# Wave Breaking over Sloping Beaches using a Coupled Boundary Integral-Level Set Method [*]

M. Garzon[†], D. Adalsteinsson[‡], L. Gray[§], and J.A. Sethian[¶]

December 8, 2003

**Abstract**

We present a numerical method for tracking breaking waves over sloping beaches. We use a fully non-linear potential model for incompressible, irrotational and inviscid flow, and consider the effects of beach topography on breaking waves. The algorithm uses a Boundary Element Method (BEM) to compute the velocity at the interface, coupled to a Narrow Band Level Set Method to track the evolving air/water interface, and an associated extension equation to update the velocity potential both on and off the interface. The formulation of the algorithm is applicable to two and three dimensional breaking waves; in this paper, we concentrate on two-dimensional results showing wave breaking and rollup, and perform numerical convergence studies and comparison with previous techniques.

## 1 Introduction and overview

The flow pattern of wave breaking over sloping beach floors has been extensively studied, both experimentally and by means of numerical simulation, due to the interest in surf-zone dynamics, sediment transport problems, and impact forces on off-shore and near-shore structures.

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

[†] Dept. of Applied Mathematics, Univ. of Oviedo, Spain

[‡] Dept. of Mathematics, Univ. of North Carolina, Chapel Hill

[§] Computer Science and Mathematics Division, Oak Ridge National Laboratory

[¶] Dept. of Mathematics, University of California, Berkeley, and Mathematics Department, Lawrence Berkeley National Laboratory.

The most commonly used mathematical models, based on various assumptions, are the non-linear shallow water equations, the non-linear Boussinesq models and the non-linear fully potential models (see for example [16], [34], [24], [14]). 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 ([18]). Slightly different approaches which also include turbulent effects can be found in ([10]). The physical validity of these various models to accurately predict wave breaking is difficult to assess, since physical experiments (laser 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 as 'Fully non-linear potential model' (FNPM) and are able to model strongly non-linear waves. This model has been extensively used by, for example Grilli et al. ([14], [15], [17]), 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 regridding 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 that avoid these regridding and topological issues are available. One option is Level Set Methods, introduced by Osher and Sethian [23]. A large collection of simulations have been performed coupling level set methods to Chorin's projection method ([9]) 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 ([7, 31, 33, 30, 35]). In this approach, boundary conditions are required for both fluids. However, it can difficult to specify appropriate numerical boundary conditions for the air region in open domains that do not adversely affect the calculation.

In this paper we present a different approach, 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 idea of the present work is to apply this approach to the numerical solution of the FNPM for two-dimensional waves shoaling over a constant and sloping bottom. 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. The formulation is unchanged in three dimensions, offering the possibility of computing complex breaking wave motions.

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 that equals the correct velocity potential along the interface; here, we were motivated by the recent work in [3] for transport and diffusion of material quantities on propagating interfaces using level set methods. To advance the position of the interface, first a nodal representation is extracted from the level set function, along with the velocity potential at these nodes. Then, a boundary integral method is used to compute the velocity at each of these nodes. This velocity is then extended throughout the narrow band, and both 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. This process is then repeated.

## 2 The Governing Equations

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 $t$ (see Figure 1).

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

$$\begin{aligned} \nabla \mathbf{u} &= 0 \ \text{ in } \ \Omega(t) \qquad (1) \\ \mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u} &= \frac{-\nabla p}{\rho} + \mathbf{b} \ \text{ in } \ \Omega(t) \qquad (2) \end{aligned}$$

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 $\rho$ is the fluid density.

Since the motion is irrotational, fluid particles do not rotate and vorticity vanishes everywhere in the field of flow. This means that the velocity field

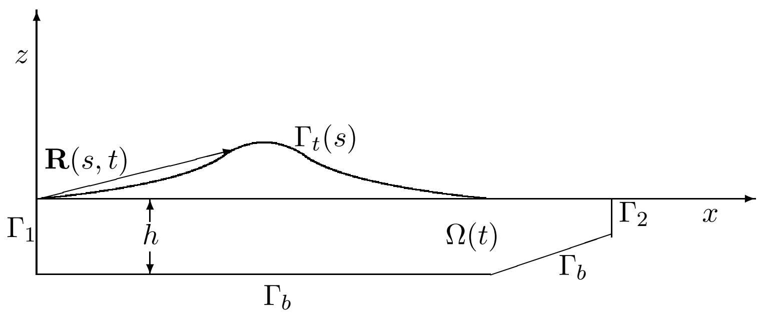

Figure 1: The domain

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 $\mathbf{b} = -g\mathbf{z}$, being $\mathbf{z}$ a unit vector in the vertical direction ($\mathbf{z} = \nabla z$), the momentum equation (2) reduces to the so-called Bernoulli's 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 $\phi \to \phi + \int_0^s C(s)ds$, which does not affect the velocity field, and, together with the transformation $p \to p - p_a$, which does not affect the basic Euler equations (they only depend upon $\nabla p$), we have that

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

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 the stress tensor between water and air leads to $p = p_a$, and thus
$$\phi_t + \frac{1}{2}(\nabla\phi\cdot\nabla\phi) + gz = 0 \text{ on } \Gamma_t(s)\ .$$
2. If $\mathbf{R}(s,t) = (X(s,t), Y(s,t))$ is the position vector of a fluid particle on the free surface, we have the kinematic boundary condition
$$\mathbf{R}_t(s,t) = \mathbf{u}(\mathbf{R}(s,t),t) \text{ on } \Gamma_t(s)$$
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:

$$\begin{aligned}
\mathbf{u} &= \nabla\phi \ \text{ in } \ \Omega(t) && (4)\\
\Delta\phi &= 0 \ \text{ in } \ \Omega(t) && (5)\\
\mathbf{R}_t &= \mathbf{u} \ \text{ on } \ \Gamma_t(s) && (6)\\
\frac{D\phi}{Dt} &= -gz + \frac{1}{2}(\nabla\phi\cdot\nabla\phi) \ \text{ on } \ \Gamma_t(s) && (7)\\
\phi_n &= 0 \ \text{ on } \ \Gamma_b\cup\Gamma_1\cup\Gamma_2 && (8)
\end{aligned}$$

with the material derivative defined in the standard way as

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

## 3 Embedding the Equations of Motion in a Level Set framework

Level set [23] 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 [25, 26] and on the link between front propagation and hyperbolic conservation laws discussed in [27]. 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 for 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 that

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

For the wave problem, let $\Omega_1$ be a fictitious fixed rectangular 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, parametrize the curve that represents the initial position of the front by its arclength: $s \rightarrow \Gamma_0(s)$. Let be $\mathbf{u}(x, z, t)$ the velocity field, the trajectory of a fluid particle at initial position $s$ is given by the solution of

$$\begin{aligned} \mathbf{R}_t(s,t) &= \mathbf{u}(\mathbf{R}(s,t),t) && (10) \\ \mathbf{R}(s,0) &= (X(s,0), Z(s,0)) \ . && (11) \end{aligned}$$

For any $t > 0$ the free boundary curves are parametrized with the same parameter $s$, $s \rightarrow \Gamma_t(s)$, 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,

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

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

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

Applying the chain rule, we have

$$G_t + \mathbf{u} \cdot \nabla G = \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - gz, \qquad (12)$$

which again holds on $\Gamma_t(s)$. Note that $\mathbf{u}$ and the right hand side of equation (12) are only defined on $\Gamma_t(s)$. In order to solve equation (12) embedded in the whole domain $\Omega_1$, we need to extend these variables off the front; this is discussed below.

