

# FULL PARTICLE PIC MODELLING OF THE SURF AND SWASH ZONES

D. M. Kelly<sup>1</sup>

In this paper a hybrid Eulerian Lagrangian solver based on the full-particle Particle-In-Cell (PIC) method is outlined. The solver is capable of simulating incompressible free-surface flows in domains with arbitrary, free-slip, solid boundaries. The flexibility of the approach allows for simulation of wetting and drying and pooling as well as wave breaking, splash-up over complex obstacles and the overtopping of coastal structures. The method has been validated for a wide variety of test cases and results are in good agreement with the numerical and experimental results of other researchers.

*Keywords: Surf; Swash; Wave breaking; Run-up; Topography; Structures*

## INTRODUCTION

The inner surf and swash zone is a region of the nearshore that is characterised by intense hydrodynamic activity. The hydrodynamics of the inner surf and swash zones have traditionally been modelled using the Boussinesq and/or non-linear shallow water (NLSW) equations (see e.g. (Karambas and Tozer, 2003)). A Boussinesq approach is suitable for modelling waves in both the surf and swash zone through the specification of a suitable breaking criterion and energy dissipation mechanism; however, the specification of such criteria is necessarily somewhat *ad-hoc*. Although the NLSW equations handle wave breaking naturally, through shock wave theory, they do not account for the dispersive effects that occur before wave breaking. As a consequence a number of researchers have attempted to marry together the Boussinesq and NLSW equations using a ‘switch’, typically based on local wave steepness, to alternate between the two (see e.g. (Tonelli and Petti, 2009)). Recently, with the advent of cheap computational power, the Navier Stokes equations have become increasingly popular to model this region of the nearshore. These equations are valid for the entire nearshore region without the need for the *ad-hoc* parameterisation of wave breaking or a ‘switch’ function. Whilst pure Eulerian methods have been used (e.g. (Jiang et al., 2011)) to solve the Navier Stokes equations a Lagrangian approach is better suited to free surface flows as it handles both wave breaking and wetting drying processes naturally. Pure Lagrangian methods such as the Smooth Particle Hydrodynamic (SPH) and Moving-Particle Semi-Implicit (MPS) schemes have become increasingly popular (Rogers et al., 2008; Seichi et al., 1998). Of these, SPH has become the Lagrangian method of choice for nearshore dynamics and has been used to simulate complex free surface flows with considerable success (see e.g. (Dalrymple and Rogers, 2006)).

The motivation behind the work presented here is to develop a robust useable code that has the flexibility of SPH but the efficiency of an Eulerian approach. The full particle PIC algorithm described here uses the fractional step method for time integration employing the projection method of (Chorin, 1968). The projection technique enforces incompressibility and allows the velocity and pressure fields to be computed distinctly. Solution of the Poisson equation is effected on the mesh and can therefore be optimised.

## GOVERNING EQUATIONS

In vector notation the governing equations for incompressible homogeneous flow have the following form (Chorin and Marsden, 1993):

$$\nabla \cdot \vec{u} = 0, \quad (1)$$

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} = \vec{f} - \frac{1}{\rho} \nabla p + \nu \nabla^2 \vec{u}. \quad (2)$$

Equation (1) is the continuity equation for an incompressible fluid and equations (2) are the Navier Stokes equations for the momentum of a viscous fluid.

$\vec{u} = [u, v]^T$  is the velocity field in two spatial dimensions ( $u$  and  $v$  being the horizontal and vertical components respectively),  $p$  is pressure,  $\vec{f} = [g_x, g_y]^T$  represents the vector of body forces acting on the water due to gravity,  $\rho$  is the water density and  $\nu$  is the kinematic viscosity of water.

