rho \int_{\partial\Omega(t)} \phi \frac{\partial\phi}{\partial n} ds = \frac{1}{2}\rho \int_{\Gamma_t(s)} \phi \frac{\partial\phi}{\partial n} ds,$$

where the divergence theorem has been applied to the three formulas and we have used the fact that  $\partial\phi/\partial n = 0$  on  $\Gamma_b, \Gamma_1, \Gamma_2$  for the kinetic energy formula. These integrals are approximated by a composite trapezoidal rule, using the values of the quantities at the free boundary BEM nodes. Note that LS nodes could have been used for  $m(t)$  and  $E_p(t)$  approximations but we also used BEM nodes for simplicity. The components of the normal vector to the free surface are computed using the level set embedding function to obtain surface geometrical variables.

A common procedure to study the accuracy and convergence properties of the discretized equations with respect to the mesh sizes and the time step is by means of an analytical solution. A solitary wave propagating over a constant depth is a traveling wave that moves in the  $x$  direction with speed equal to the celerity of the wave ( $c$ ). The velocity potential and the velocity on the front as functions of  $x$  are also translated with the same speed  $c$ . Therefore, in this case, by calculating initial wave data with Tanaka's method and translating it, we are able to compute the  $L_2$  norms of the errors for the various magnitudes. For the case of a solitary wave shoaling over a sloping bottom, the accuracy can only be checked looking at the mass and energy conservation properties and comparing breaking wave characteristics obtained here with those reported elsewhere, for example in [16].

## 6. Numerical results

The system of equations to be discretized is a nonlinear system of strongly coupled partial differential equations. First order in time and second order in space schemes are used for equation (16); first order in time and in space schemes are used for equation (17); and a first order BEM solver is used for the velocity updating.

To study the convergence properties of this method and its capability to predict wave breaking characteristics, the numerical results corresponding to the following physical settings are presented: A solitary wave propagating over a constant depth and the shoaling and breaking of a solitary wave propagating over various sloping bottoms.

### 6.1 Constant depth test

In order to tune the discretization parameters and see how they affect numerical accuracy we performed a series of numerical tests with a solitary wave of  $H_0 = 0.5$  m (wave height at the

crest) propagating over a constant depth of 1 m. The wave crest is initially located at  $x = 6.5$  m and the domain has  $L = 15$  m of length. In what follows, the units are taken as meters and seconds for length and time, respectively.

Let  $\Omega_1 = [0, 15] \times [-0.3, 1]$  be the fictitious domain that contains the free boundary for all  $t \in [0, 0.5]$ ,  $\Delta x = \Delta z$  the grid size and  $\Delta t$  the time step. To discretize  $\partial\Omega(t)$ , in order to generate the input BEM file, a variable mesh size is used:  $\Delta l = 0.1$  for  $\Gamma_1$  and  $\Gamma_2$ ,  $\Delta l = 0.2$  for  $\Gamma_b$ , and the regridding parameters for  $\Gamma_s(t)$  are chosen to be  $d_0 = 0.005$ ,  $f_0 = 10$ . This gives 193 BEM nodes on the moving front and 98 nodes on the fixed boundaries.

The mesh size  $\Delta x = \Delta z$  for  $\Omega_1$  should be chosen in order to achieve accurate interpolated values of front position and potential on the front. For the time step selection, a first limitation is the CFL condition. While this condition is enough for the stability of the numerical approximation of equations (16) and (17), the accuracy in the numerical solution of equation (17) requires a smaller time step. This is due to the fact that  $G$  and the source term  $f$ , for this particular wave problem, develop high gradients in  $\Omega_1$ . Therefore we present the results for the following test cases:

- (a)  $\Delta x = 0.1$ ,  $\Delta t = 0.01$ .
- (b)  $\Delta x = 0.1$ ,  $\Delta t = 0.001$ .
- (c)  $\Delta x = 0.01$ ,  $\Delta t = 0.001$ .
- (d)  $\Delta x = 0.01$ ,  $\Delta t = 0.0001$ .

For given solitary wave parameters ( $H_0$  and length  $L$  in the  $x$  direction) Tanaka's method gives us the initial wave magnitudes, front location, velocity potential, velocity components at front points and wave celerity  $c$ . At any time  $t$ , let  $(x_{ex}, z_{ex})$ ,  $\phi_{ex}$ ,  $u_{ex}$ ,  $v_{ex}$  be the values of these variables obtained by translating initial values a distance  $ct$  along the  $x$  direction and spline interpolating at LS nodes. Denote by  $(x_c, z_c)$ ,  $\phi_c$ ,  $u_c$ ,  $v_c$  the computed values at LS nodes, and by  $L_2(z) = \|z_c - z_{ex}\|_{L_2(\Gamma_s(t))}$ ,  $L_2(\phi) = \|\phi_c - \phi_{ex}\|_{L_2(\Gamma_s(t))}$ ,  $L_2(u) = \|u_c - u_{ex}\|_{L_2(\Gamma_s(t))}$  and  $L_2(v) = \|v_c - v_{ex}\|_{L_2(\Gamma_s(t))}$  the  $L_2$  norms of the errors. Table 1 shows these errors at the final time  $t = 0.5$  for the various test cases.

TABLE 1  
Values of the  $L_2$  error norms at  $t = 0.5$

Test	$L_2(z)$	$L_2(\phi)$	$L_2(u)$	$L_2(v)$
(a)	0.007239	0.095254	0.025147	0.025856
(b)	0.009762	0.021451	0.039635	0.035685
(c)	0.001476	0.011363	0.0099744	0.009356
(d)	0.001699	0.00424601	0.0106674	0.010188

Figures 2 and 3 show  $L_2(z)$ ,  $L_2(\phi)$ ,  $L_2(u)$ ,  $L_2(v)$  versus time for cases (c) and (d) respectively. As observed from these results, the  $L_2$  error norm in front location and velocity components decreases with mesh size ( $\Delta x$ ) but not with the time step. Only the velocity potential gains accuracy when  $\Delta t$  is reduced according to the above mentioned facts.

Regarding wave mass and energy conservation, at each time step we calculate  $m(t)$  and  $E(t)$  as explained in 4.7. Figures 4 and 5 show the values of  $|m(t) - m(0)|$  and  $|E(t) - E(0)|$  versus time and the same behavior of these quantities with respect discretization parameters is observed.

