Some free boundary problems in potential flow regime using a based Level Set method *

M. Garzon[†], N. Bobillo-Ares[‡], and J.A. Sethian[§]

September 12, 2008

Abstract

Recent advances in the field of fluid mechanics with moving fronts are linked to the use of Level Set Methods, a versatile mathematical technique to follow free boundaries which undergo topological changes. A challenging class of problems in this context are those related to the solution of a partial differential equation posed on a moving domain, in which the boundary condition for the PDE solver has to be obtained from a partial differential equation defined on the front. This is the case of potential flow models with moving boundaries. Moreover the fluid front will possibly be carrying some material substance which will diffuse in the front and be advected by the front velocity, as for example the use of surfactants to lower surface tension. We present a Level Set based methodology to embed this partial differential equations defined on the front in a complete Eulerian framework, fully avoiding the tracking of fluid particles and its known limitations. To show the advantages of this approach in the field of Fluid Mechanics we present in this work one particular application: the numerical approximation of a potential flow model to simulate the evolution and breaking of a solitary wave propagating over a slopping bottom and compare the level set based algorithm with previous front tracking models.

1 Introduction

In this chapter we present a class of problems in the field of fluid mechanics that can be modeled using the potential flow assumptions, that is, inviscid and incompresible

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

[†] Dept. de Matemáticas, Univ. of Oviedo, Spain

[‡] Dept. de Matemaáticas, Univ. of Oviedo, Spain

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

fluids moving under an irrotational velocity field. The inviscid constraint will limit us to water flow problems, but sometimes this poses also a challenge as viscosity is a slow down flow mechanism. The irrotational character is quite stronger and will limit us to non turbulent flows. Yet, when moving boundaries are present, the boundary condition is a non linear partial differential equation and this fact is what makes the problem interesting, from the mathematical/computational point of view as well as the various industrial aplications. Frecuently in the literature this model is called the fully non linear potential flow model (FNPFM). Several interesting and rather complicated phenomenon are described using the FNPFM, as for example, Helle-Shaw flows, jet evolution and drop formation, sprays and electrosprays, wave propagation and breaking machanisms, etc, see [21], [22], [30], [13].

Level Set Methods (LSM) [31], [33], [34] [37] are widely used in the field of fluid mechanics as in many other fields, such as: medical imaging, semiconductor manufacturing, ink jet printing, seismology, etc. The LSM is a powerfull mathematical tool to move interfaces, once its velocity is known. In fluid mechanics the interface velocity is obtained by solving the partial diferential equations system used to model the fluid/fluids flow. The LSM is based on embeding the moving front as the zero level set of one higher dimensional function. By doing so, the problem can be formulated in a complete Eulerian description and topologycal changes of the free surface are authomatically included. The equation for the motion of the level set function is an initial value hyperbolic partial differential equation, which can be easily approximated using upwind finite differences schemes.

Recently, the LSM has been extended to formulate problems involving the transport and diffusion of material quantities, see [3]. In [3] model equations and algorithms are presented together with the corresponding test examples and convergence studies. Inspired by this advance, it was realized that the nonlineal boundary condition in potential flow problem could also be embeded using level set based methods. As a result, the FNPFM can also be formulated with an Eulerian description with the associated computational advantages: very simple first order numerical schemes give enough accuracy and resolution. Two difficult problems that have been already approximated using this novel algorithm are: wave breaking over sloping beaches [16], [17] and the Rayleigh taylor instability of a water jet [20]. Moreover, related to drop formation and wave breaking, it has been recently reported in the literature [46], [45] that the presence of surfactants on the fluid surface affects the flow patterns. This complicated physical settings can be mathematically described using the models described in this chapter.

This chapter is organized as follows: In section 2 we have made an effort to obtain dynamic equations valid for any spatial coordinate system. To do this, the derivation of the equations should use only objects defined in an intricsic way (i.e., independent of any coordinate system). At the same time, to be tunned with the level set phylosophy, we have avoided as much as possible the " material description" (Lagrangean coordinates). The geometric magnitudes has been defined using the level sets and scalar fields in the space. In section 3 a brief description of the Levels Set Method is given using also the intrinsic approach. Section 4 is devoted to describe two particular potential flow models: the first one related to drop formation in the presence of surfactants, which combines all the models derived in section 2. The code development for this 3D problem is quite chalenging and numerical results are not yet at hand. Instead, the wave breaking problem can be modeled in 2D, being the corresponding algorithm easier to implement. Thus, in section 5, we present the numerical approximation and algorithm for the wave breaking problem. Numerical results and accuracy tests are also presented in section 6. Precise definitions of certain needed geometrical tools, throughout used in this chapter, are shown in Apendix III.

2 Some physical models

Some physical systems in the field of mechanics of continuous media will be presented here. We are particulary interested on certain models that can be reformulated using the Level Set Method (LSM) techniques. The brief derivation of known physical laws is used also as a pretext to introduce some preliminary concepts and notation.

2.1 Kinematic relationships

Reference configuration. The configuration of a continuous medium at certain time t is known when the position of each particle is specified. We name Ω_t the space region occupied by the continuos medium at that time.

The continuos medium kinematics requires the movement description of each particle. To this aim we need:

i. Label the particles.

ii. Specify the movement of each particle.

The first step is done considering the configuration at an arbitrary instant t_0 (reference configuration). Particles are marked by the point $P_0 \in \Omega_{t_0}$ they occupy. Points in Ω_{t_0} are good labels because they are in a 1 to 1 correspondence with the particles ("particles can not penetrate on each other"). In what follows we will abreviate the phrase "particle with label P_0" by "particle P_0".

Once all the particles are labeled, it is now possible to undertake the second step. Let $P_0 \in \Omega_{t_0}$ be a particle. Its position P at instant t is given by the function:

$$P = R(P_0, t), \quad P \in \Omega_t, P_0 \in \Omega_{t_0}. \tag{1}$$

According to the reference configuration definition, we have:

$$R(P_0, t_0) = P_0. \tag{2}$$

The mapping R_t, $R_t(P_0) := R(P_0, t) = P$, must be invertible:

$$P_0 = R_t^{-1}(P) \in \Omega_{t_0}, \;\; P \in \Omega_t. \tag{3}$$

Lagrangean/Eulerian descriptions. Any tensor field $\mathbf{w}$ may be described in two ways, using (1):

$$\mathbf{w} = w(P, t) = w(R(P_0, t), t) = w_0(P_0, t). \tag{4}$$

Function $w(P, t)$ corresponds to the so called Eulerian description and $w_0(P_0, t)$ corresponds to the Lagrangean description. As a consequence any tensorial field $\mathbf{w}$ admits two time partial derivatives. The "spatial" derivative, corresponding to the Eulerian description:

$$\partial_t \mathbf{w} := \left. \frac{\mathrm{d}}{\mathrm{d}\epsilon} w(P, t + \epsilon) \right|_{\epsilon=0}, \tag{5}$$

measures the variation rate with time of $\mathbf{w}$ from a fixed point in the space. The "convective" derivative, corresponding to the Lagrangean description:

$$\mathrm{D}_t \mathbf{w} := \left. \frac{\mathrm{d}}{\mathrm{d}\epsilon} w_0(P_0, t + \epsilon) \right|_{\epsilon=0}, \tag{6}$$

gives the variation rate of $\mathbf{w}$ following the particle P_0.

Velocity field The velocity $\mathbf{u} = u(P_0, t)$ of the particle P_0 is obtained using the convective derivative ("following the particle") of the position $P = R(P_0, t)$:

$$\mathbf{u} = \mathrm{D}_t P, \quad P = R(P_0, t). \tag{7}$$

Obiously, $\mathbf{u}$ admits both descriptions:

$$\mathbf{u} = u(P, t) = u_0(P_0, t), \quad P = R(P_0, t). \tag{8}$$

Given an arbitrary tensor field $\mathbf{w}$, its spatial and convective derivatives are related using the calculus chain rule and the definition (7):

$$\mathrm{D}_t \mathbf{w} = \partial_t \mathbf{w} + \partial_{\mathbf{u}} \mathbf{w}. \tag{9}$$

Here, $\partial_{\mathbf{u}} \mathbf{w}$ designates the directional derivative of $\mathbf{w}$ along $\mathbf{u}$ (see Apendix III). The acceleration of particle P_0 is obtained by the convective derivative of the velocity field. Using (9), we have:

$$\mathrm{D}_t \mathbf{u} = \partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}. \tag{10}$$

Transport of a vector due to a moving medium. A fluid particle is located at point[1] P at time t. After a lapse of time Δt, the same particle is at point $R(P, t+\Delta t)$. Clearly, the function R must verify that $R(P, t+0) = P$. A nearby particle at same time t is located at $P+\epsilon\mathbf{a}$, and at $t+\Delta t$ is at point $R(P+\epsilon\mathbf{a}, t+\Delta t)$. We have again $P+\epsilon\mathbf{a} = R(P+\epsilon\mathbf{a}, t+0)$. The vector $\epsilon\mathbf{a}$ that connects both particles varies as they move. Denote by $\mathrm{D}_t\epsilon\mathbf{a}$ its rate of change with time:

$$\begin{aligned}
\mathrm{D}_t\epsilon\mathbf{a} &= \lim_{\Delta t\to 0}\frac{1}{\Delta t}\left[(R(P+\epsilon\mathbf{a}, t+\Delta t) - R(P, t+\Delta t)) - (R(P+\epsilon\mathbf{a}, t) - R(P, t))\right] \\
&= \lim_{\Delta t\to 0}\frac{R(P+\epsilon\mathbf{a}, t+\Delta t) - R(P+\epsilon\mathbf{a}, t)}{\Delta t} - \lim_{\Delta t\to 0}\frac{R(P, t+\Delta t) - R(P, t)}{\Delta t}.
\end{aligned}$$

The first term of the right hand side of previous equation is, by definition, the particle velocity at $P+\epsilon\mathbf{a}$, $u(P+\epsilon\mathbf{a}, t)$, and the second term the particle velocity at P, $u(P, t)$. Thus we have:

$$\mathrm{D}_t\epsilon\mathbf{a} = u(P+\epsilon\mathbf{a}, t) - u(P, t).$$

Letting $\epsilon \to 0$, we obtain the rate of change with time of an infinitesimal vector dragged by the medium:

$$\mathrm{D}_t\mathbf{a} := \lim_{\epsilon\to 0}\frac{1}{\epsilon}\mathrm{D}_t\epsilon\mathbf{a} = \lim_{\epsilon\to 0}\frac{u(P+\epsilon\mathbf{a}, t) - u(P, t)}{\epsilon} = \frac{\mathrm{d}}{\mathrm{d}\epsilon}u(P+\epsilon\mathbf{a}, t)\bigg|_{\epsilon=0} = \partial_{\mathbf{a}}\mathbf{u}. \quad (11)$$

We denote $\partial_{\mathbf{a}}$ the operator that performs the directional derivative along the vector $\mathbf{a}$ (see Apendix III).

Fluid volume change as it is tansported by the velocity field. Let $\mathbf{a}, \mathbf{b}$ and $\mathbf{c}$ be three small vectors with origin at point P. The volume of the parallelepiped spanned by vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ is given by the trilineal alternate form

$$\delta V = [\mathbf{a}, \mathbf{b}, \mathbf{c}] = \mathbf{a}\cdot\mathbf{b}\times\mathbf{c}.$$

The rate of change of this volume when the particles located in its vertexes move is $\mathrm{D}_t\delta V$, and thus we have

$$\mathrm{D}_t\delta V = \mathrm{D}_t[\mathbf{a}, \mathbf{b}, \mathbf{c}] = [\mathrm{D}_t\mathbf{a}, \mathbf{b}, \mathbf{c}] + [\mathbf{a}, \mathrm{D}_t\mathbf{b}, \mathbf{c}] + [\mathbf{a}, \mathbf{b}, \mathrm{D}_t\mathbf{c}].$$

Using now (11) we get

$$\mathrm{D}_t\delta V = [\partial_{\mathbf{a}}\mathbf{u}, \mathbf{b}, \mathbf{c}] + [\mathbf{a}, \partial_{\mathbf{b}}\mathbf{u}, \mathbf{c}] + [\mathbf{a}, \mathbf{b}, \partial_{\mathbf{c}}\mathbf{u}],$$

[1] For this calculation we use here the configuration at t as the reference configuration.

which is also a trilineal alternate form. As in the tridimensional space all these forms are proportional, we can set

$$\mathrm{D}_t[\mathbf{a},\mathbf{b},\mathbf{c}] = (\operatorname{div}\mathbf{u})[\mathbf{a},\mathbf{b},\mathbf{c}], \tag{12}$$

which gives us an intrinsic definition of the divergence of the field $\mathbf{u}$. If the continuous medium is incompressible, the volume δV does not change, $\mathrm{D}_t\delta V = 0$, and we arrive at the incompressibility condition

$$\operatorname{div}\mathbf{u} = 0. \tag{13}$$

2.2 Dynamic relationships

Conservation of mass. Denote by $\rho = \varrho(P,t)$ the volumetric mass density of the continuous medium at point P and at time t. The rate of change of the mass in a small volume δV dragged by the velocity field is, using definition (12),

$$\mathrm{D}_t(\rho\delta V) = (\mathrm{D}_t\rho)\delta V + \rho\mathrm{D}_t\delta V = (\mathrm{D}_t\rho + \rho\ \operatorname{div}\mathbf{u})\delta V.$$

The mass conservation law is thus

$$\mathrm{D}_t\rho + \rho\operatorname{div}\mathbf{u} = 0. \tag{14}$$

Applying general formula (9) to ρ, we have $\mathrm{D}_t\rho = \partial_t\rho + \partial_{\mathbf{u}}\rho$. In the case of an homogeneous and incompressible medium with uniform initial density ρ_0, using equations (14) and (13), we have $\mathrm{D}_t\rho = 0$ which gives $\varrho(P,t) = \rho_0$.