---

<sup>1</sup>Research Engineer, HR Wallingford, Wallingford, OX10 8BA, United Kingdom

## NUMERICAL MODEL

### Full Particle PIC

The classical PIC method was developed by Harlow and co-workers at the Los Alamos National Laboratory, USA for compressible flows; refer to (Harlow, 1964) for a detailed description. Although the classical PIC method is ingenious, it suffers from severe numerical diffusion due to two transfers of the velocity field; velocity interpolation is carried out first from the particles to the grid and then back to the particles every time-step. To reduce the diffusion (Brackbill and Ruppel, 1986) suggested incrementing the particle velocity field based on the change in the grid velocity field once the Eulerian computation has been conducted. This approach, known as full particle PIC, effectively obviates the numerical diffusion found in classical PIC. The full particle PIC approach was first applied to incompressible fluids with grid-aligned boundaries by (Zhu and Bridson, 2005). Here the method of (Zhu and Bridson, 2005) is modified to include simpler tracking of the free surface boundary and the ability to deal with arbitrarily shaped solid boundaries via a finite volume (FVM) cut-cell approach. Below the basic methodology is outlined, further detail can be found in (Kelly, 2012).

### Algorithm

(Chorin, 1968) proposed the use of time-operator-splitting for the Navier Stokes equations in order to decouple  $\vec{u}$  and  $p$ . Chorin's projection approach enforces incompressibility and allows the velocity and pressure fields to be computed distinctly. The scheme described here is similar to that of (Zhu and Bridson, 2005) and splits the solution of equations (1) and (2) into four distinct sub-steps; the first three steps are effected in a purely Eulerian manner and are carried out on the grid while the last step is purely Lagrangian and carried out on the particles. The Eulerian steps use fractional increments in time to 1. apply body forces, 2. apply velocity diffusion, 3. project the obtained, tentative, velocity field ( $\vec{u}$ ) onto the nearest divergence free velocity field using Helmholtz-Hodge decomposition. In order to achieve this step the pressure Poisson equation must be solved. The final step, step 4., is the Lagrangian advection step.

### Eulerian Steps

Following (Harlow and Welch, 1965) a staggered grid is used where the pressure is stored at the centre of the grid cells and the  $u$  and  $v$  components of the velocity are stored on the vertical and horizontal cell sides respectively. Such a grid ensures that second-order central differencing can be employed without introducing any false zero derivatives. The purpose of the underlying grid is to increase efficiency when accounting for the body forces, applying the boundary conditions and ensuring that the fluid remains incompressible. In the following description the usual  $i, j$  subscript notation is used to define variables at cell centres. Variables at cell sides are defined by  $i \pm \frac{1}{2}$  and  $j \pm \frac{1}{2}$  for vertical and horizontal sides respectively. The first stage in the grid calculation is to interpolate the velocity field from the particles onto the grid. The change in the velocity field over a time step due to the body forces is then computed and the velocities diffused to give the tentative velocity field,  $\vec{u}$ , at each grid point. The tentative velocity field now needs to be projected onto the nearest divergence free velocity field to ensure that the flow remains incompressible. To do this the pressure Poisson equation is first solved and the pressure gradient is then subtracted from the tentative velocity field.

The pressure Poisson equation is constructed following (Chorin and Marsden, 1993). Once the body forces and velocity diffusion terms have been integrated, ignoring the convection term, the remaining term from (2) is:

$$\frac{\Delta \vec{u}}{\Delta t} = \rho^{-1} \nabla p \quad (3)$$

as  $\rho$  and  $\Delta t$  are constant it is possible to introduce  $\varphi = \Delta t \rho^{-1} p$  to give

$$\vec{u} = \vec{u} + \nabla \varphi \quad (4)$$

Taking the divergence of both sides of (4) and recalling that the divergence of the velocity field at the next time step,  $\vec{u}^{n+1}$ , is zero gives a Poisson equation for  $\varphi$ :

$$\nabla^2 \varphi = -\nabla \cdot \vec{u}. \quad (5)$$

Boundary conditions are applied when the Poisson equation is solved.

### Lagrangian Step

The Lagrangian step displaces the particles in the, divergence free, velocity field. As the water mass is stored on the particles, as long as incompressibility is enforced accurately (i.e. the divergence of the velocity field vanishes to within a preset tolerance level), there is no mass diffusion and practically no loss of fluid volume. Note that in order to convect the velocity field the particles are moved through the grid velocity field transporting their, newly incremented, particle velocity values. Using bilinear interpolation the grid velocity is interpolated at the particle positions to give  $\vec{u}_p$  and the particles are displaced according to