Next, to see if we gain accuracy in the velocity calculations by increasing the number of BEM nodes, we take  $\Delta l = 0.05$  on  $\Gamma_1$  and  $\Gamma_2$ ,  $\Delta l = 0.1$  on  $\Gamma_b$ , and  $d_0 = 0.001$ ,  $f_0 = 5$  on  $\Gamma_s(t)$ .

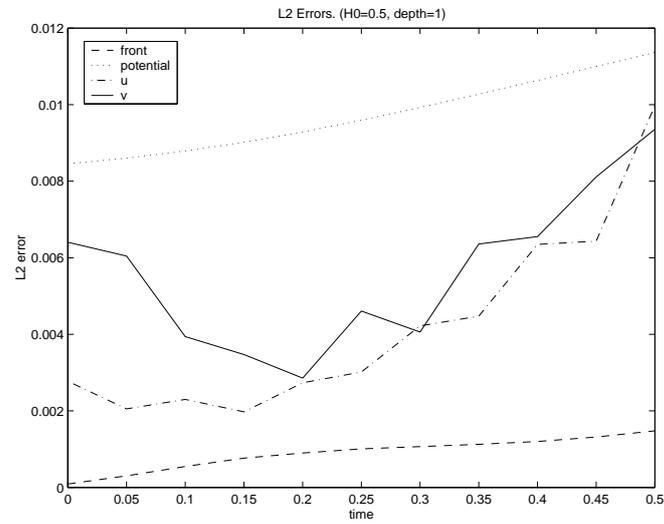


FIG. 2.  $L_2(z)$ ,  $L_2(\phi)$ ,  $L_2(u)$ ,  $L_2(v)$  vs time for case (c).

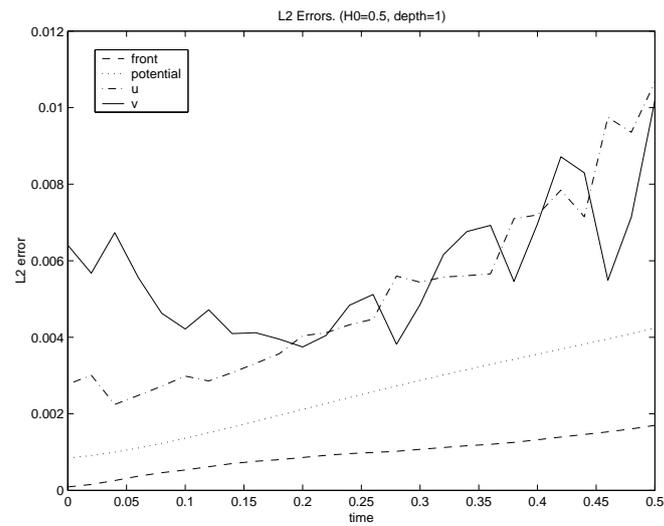


FIG. 3.  $L_2(z)$ ,  $L_2(\phi)$ ,  $L_2(u)$ ,  $L_2(v)$  vs time for case (d).

This gives 1720 BEM nodes on the moving front and 196 nodes for the fixed boundaries. For this discretization of the BEM boundary we run two more cases:

- (e)  $\Delta x = 0.01$ ,  $\Delta t = 0.001$ .
- (f)  $\Delta x = 0.01$ ,  $\Delta t = 0.0001$ .

Values of the  $L_2$  error norms for cases (e) and (f) are almost identical to those obtained for cases (c)

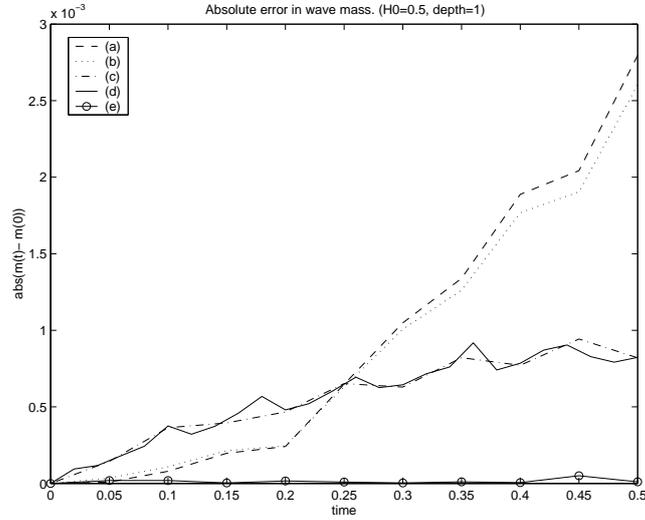


FIG. 4. Absolute error in wave mass.

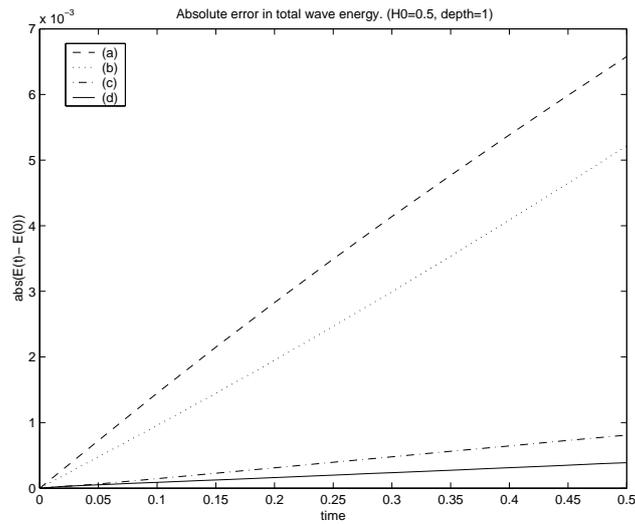


FIG. 5. Absolute error in total wave energy.

and (d) respectively, which means that accuracy in velocity is not gained by increasing the number of BEM nodes. However, as is shown in Figure 5,  $|m(t) - m(0)|$  has decreased by almost an order of magnitude due to the accuracy in front position and the improvement in the integral approximation to calculate  $m(t)$ . Figure 6 shows for case (e) the absolute errors in  $E_p(t)$ ,  $E_k(t)$ ,  $E(t)$  versus time and, in agreement with the previous discussion, the kinetic energy is much less accurate than the potential energy.