Conservation of the momentum associated with a small piece of continuous medium. From Newton's second law applied to a fluid volume V we get the relation

$$\mathrm{D}_t\int_V \mathbf{u}\,\rho\mathrm{d}V = \int_V \mathbf{g}\,\rho\mathrm{d}V + \int_{\partial V}\tau(\mathrm{d}\mathbf{s}). \tag{15}$$

The term in left hand side of this equation is the rate of change with time of the momentum associated with volume V when dragged by the continuous medium. The first term in the right hand side corresponds to the volumetric forces inside V, generated by a vector field per unit mass $\mathbf{g}$, usually the gravitational field. The second term represents the "contact" forces applied by the rest of the medium over the part in V. The Cauchy's tensor τ is a linear operator field that is obtained from specific relationships which depend on the material, the so called constitutive relations. We are interested in inviscid fluids which verify the Pascal's law:

$$\tau(\mathrm{d}\mathbf{s}) = -p\mathrm{d}\mathbf{s},$$

where p is the pressure scalar field. Green's formula,

$$\int_{\partial V} -p\,\mathrm{d}\mathbf{s} = \int_V -\nabla p\,\mathrm{d}V,$$

shows that contact forces may be computed as a kind of volume forces with density $-\nabla p$. For a small volume δV dragged by the fluid, equation (15) can be written:

$$\mathrm{D}_t(\mathbf{u}\,\rho\delta V) = (\mathbf{g}\,\rho - \nabla p)\delta V. \tag{16}$$

Due to the mass conservation law, $\mathrm{D}_t(\rho\delta V) = 0$, equation (16) leads to the Euler equation:

$$\mathrm{D}_t\mathbf{u} = \partial_t\mathbf{u} + \partial_{\mathbf{u}}\mathbf{u} = \mathbf{g} - \frac{1}{\rho}\nabla p. \tag{17}$$

If $\mathbf{g}$ is a uniform field it comes from the gradient of a potential function:

$$\mathbf{g} = -\nabla U(P),\ \ U(P) = -\mathbf{g}\cdot(P-O),$$

where $P-O$ is the position vector of the point P.

2.3 Potential flow

Assuming an irrotational flow regime, curl $\mathbf{u} = 0$, there exists an scalar field ϕ such that

$$\mathbf{u} = \nabla\phi. \tag{18}$$

Outside of the fluid domain, and separated by a free boundary, there is a gas at pressure p_a that is assumed to be constant. This means that, within the gas, the time needed to restore the equilibrium is very small compared with the time evolution of the fluid. Therefore, at the fluid free boundary, the boundary condition is just:

$$p = p_a. \tag{19}$$

Using the vectorial relationship $\nabla\mathbf{u}^2/2 = \partial_{\mathbf{u}}\mathbf{u} + \mathbf{u}\times(\text{curl}\,\mathbf{u})$, and relations (18) and (13) we have

$$\nabla\left(\partial_t\phi + \frac{1}{2}\mathbf{u}^2 + \frac{p}{\rho} + U\right) = 0.$$

Performing the first integration,

$$\partial_t\phi + \frac{1}{2}\mathbf{u}^2 + \frac{p}{\rho} + U = C(t),$$

where $C(t)$ is an arbitrary function of time, which can be choosen in such a way that the previous relation can be written:

$$\partial_t\phi + \frac{1}{2}\mathbf{u}^2 + \frac{p-p_a}{\rho} + U = 0.$$

Now using the obvious relation $\partial_t\phi + \mathbf{u}^2 = \partial_t\phi + \partial_\mathbf{u}\phi = \mathrm{D}_t\phi$, we finally obtain

$$\mathrm{D}_t\phi - \frac{1}{2}\mathbf{u}^2 + \frac{p-p_a}{\rho} + U = 0. \tag{20}$$

2.4 Advection

On the surface of a continuous medium with a known movement a certain substance is distributed, which will be named as "charge". This is adhered to the fluid particles and it is transported by them. In this way a set of particles will always carry the same amount of "charge". This phenomenon is called advection.

The continuous medium surface is implicitly described as the zero level set of a certain scalar function $\Psi = \psi(P,t)$:

$$\Gamma_t = \{Q|\psi(Q,t) = 0\}. \tag{21}$$

Vectors $\mathbf{a}$ tangent to the surface are characterized by the condition

$$\partial_\mathbf{a}\Psi = \mathbf{a}\cdot\nabla\Psi = 0;$$

so, the tangent vectorial plane at each point of the surface is given by the normal unit vector[2]

$$\mathbf{n} = \frac{\nabla\Psi}{|\nabla\Psi|}.$$

The function ψ by itself does not specify the particle movement on the surface, just its shape. We need to add the information about how these particles move, e.g., specifying the velocity field on the surface

$$Q\in\Gamma_t,\quad \mathbf{u} = u(Q,t).$$

A small vector $\mathbf{a}$ connecting two nearby particles on the surface and dragged by them as they move, has a rate of change given by (11),

$$\mathrm{D}_t\mathbf{a} = \partial_\mathbf{a}\mathbf{u}. \tag{22}$$

Note that $\mathbf{a}$ is a tangent vector, $\mathbf{n}\cdot\mathbf{a} = 0$.

[2] Ψ must increase from the interior to the exterior of the surface to get $\mathbf{n}$ outwards.

Surface areas. Using the normal vector to the surface, $\mathbf{n}$, a 2–form to calculate surface areas can be constructed:[3]

$$\omega(\mathbf{a},\mathbf{b}) := [\mathbf{n},\mathbf{a},\mathbf{b}] = \mathbf{n}\cdot\mathbf{a}\times\mathbf{b},$$

where $\omega(\mathbf{a},\mathbf{b})$ is the area spanned by tangent vectors $\mathbf{a},\mathbf{b}$, and $[\mathbf{n},\mathbf{a},\mathbf{b}]$ the volume form in the 3D space.

As the tangent vectors $\mathbf{a},\mathbf{b}$ are transported by the surface movement, the parallelogram area associated to them changes. The rate of change with time is easily obtained:

$$\mathrm{D}_t\omega(\mathbf{a},\mathbf{b}) = (\mathrm{D}_t\mathbf{n})\cdot\mathbf{a}\times\mathbf{b} + \mathbf{n}\cdot(\mathrm{D}_t\mathbf{a})\times\mathbf{b} + \mathbf{n}\cdot\mathbf{a}\times\mathrm{D}_t\mathbf{b}.$$

First term of the right hand side of previous equation is zero since $\mathbf{a}\times\mathbf{b}$ is a normal vector and $\mathrm{D}_t\mathbf{n}$ is tangent: indeed, as $\mathbf{n}^2 = 1$, we have $\mathrm{D}_t\mathbf{n}^2 = 2\mathbf{n}\cdot\mathrm{D}_t\mathbf{n} = 0$.

Using (22) we have

$$\mathrm{D}_t\omega(\mathbf{a},\mathbf{b}) = \partial_{\mathbf{a}}\mathbf{u}\cdot\mathbf{b}\times\mathbf{n} + \partial_{\mathbf{b}}\mathbf{u}\cdot\mathbf{n}\times\mathbf{a}.$$

This expression is bilinear and alternate with respect the tangent vectors. It must be, at each point on the surface, proportional to the 2–form ω. We denote by $\mathrm{Div}\,\mathbf{u}$, "surface divergence", the proportionality coefficient:

$$\partial_{\mathbf{a}}\mathbf{u}\cdot\mathbf{b}\times\mathbf{n} + \partial_{\mathbf{b}}\mathbf{u}\cdot\mathbf{n}\times\mathbf{a} := (\mathrm{Div}\,\mathbf{u})\;\omega(\mathbf{a},\mathbf{b}) \tag{23}$$

This definition of $\mathrm{Div}\,\mathbf{u}$ does not depend upon the choice of tangent vectors $\mathbf{a}$ and $\mathbf{b}$. In Apendix I, the expression for the surface divergence of an arbitrary vector field $\mathbf{w}$ using rectangular coordinates is shown.

Advection law. Now, let be

$$\sigma = \sigma(Q,t), \quad Q\in\Gamma_t$$

the "charge" surface density. The "charge" δq carried by a small parallelogram, spanned by two small tangent vectors $(\mathbf{a},\mathbf{b})$, of area $\omega(\mathbf{a},\mathbf{b})$ is

$$\delta q = \sigma\;\omega(\mathbf{a},\mathbf{b}).$$

[3] The surface area definition is not made using the Gram determinant of two tangent vectors, because this procedure involves a particular parametrization of the surface. Instead, a 2-form is defined from the volume form in space ("the parallelogram area is the volume of a rectangular prism of unit height").

As the “charge” is conserved, the advection law is

$$\mathrm{D}_t\ \delta q = 0.$$

Now, by definition (23), we have

$$\mathrm{D}_t(\sigma\omega) = (\mathrm{D}_t\sigma)\omega + \sigma\mathrm{D}_t\omega = (\mathrm{D}_t\sigma + \sigma\,\mathrm{Div}\,\mathbf{u})\omega. \tag{24}$$

Hence we arrive to the intrinsic equation for the advection phenomena:

$$\mathrm{D}_t\sigma + \sigma\,\mathrm{Div}\,\mathbf{u} = 0.$$

2.5 Advection-Difusion

Next, we are going to assume that the “charge” diffuses along particles on the surface according to the Fick’s law:[4]

$$\mathbf{j} = -\alpha\nabla\sigma,$$

where α is the diffusion coeficient, $\mathbf{j}$ is the “charge” flux and $\nabla\sigma$ is the “charge” surface density gradient. As σ is only defined on the surface Γ_t, $\nabla\sigma$ is only defined for tangent vectors:

$$\nabla\sigma\cdot\mathbf{a} := \partial_{\mathbf{a}}\sigma, \quad \mathbf{a}\ \text{tangent vector.}$$

On the surface Γ_t let us consider a surface region S, bounded by a curve ∂S. Let be $\boldsymbol{\nu}$ the unit vector field tangent to Γ_t and ortogonal to the curve ∂S at each point. The “charge” that leaves the surface per unit time is the outward flux through the boundary ∂S:

$$\int_{\delta S}\mathbf{j}\cdot\boldsymbol{\nu}\,\mathrm{d}l = -\int_{\partial S}\mathbf{j}\cdot\mathbf{n}\times\mathrm{d}\mathbf{l} = \int_{\partial S}\mathbf{n}\times\mathbf{j}\cdot\mathrm{d}\mathbf{l}.$$

Applying now Stokes’ theorem, we have

$$\int_{\partial S}\mathbf{n}\times\mathbf{j}\cdot\mathrm{d}\mathbf{l} = \int_S A\ \omega(\mathrm{d}_1P, \mathrm{d}_2P). \tag{25}$$

The 2-form of the surface integral is obtained using the intrinsic formula

$$A\ \omega(\mathbf{a},\mathbf{b}) = \partial_{\mathbf{a}}(\mathbf{n}\times\mathbf{j}\cdot\mathbf{b}) - \partial_{\mathbf{b}}(\mathbf{n}\times\mathbf{j}\cdot\mathbf{a}). \tag{26}$$

We interpret $A\ \omega(\mathbf{a},\mathbf{b})$ as the “charge” per unit time that abandons by diffusion the small parallelogram spanned by the tangent vectors $(\mathbf{a},\mathbf{b})$.