$$\frac{D\vec{x}_p}{Dt} = \vec{u}_p, \quad (6)$$

To ensure stability the ODE (6) is solved using a Runge Kutta method that is third-order accurate in time. Finally, after they have been convected, velocities must be transferred from the particles to the Eulerian grid. Also, once the velocity field has been updated on the grid,  $\frac{d\vec{u}}{dt}$  must be computed and interpolated to the particle positions in order to increment the particle velocities. To transfer velocities from the particles to the grid, a weighted sum of the particle velocity components used:

$$\vec{u}_g = \sum_p \sigma^{-1} \vec{u}_p S_n(\vec{x}_p - \vec{x}_g), \quad (7)$$

where  $S_n$  is an assignment function with bounded support and  $\sigma$  is a weighting function (see (Brackbill and Ruppel, 1986) for further details). The subscripts  $p$  and  $g$  represent values on particles and at the correct (in relation to the velocity component) cell edges respectively. For this work a linear assignment function was used to transfer particle velocities to the grid. Particle velocity is incremented by interpolating the velocity change from the grid to the particle position using bi-linear interpolation.

### BOUNDARY CONDITIONS

Discretisation of the pressure Poisson equation is undertaken in such a manner that it incorporates the treatment of the free surface boundary and any arbitrary solid boundaries that may be present in the computational domain.

#### Free Surface Boundary

The simplest boundary condition to impose is the free surface boundary condition on which the pressure is set to zero. Tracking of the free surface is performed using the particles. The free surface boundary is identified using an algorithm based on the particle number density approach (Seiichi et al., 1998). Second-order accuracy for solution of the Poisson equation at the free surface boundary is ensured by employing the approach of (Gibou et al., 2002).

#### Solid Boundaries

For the staggered Cartesian mesh described above free-slip conditions are simple to implement at solid boundaries that are aligned with the underlying grid. Solid boundaries that are not grid aligned are more complex and there are a range of techniques that have been applied in the context of Marker and Cell (MAC) solvers see e.g. (Viecelli, 1969). Such methods require programming that is relatively involved. Here the finite volume method of (Noh, 1964) is used to implement a cut-cell type approach for arbitrary solid boundaries. This technique is straightforward and has been analysed in detail by (Ng et al., 2009) who showed it is convergent and second-order accurate in two spatial dimensions. Using the divergence theorem, and putting the contour integrals into discrete form, the FVM formulation of (5) is given by:

$$\begin{aligned} & S_{i-\frac{1}{2},j} \cdot \frac{(\varphi_{i,j} - \varphi_{i-1,j})}{\Delta x} + S_{i+\frac{1}{2},j} \cdot \frac{(\varphi_{i,j} - \varphi_{i+1,j})}{\Delta x} + \\ & S_{i,j-\frac{1}{2}} \cdot \frac{(\varphi_{i,j} - \varphi_{i,j-1})}{\Delta y} + S_{i,j+\frac{1}{2}} \cdot \frac{(\varphi_{i,j} - \varphi_{i,j+1})}{\Delta y} = \\ & -S_{i+\frac{1}{2},j} \cdot \tilde{u}_{i+\frac{1}{2},j} + S_{i-\frac{1}{2},j} \cdot \tilde{u}_{i-\frac{1}{2},j} - S_{i,j+\frac{1}{2}} \cdot \tilde{v}_{i,j+\frac{1}{2}} + S_{i,j-\frac{1}{2}} \cdot \tilde{v}_{i,j-\frac{1}{2}}, \end{aligned} \quad (8)$$

where  $S$  is the length fraction of a cell face that is open to water. As (8) is a symmetric positive definite linear system of equations it can be solved using one of the multitude of techniques developed for such systems. Currently the model employs a dimensional splitting approach to solve the Poisson equation. The

Length fractions open to flow in cut-cells are determined employing an algorithm based on a signed difference type approach (Osher and Fedkiw, 2002).

The projection step uses the pressure gradient to correct the tentative velocity field so that it is divergence free according to:

$$\vec{u}^{n+1} = \vec{u} - \nabla p, \forall i \pm \frac{1}{2}, j \pm \frac{1}{2}. \quad (9)$$

It is worth noting that it is often necessary to extrapolate velocities outside the water this is achieved using a variation of the approach detailed in (Aslam, 2004).