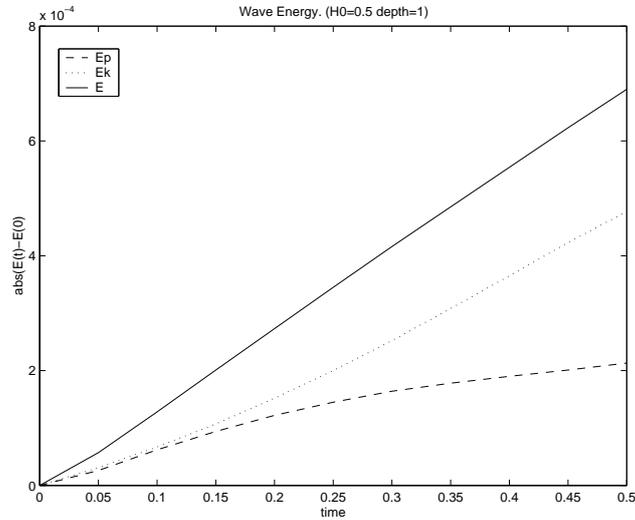


FIG. 6. Absolute error in potential, kinetic and total energy. Case (e).

From these numerical experiments we conclude that the proposed algorithm converges, but we do not achieve exactly first order convergence with respect to discretization parameters. This is due to the strong interdependence of the equations. Note that  $f$  depends nonlinearly on  $\mathbf{u}$  and linearly on  $z$  and that the boundary condition imposed on  $\Gamma_s(t)$  for the BEM solver builds up numerical and round off error as we step forward in time; we note that the level set approach is stable and robust with respect to these small sawtooth instabilities resulting from velocity calculations on very closely spaced nodes, and the use of filtering or smoothing was not required.

Case (c) discretization parameters give sufficient accuracy in the following sense: the results show that the absolute error in mass and energy conservation is approximately .0008 for mesh sizes that are not too restrictive. We show wave profiles, velocity potential and velocity components for various times in Figures 7, 8 and 9 respectively.

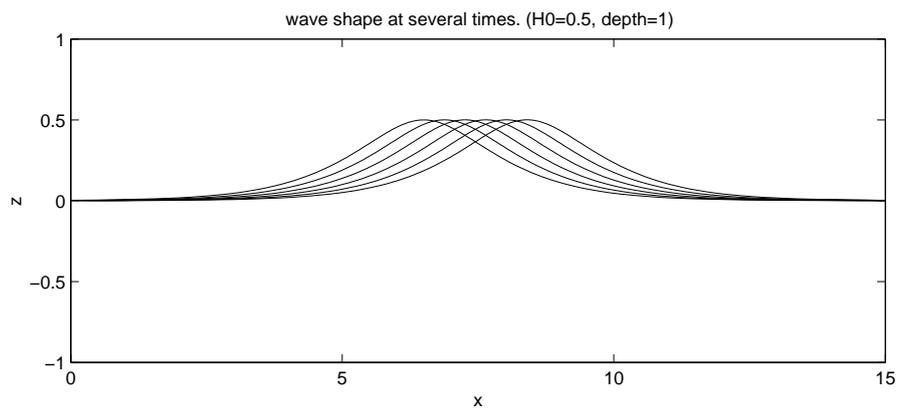
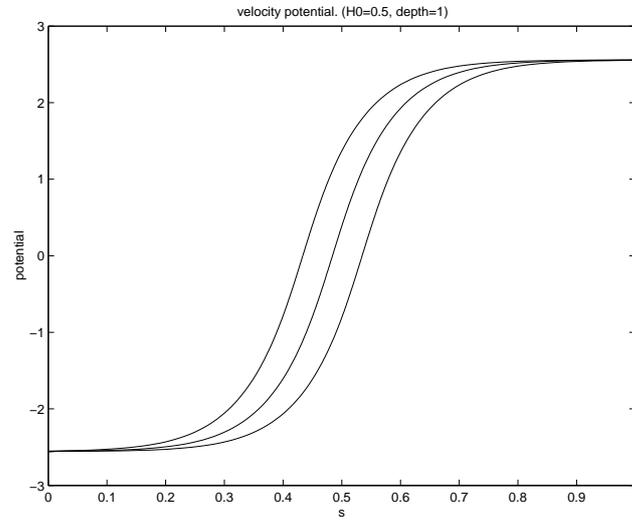
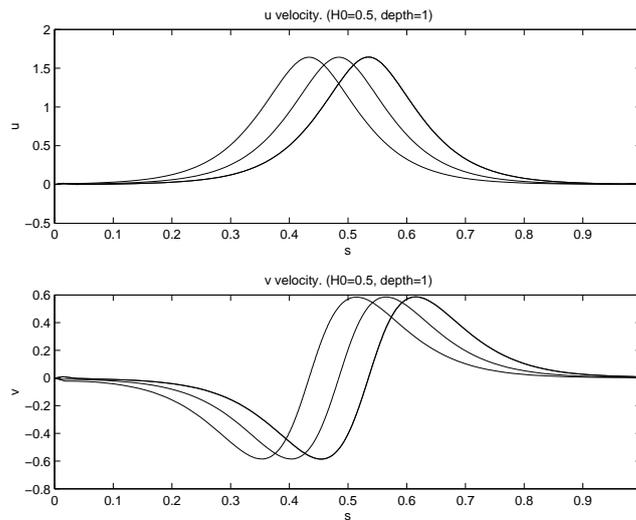


FIG. 7. Front location at  $t = 0, 0.1, 0.2, 0.3, 0.4, 0.5$ . Case (c).

FIG. 8. Velocity potential at  $t = 0, 0.25, 0.5$ . Case (c).FIG. 9. Velocity components at  $t = 0, 0.25, 0.5$ . Case (c).