In summary, the Level Set model equations, written in a complete Eulerian framework, are

$$
\begin{aligned}
\mathbf{u} &= \nabla\phi \text{ in } \Omega(t) & (13)\\
\Delta\phi &= 0 \text{ in } \Omega(t) & (14)\\
\Psi_t + \nabla\Psi\cdot\mathbf{u}_{\text{ext}} &= 0 \text{ in } \Omega_1. & (15)\\
G_t + \mathbf{u}_{\text{ext}}\cdot\nabla G &= f_{\text{ext}} \text{ in } \Omega_1 & (16)\\
\phi_n &= 0 \text{ on } \Gamma_b\cup\Gamma_1\cup\Gamma_2 & (17)
\end{aligned}
$$

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

# 4 Numerical Approximations and Algorithms

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

## 4.1 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 (15) and (16) 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 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 the stream function and velocity potential we have

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

Under sensible assumptions about the smoothness of $\phi$ and $\psi$, these are just the Cauchy-Riemman equations which are satisfied by the real and imaginary parts of the complex potential function $W = \phi + i\psi$. It is an analytical function of the complex variable $Z = x+iz$ in the domain occupied by the fluid. By interchanging the role of the variables $Z$ and $W$, we can take $\phi$ and $\psi$ as independent variables, as $W = \phi+i\psi$ 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

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

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

$$\begin{aligned} \frac{dq^3}{d\phi} &= -\frac{3}{F^2}\sin\theta \quad \text{on } \psi = 1 && (18) \\ \theta &= 0 \quad \text{on } \psi = 0, && (19) \end{aligned}$$

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

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 region of the unit strip $0 < \psi < 1$, decays at infinity, and satisfies the boundary conditions (18) and (19). Tanaka's method provides a way to solve the previous outlined equations in terms of the new variables $\tau$, $\theta$ and a full description of the algorithm can be found in [32].

### 4.2 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 [23], we use second order in space upwind differences to approximate the gradient in the level set equation, and a first order time scheme to update the solution. For boundary conditions, homogeneous flux boundary conditions are usually chosen. This has been 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 using the

Fast Marching Method [28], 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, required to update the fixed grid Eulerian partial differential equations, we use the extension methodology developed by Adalsteinsson and Sethian [2]. The idea of building extension velocities was first introduced in [19]; in that approach, the extension velocity at any grid point in the domain was taken as equal to the velocity on the closest point on the front itself. As shown in [6], this is equivalent to solving the equation $\nabla V \cdot \nabla \Psi = 0$ for the velocity $V$, and in [6] this 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 further developed to build extension values for arbitrary material quantities (not just velocities) whose evolution affects the underlying interface dynamics.

### 4.3 The Boundary Integral equation and the BEM approximation

A first order boundary element method is used to approximate equation (13). 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 cleary simpler than remeshing the entire domain.

The Laplace equation for the velocity potential (14) is solved by approximating the corresponding boundary integral equation. Boundary conditions are given by (17) and, on the free boundary, at each time step, by the updated potential velocity given by equation (16). The solution of the integral equation completes the knowledge of the potential and its normal derivative on the boundary, and the potential gradient on the free surface (the velocity $\mathbf{u}$) is then computed in a post-processing step.

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

$$\mathcal{P}(P) = \phi(P) + \lim_{P_I \to P} \int_{\Sigma} \left[ \phi(Q) \frac{\partial G}{\partial \mathbf{n}}(P_I, Q) - G(P_I, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) \right] \mathrm{d}Q = 0 \ , \tag{20}$$

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

$$G(P, Q) = -\frac{1}{2\pi} \log(r) \ . \tag{21}$$

The integral equation is usually written with the $\frac{\partial G}{\partial \mathbf{n}}$ singular integral evaluated as a Cauchy Principal Value (CPV), resulting in an 'interior angle' coefficient $c(P)$ multiplying the leading $\phi(P)$ term [4, 5]. The reason for employing the seemingly more complicated limit process will become clear in the discussion of gradient evaluation. The exterior limit equation

$$\lim_{P_E \to P} \int_{\Sigma} \left[ \phi(Q) \frac{\partial G}{\partial \mathbf{n}}(P_E, Q) - G(P_E, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) \right] \mathrm{d}Q = 0 \ . \tag{22}$$

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 Eq. (20) 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

$$\int_{\Sigma} \hat{\psi}_k(P) \mathcal{P}(P) \, \mathrm{d}P = 0 \ , \tag{23}$$

where the weight functions $\hat{\psi}_k(P)$ are comprised of all shape functions which are non-zero at a particular node $P_k$ [4]. 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 [11].

As noted above, for the wave problem and moving boundary problems in general, knowledge of the normal flux is not sufficient, the complete gradient of $\phi$ is required to compute the surface velocity. The remainder of this section will briefly discuss the algorithm for computing this gradient.

¿From Eq. (20) a gradient component in the direction $\mathbf{E}_k$ can be expressed as

$$\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \lim_{P_I \to P} \int_{\Sigma} \left[ \frac{\partial G}{\partial \mathbf{E}_k}(P_I, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q) \frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_I, Q) \right] \mathrm{d}Q \ . \tag{24}$$

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 [13, 21, 22, 20]. As described in [12], a Galerkin approximation of this equation,

$$
\begin{aligned}
&\int_\Sigma \hat{\psi}_k(P)\frac{\partial \phi(P)}{\partial \mathbf{E}_k}\,\mathrm{d}P = \\
&\lim_{P_I\to P}\int_\Sigma \hat{\psi}_k(P)\int_\Sigma \left[\frac{\partial G}{\partial \mathbf{E}_k}(P_I,Q)\frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q)\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_I,Q)\right]\mathrm{d}Q\,\mathrm{d}P
\end{aligned}
\tag{25}
$$

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

Interpolating $\partial\phi(P)/\partial\mathbf{E}_k$ in terms of the linear shape functions results in a simple system of 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 of Eq. (25) are quite expensive.