The final step carried out on the grid is to compute:

$$\Delta \vec{u}_g = \vec{u}_g^{n+1} - \vec{u}_g^n, \forall i \pm \frac{1}{2}, j \pm \frac{1}{2}. \quad (10)$$

This change in velocity is then interpolated, using bilinear interpolation (Press et al., 2007), at each particle location  $\vec{x}_p$  and the velocity stored on each particle is incremented by  $\Delta \vec{u}_p$ .

## RESULTS AND DISCUSSION

### Solitary Wave in a Walled Tank

This first test allows a comparison of model predictions of solitary wave run-up on a vertical wall with the laboratory results from the benchmark experiment of (Camfield and Street, 1969). This test has previously been used by (Chan and Street, 1970) and (Gray and Monaghan, 1998) to validate the MAC and SPH methods respectively. A solitary wave of amplitude  $H$  propagating through water of depth  $d$  is considered. Following (Gray and Monaghan, 1998) the solitary wave initial conditions were given by the Kortweg-de Vries (KDV) equation:

$$\eta = \frac{H}{\cosh^2\left(\frac{3Hx}{4d^3}\right)}, \quad (11)$$

$\eta$  is the free surface and  $x$  is distance parallel to the undisturbed surface. For the full particle PIC simulations a variable particle spacing of  $\frac{d}{25}$  after (Monaghan and Kos, 1999) was employed. Once the particle spacing was computed a suitable Eulerian mesh was created for the particular particle density that allowed for an initial seeding of four particles per fluid cell. Figure 1 compares predictions of solitary wave run-up on a vertical wall with the laboratory results from the benchmark experiment of (Camfield and Street, 1969). A number of different ratios of solitary wave amplitude ( $H$ ) to still water depth ( $d$ ) have been simulated. The agreement between the simulated and experimental run-up ( $R$ ) is clearly very good. It is worth noting that for small amplitude waves, i.e. those for which  $Hd^{-1} < 0.2$ , full particle PIC simulates reflection from the wall well with the incident and reflected waves agreeing closely.

### Solitary Wave Interacting With a Semi-Circular Breakwater

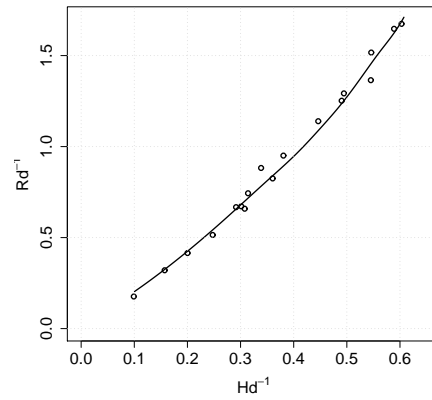
This simulation involves the formation of a backwards breaking wave as a solitary wave passes over a semi-circular breakwater. The test is interesting as it validates the FVM cut-cell treatment of non-grid aligned solid boundaries, in this case the semi-circular breakwater.

Non-dimensionalisation is achieved following (Cooke et al., 1990) with lengths scaled by depths,  $h$ , and time scaled according to  $(hg^{-1})^{\frac{1}{2}}$  where  $g$  is acceleration due to gravity. Four particles were initially seeded per water cell; cells that were initially part air had any particles in the air fraction of the cell removed. The domain extended by 55 dimensionless length units up- and downstream of the breakwater centre. For this test the solitary wave initial conditions were again given by the KDV equation:

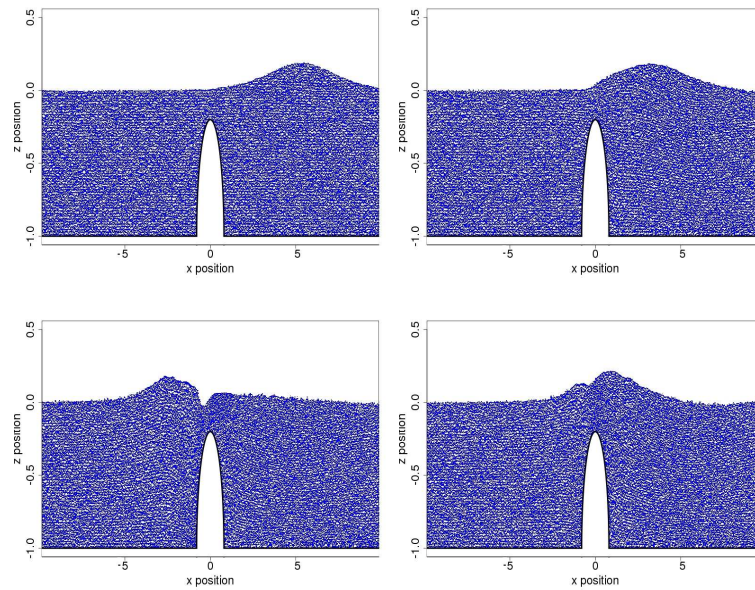
$$\eta = \frac{H}{\cosh^2\left(\frac{3Hx}{4h^3}\right)}, \quad (12)$$