[4] Fick’s diffusion law applies when the “charge” particles move randomly without any preferential direction (Brownian movement).

Now it is straightforward to set the condition for the advection-diffusion mechanism

$$\left\{\begin{array}{c}\text{“charge” rate of change}\\ \text{within the tangent}\\ \text{parallelogram } (\mathbf{a},\mathbf{b})\end{array}\right\} = -\left\{\begin{array}{c}\text{“charge” that leaves}\\ \text{the parallelogram}\\ \text{by diffusion}\end{array}\right\},$$

that is

$$\mathrm{D}_t(\sigma\omega(\mathbf{a},\mathbf{b})) = -A\,\omega(\mathbf{a},\mathbf{b}).$$

In Appendix II the following expression for A is obtained:

$$A = \mathrm{Div}\,\mathbf{j} - (\mathrm{Div}\,\mathbf{n})\,\mathbf{j}\cdot\mathbf{n}.$$

Hence, using (24) we arrive at the general equation for the advection-diffusion model:

$$\left\{\begin{array}{ccc}\mathrm{D}_t\sigma + \sigma\ \mathrm{Div}\,\mathbf{u} & = & -\,\mathrm{Div}\,\mathbf{j} + (\mathrm{Div}\,\mathbf{n})\,\mathbf{j}\cdot\mathbf{n}\\ \mathbf{j} & = & -\alpha\,\nabla\sigma\end{array}\right.$$

or

$$\mathrm{D}_t\sigma + \sigma\ \mathrm{Div}\,\mathbf{u} = \alpha\ \mathrm{Div}\,\nabla\sigma - \alpha(\mathrm{Div}\,\mathbf{n})\ \nabla\sigma\cdot\mathbf{n} \tag{27}$$

The Cartesian expressions[5] for $\mathrm{Div}\,\mathbf{u}$, $\mathrm{Div}\,\nabla\sigma$ and $\mathrm{Div}\,\mathbf{n}$ are:

$$\mathrm{Div}\,\mathbf{n} = (\delta_{ij} - n_i n_j)\partial_j n_i = \frac{1}{|\nabla\Psi|^3}\left[(\nabla\Psi)^2\partial_i\partial_i\Psi - \partial_j\Psi\partial_i\Psi\partial_j\partial_i\Psi\right], \tag{28}$$

$$\mathrm{Div}\,\nabla\sigma = \frac{1}{|\nabla\Psi|^2}\left[(\nabla\Psi)^2\partial_i\partial_i\sigma - \partial_i\Psi\partial_j\Psi\partial_i\partial_j\sigma\right], \tag{29}$$

$$\mathrm{Div}\,\mathbf{u} = \frac{1}{|\nabla\Psi|^2}\left[(\nabla\Psi)^2\partial_i u_i - \partial_i\Psi\partial_j\Psi\partial_i u_j\right]. \tag{30}$$

Expanding the implicit sumatories we obtain the following expressions for the 3D

[5] In the following expressions we use only subscripts because orthonormal bases coincides with their corresponding reciprocal ones. Then, the position of the indices becomes irrelevant.

space $(i,j=1,2,3)$:

$$\begin{aligned}
\mathrm{Div}\,\mathbf{n} &= \frac{1}{|\nabla\Psi|^3}\big[(\partial_1\Psi)^2(\partial_2^2\Psi+\partial_3^2\Psi)+(\partial_2\Psi)^2(\partial_1^2\Psi+\partial_3^2\Psi)+ \\
&\quad +(\partial_3\Psi)^2(\partial_1^2\Psi+\partial_2^2\Psi)-2\partial_1\Psi\partial_2\Psi\partial_1\partial_2\Psi- \\
&\quad -2\partial_1\Psi\partial_3\Psi\partial_1\partial_3\Psi-2\partial_2\Psi\partial_3\Psi\partial_2\partial_3\Psi\big]\,, \qquad (31)\\
\mathrm{Div}\,\nabla\sigma &= \frac{1}{|\nabla\Psi|^2}\big[(\partial_1\Psi)^2(\partial_2^2\sigma+\partial_3^2\sigma)+(\partial_2\Psi)^2(\partial_1^2\sigma+\partial_3^2\sigma)+ \\
&\quad +(\partial_3\Psi)^2(\partial_1^2\sigma+\partial_2^2\sigma)-2\partial_1\Psi\partial_2\Psi\partial_1\partial_2\sigma- \\
&\quad -2\partial_1\Psi\partial_3\Psi\partial_1\partial_3\sigma-2\partial_2\Psi\partial_3\Psi\partial_2\partial_3\sigma\big]\,, \qquad (32)\\
\mathrm{Div}\,\mathbf{u} &= \frac{1}{|\nabla\Psi|^2}\big[(\partial_1\Psi)^2(\partial_2u_2+\partial_3u_3)+(\partial_2\Psi)^2(\partial_1u_1+\partial_3u_3)+ \\
&\quad +(\partial_3\Psi)^2(\partial_1u_1+\partial_2u_2)-\partial_1\Psi\partial_2\Psi(\partial_1u_2+\partial_2u_1)- \\
&\quad -\partial_1\Psi\partial_3\Psi(\partial_1u_3+\partial_3u_1)-\partial_2\Psi\partial_3\Psi(\partial_2u_3+\partial_3u_2)\big]\,. \qquad (33)
\end{aligned}$$

To obtain the formulas for the plane we assume axial symmetry in the direction 3:

$$\partial_3\Psi=0,\quad \partial_3^2\Psi=0,\quad \partial_3\sigma=0,\quad u_3=0,\quad \partial_3u_i=0,$$

$$|\nabla\Psi|^2=(\partial_1\Psi)^2+(\partial_2\Psi)^2.$$

Inserting these values in (31), (32) and (33) we get:

$$\begin{aligned}
\mathrm{Div}\,\mathbf{n} &= \frac{1}{|\nabla\Psi|^3}\big[(\partial_1\Psi)^2(\partial_2^2\Psi)+(\partial_2\Psi)^2(\partial_1^2\Psi)-2\partial_1\Psi\partial_2\Psi\partial_1\partial_2\Psi\big]\,, \qquad (34)\\
\mathrm{Div}\,\nabla\sigma &= \frac{1}{|\nabla\Psi|^2}\big[(\partial_1\Psi)^2(\partial_2^2\sigma)+(\partial_2\Psi)^2(\partial_1^2\sigma)-2\partial_1\Psi\partial_2\Psi\partial_1\partial_2\sigma\big]\,, \qquad (35)\\
\mathrm{Div}\,\mathbf{u} &= \frac{1}{|\nabla\Psi|^2}\big[(\partial_1\Psi)^2(\partial_2u_2)+(\partial_2\Psi)^2(\partial_1u_1)- \\
&\quad -2\partial_1\Psi\partial_2\Psi(\partial_1u_2+\partial_2u_1)\big]\,. \qquad (36)
\end{aligned}$$

3 The Level Set Method

The Level Set method is a mathematical tool developed by Osher and Sethian [31] to follow interfaces which move with a given velocity field. The key idea is to view the moving front as the zero level set of one higher dimensional function called the level set funcion. The main advantage of this approach comes when the moving boundary changes topology, and thus a simple connected domain splits into separated disconnected domains.

Let be Γ_t the set of points lying in the surface boundary at time t. This surface is defined through the zero level set of the scalar field $\Psi = \psi(P,t)$:

$$\Gamma_t = \{Q|\psi(Q,t) = 0\}. \tag{37}$$

To identify the fluid particles, the configuration at t_0 (reference configuration) is used:

$$\Gamma_{t_0} = \{Q_0|\psi(Q_0,t_0) = 0\}. \tag{38}$$

The particle movement is specified through the function

$$Q = R(Q_0,t), \tag{39}$$

which gives the position $Q \in \Gamma_t$ of the fluid particle $Q_0 \in \Gamma_{t_0}$. The particle Q_0 velocity is calculated using the convective derivative D_t ("following the particle"):

$$\mathbf{u} = D_t Q = \frac{\mathrm{d}}{\mathrm{d}\epsilon} R(Q_0, t+\epsilon)\bigg|_{\epsilon=0}. \tag{40}$$

According to definition (37), we have $\psi(R(Q_0,t),t) = 0$. Deriving with respect to time and applying the chain rule, we obtain

$$\partial_t \Psi + \mathbf{u} \cdot \nabla \Psi = 0. \tag{41}$$

which has to be completed with the value of the level set function at time $t = 0$, usually set to be the signed distance function to the initial front,

$$\Psi(P,0) = s(P)d(P),$$

being $d(P)$ the distance from the point P to the surface at the initial configuration Γ_0, $s(P) = -1$ if $P \in \Omega_0$ and $s(P) = +1$ if $P \notin \Omega_0$.

Now, if we take a fixed 3D domain Ω_D that contains the free surface for all times, we can define the initial value problem for the level set function Ψ posed on Ω_D:

$$\partial_t \Psi + \mathbf{u} \cdot \nabla \Psi = 0 \text{ in } \Omega_D \tag{42}$$

$$\Psi(P,0) = s(P)d(P) \text{ in } \Omega_D \tag{43}$$

A graphical interpretation of the level set function evolution is depicted in figure 1 Equation (42) moves all the level set of Ψ, not just the zero level set, and in many physical applications the front velocity is just defined for points lying on the free boundary. Therefore for this equation to be valid on the whole domain we have to extend the velocity $\mathbf{u}$ off the front. There exist several extension procedures which will be briefly comented below.

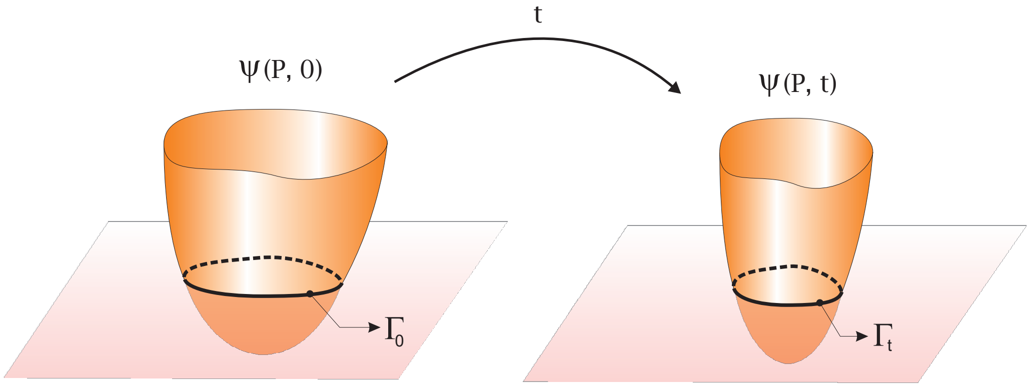

Figure 1: Evolution of the level set function

3.1 Extension of functions defined on the front

Let us consider a classical result from functional analysis: suppose a domain Ω bounded by a closed surface $\partial\Omega$. If for $k \geq 1$ the surface $\partial\Omega \in \mathcal{C}^k$, then for all functions $F(\mathbf{x}) \in \mathcal{C}^k(\partial\Omega)$ there exists a function $F_{\text{ext}}(\mathbf{x}) \in \mathcal{C}^k(\bar{\Omega})$ such that $F_{\text{ext}}|_{\partial Q} = F(\mathbf{x})$.

Practically there are several ways to extend any magnitude F defined on the front onto Ω_D. As shown in [10] for the numerical stability of the level set equation it is convenient to preserve Ψ as a signed distance function, which is characterized by the property $|\nabla\Psi| = 1$. One way is to perform reinitializations of the level set function at choosen times. If it is frecuently done, it will smooth the solution aborting the development of small structures and affecting mass conservation. For the potential flow problems presented in this chapter we follow the strategy introduced in [2]. The idea is to extrapolate F given at the front along its gradient. Mathematically the extended variable F_{ext} is the solution of

$$\nabla F_{\text{ext}} \cdot \nabla\Psi = 0. \tag{44}$$

It is straightforward to show that this choice maintains the signed distance function for the level sets of Ψ for all times. For the numerical approximation we proceed as follows: given a level set function Ψ at time n, namely Ψ^n, one first obtains a distance function $\hat{\Psi}^n$ around the zero level set. Simultaneous with this construction, the extended quantity F_{ext} is obtained satisfying Eq. (44). For a complete explanation of this extension method see [2].