The computational cost of this procedure can be significantly reduced by exploiting the exterior limit equation, Eq. (22). It appears to be useless for computing tangential derivatives for, lacking the free term, the corresponding derivative equation takes the form

$$
0 = \lim_{P_E\to P}\int_\Sigma \left[\frac{\partial G}{\partial \mathbf{E}_k}(P_E,Q)\frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q)\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}(P_E,Q)\right]\mathrm{d}Q\ , \tag{26}
$$

and the derivatives obviously do not appear. However, subtracting this equation from Eq. (24) yields (with shorthand notation)

$$
\frac{\partial \phi(P)}{\partial \mathbf{E}_k} = \left\{\lim_{P_I\to P} - \lim_{P_E\to P}\right\}\int_\Sigma \left[\frac{\partial G}{\partial \mathbf{E}_k}\frac{\partial \phi}{\partial \mathbf{n}}(Q) - \phi(Q)\frac{\partial^2 G}{\partial \mathbf{E}_k \partial \mathbf{n}}\right]\mathrm{d}Q\ . \tag{27}
$$

The advantage of this formulation is that now *only the terms that are discontinuous crossing boundary* contribute to the integral. In particular, all non-singular integrations, by far the most time consuming, drop out. The calculation of the right hand side in Eq. (27) 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 the Galerkin form of Eq. (27) can be found in [12].

### 4.4 The velocity potential updating

The potential equation (16) is a convection equation with a strong non-linear source term, and homogeneous Newmann boundary conditions are imposed on the boundary of $\Omega_1$. To update this equation in time, note that it is similar to (15) except that it has a nonlinear source term, and therefore similar schemes can be employed. 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

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

are the backward and forward finite approximation for the derivative in the $x$ direction (and the same expressions for $D_{i,j}^{-z}$ and $D_{i,j}^{+z}$.) Note that for simplicity we have written $u, v, G, f$ instead of $u_{\text{ext}}, v_{\text{ext}}$, $G_{\text{ext}}, f_{\text{ext}}$, and we have employed 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 reinitialize $G_{i,j}^n$.

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 choose to 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 quantities defined on the front off the front works very well for the velocity front, because it maintains the signed distance function for the level sets of $\Psi$. However, for the wave problem, there are large variations in $f$ along the front together with its topological structure when overturning, the previous method creates strong $G$ and $f$ gradients in $\Omega_1$. This fact strong limits the ratio between grid spacing in $\Omega_1$ and the time step needed to maintain accuracy (see the section on numerical experiments).

### 4.5 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 [8] 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 in 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.

### 4.6 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 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 (14), using the Boundary Element Method. This yields the velocity $\mathbf{u}$ and source term $f$ on the front nodes.

4. Extend $\mathbf{u}$ and $f$ off the front onto $\Omega_1$.

5. Update $G$ using (16) in $\Omega_1$.

6. Move the front with velocity $\mathbf{u}$ using (15) in $\Omega_1$

7. Interpolate (bi-cubic interpolation) $G$ from grid points of $\Omega_1$ to the front nodes to obtain new boundary conditions for (14). 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 $G^{n+1}$ in $\Omega_1$ 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 $\Phi^{n+1}$.

### 4.7 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 each time step. The wave mass above $z = 0$ is given by

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

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

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

which is the potential energy with respect to $z = 0$. For the kinetic energy

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

where the divergence theorem has been applied to the three formulas and we have used the fact $\frac{\partial \phi}{\partial n} = 0$ on $\Gamma_b, \Gamma_1, \Gamma_2$. These integrals are approximated by a composite trapezoidal rule, using the values of the quantities on the free boundary BEM nodes. 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 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 characteristic obtained here with those reported elsewhere, for example in [15].

# 5 Numerical Results

The system of equations to be discretized is a non-linear system of strongly coupled partial differential equations. First order in time and second order in space schemes are used for equation (15); first order in time and in space schemes are used for equation (16); 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.

## 5.1 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 $\partial\Omega(t)$ for the input to the boundary integral calculation, 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. We ran 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 a given solitary wave parameters ($H_0$ and length L in the $x$ direction) Tanaka's method gives 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 in LS nodes. Denote by $(x_c, z_c)$, $\phi_c$, $u_c$, $v_c$ the computed values at LS nodes, $L^2(z) = \| z_c - z_{ex} \|_{L^2(\Gamma_s(t))}$, $L^2(\phi) = \| \phi_c - \phi_{ex} \|_{L^2(\Gamma_s(t))}$, $L^2(u) = \| u_c - u_{ex} \|_{L^2(\Gamma_s(t))}$ and $L^2(v) = \| v_c - v_{ex} \|_{L^2(\Gamma_s(t))}$ the $L^2$ norm of the errors. Table 1 shows these errors at the final time $t = 0.5$ for the various test cases.

| 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 |

Table 1: Values of the $L^2$ error norms at $t = 0.5$

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 (possibly due to the accumulation of round off errors). Only the velocity potential gains accuracy when $\frac{\Delta x}{\Delta t} = 100$ and this is due to the high $G$ and $f$ gradients occurring on $\Omega_1$.

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 behaviour of these quantities with respect to 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 $\Gamma_s(t)$. This gives 1720 BEM nodes on the moving front and 196 nodes for the fixed boundaries. For this discretization of the 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 case (e) and (f) are almost identical to those obtained for case (c) and (d) respectively, which means that accuracy in velocity is not gained by increasing the number of bem nodes. However, as 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.

¿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 smothing was not required.

Case (c) discretization parameters give sufficient accuracy and we show wave profiles, velocity potential and velocity components for various times in Figures 7, 8 and 9 respectively.

### 5.2 Sloping bottom test

A solitary wave propagating over a sloping bed changes its shape gradually, slightly increasing maximum height and front steepness, until 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 occurence. Beyond the BP the wave tip develops, with velocities much bigger than the wave celerity, causing wave overturning and the subsequent 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 the 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 ([14]). 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 carried out, 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 implemented 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 $\Phi(s,t)$ extensions have been performed every 1000 time steps.

Table 2 shows the breaking characteristics for the test cases. Grilli et all reported in ([14]) 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 case (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. The maximum absolute error in wave mass is 0.01 before $BP$, and 0.02 beyond BP, and the maximum absolute error in total wave energy is 0.02 near the BP. Although these errors could be improved by increasing

| 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 |

Table 2: Breaking characteristics

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 (which can be significantly reduced using fast methods). 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.

¿From these numerical experiments we conclude that the numerical method presented here is capable of reproducing wave shoaling and breaking until 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 ([14]). This is not surprising due to the fact that in ([14]) 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.

### 5.3 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 $A_b$ denote 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 breaker of case (a) and (b). In Figures 18 and 19 we show wave profiles for various times corresponding to case (c) and (d) respectively. Measurements for the mass and total energy conservation behave similarly 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.

| 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 |

Table 3: Breaking characteristics

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.

To summarize, we have built a coupled level-set boundary element algorithm for modeling 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.

**Acknowledgements:**

All work was performed at the Lawrence Berkeley National Laboratory, and the Mathematics Dept. 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. First author was partially supported by the Spanish DGI Project BFM2000-1324 and would like to thank Nilo Bobillo and Omar Menedez for their valuable collaboration.