where all variables are dimensionless and  $\eta$  is the free surface,  $x$  is distance parallel to the undisturbed surface,  $H$  is the solitary wave amplitude and  $h$  is the initial still water depth.

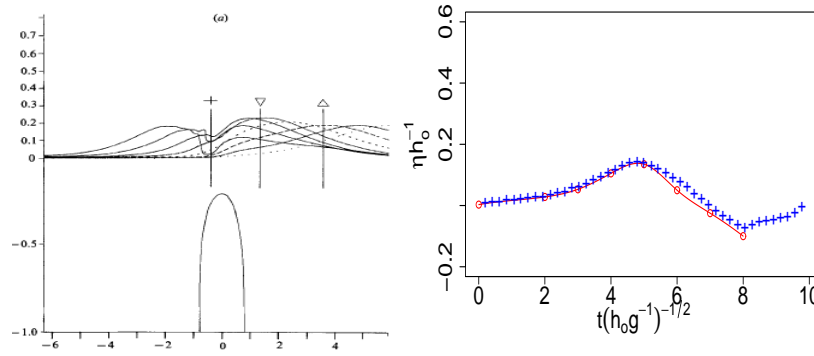
Figure 2 shows a comparison of the free surface with the results of Example 3 of (Cooke et al., 1990). In order to facilitate comparison with the results of (Cooke et al., 1990) there is a vertical exaggeration



**Figure 1:** The run-up  $R$  of a solitary wave on a vertical wall. Comparison between full particle PIC results (line) and the experimental data of Camfield and Street (1969) (circles) for various amplitude to depth ratios. Results are presented non-dimensionalised with respect to the still water depth  $d$ .



**Figure 2:** Particle plot showing free surface for Cooker et al. (1990) Example 3. Clockwise from top left snapshots are at dimensionless times 0, 2, 4, 6. Vertical exaggeration = 8.