4 Examples of potential flow models with moving boundaries

In this section the governing equations of two interesting physical problems will be formulated using a level set framework. First, drop formation is a complex 3D phenomena driven mainly by capillary forces, which can be modeled using the potential flow assumptions. It is well known that the presence of surface surfactants lowers the surface tension affecting drops shape. Secondly, propagation and wave breaking over sloping beaches can also be modeled with the potential flow equations, which are valid until the jet of the wave impinges against that flat water surface. In this case we can formulate the equations in 2D taking a vertical section of the beach, which facilitates the algorithm and code development. The wave numerical simulatons will be presented in section 6.

4.1 Governing equations for surface tension driven flows with material advection-diffusion

Let Ω_t be the $3D$ closed fluid domain surrounded by air and Γ_t the free surface boundary at time t. Suppose that initially a certain amount of surfactants, which are assumed to be insoluble in water, are uniformily distributed on the surface (see Figure 2).

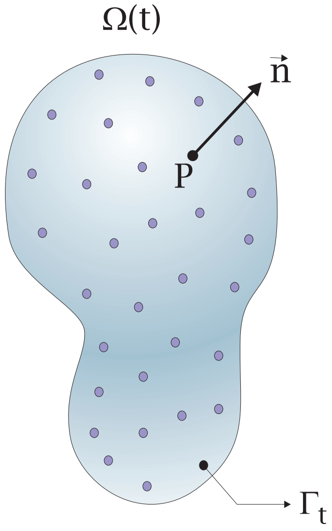

Figure 2: A fluid volume with surface surfactants

For an incompressible and inviscid fluid, the governing equations are the Euler equations (16). On the free boundary the following partial differential equations apply:

- The advection-difusion equation for the surfactant is (27):

$$D_t\sigma + \sigma \text{Div }\mathbf{u} = \alpha(\text{Div}\nabla\sigma - \kappa \; \nabla\sigma \cdot \mathbf{n}) \quad \text{on } \Gamma_t,$$

  where σ is the surface density of the surfactant, $\mathbf{u}$ is the free boundary velocity, α is the surface diffusion coeficient, $\kappa = \text{Div } \mathbf{n} = \frac{1}{R_1} + \frac{1}{R_2}$ and R_1, R_2 the principal radii of curvature of Γ_t at each point.

- Continuity of the stress tensor between water and air leads to the balance of the surface tension forces, $p = p_a + \gamma(\frac{1}{R_1} + \frac{1}{R_2})$, where γ is the surface tension coeficient that may depend on the surfactant concentration σ. Thus Eq. (20) becomes

$$\partial_t\phi + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) + \frac{\gamma}{\rho}\kappa + U = 0 \;\text{ on }\; \Gamma_t.$$

- Finally, if $Q = R(Q_0, t)$ is the position of a fluid particle Q_0 on the free surface, the definition (40) states

$$D_tQ = u(Q, t), \quad Q \in \Gamma_t. \tag{45}$$

The complete model equations in 3D are therefore,

$$\begin{aligned}
\mathbf{u} &= \nabla\phi \;\text{ in }\; \Omega_t && (46)\\
\Delta\phi &= 0 \;\text{ in }\; \Omega_t && (47)\\
D_tQ &= \mathbf{u} \;\text{ on }\; \Gamma_t && (48)\\
D_t\phi &= -U + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - \frac{\gamma}{\rho}\kappa \;\text{ on }\; \Gamma_t && (49)\\
D_t\sigma &= -\sigma \text{Div }\mathbf{u} + \alpha(\text{Div}\nabla\sigma - \kappa \; \nabla\sigma \cdot \mathbf{n}) \quad \text{on } \Gamma_t. && (50)
\end{aligned}$$

This is the Lagrangean-Eulerian formulation of he model equations. Classical methods to approximate this set of equations are the so called "front tracking methods", where a fixed number of markers are chosen initially and the trajectories of this markers are followed as time evolves. This method suffers serious difficulties when the free boundary changes topology. Thus it is computational very convenient to develop an approach based upon a Level set formulation.

Level set framework

Equation (48) can be directly formulated as the level set Eq. (41). For the velocity field $u(Q,t)$, the trajectory of a fluid particle at initial position Q_0 is given by the solution of

$$\begin{aligned} D_t Q &= u(R(Q_0,t),t), \\ R(Q_0,0) &= Q_0 \ . \end{aligned} \tag{51}$$

Next, Let Ω_D be a fixed 3D domain that contains the free surface for all times and let $G(P,t)$ and $S(P,t)$ be two functions defined on Ω_D such that for every $Q \in \Gamma_t$

$$G(Q,t) = \phi(Q,t) \ , \tag{52}$$

$$S(Q,t) = \sigma(Q,t) \ . \tag{53}$$

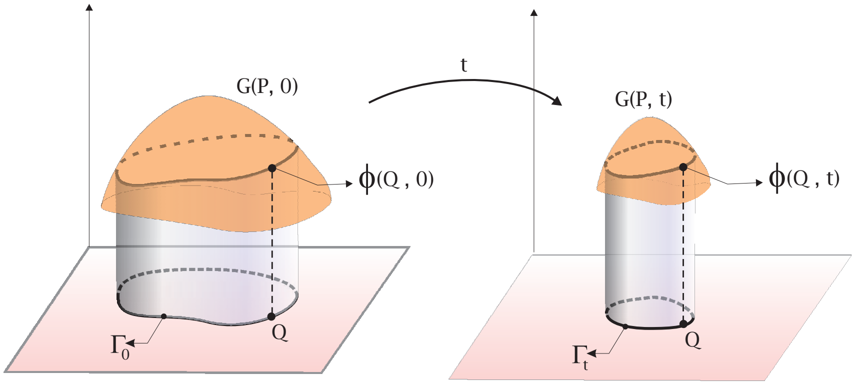

Figure 3: Extension of the velocity potential off the front

It is important to remark here that $G(P,t)$ and $S(P,t)$ are auxiliary functions defined in Ω_D that can be chosen arbitrarily, the only restriction is that they equal $\phi(Q,t)$ and $\sigma(Q,t)$ on Γ_t respectively. Figure 3 gives an interpretation of this property for a moving curve in 2D. Applying the chain rule in both identities (52) and (53) we obtain

$$\partial_t G + \mathbf{u} \cdot \nabla G = -U + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - \frac{\gamma}{\rho}\kappa, \tag{54}$$

$$\partial_t S + \mathbf{u} \cdot \nabla S = -\sigma \mathrm{Div}\ \mathbf{u} + \alpha(\mathrm{Div}\nabla\sigma - \kappa\ \nabla\sigma \cdot \mathbf{n}). \tag{55}$$

which holds on Γ_t. Note that $\mathbf{u}$ and the right hand side of Eq. (54) and Eq. (55) are only defined on Γ_t, and thus, in order to solve these equations over the fixed domain Ω_D, these variables must be extended off the front. This strategy has been discussed in Section 3. Naming

$$f = -U + \frac{1}{2}(\nabla\phi \cdot \nabla\phi) - \frac{\gamma}{\rho}\kappa,$$

$$h = -\sigma \text{Div}\, \mathbf{u} + \alpha(\text{Div}\nabla\sigma - \kappa\, \nabla\sigma \cdot \mathbf{n}),$$

the system of equations, written in a complete Eulerian framework, is

$$\begin{aligned} \mathbf{u} &= \nabla\phi \text{ in } \Omega_t && (56)\\ \Delta\phi &= 0 \text{ in } \Omega_t && (57)\\ \Psi_t + \mathbf{u}_{\text{ext}} \cdot \nabla\Psi &= 0 \text{ in } \Omega_D. && (58)\\ G_t + \mathbf{u}_{\text{ext}} \cdot \nabla G &= f_{\text{ext}} \text{ in } \Omega_D && (59)\\ S_t + \mathbf{u}_{\text{ext}} \cdot \nabla S &= h_{\text{ext}} \text{ in } \Omega_D && (60) \end{aligned}$$

Here the subscript “ext” denotes the extension of f, h and $\mathbf{u}$ onto Ω_D.

4.2 Governing equations for the wave breaking problem

We now derive our coupled level set/extension potential equations for breaking waves in two dimensions for which a numerical approximation will be also presented. Let Ω_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 Γ_t the free boundary at time t (see Figure 4.2).

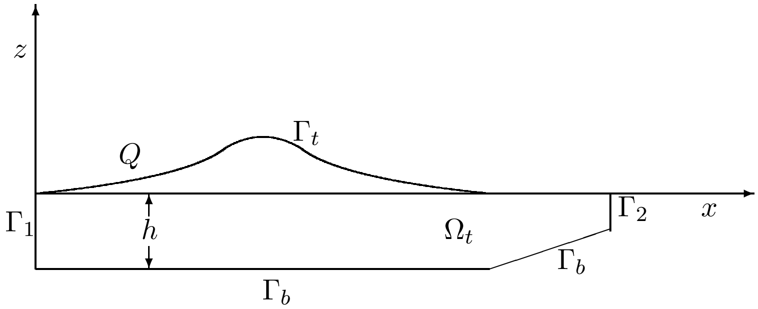

Figure 4: The domain

We assume also an inviscid and incompresibble fluid, and capillary forces are disregarded on the free boundary curve. The model equations in the Lagragian-Eulerian formulation are therefore:

$$\begin{aligned}
\mathbf{u} &= \nabla\phi \text{ in } \Omega_t & (61)\\
\Delta\phi &= 0 \text{ in } \Omega_t & (62)\\
D_t Q &= \mathbf{u} \text{ on } \Gamma_t & (63)\\
D_t\phi &= -gz + \frac{1}{2}(\nabla\phi\cdot\nabla\phi) \text{ on } \Gamma_t & (64)\\
\phi_n &= 0 \text{ on } \Gamma_b \cup \Gamma_1 \cup \Gamma_2, & (65)
\end{aligned}$$

Let Ω_1 be a fixed 2D domain that contains Γ_t for all times. Following the same embedding procedure for the potential function as in previous section, we obtain the complete 2D Eulerian formulation:

$$\begin{aligned}
\mathbf{u} &= \nabla\phi \text{ in } \Omega_t & (66)\\
\Delta\phi &= 0 \text{ in } \Omega_t & (67)\\
\Psi_t + \mathbf{u}_{\text{ext}}\cdot\nabla\Psi &= 0 \text{ in } \Omega_1. & (68)\\
G_t + \mathbf{u}_{\text{ext}}\cdot\nabla G &= f_{\text{ext}} \text{ in } \Omega_1 & (69)\\
\phi_n &= 0 \text{ on } \Gamma_b \cup \Gamma_1 \cup \Gamma_2 & (70)
\end{aligned}$$

being here $f = \frac{1}{2}(\nabla\phi\cdot\nabla\phi) - gz$ and f_{ext} the extension of f onto Ω_1.

5 Numerical Approximations and Algorithms

In this section, we provide overviews of the numerical schemes used to approximate the wave model equations. The integral formulation of Eq. (66) is approximated using a liner boundary element method (BEM), which will provide the velocity of the front node representation. More detailed discussions of the various components may be found in the cited references.

5.1 Initialization

The initial front position Γ_0 and initial velocity potential $\phi(Q,0)$, $Q \in \Gamma_0$, are needed to solve equations (68) and (69) 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. 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

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

Under sensible assumptions about the smoothness of ϕ and ψ, these are just the Cauchy-Riemman equations which are satisfied by the real and imaginary parts of the function $W = \phi + i\psi$, which is called the complex potential and it is a 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 ϕ and ψ 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 θ 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 θ as:

$$\begin{aligned} \frac{dq^3}{d\phi} &= -\frac{3}{F^2}\sin\theta \quad \text{on } \psi = 1 & (71) \\ \theta &= 0 \quad \text{on } \psi = 0, & (72) \end{aligned}$$

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 ω 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 (71) and (72). Tanaka's method provides a way to solve the previous outlined equations in terms of the new variables τ, θ and a full description of the algorithm can be found in ([40]).

5.2 The level set and velocity potential updating

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 [31], 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, which are implemented by creating an extra layer of ghost cells around the domain whose values are simply direct copies of the Ψ 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 [36], 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 [26]; 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 [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.

The potential equation (69) is a convection equation with a strong non-linear source term, and homogeneous Newmann boundary conditions are imposed on the boundary of Ω_1. To update in time this equation, note that it is similar to (68) 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

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

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

are the backward and forward finite approximation for the derivative in the x direction (we have the same expressions for 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 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 $\phi^n(x, z, n\Delta t)$ to obtain a better value of $G_{i,j}^n$.