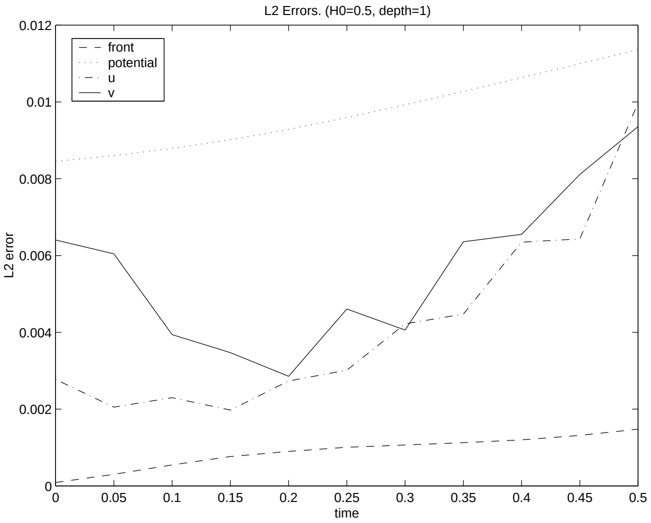

Figure 2: $L^2(z), L^2(\phi), L^2(u), L^2(v)$ vs time for case (c)

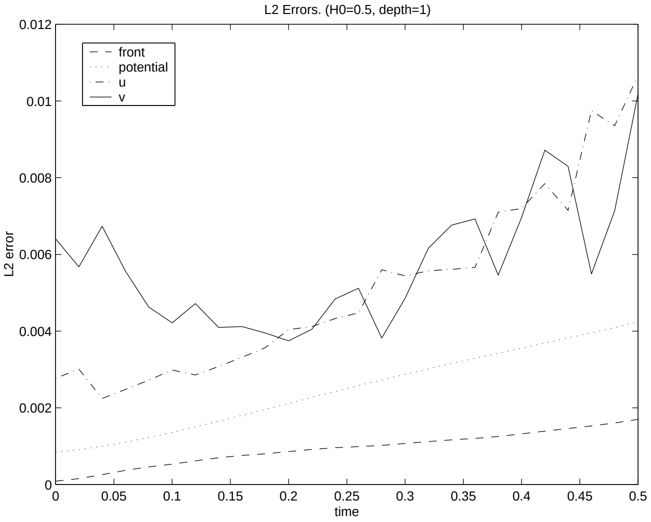

Figure 3: $L^2(z), L^2(\phi), L^2(u), L^2(v)$ vs time for case (d)

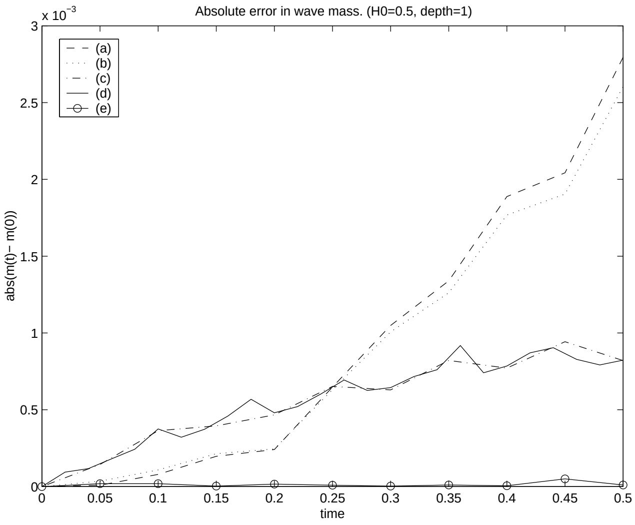

Figure 4: Absolute error in wave mass

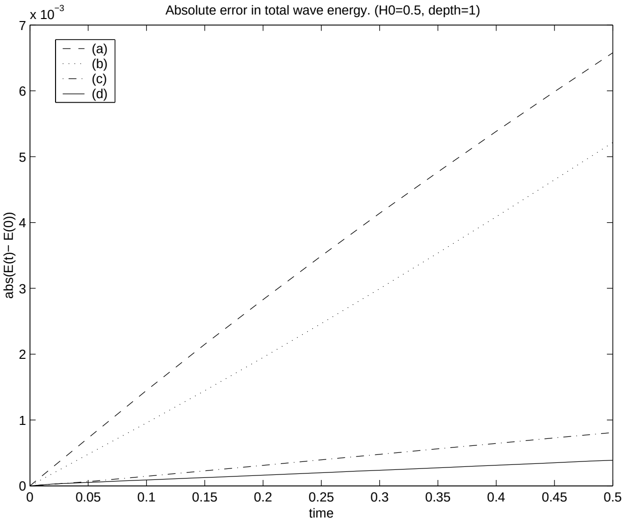

Figure 5: Absolute error in wave total energy

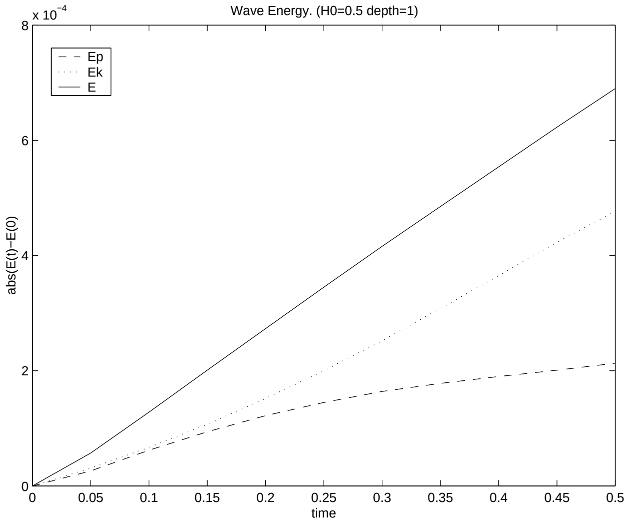

Figure 6: Absolute error in potential, kinetic and total energy. Case (e)