## 6.2 Sloping bottom test

A solitary wave propagating over a sloping bed changes its shape gradually, slightly increasing maximum height and front steepness, till a point where a vertical front tangent is reached. This is usually called the breaking point  $BP = (t_{bp}, x_{bp}, z_{bp})$ , where  $x_{bp}$  represents the  $x$  coordinate,  $z_{bp}$  the height at  $x_{bp}$ , and  $t_{bp}$  the time of occurrence. Beyond the BP the wave tip develops, with velocities much bigger than the wave celerity, causing the wave overturning and the subsequent

TABLE 2  
*Breaking characteristics*

Test	$t_{bp}$	$x_{bp}$	$z_{bp}$	$t_{ep}$	$x_{ep}$
(a)	2.76	17.39	0.674	3.36	20.2
(b)	2.34	15.20	0.662	2.90	17.8

falling of the jet toward the flat water surface. Denote this endpoint as EP =  $(t_{ep}, x_{ep}, z_{ep})$ . Total wave mass and total energy should be theoretically conserved until EP. However beyond the BP, a loss in potential energy and the corresponding gain in kinetic energy is expected, due to the large velocities on the wave jet.

Wave breaking characteristics change, mainly according to initial wave amplitude ( $H_0$ ) and bottom topography. To study how our numerical method predicts wave breaking we run the following test cases:

- (a)  $H_0 = 0.6$ ,  $L = 25$ , slope = 1 : 22,  $x_c = 6.05$ ,  $x_s = 6$ ,
- (b)  $H_0 = 0.6$ ,  $L = 18$ , slope = 1 : 15,  $x_c = 5.55$ ,  $x_s = 5.4$ ,

and compare the results obtained here for case (b) with those reported in [15]. Here  $x_c$  denotes the  $x$  coordinate at the crest for the initial wave, and  $x_s$  the  $x$  coordinate where the bottom slope starts.

A series of numerical experiments have been made, and optimal discretization parameters found are:  $\Delta x = 0.01$ ,  $\Delta t = 0.0001$  and  $d_0 = 0.005$ ,  $f_0 = 10$  (approximately 193 BEM nodes) for all cases. Front regridding has been made according to maximum height before the BP and according to maximum velocity modulus beyond BP. Beyond the BP, and due to the complex topography of the wave front, reinitialization of  $\Psi$  and new  $\Phi(s, t)$  extension has been performed every 1000 time steps.

Table 2 shows the breaking characteristics for the test cases. Grilli et al. reported in [15] for test (b) values of  $t_{bp} = 2.41$ ,  $x_{bp} = 15.64$  and  $z_{bp} = 0.67$ . The discrepancies can be attributed to the slightly different position of the initial wave ( $x_c = 5.5$ ) and the higher order approximations used in their Lagrangian-Eulerian formulation.

In Figure 10 we show  $m(t)$  versus time for cases (a) and (b), and Figures 11 and 12 show the evolution of  $E_p$ ,  $E_k$  and  $E$  with time for cases (a) and (b) respectively. Maximum absolute error in wave mass is 0.01 before BP and 0.02 beyond BP, and maximum absolute error in total wave

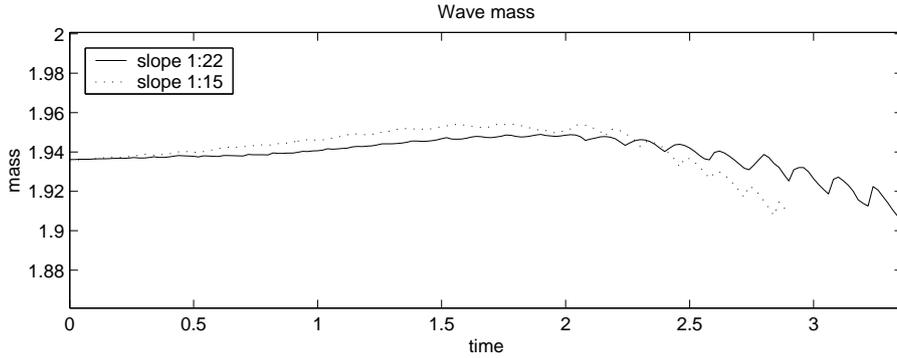


FIG. 10. Wave mass vs time. Cases (a) and (b).

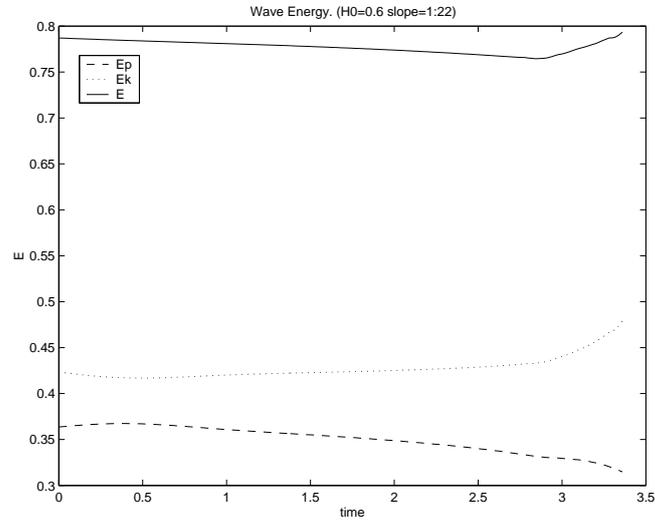


FIG. 11. Wave energy. Case (a).

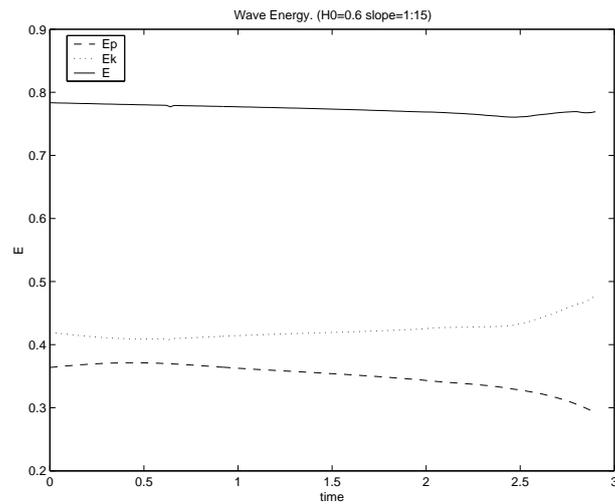


FIG. 12. Wave energy. Case (b).

energy is 0.02 near the BP. Although these errors could be improved by increasing the number of BEM nodes on the free boundary (as shown in the constant depth cases), it would require considerably more CPU time per run due to the high cost of the BEM solver. Regarding the evolution of the potential and kinetic energy of the wave we observe the expected behavior beyond the BP.