A key issue is how one obtains f_{ext} in the grid points of Ω_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_{ext}. This algorithm for extending quantities defined on the front off the front works very well for the velocity field in the case of equation (68), because it maintains the signed distance function for the level sets of Ψ. However, regarding equation (69) for this particular wave problem, and due to the high variations of f along the front together with its topological structure when overturning, the previous method creates strong G and f gradients in Ω_1. This fact limits the grid spacing in Ω_1 and the time step needed to maintain accuracy (see the section on numerical experiments).

5.3 The Boundary Integral equation and the BEM approximation

A first order boundary element method is used to approximate equation (66). 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 (67) is solved by approximating the corresponding boundary integral equation. Boundary conditions are given by (70) and, on the free boundary, at each time step, by the updated potential velocity given by equation (69). 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

$$\mathcal{P}(P) = \phi(P) + \lim_{P_I \to P} \int_{\Sigma} \left[\phi(Q) \frac{\partial \mathcal{G}}{\partial \mathbf{n}}(P_I, Q) - \mathcal{G}(P_I, Q) \frac{\partial \phi}{\partial \mathbf{n}}(Q) \right] \mathrm{d}Q = 0 \ , \qquad (73)$$

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

$$\mathcal{G}(P,Q) = -\frac{1}{4\pi}\log(r)\ . \tag{74}$$

The integral equation is usually written with the $\frac{\partial\mathcal{G}}{\partial\mathbf{n}}$ singular integral evaluated as a Cauchy Principal Value (CPV), resulting in a '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

$$\lim_{P_E\to P}\int_\Sigma \left[\phi(Q)\frac{\partial\mathcal{G}}{\partial\mathbf{n}}(P_E,Q) - \mathcal{G}(P_E,Q)\frac{\partial\phi}{\partial\mathbf{n}}(Q)\right]\,\mathrm{d}Q = 0\ . \tag{75}$$

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. (73) 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

$$\int_\Sigma \psi_k(P)\mathcal{P}(P)\,\mathrm{d}P = 0\ , \tag{76}$$

where the weight functions $\hat{\psi}_k(P)$ are comprised of all shape functions which are non-zero at a particular node P_k [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 [14].

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 ϕ is required to compute the surface velocity. The remainder of this section will present the algorithm for computing this gradient.

From Eq. (73) a gradient component can be expressed as

$$\frac{\partial\phi(P)}{\partial\mathbf{E}_k} = \lim_{P_I\to P}\int_\Sigma \left[\frac{\partial\mathcal{G}}{\partial\mathbf{E}_k}(P_I,Q)\frac{\partial\phi}{\partial\mathbf{n}}(Q) - \phi(Q)\frac{\partial^2\mathcal{G}}{\partial\mathbf{E}_k\partial\mathbf{n}}(P_I,Q)\right]\,\mathrm{d}Q\ . \tag{77}$$

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 [28, 29, 27]. As described in [15], a Galerkin approximation of this equation,

$$\int_\Sigma \hat{\psi}_k(P)\frac{\partial\phi(P)}{\partial\mathbf{E}_k}\,\mathrm{d}P = \tag{78}$$
$$\lim_{P_I\to P}\int_\Sigma \hat{\psi}_k(P)\int_\Sigma\left[\frac{\partial\mathcal{G}}{\partial\mathbf{E}_k}(P_I,Q)\frac{\partial\phi}{\partial\mathbf{n}}(Q)-\phi(Q)\frac{\partial^2\mathcal{G}}{\partial\mathbf{E}_k\partial\mathbf{n}}(P_I,Q)\right]\mathrm{d}Q\,\mathrm{d}P$$

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 Σ; 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, Eq. (75). It appears to be useless for computing tangential derivatives for, lacking the free term, the corresponding derivative equation takes the form

$$0=\lim_{P_E\to P}\int_\Sigma\left[\frac{\partial\mathcal{G}}{\partial\mathbf{E}_k}(P_E,Q)\frac{\partial\phi}{\partial\mathbf{n}}(Q)-\phi(Q)\frac{\partial^2\mathcal{G}}{\partial\mathbf{E}_k\partial\mathbf{n}}(P_E,Q)\right]\mathrm{d}Q\ , \tag{79}$$

and the derivatives obviously do not appear. However, subtracting this equation from Eq. (77) yields (with shorthand notation)

$$\frac{\partial\phi(P)}{\partial\mathbf{E}_k}=\left\{\lim_{P_I\to P}-\lim_{P_E\to P}\right\}\int_\Sigma\left[\frac{\partial\mathcal{G}}{\partial\mathbf{E}_k}\frac{\partial\phi}{\partial\mathbf{n}}(Q)-\phi(Q)\frac{\partial^2\mathcal{G}}{\partial\mathbf{E}_k\partial\mathbf{n}}\right]\mathrm{d}Q\ . \tag{80}$$

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. (80) 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 Eq. (80) can be found in [15].

5.4 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 Ψ. In order to extract the front, it is possible to construct first order and second order approximations of the interface using local data of Ψ 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 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.

5.5 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 $\phi(Q, 0)$ on Γ_0.
2. Extend $\phi(Q, 0)$ onto the grid points of Ω_1 to initialize G.
3. Generate Ω_t and solve (67), 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 Ω_1.
5. Update G using (69) in Ω_1.
6. Move the front with velocity $\mathbf{u}$ using (68) in Ω_1
7. Interpolate (bi-cubic interpolation) G from grid points of Ω_1 to the front nodes to obtain new boundary conditions for (67). Go back to step 3 and repeat forward in time.

A more detailed algorithm including regridding is:
Initialization: Given $\Gamma^0 = \Gamma_0, \phi^0 = \phi(Q, 0)$

1. Calculate Ψ^0 and LS nodes.

2. Extend ϕ^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 Ω_1.

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

1. Calculate Ψ^{n+1} and LS nodes.
2. Calculate G^{n+1} in Ω_1 grid points.
3. Redistribute LS nodes to obtain BEM nodes.
4. Interpolate G on BEM nodes to find ϕ^{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 Ω_1. Go to step 1 and repeat.
7. If reinitialization
   (a) Take LS nodes and reinitialize Ψ^{n+1}.
   (b) Take BEM nodes and extend ϕ^{n+1}.

5.6 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

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

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

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

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

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

where the divergence theorem has been applied to the three formulas and we have used the fact $\frac{\partial\phi}{\partial n} = 0$ on $\Gamma_b, \Gamma_1, \Gamma_2$ for the kinetic energy formula. These integrals are approximated by a composite trapezoidal rule, using the values of the quantities on the free boundary BEM nodes. Note that LS nodes could have been used for $m(t)$ ans $E_p(t)$ 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 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 [22].

6 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 (68); first order in time and in space schemes are used for equation (69); 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 heigth 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 Δt 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 Γ_1 and Γ_2, $\Delta l = 0.2$ for Γ_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 Ω_1 should be choosen 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 (68) and (69), the accuracy in the numerical solution of equation (69) requires a smaller time step. This is due to the fact that G and the source term f, for this particular wave problem, develops high gradients in Ω_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 a 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}), ϕ_{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), ϕ_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 5 and 6 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 (Δx) but not with the time step. Only the velocity potential gains accuracy when Δt is reduced accordingly 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 7 and 8 show the values of $|m(t) - m(0)|$ and $|E(t) - E(0)|$ versus time and same behaviour 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 Γ_1 and Γ_2, $\Delta l = 0.1$ on Γ_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 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 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 is shown in Figure 8, $|m(t) - m(0)|$ 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 9 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 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 10, 11 and 12 respectively.

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 heigth 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 the 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 lost 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 ([21]). 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 Ψ 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 all reported in ([21]) 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 Lagrangean-Eulerian formulation.

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

In Figure 13 we show $m(t)$ versus time for case (a) and (b) and Figures 14 and 15 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 energy is 0.02 near the BP. Although this 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 16 shows wave shape for case (a) at $t = 0, 1, 2, 2.76, 2.94, 2.14, 3.34$ and Figure 17 shows wave shape for case (b) at $t = 0, 1, 2, 2.34, 2, 48, 2.68, 2.90$. In Figures 18 and 19 we show in more detail the wave profiles from the BP to the EP for cases (a) and (b) respectively. Finally in Figure 20 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 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 ([21]). This is not surprising due to the fact that in ([21]) 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 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 21 and 22 we show wave profiles for various times corresponding to case (c) and (d) respectively. Measurements for the mass and total energy conservation behave similar to previous cases. In Figure 23 we show the evolution of wave mass for cases (c) and (d). Finally, Figures 24 and 25 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.

7 Conclusion

To summarize, in this chapter we have derived some physical models related to moving interfaces in an intricsic way, that is, independent of any coordinate system. Based on these models a complete Eulerian description of potential flow problems for a single fluid, with or without advection-diffusion of material quatities on the front has been stablished. For the case of two-dimensional breaking waves over sloping beaches a coupled level set-boundary integral algorithm has been developed. Numerical results and convergence tests show that the novel algorithm gives enough accuracy just using first order numerical schemes.

Acknowledgements:

All work was performed at the Lawrence Berkeley National Laboratory, and the Mathematics Dept. of the University of California at Berkeley. First author was partially supported by the Spanish Project MTM2004-05417. Second author was supported by Spanish CGL2005-02233-BTE project funded by Ministerio de Educació n y Ciencia and Fondo Europeo de Desarrollo Regional (FEDER).

Work supported by the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Appendix I. The surface divergence in rectilinear coordinates

Given a vector field $\mathbf{w}$, we want to find an expression for the surface divergence $\operatorname{Div}\mathbf{w}$ using rectilinear coordinates. We start from the definition:

$$\omega(\mathbf{a},\mathbf{b})\operatorname{Div}\mathbf{w} := \partial_{\mathbf{a}}\mathbf{w}\cdot\mathbf{b}\times\mathbf{n} + \partial_{\mathbf{b}}\mathbf{w}\cdot\mathbf{n}\times\mathbf{a}, \tag{81}$$

being $\mathbf{a}$ and $\mathbf{b}$ arbitrary tangent vectors to the surface, $\omega(\mathbf{a}, \mathbf{b})$ the area of the corresponding parallelogram and $\mathbf{n}$ the unit vector normal to the surface at the same point.

To abbreviate the computations we use indices for basis vectors:[6]

$$\mathbf{a}_1 = \mathbf{a}, \quad \mathbf{a}_2 = \mathbf{b}, \quad \mathbf{a}_3 = \mathbf{n}. \tag{82}$$

The reciprocal basis, designed by $\mathbf{a}^i$ $(i = 1, 2, 3)$,

$$\mathbf{a}^i \cdot \mathbf{a}_j = \delta^i_j, \quad i, j = 1, 2, 3, \tag{83}$$

is calculated by the formulae:

$$\mathbf{a}^1 = \frac{\mathbf{a}_2 \times \mathbf{a}_3}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]}, \; \mathbf{a}^2 = \frac{\mathbf{a}_3 \times \mathbf{a}_1}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]}, \; \mathbf{a}^3 = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]} = \mathbf{n}. \tag{84}$$

According to definition (81), we have for $\operatorname{Div} \mathbf{w}$:

$$\begin{aligned} \operatorname{Div} \mathbf{w} &= \frac{\mathbf{a}_2 \times \mathbf{a}_3}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]} \cdot \partial_{\mathbf{a}_1} \mathbf{w} + \frac{\mathbf{a}_3 \times \mathbf{a}_1}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]} \cdot \partial_{\mathbf{a}_2} \mathbf{w} \\ &= \mathbf{a}^1 \cdot \partial_{\mathbf{a}_1} \mathbf{w} + \mathbf{a}^2 \cdot \partial_{\mathbf{a}_2} \mathbf{w} = \mathbf{a}^\alpha \cdot \partial_{\mathbf{a}_\alpha} \mathbf{w}. \end{aligned} \tag{85}$$

In the last expression and below we will use the summation convection: when in a monomial expression we have two repeated indices it must be interpreted as a summation, from 1 to 2 for greek indices and from 1 to 3 for latin indices.

Notice that the basis $\mathbf{a}_i$ is in general different in each surface point. We want now to express $\operatorname{Div} \mathbf{w}$ using the components and coordinates in a fixed basis (global) $\mathbf{e}_i$ and the reciprocal one $\mathbf{e}^j$, defined[7] by the nine equations $\mathbf{e}^j \cdot \mathbf{e}_i = \delta^j_i$.