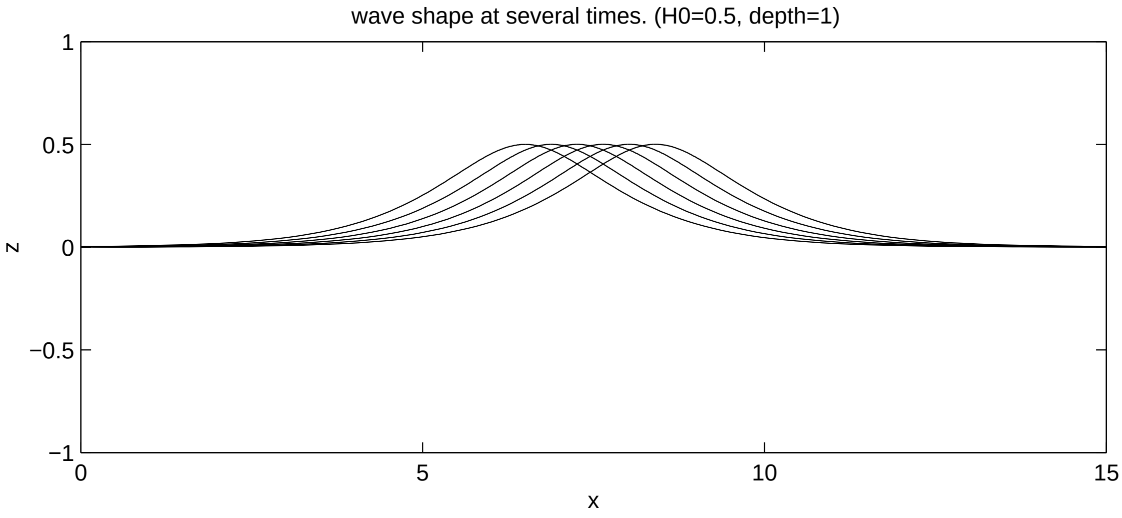

Figure 7: Front location at $t = 0, 0.1, 0.2, 0.3, 0.4, 0.5$. Case (c)

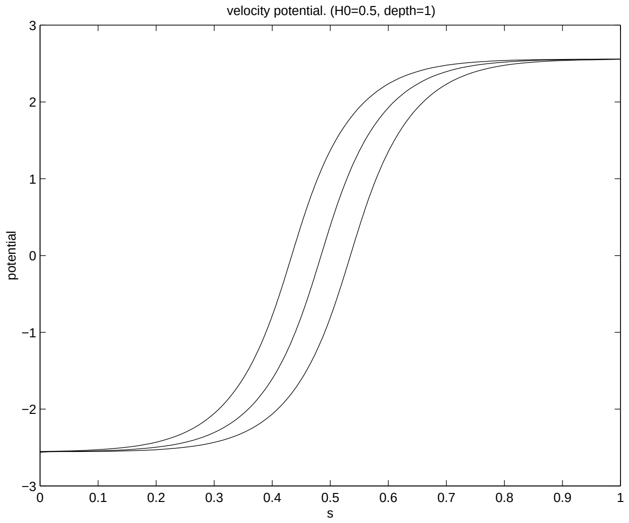

Figure 8: Velocity potential at $t = 0, 0.25, 0.5$. Case (c)

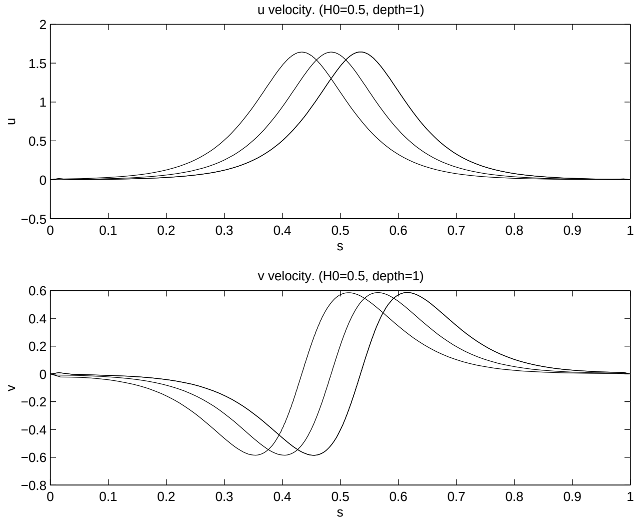

Figure 9: Velocity components at $t = 0, 0.25, 0.5$. Case (c)

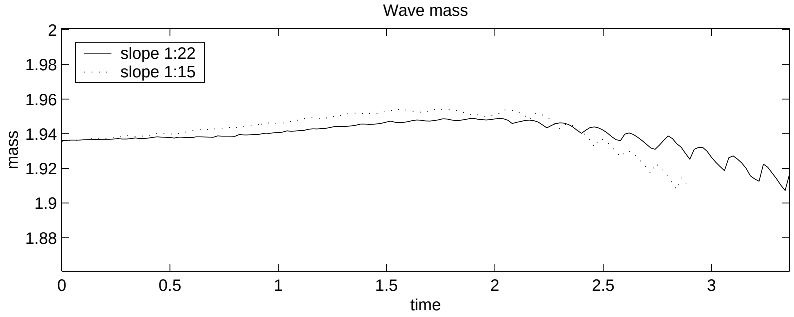

Figure 10: Wave mass vs time. Case (a) and (b)

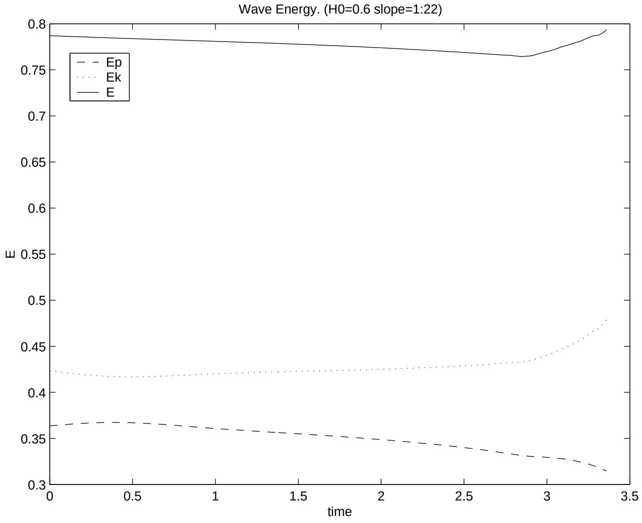

Figure 11: Wave energy. Case (a)

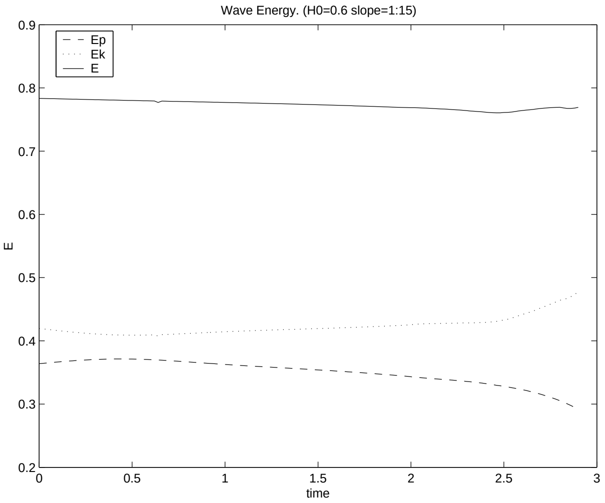

Figure 12: Wave energy. Case (b)