Figure 13 shows wave shape for case (a) at  $t = 0, 1, 2, 2.76, 2.94, 2.14, 3.34$  and Figure 14 shows wave shape for case (b) at  $t = 0, 1, 2, 2.34, 2.48, 2.68, 2.90$ . In Figures 15 and 16 we show in more detail the wave profiles from the BP to the EP for cases (a) and (b) respectively. Finally, in Figure 17 the front BEM nodes for case (a) and time 3.34 are shown.

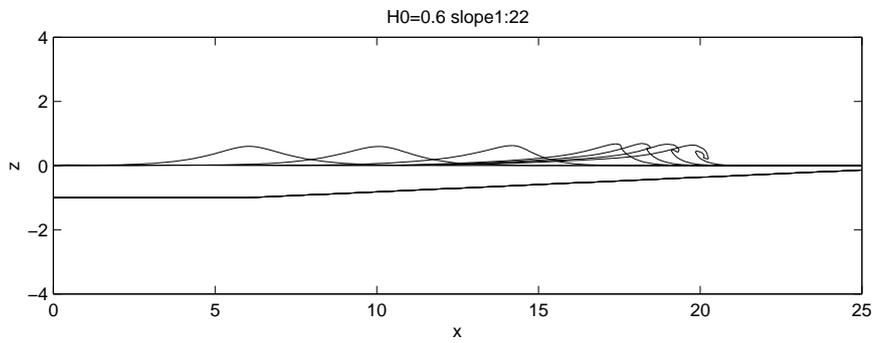


FIG. 13. Wave shape at various times. Case (a).

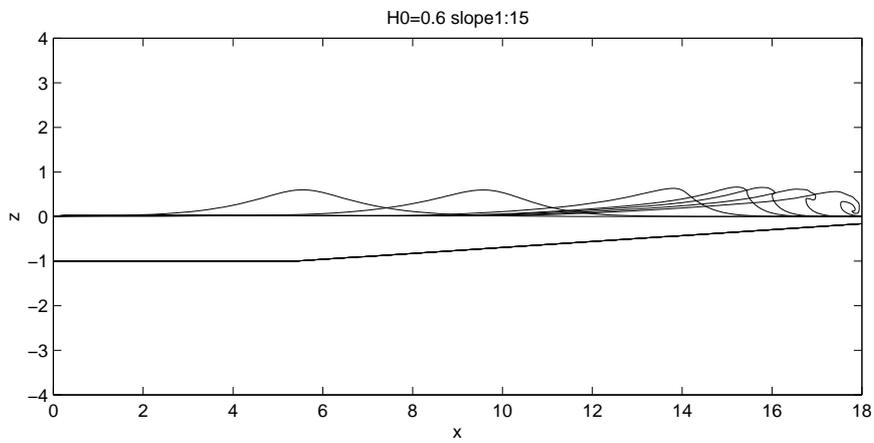


FIG. 14. Wave shape at various times. Case (a).

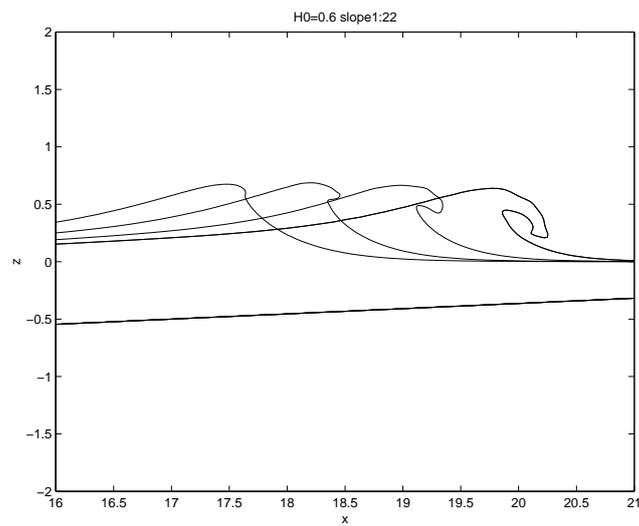


FIG. 15. Wave shape at various times. Case (a).

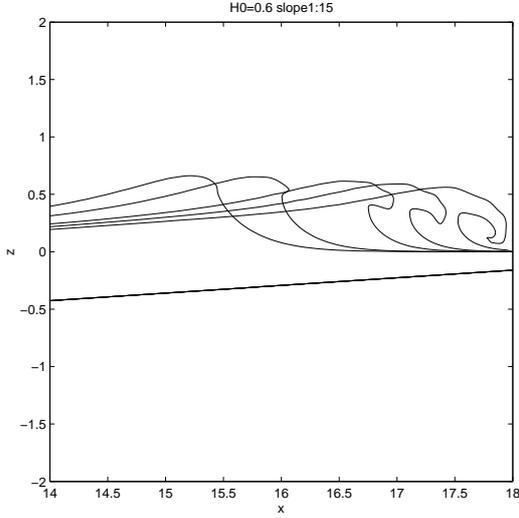
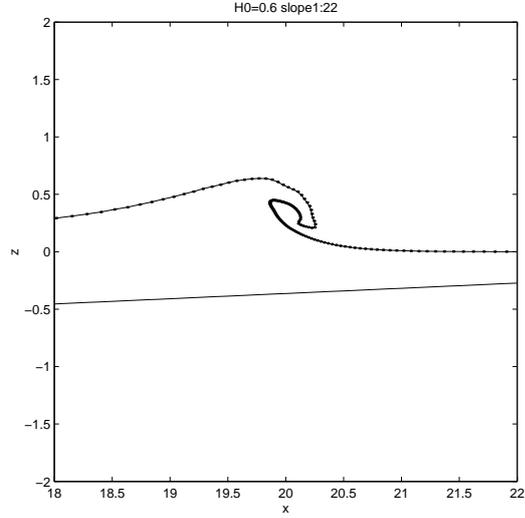


FIG. 16. Wave shape at various times. Case (b).

FIG. 17. Front BEM nodes at  $t=3.34$ . Case (a).