We set:

$$\mathbf{a}_i = h^j_i \mathbf{e}_j, \; \mathbf{a}^i = f^i_k \mathbf{e}^k, \; \mathbf{w} = w^j \mathbf{e}_j; \quad (f^i_k h^k_j = \delta^i_j). \tag{86}$$

Substituting this expressions in the last term of (85), we have:

$$\operatorname{Div} \mathbf{w} = f^\alpha_k \mathbf{e}^k \cdot \partial_{h^i_\alpha \mathbf{e}_i} w^j \mathbf{e}_j = f^\alpha_j h^i_\alpha \partial_i w^j. \tag{87}$$

Considering that $n_j = \mathbf{a}^3 \cdot \mathbf{e}_j = f^3_j$ and $n^i = \mathbf{a}_3 \cdot \mathbf{e}^i = h^i_3$, the coefficient in the previous result becomes:

$$f^\alpha_j h^i_\alpha = f^k_j h^i_k - f^3_j h^i_3 = \delta^i_j - n_j n^i. \tag{88}$$

[6] Latin indices i, j,... go through the values 1, 2 y 3 and greek indices α, β,... go through 1 y 2. The vectors $\mathbf{a}_i$ $(i = 1, 2, 3)$ must accomplish the condition $[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3] > 0$.

[7] When $\mathbf{e}_i$ is an orthonormal basis (Cartesian coordinates) then it coincides with the corresponding reciprocal basis: $\mathbf{e}^i = \mathbf{e}_i$.

Substituting this result in equation (87), the searched expression is obtained:

$$\operatorname{Div} \mathbf{w} = \left(\delta^i_j - n_j n^i\right) \partial_i w^j. \tag{89}$$

Notice that, as we have anticipated, the final result does not depend on the selected tangent vectors $\mathbf{a}$ and $\mathbf{b}$.

Appendix II. The differential operator A

We are going to show that the differential operator

$$A = \frac{1}{\omega(\mathbf{a},\mathbf{b})} \left(\partial_{\mathbf{a}}(\mathbf{n} \times \mathbf{j} \cdot \mathbf{b}) - \partial_{\mathbf{b}}(\mathbf{n} \times \mathbf{j} \cdot \mathbf{a})\right), \tag{90}$$

appearing in (25), may be written using surface divergences of $\mathbf{j}$ and $\mathbf{n}$. To do that, we perform in the previous definition the indicated derivatives,

$$\begin{aligned} A = {} & \frac{1}{\omega(\mathbf{a},\mathbf{b})} \left(\mathbf{b} \times \mathbf{n} \cdot \partial_{\mathbf{a}}\mathbf{j} + \mathbf{n} \times \mathbf{a} \cdot \partial_{\mathbf{b}}\mathbf{j}\right) + \\ & + \frac{1}{\omega(\mathbf{a},\mathbf{b})} \left(\mathbf{b} \times \partial_{\mathbf{a}}\mathbf{n} + \partial_{\mathbf{b}}\mathbf{n} \times \mathbf{a}\right) \cdot \mathbf{j}. \end{aligned}$$

According to definition (81), the first term is $\operatorname{Div} \mathbf{j}$. Let be:

$$A = \operatorname{Div} \mathbf{j} + B. \tag{91}$$

$(\alpha = 1,2)$ es un vector tangente a la superficie, tenemos In order to identify de second term B, we select the basis $\mathbf{a}_i$, following the specified notation in (82). As $\partial_{\mathbf{a}_\alpha}\mathbf{n}$ $(\alpha = 1,2)$ are tangent vectors to the surface, we can set

$$\partial_{\mathbf{a}_\alpha}\mathbf{n} = N^\beta_\alpha \mathbf{a}_\beta, \quad \alpha = 1,2. \tag{92}$$

Also, for the two terms in B we obtain:

$$\begin{aligned} \mathbf{j} \cdot \frac{\mathbf{b} \times \partial_{\mathbf{a}}\mathbf{n}}{\omega(\mathbf{a},\mathbf{b})} &= \mathbf{j} \cdot \frac{\mathbf{a}_2 \times N^1_1 \mathbf{a}_1}{\omega(\mathbf{a}_1,\mathbf{a}_2)} = -N^1_1 \mathbf{j} \cdot \mathbf{n}, \\ \mathbf{j} \cdot \frac{\partial_{\mathbf{b}}\mathbf{n} \times \mathbf{a}}{\omega(\mathbf{a},\mathbf{b})} &= -N^2_2 \mathbf{j} \cdot \mathbf{n}. \end{aligned}$$

Therefore:

$$B = -N^\alpha_\alpha \mathbf{j} \cdot \mathbf{n}. \tag{93}$$

divind), obtenemos: On the other hand, as $N^\beta_\alpha = \mathbf{a}^\beta \cdot \partial_{\mathbf{a}_\alpha}\mathbf{n}$, making $\alpha = \beta$, summing and using the result (85), we have:

$$N^\alpha_\alpha = \mathbf{a}^\alpha \cdot \partial_{\mathbf{a}_\alpha}\mathbf{n} = \operatorname{Div} \mathbf{n}. \tag{94}$$

Finally, substituting this result in (93) we arrive to the searched expression:

$$A = \operatorname{Div} \mathbf{j} - (\operatorname{Div} \mathbf{n}) \mathbf{j} \cdot \mathbf{n}. \tag{95}$$

Appendix III. Useful definitions

Points and vectors. In our euclidean space we can define two useful operations. Given a point P and a displacement vector $\mathbf{a}$, we define $P + \mathbf{a}$ as the point that results translating point P by vector $\mathbf{a}$. Also, given two points A and B, we define $A - B$ as a vector $\mathbf{c}$, so that:

$$A - B := \mathbf{c} \quad \Leftrightarrow \quad B + \mathbf{c} = A. \tag{96}$$

Directional derivative. Given a tensor field $\mathbf{w} = w(P, t)$, function of the position P (a point of our Euclidean space) and the time t, we define the directional derivative along the vector $\mathbf{a}$ as

$$\partial_{\mathbf{a}}\mathbf{w} := \frac{\mathrm{d}}{\mathrm{d}\epsilon} w(P + \epsilon\mathbf{a}, t)\bigg|_{\epsilon=0}. \tag{97}$$

The result $\partial_{\mathbf{a}}\mathbf{w}$ is a tensor of the same rank that $\mathbf{w}$. Differentiability of the field $\mathbf{w}$ implies that $\partial_{\mathbf{a}}\mathbf{w}$ is a linear function of the vector $\mathbf{a}$.

When $\mathbf{w}$ is a vector field, we use, as customary, the special notation:

$$\mathbf{a} \cdot \nabla\mathbf{w} := \partial_{\mathbf{a}}\mathbf{w}. \tag{98}$$

In this case, $\partial_{\mathbf{a}}\mathbf{w}$ is a linear operator acting on the vector $\mathbf{a}$.

Gradient of a scalar field. Let us consider a scalar field $\phi = \varphi(P, t)$. As $\partial_{\mathbf{a}}\phi$ is a real valued linear function on the argument $\mathbf{a}$, we can define the gradient vector field, $\nabla\phi$,

$$\mathbf{a} \cdot \nabla\phi := \partial_{\mathbf{a}}\phi. \tag{99}$$

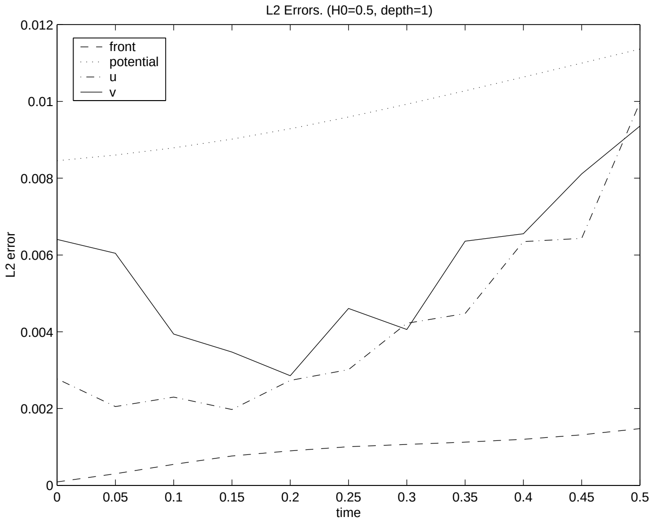

Figure 5: $L_2(z), L_2(\phi), L_2(u), L_2(v)$ vs time for case (c)

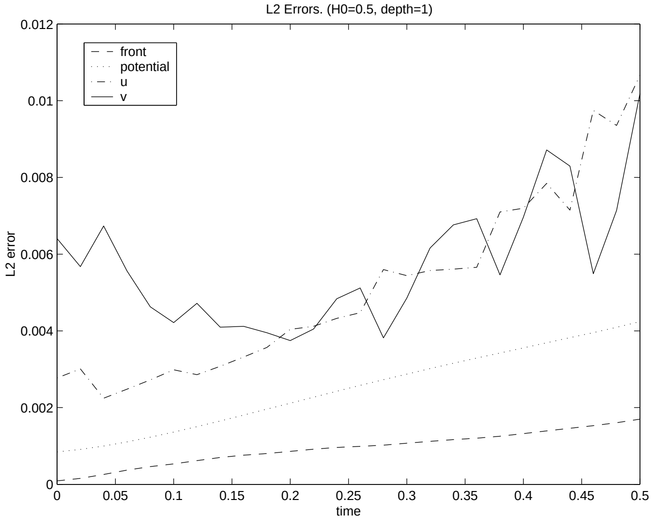

Figure 6: $L_2(z), L_2(\phi), L_2(u), L_2(v)$ vs time for case (d)

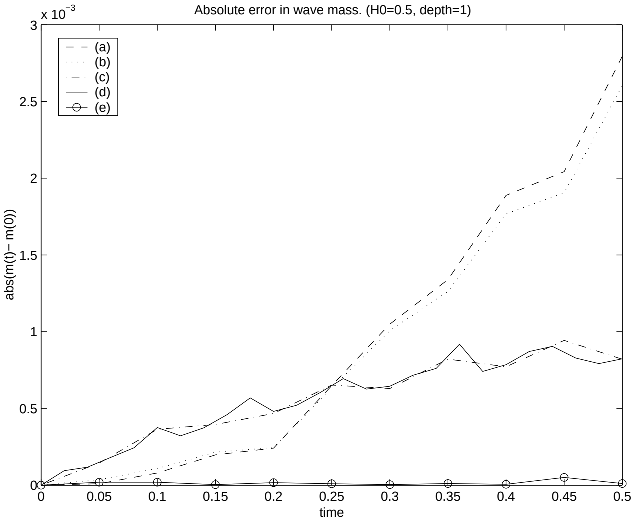

Figure 7: Absolute error in wave mass

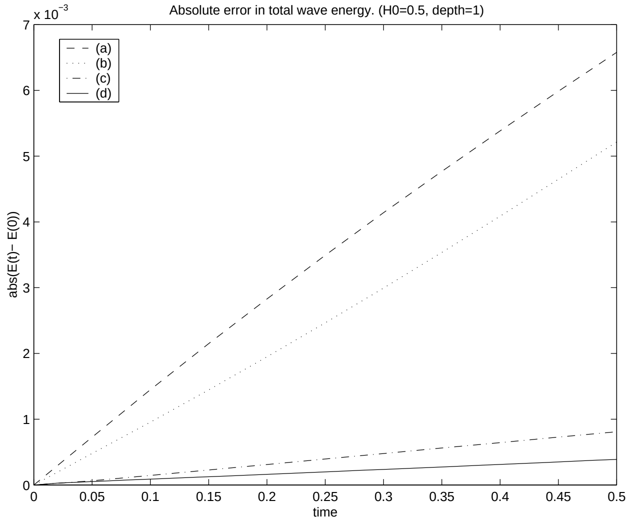

Figure 8: Absolute error in wave total energy

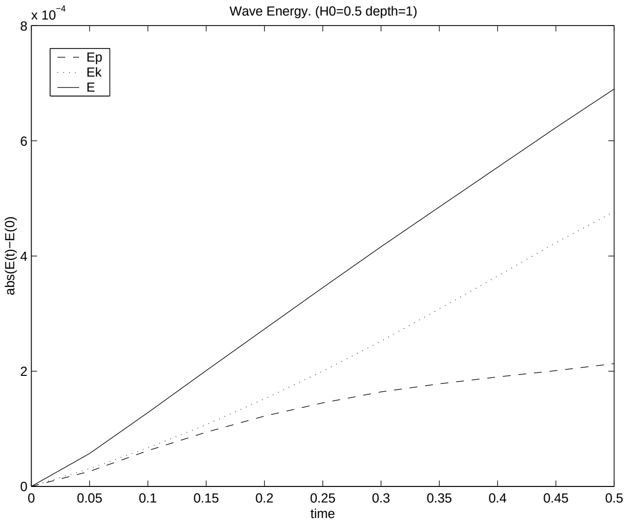

Figure 9: Absolute error in potential, kinetic and total energy. Case (e)