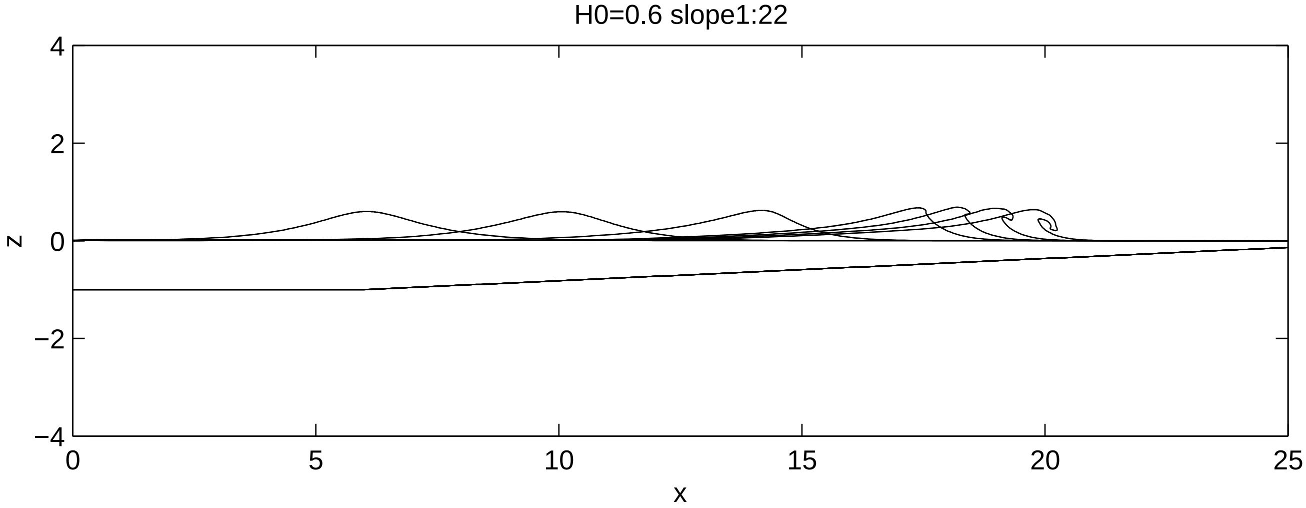

Figure 13: Wave shape at various times. Case (a)

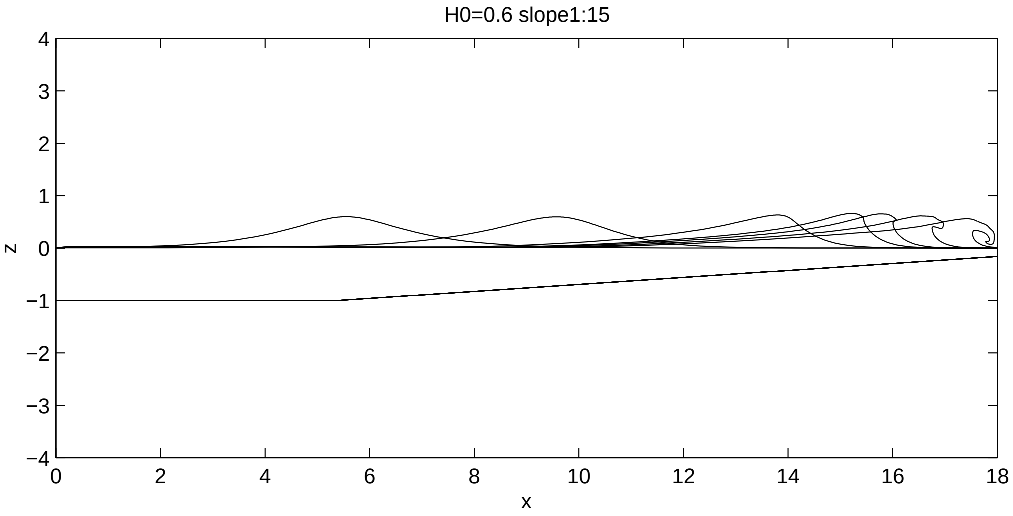

Figure 14: Wave shape at various times. Case (a)

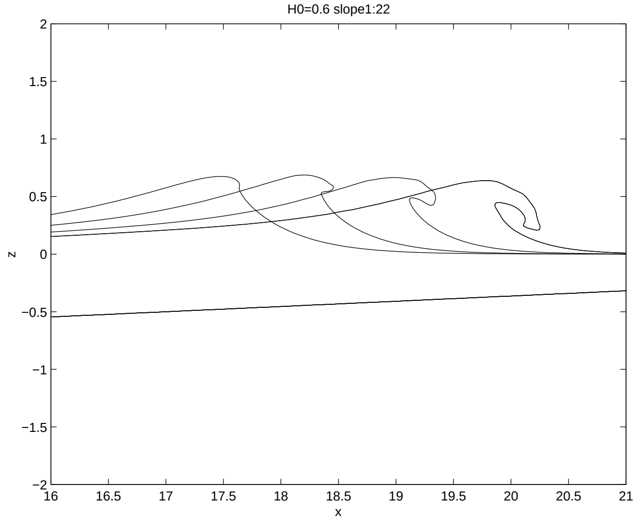

Figure 15: Wave shape at various times. Case (a)

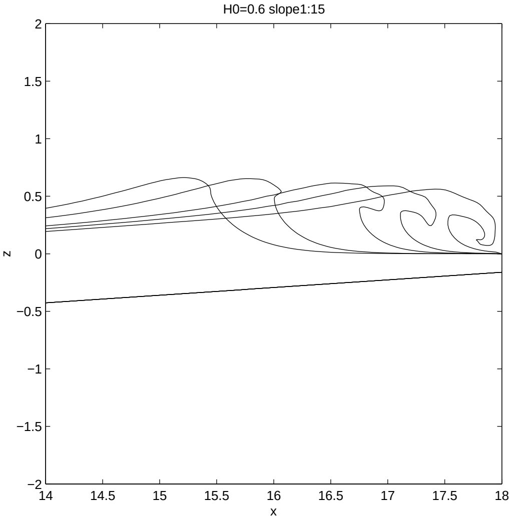

Figure 16: Wave shape at various times. Case (b)

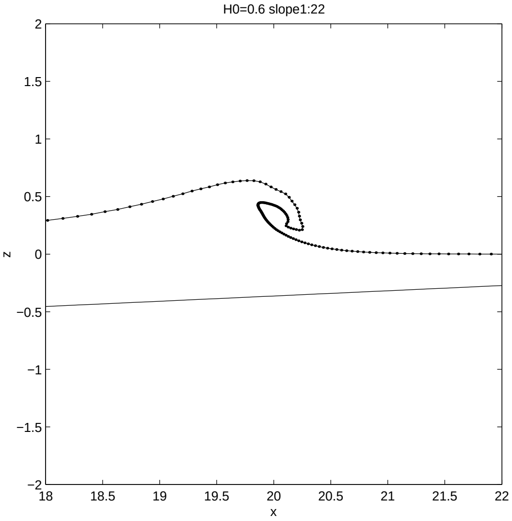

Figure 17: Front BEM nodes at t=3.34. Case (a)