From these numerical experiments we conclude that the numerical method presented here is capable of reproducing wave shoaling and breaking till the touchdown of the wave jet. Considering that we use only first order approximations of the model equations, a piecewise linear approximation of the free boundary, and a first order linear BEM, the results are quite accurate. The absolute errors in mass and energy seem to be higher than those reported in [15]. This is not surprising due to the fact that in [15] a higher order BEM is used (both higher order elements to define local interpolation between nodes and spline approximation of the free boundary geometry), and time integration for the free boundary conditions is at least second order in time.

### 6.3 Sinusoidal bottom test

To see how wave shape and breaking characteristics change with bottom topography, we consider two more tests, this time with a sinusoidal shape bottom:

(c)  $H_0 = 0.6$ ,  $L = 25$ ,  $x_c = 6.05$ ,  $A_b = 0.5$ ,  $h_{\min} = 0.5$ ,

(d)  $H_0 = 0.6$ ,  $L = 25$ ,  $x_c = 6.05$ ,  $A_b = 0.8$ ,  $h_{\min} = 0.2$ ,

where  $A_b$  denotes the amplitude of the sinusoidal function that represents the bottom and  $h_{\min}$  the minimum depth.

As can be seen in Table 3, the breaking characteristics are considerably different for these simulations, and, in particular, case (c) behaves like a spilling breaker rather than the plunging

TABLE 3  
*Breaking characteristics*

Test	$t_{bp}$	$x_{bp}$	$z_{bp}$	$t_{ep}$	$x_{ep}$
(c)	1.6	12.5	0.71	1.96	14.1
(d)	1.0	10.5	0.55	1.38	13.6

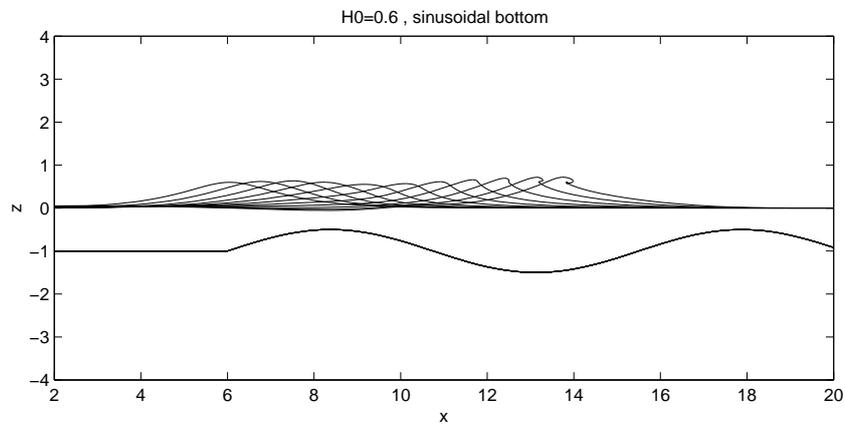


FIG. 18. Wave shape at various times. Case (c).

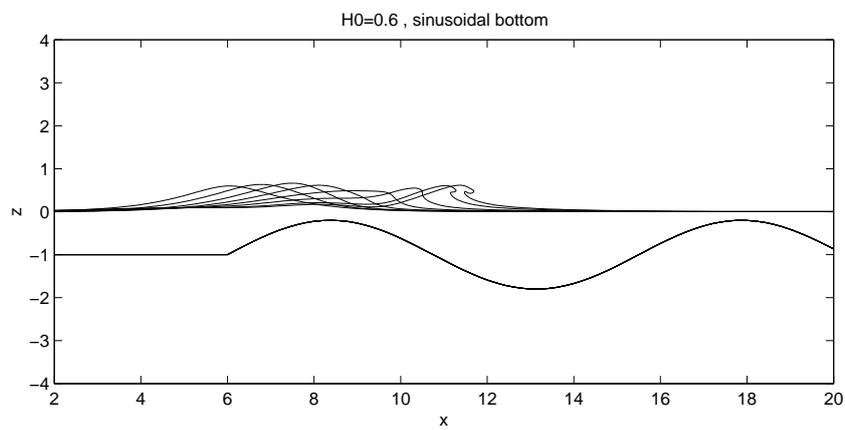


FIG. 19. Wave shape at various times. Case (d).

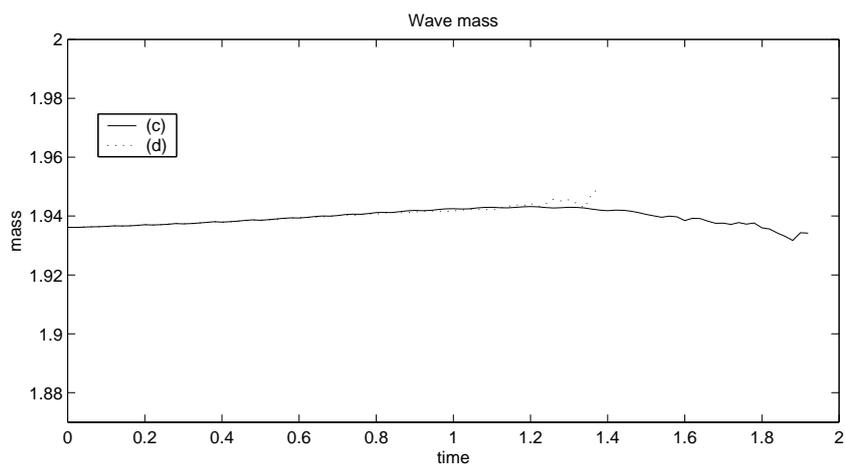


FIG. 20. Wave mass vs time. Case (c) and (d).

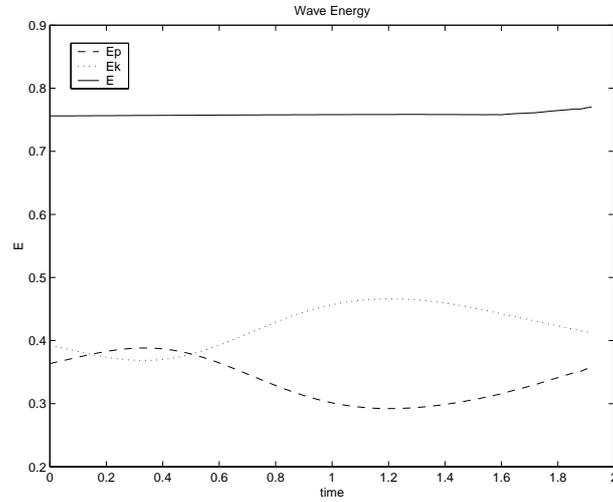


FIG. 21. Wave energy. Case (c).