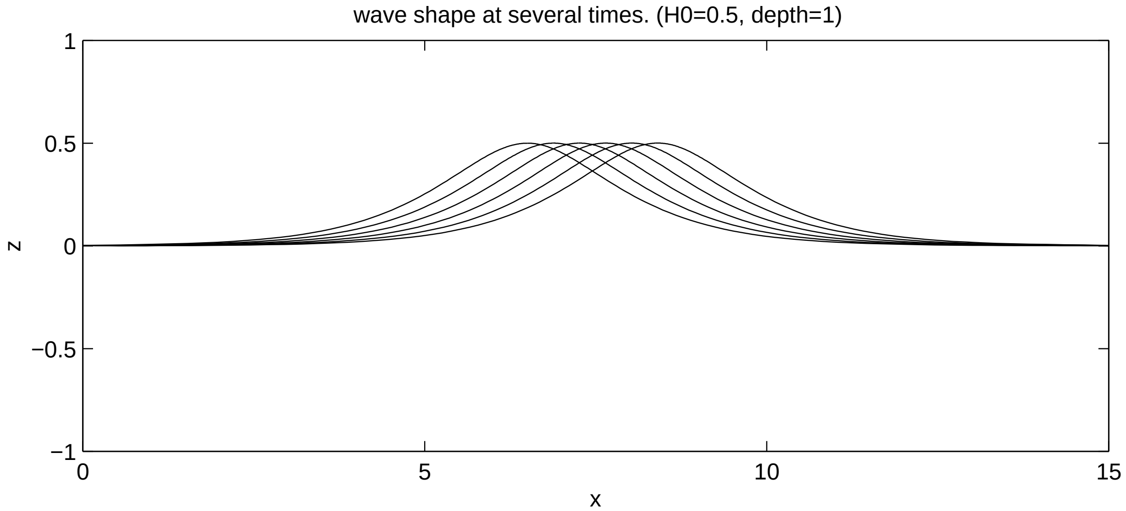

Figure 10: Front location at $t = 0, 0.1, 0.2, 0.3, 0.4, 0.5$. Case (c)

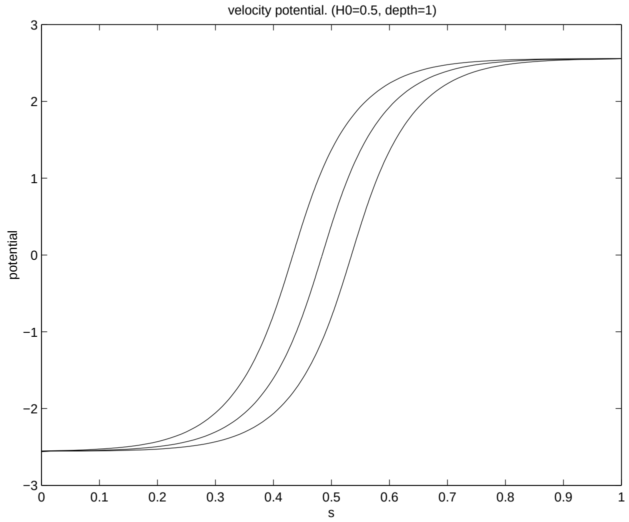

Figure 11: Velocity potential at $t = 0, 0.25, 0.5$. Case (c)

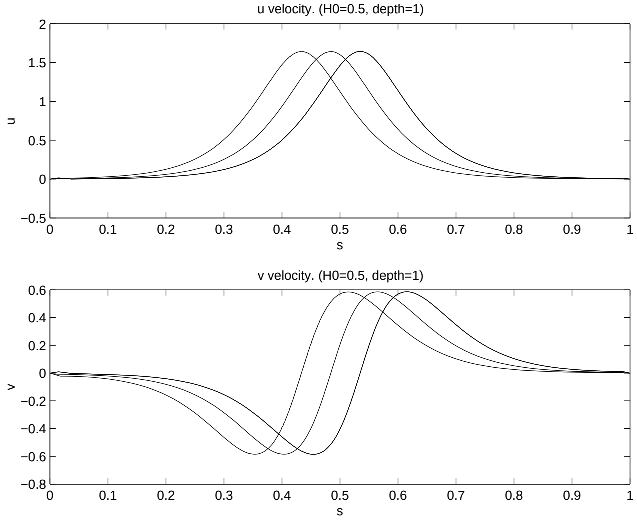

Figure 12: Velocity components at $t = 0, 0.25, 0.5$. Case (c)

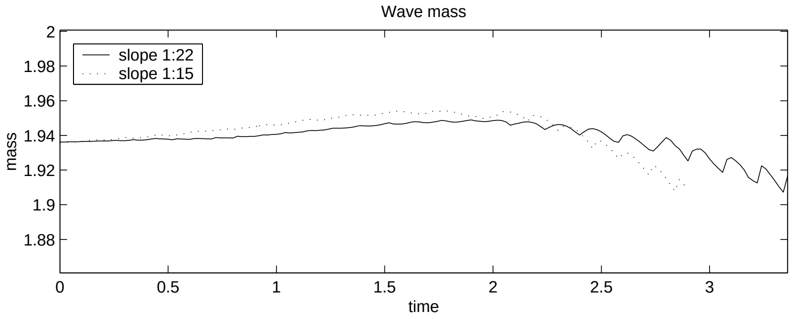

Figure 13: Wave mass vs time. Case (a) and (b)

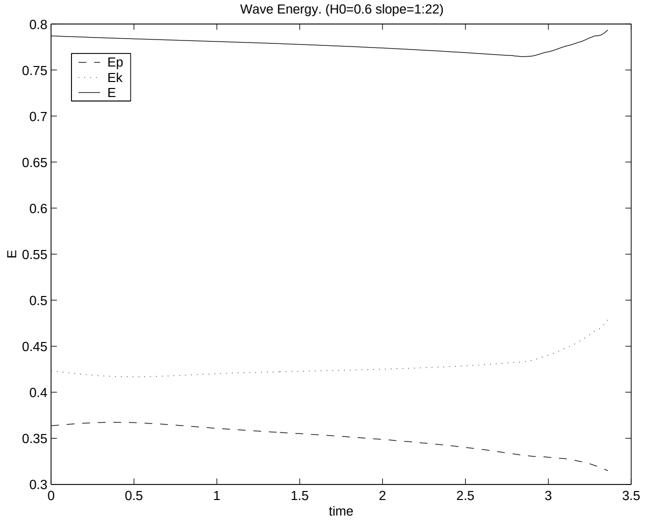

Figure 14: Wave energy. Case (a)

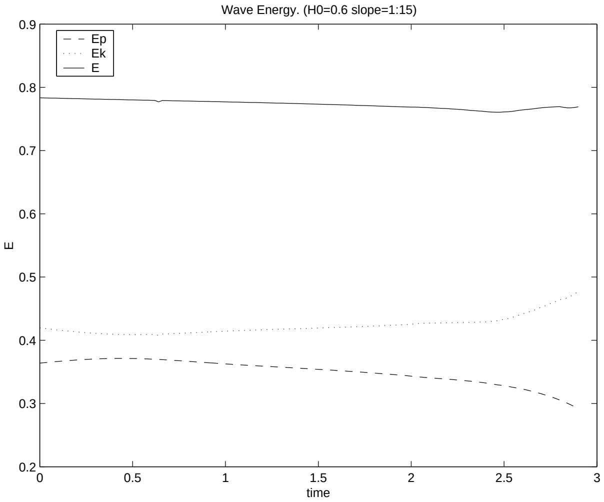

Figure 15: Wave energy. Case (b)

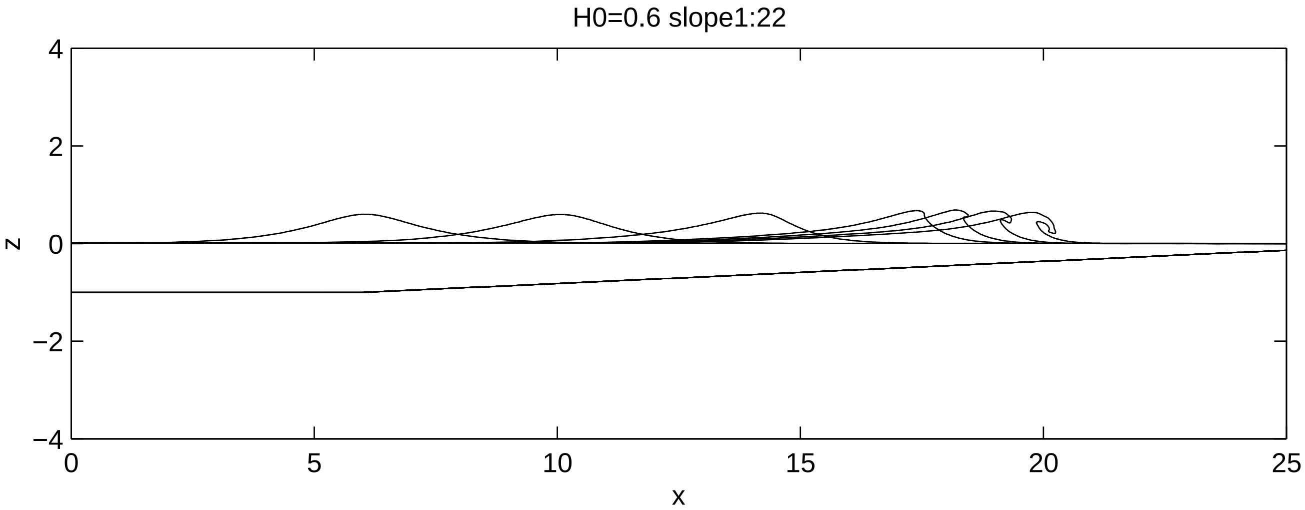

Figure 16: Wave shape at various times. Case (a)

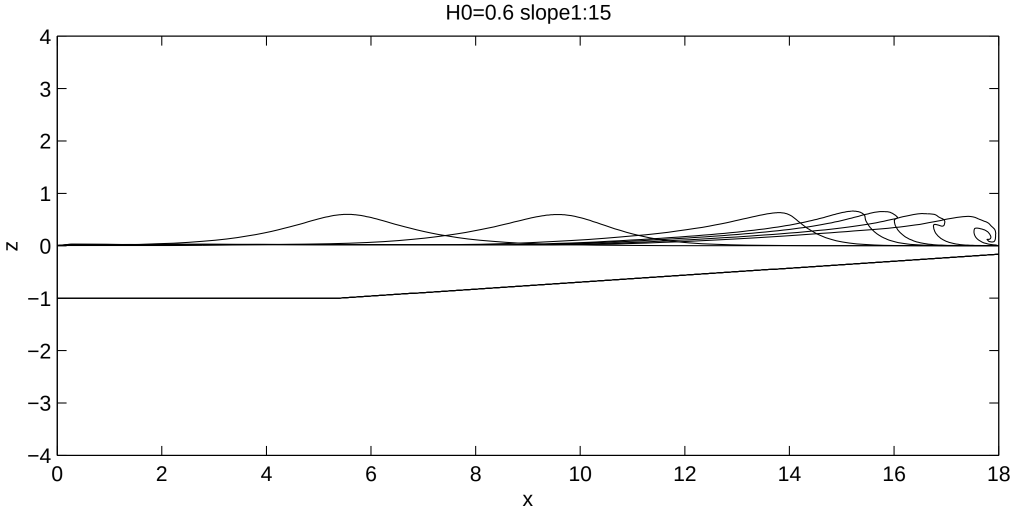

Figure 17: Wave shape at various times. Case (a)