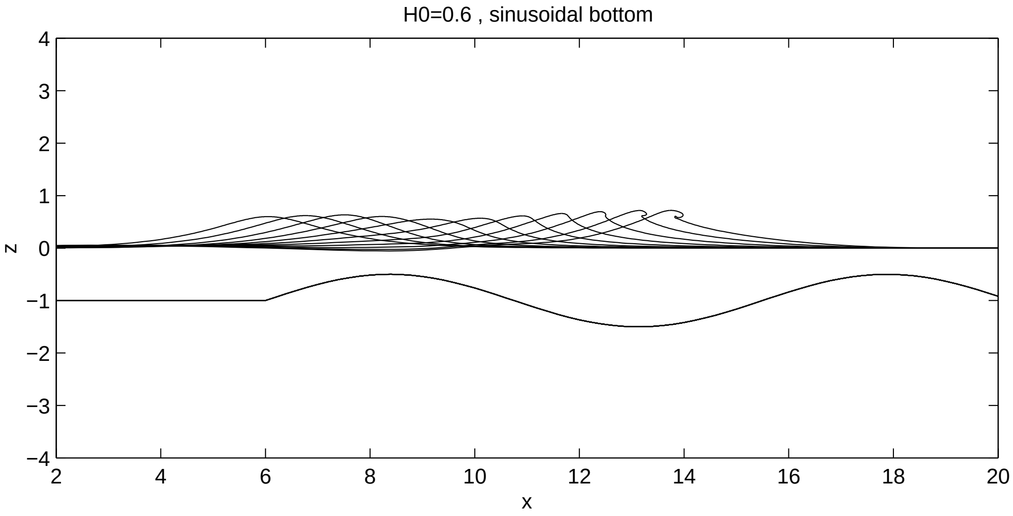

Figure 18: Wave shape at various times. Case (c)

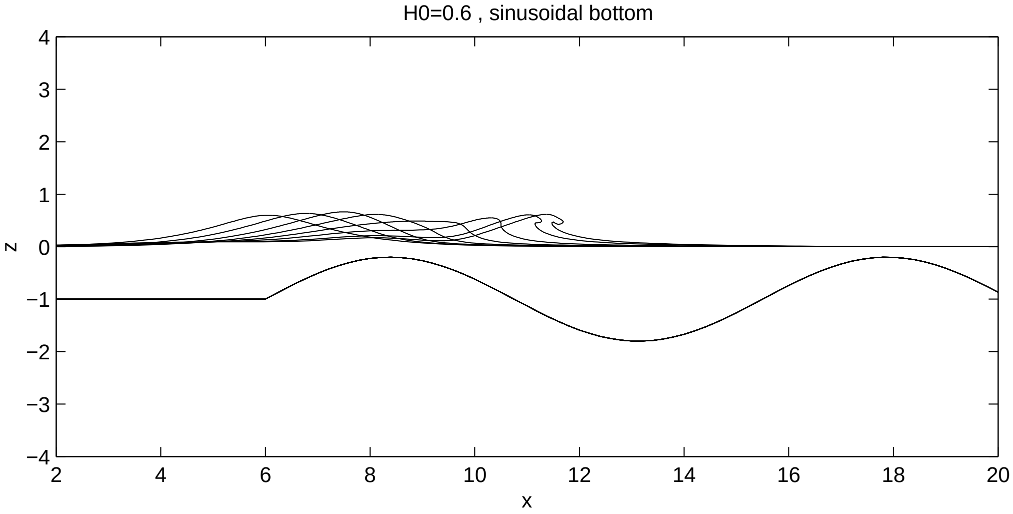

Figure 19: Wave shape at various times. Case (d)

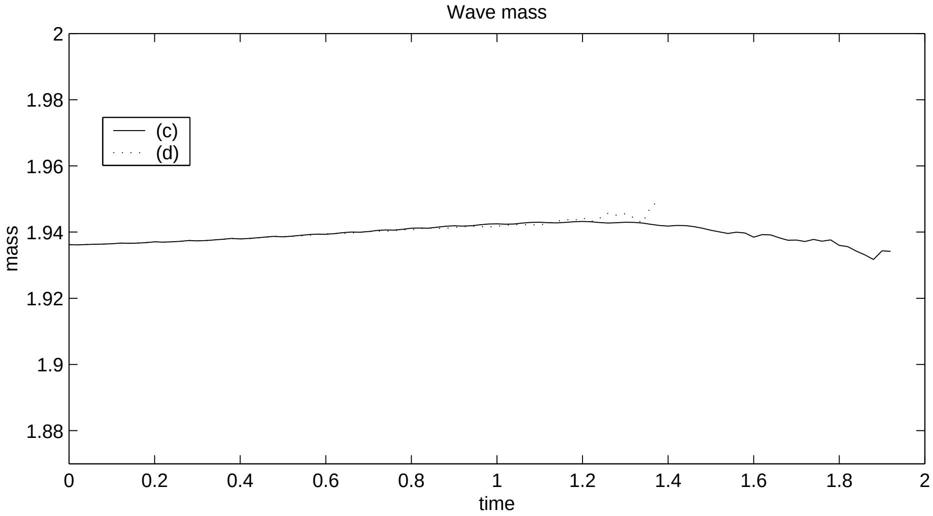

Figure 20: Wave mass vs time. Case (c) and (d)

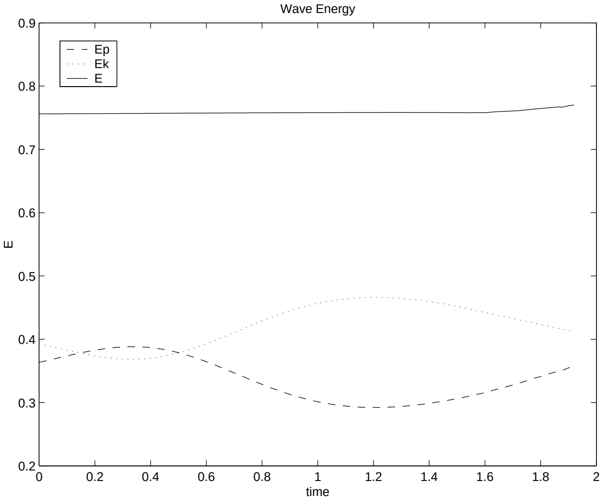

Figure 21: Wave energy. Case (c)

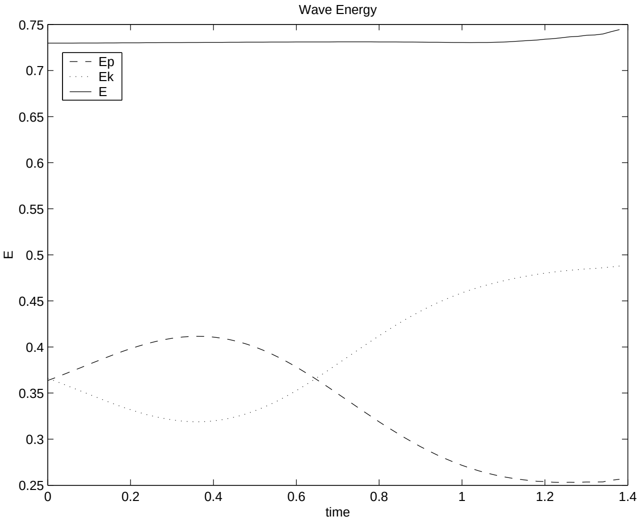

Figure 22: Wave energy. Case (d)

# References

[1] Adalsteinsson, D., and Sethian, J.A., *A Fast Level Set Method for Propagating Interfaces*, J. Comp. Phys., 118, 2, pp. 269–277, 1995.