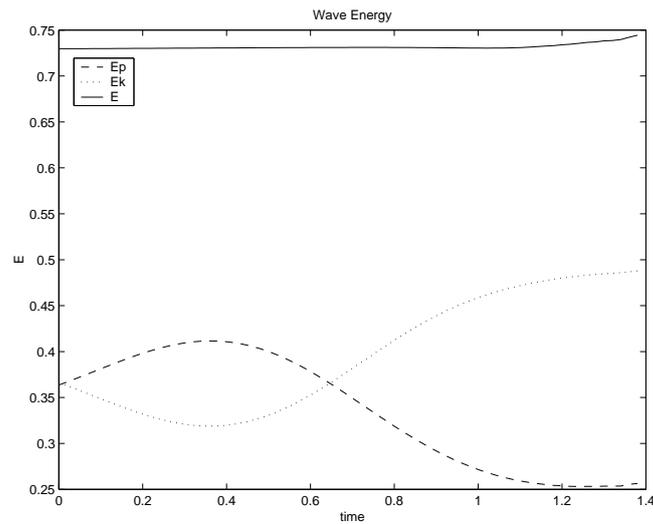


FIG. 22. Wave energy. Case (d).

breaker of cases (a) and (b). Figures 18 and 19 show wave profiles for various times corresponding to cases (c) and (d) respectively. Measurements for the mass and total energy conservation behave similar to previous cases. In Figure 20 we show the evolution of wave mass for cases (c) and (d). Finally, Figures 21 and 22 show the evolution of  $E_p$ ,  $E_k$  and  $E$  corresponding to cases (c) and (d) respectively.

These results show that, in response to the bottom topography, wave height follows a sinusoidal curve, as does the potential and kinetic wave energies, with an amplitude related to the sinusoidal bottom amplitude.

#### 6.4 Accuracy comments

The accuracy in our calculations is essentially first order. To recall our procedure, we first obtain the level set nodes in the front, and we then find the zero level set using linear interpolation from mesh values. The velocity potential is bicubically interpolated from the mesh onto BEM nodes, and then the velocity components at level set nodes are obtained by linear interpolation from velocities computed on BEM nodes. We note that we do not see a decrease in error with decreasing time step, most probably because interpolation errors play a dominant role. Only the velocity potential error decreases with decreasing time step, and this is probably due to the bicubic interpolation of this variable.

Additionally, we have used a first order boundary element method in these simulations. Indeed, we later implemented a second order version in both space and time for the level set update and for the update of the potential  $G$  equation. However, our results show that this additional accuracy was overshadowed by the error in the boundary element solver. We have recently built a cubic Hermite second order method, which is now being incorporated into our electrospray simulations, and we will report on this work elsewhere.

To summarize, we have built a coupled level set–boundary element algorithm for modeling a class of free boundary problems, in particular, two-dimensional breaking waves over sloping beaches. The algorithm rests on a fully nonlinear potential model for a single fluid with appropriate boundary conditions, with both the interface location and the velocity potential recast as an embedded function throughout the domain. The use of a boundary integral method avoids far-field boundary conditions for the air, and the use of a level set method avoids complex gridding. The formulation is unchanged in three dimensions; we shall report elsewhere on the extension of this approach to three-dimensional flow, as well as introduce a new model for what happens when the breaking wave reconnects with the surface.

#### Acknowledgments

All work was performed at the Lawrence Berkeley National Laboratory, and the Mathematics Department of the University of California at Berkeley. The authors would like to thank S. T. Grilli for the use of an initialization computer code of Tanaka’s method, and for helpful comments. The first author was partially supported by the Spanish DGI Project BFM 00-1324 and would like to thank Nilo Bobillo and Omar Menendez for their valuable collaboration. The third author was supported by the Applied Mathematical Sciences Research Program of the Office of Mathematical, Information, and Computational Sciences, U.S. Department of Energy, under contract DE-AC05-00OR22725 with UT-Battelle, LLC.

#### REFERENCES

1. ADALSTEINSSON, D. & SETHIAN, J. A. A fast level set method for propagating interfaces. *J. Comput. Phys.* **118** (1995), 269–277. Zbl 0823.65137 MR 1329634
2. ADALSTEINSSON, D. & SETHIAN, J. A. The fast construction of extension velocities in level set methods. *J. Comput. Phys.* **148** (1999), 2–22. Zbl 0919.65074 MR 1665209
3. ADALSTEINSSON, D. & SETHIAN, J. A. Transport and diffusion of material quantities on propagating interfaces via level set methods. *J. Comput. Phys.* **185** (2002), 271–288. Zbl 1047.76093 MR 2010161