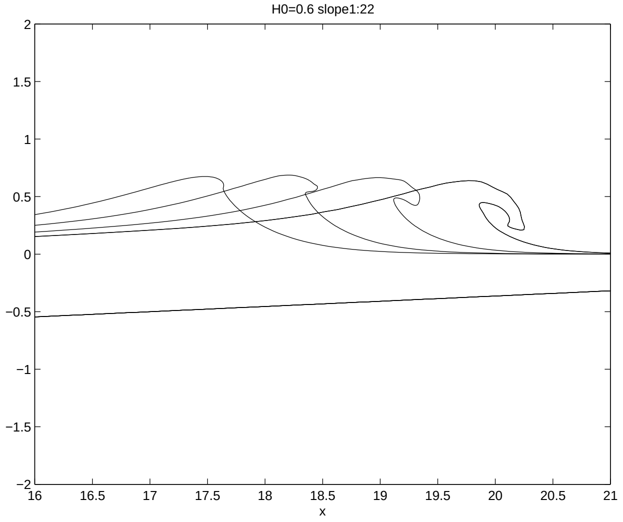

Figure 18: Wave shape at various times. Case (a)

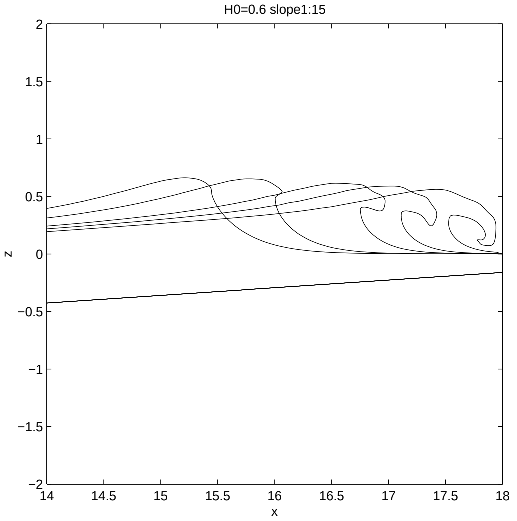

Figure 19: Wave shape at various times. Case (b)

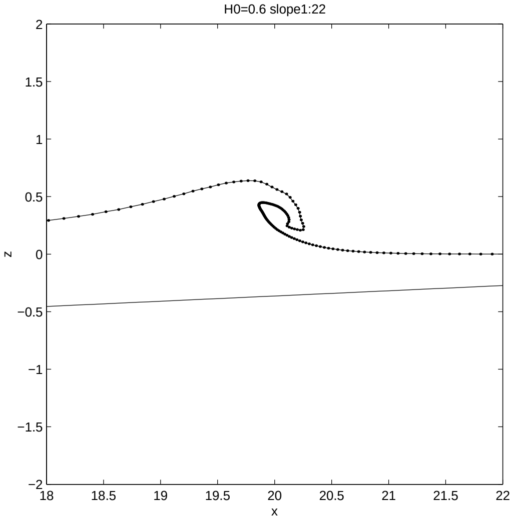

Figure 20: Front BEM nodes at t=3.34. Case (a)

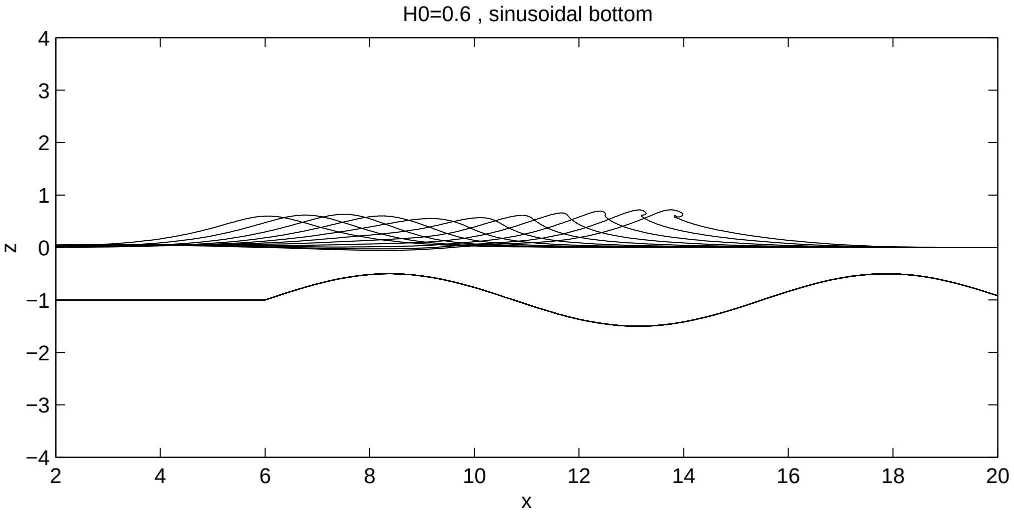

Figure 21: Wave shape at various times. Case (c)

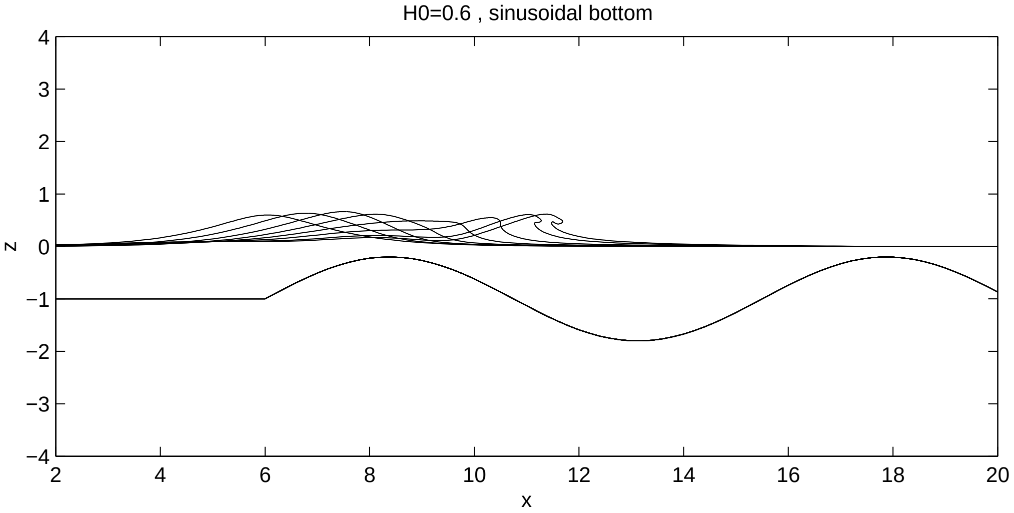

Figure 22: Wave shape at various times. Case (d)

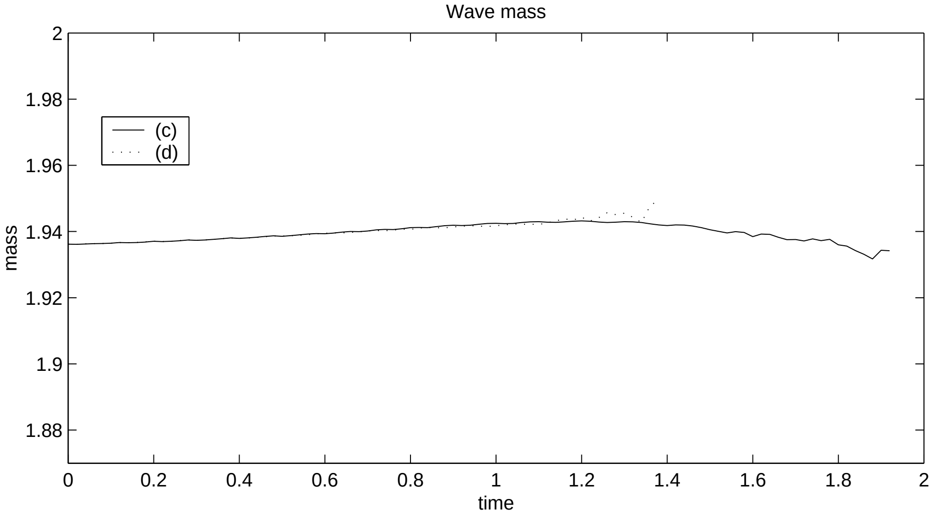

Figure 23: Wave mass vs time. Case (c) and (d)

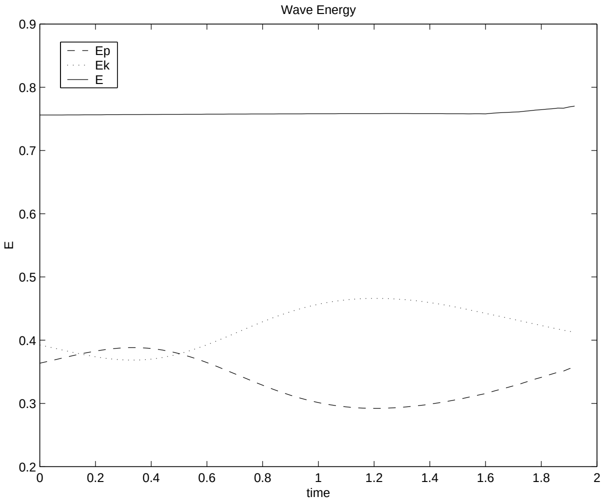

Figure 24: Wave energy. Case (c)

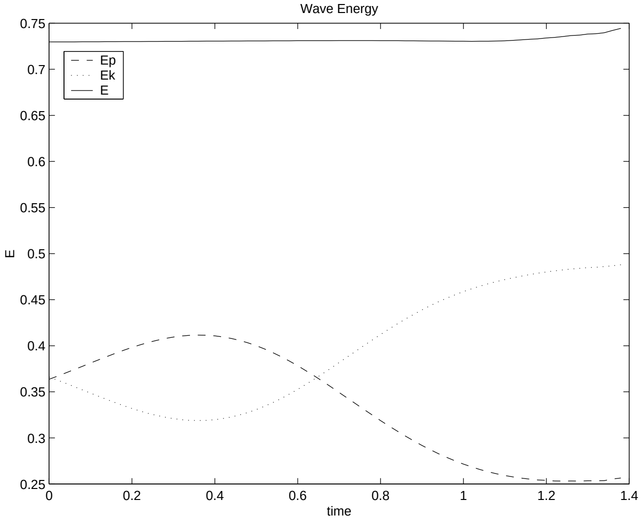

Figure 25: Wave energy. Case (d)

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] Beale, J. Thomas, Hou, Thomas Y., Lowengrub, John, *Convergence of a boundary integral method for water waves*, SIAM J. Numer. Anal., 33, 5, pp.1797-1843, 1996.

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

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

[7] 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.

[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, pp. 449–64, 1996.

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

[10] Chopp, D.L., *Computing minimal surfaces via level set curvature flow.* J. Comput. Phys. 106, pp. 77–91, 1993.

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

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

[13] Eggers, Jen, *Nonlinear dynamics and breakup of free-surface flows*, Rev. Mod. Phys., 69, 3, pp.865-929, 1997.

[14] 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.

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

[16] M. Garzon, D. Adalsteinsson, L. J Gray and J. A Sethian, *A coupled level set-boundary integral method for moving boundaries simulations*, Interfaces and Free Boundaries, 7, 277-302, 2005.

[17] M. Garzon, J.A. Sethian, *Wave breaking over sloping beaches using a coupled boundary integral-level set method*, International Series of Numerical methods, 154, 189-198, 2006.

[18] L. J. Gray, M. Garzon, *On a hermite boundary integral approximation*, Computers and Structures, 83, 889-894 (2005).

[19] *Integral analysis for the axisymetric laplace equation*, L. J. Gray, M. Garzon, V. Mantic, and E. Graciani, International Journal For Numerical Methods in Engineering, 66, 2014-2034, 2005.

[20] M. Garzon, J.A. Sethian, L. Gray, *Numerical solution of non-viscous pinch off using a coupled level set boundary integral method*, Proceedings in Applied Mathematics and Mechanics, 2007.

[21] 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).

[22] 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).

[23] 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.

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

[25] 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).

[26] 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.

[27] 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.

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

[29] 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.

[30] Notz, Patrick K., and Basaran, Osman A., *Dynamics of drop formation in an electric field*, J. Colloid Interface Sci., 213, p.. 218-237, 1999.

[31] 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.

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

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

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

[35] 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.

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

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

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

[39] 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.

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

[41] Yan, Fang, Farouk, Baktier, and Ko, Frank, *Numerical modeling of an electrostatically driven liquid meniscus in the cone-jet mode*, Aerosol Science, 34, pp. 99-116, 2003,

[42] 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.

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

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

[45] Xinan Liu, J. H. Duncan *The effects of surfactants on spilling breaking waves* Nature International weekly journal of science, 421, pp. 520-523, 2003.

[46] Maik J. Geerkena, Rob G.H. Lammertink and Matthias Wessling *Interfacial aspects of water drop formation at micro-engineered orifices* Journal of Colloid Interface Science, 312, pp. 460-469, 2007.