[2] Adalsteinsson, D., and Sethian, J.A., *The Fast Construction of Extension Velocities in Level Set Methods*, 148, J. Comp. Phys., 1999, pp. 2-22.

[3] Adalsteinsson, D., and Sethian, J.A., *Transport and Diffusion of Material Quantities on Propagating Interfaces via Level Set Methods*, J. Comp. Phys, 185, 1, pp. 271-288, 2002.

[4] Bonnet M. *Boundary Integral Equation Methods for Solids and Fluids*, Wiley and Sons, England, 1995.

[5] *Brebbia C. A.,Telles J. C. F. and Wrobel L. C.*, Boundary Element Techniques, SV, BNY, 1984.

[6] Chen, S., Merriman, B., Osher, S., Smereka P., *A simple level set method for solving Stefan problems.* J. Comput. Phys.135, pp. 8–29, 1997.

[7] Chang, Y.C., Hou, T.Y., Merriman, B., Osher, S.J., it A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, J. Comput. Phys., 124, pp. 449–64, 1996.

[8] Chopp, D.L., *Some Improvements of the Fast Marching Method.* SIAM J.Sci Comput.. 23, pp. 230–244, 2001.

[9] Chorin, A.J., *Numerical solution of the Navier-Stokes equations.* Math. Comput. 22, pp. 745–62, 1968.

[10] Christensen, E.D., Deigaard, R., *Large Eddy Simulation of Breaking Waves. Coastal Engineering 42 (2001) 53-86.*

[11] 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, Computational Mechanics Publishers, chapter 2, pp 33-84, 1998.

[12] Gray L. J., Phan A. -V and Kaplan T., *Boundary Integral Evaluation of Surface Derivatives*, SIAM J. Sci. Comput.,submitted, 2004.

[13] Gray L. J. *Evaluation of hypersingular integrals in the boundary element method*,

[14] Grilli, S.T., Guyenne, P., and Dias, F., *A Fully Non-linear Model for Three Dimensional Overturning Waves Over an Arbitrary Bottom.* International Journal for Numerical Methods in Fluids 35:829-867pp (2001).

[15] Grilli, S.T., Svendsen, I.A., and Subramanya, R., *Breaking Criterion and Characteristics For Solitary Waves On Slopes.* Journal Of Waterway, Port, Coastal, and Ocean Engineering (June 1997).

[16] Grilli, S.T., *Modeling Of Non-linear Wave Motion In Shallow Water.* In Computational Methods for Free and Moving Boundary Problems in Heat and Fluid Flow. Wrobel LC, Brebbia CA (eds). Computational Mechanics Publishers: Southampton, 1995:91-122.

[17] Grilli, S.T., Subramanya, R., *Numerical Modeling of Wave Breaking Induced by Fixed or Moving Boundaries.* Computational Mechanics 1996; 17:374-391.

[18] Lin, P., Chang, K., and Liu, P.L., *Runup and Rundown of Solitary Waves on Sloping Beaches.* Journal Of Waterway, Port, Coastal, and Ocean Engineering (Sep/Oct 1999).

[19] Malladi R., Sethian J.A., Vemuri B.C., *Shape Modeling with Front Propagation: A Level Set Approach* IEEE Trans. on Pattern Analysis and Machine Intelligence, 17, 2, pp. 158–175, 1995.

[20] Martin P. A., Rizzo F. J. and Cruse T. A., *Smoothness-relaxation strategies for singular and hypersingular integral equations*, Int. J. Numer. Meth. Engrg., Vol 42, pp 885-906, 1998.

[21] Martin P. A, and Rizzo F. J., *On boundary integral equations for crack problems*, Proc. R. Soc. Lond., Vol A421, pp 341-355, 1989.

[22] Martin P. A, and Rizzo F. J., *Hypersingular integrals: how smooth must the density be?*, Int. J. Numer. Meth. Engrg., Vol 39, pp 687-704, 1996.

[23] Osher, S., and Sethian, J.A., *Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton–Jacobi Formulations*, Journal of Computational Physics, 79, pp. 12–49, 1988.

[24] Peregrine, D.H., *Breaking Waves on Beaches.* Annual Review in Fluid Mechanics 1983; 15:149-178.

[25] Sethian, J.A., *An Analysis of Flame Propagation*, Ph.D. Dissertation, Dept. of Mathematics, University of California, Berkeley, CA, 1982.

[26] Sethian, J.A., *Curvature and the Evolution of Fronts*, Comm. in Math. Phys., 101, pp. 487–499, 1985.

[27] Sethian, J.A., *Numerical Methods for Propagating Fronts*, in Variational Methods for Free Surface Interfaces, Eds. P. Concus and R. Finn, Springer-Verlag, NY, 1987.

[28] Sethian, J.A., *A Fast Marching Level Set Method for Monotonically Advancing Fronts*, Proc. Nat. Acad. Sci., 93, 4, pp.1591–1595, 1996.

[29] Sethian, J.A., *Level Set Methods and Fast Marching Methods.* Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press (1999).

[30] Sethian, J.A., and Smereka, P., *Level Set Methods for Fluid Interfaces*, Annual Review of Fluid Mechanics, 35, pp.341-372, 2003.

[31] Sussman, M., Smereka, P., Osher, S.J., *A level set approach to computing solutions to incompressible two-phase flow* , J. Comput. Phys., 114, pp. 146–159, 1994.

[32] Tanaka, M., *The stability of solitary waves* , Phys. Fluids, 29 (3), pp. 650–655, 1986.

[33] Yu, J-D., Sakai, S., and Sethian, J.A., *A Coupled Level Set Projection Method Applied to Ink Jet Simulation*, in press, Interfaces and Free Boundaries, 2003.

[34] Zelt. J.A., *The Run-up of Non-breaking and Breaking Solitary Waves.* Coastal Engineering, 15 (1991) 205-246.

[35] Zhu, J., Sethian, J.A., *Projection Methods Coupled to Level Set Interface Techniques,* J. Comp. Phys., 102, pp. 128–138, 1992.