4. BEALE, J. T., HOU, T. Y., & LOWENGRUB, J. Convergence of a boundary integral method for water waves. *SIAM J. Numer. Anal.* **33** (1996), 1797–1843. Zbl 0858.76046 MR 1411850
5. BONNET, M. *Boundary Integral Equation Methods for Solids and Fluids*. Wiley (1995). Zbl 0920.73001
6. BREBBIA, C. A., TELLES, J. C. F., & WROBEL, L. C. *Boundary Element Techniques*. Springer (1984). Zbl 0556.73086 MR 0934922
7. CHEN, S., MERRIMAN, B., OSHER, S., & SMEREKA, P. A simple level set method for solving Stefan problems. *J. Comput. Phys.* **135** (1997), 8–29. Zbl 0889.65133 MR 1461705
8. CHANG, Y. C., HOU, T. Y., MERRIMAN, B., & OSHER, S. J. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows. *J. Comput. Phys.* **124** (1996), 449–64. Zbl 0847.76048 MR 1383769
9. CHOPP, D. L. Some improvements of the fast marching method. *SIAM J. Sci. Comput.* **23** (2001), 230–244. Zbl 0991.65105 MR 1860913
10. CHORIN, A. J. Numerical solution of the Navier–Stokes equations. *Math. Comp.* **22** (1968), 745–762. Zbl 0198.50103 MR 0242392
11. CHRISTENSEN, E. D. & DEIGAARD, R. Large eddy simulation of breaking waves. *Coastal Engrg.* **42** (2001), 53–86.
12. EGGERS, J. Nonlinear dynamics and breakup of free-surface flows. *Rev. Mod. Phys.* **69** (1997), 865–929.
13. GRAY, L. J. Evaluation of singular and hypersingular Galerkin boundary integrals: direct limits and symbolic computation. *Singular Integrals in the Boundary Element Method*, V. Sladek and J. Sladek (eds.), Computational Mechanics Publ. (1998), Chapter 2, 33–84.
14. GRAY, L. J., PHAN, A. -V. & KAPLAN, T. Boundary integral evaluation of surface derivatives. *SIAM J. Sci. Comput.* **26** (2004), 294–312. Zbl pre02138744 MR 2114345
15. GRILLI, S. T., GUYENNE, P., & DIAS, F. A fully non-linear model for three dimensional overturning waves over an arbitrary bottom. *Internat. J. Numer. Methods Fluids* **35** (2001), 829–867. Zbl 1039.76043
16. GRILLI, S. T., SVENDSEN, I. A., & SUBRAMANYA, R. Breaking criterion and characteristics for solitary waves on slopes. *J. Waterway, Port, Coastal, and Ocean Engrg.* (June 1997).
17. GRILLI, S. T. Modeling of non-linear wave motion in shallow water. *Computational Methods for Free and Moving Boundary Problems in Heat and Fluid Flow*, L. C. Wrobel and C. A. Brebbia (eds.), Computational Mechanics Publ., Southampton (1995), 91–122.
18. GRILLI, S. T. & SUBRAMANYA, R. Numerical modeling of wave breaking induced by fixed or moving boundaries. *Comput. Mech.* **17** (1996), 374–391. Zbl 0851.76043 MR 1395439
19. LIN, P., CHANG, K., & LIU, P. L. Runup and rundown of solitary waves on sloping beaches. *J. Waterway, Port, Coastal, and Ocean Engrg.* (Sep/Oct 1999).
20. MALLADI, R., SETHIAN, J. A., & VEMURI, B. C. Shape modeling with front propagation: a level set approach. *IEEE Trans. Pattern Anal. Machine Intelligence* **17** (1995), 158–175.
21. MARTIN, P. A., RIZZO, F. J., & CRUSE, T. A. Smoothness-relaxation strategies for singular and hypersingular integral equations. *Internat. J. Numer. Meth. Engrg.* **42** (1998), 885–906. Zbl 0913.65105 MR 1630300
22. MARTIN, P. A. & RIZZO, F. J. On boundary integral equations for crack problems. *Proc. Roy. Soc. London* **A421** (1989), 341–355. Zbl 0674.73071 MR 0985268
23. MARTIN, P. A. & RIZZO, F. J. Hypersingular integrals: how smooth must the density be? *Internat. J. Numer. Meth. Engrg.* **39** (1996), 687–704. Zbl 0846.65070 MR 1377122
24. NOTZ, P. K. & BASARAN, O. A. Dynamics of drop formation in an electric field. *J. Colloid Interface Sci.* **213** (1999), 218–237.
25. OSHER, S. & SETHIAN, J. A. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations. *J. Comput. Phys.* **79** (1988), 12–49. Zbl 0659.65132 MR 0965860
26. PEREGRINE, D. H. Breaking waves on beaches. *Ann. Rev. Fluid Mech.* **15** (1983), 149–178.

27. SETHIAN, J. A. An analysis of flame propagation. Ph.D. Dissertation, Dept. of Mathematics, Univ. of California, Berkeley, CA (1982).
28. SETHIAN, J. A. Curvature and the evolution of fronts. *Comm. Math. Phys.* **101** (1985), 487–499. Zbl 0619.76087 MR 0815197
29. SETHIAN, J. A. Numerical methods for propagating fronts. *Variational Methods for Free Surface Interfaces*, P. Concus and R. Finn (eds.), Springer (1987), 155–164. Zbl 0618.65128 MR 0872900
30. SETHIAN, J. A. A fast marching level set method for monotonically advancing fronts. *Proc. Nat. Acad. Sci. U.S.A.* **93** (1996), 1591–1595. Zbl 0852.65055 MR 1374010
31. SETHIAN, J. A. *Level Set Methods and Fast Marching Methods*. Cambridge Monogr. Appl. Comput. Math., Cambridge Univ. Press (1999). Zbl 0973.76003 MR 1700751
32. SETHIAN, J. A. & SMEREKA, P. Level set methods for fluid interfaces. *Ann. Rev. Fluid Mech.* **35** (2003), 341–372. Zbl 1041.76057 MR 1967017
33. SETHIAN, J. A. & STRAIN, J. D. Crystal growth and dendritic solidification. *J. Comput. Phys.* **98** (1992), 231–253. Zbl 0752.65088 MR 1150905
34. SUSSMAN, M., SMEREKA, P., & OSHER, S. J. A level set approach to computing solutions to incompressible two-phase flow. *J. Comput. Phys.* **114** (1994), 146–159. Zbl 0808.76077
35. TANAKA, M. The stability of solitary waves. *Phys. Fluids* **29** (1986), 650–655. Zbl 0605.76025 MR 0828181
36. YAN, F., FAROUK, B., & KO, F. Numerical modeling of an electrostatically driven liquid meniscus in the cone-jet mode. *Aerosol Sci.* **34** (2003), 99–116.
37. YU, J.-D., SAKAI, S., & SETHIAN, J. A. A coupled level set projection method applied to ink jet simulation. *Interfaces Free Bound.* **5** (2003), 459–482. Zbl 1042.35059 MR 2031466
38. ZELT, J. A. The run-up of non-breaking and breaking solitary waves. *Coastal Engrg.* **15** (1991), 205–246.
39. ZHU, J. & SETHIAN, J. A. Projection methods coupled to level set interface techniques. *J. Comput. Phys.* **102** (1992), 128–138. Zbl 0751.76050