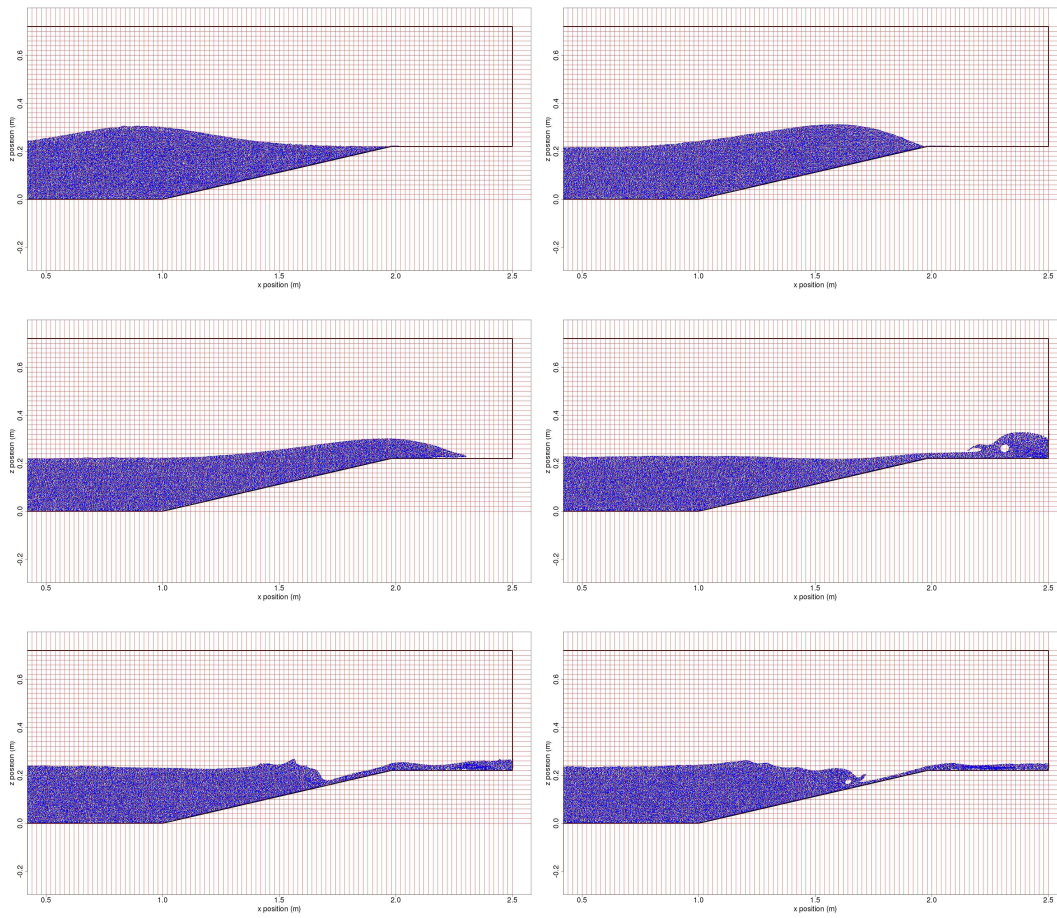


**Figure 3: Left: Numerical results of Cooker et al. 1990 experiment 3. Right: Time-series comparison of water elevation predicted by full-particle PIC model (circles) with the experimental data of Cooker et al. (1990) (crosses). Data shown is for gauge location +.**

of 8 in the plots as there is in their paper. For this case the cylinder has a dimensionless radius of 0.8 and the dimensionless wave height is 0.191. The full-particle PIC results are clearly in good qualitative agreement with the numerical results of (Cooke et al., 1990) which were obtained using a boundary integral method. Figure 3 illustrates a comparison over time of the water elevation at gauge + with the experimental results. The full particle PIC results are in excellent agreement with the experimental results up to time 6. Interestingly, in both the numerical results of (Cooke et al., 1990) and the full-particle PIC results after time 8 the wave broke backwards; however, (Cooke et al., 1990) report that although there was a marked steepening of the water surface in the experiment no backwards breaking was observed.

#### **Solitary Wave Climbing an Idealised Cretan Beach**

(Monaghan and Kos, 1999) studied the run-up and return of a solitary wave on an idealised Cretan beach. The main features of interest in the flow is the climb of the wave over the horizontal section, the formation of a plunger as the wave impacts the wall at the end of the tank and the formation of a backwash bore in the run-down. For the present study the solitary wave was generated using the KDV equation (11) with an amplitude of 8.8cm in still water of depth 21cm. Figure 4 shows snapshots of particle configurations illustrating the pertinent features of the flow. The series of snapshots clearly illustrates the development of the wave as it steepens, runs across the dry section of the beach, interacts with the wall to form a backwards breaking plunger and finally forms a turbulent bore in the backwash. Animations of the results show striking visual agreement to the results presented in (Monaghan and Kos, 1999) with all the main flow features being well captured.



**Figure 4: Particle plot showing free surface for Monaghan and Kos (1999) experiment. Snapshots are shown to illustrate key stages in the evolution of a solitary wave on an idealised Cretan beach.**

## CONCLUSIONS

In order to investigate a wide range of coastal engineering phenomena including wave breaking, wave slamming, overtopping and fluid structure interaction a new code has been developed. The code employs the full-particle PIC technique to solve the Navier Stokes equations subject to an incompressibility constraint, free-surface and free-slip boundary conditions. The code has all the functionality of pure Lagrangian techniques such as SPH whilst retaining the computational efficiency of an Eulerian grid based technique. For a wide range of test cases, three of which are presented here, results from the numerical simulations are in good qualitative agreement with those from previous numerical and experimental studies. Further quantitative validation of the method is required and is the subject of ongoing research. The code appears to be extremely efficient in terms of computational effort required in relation to alternative techniques such as weakly compressible SPH. Initial findings regarding improved computational efficiency, however, have yet to be formally qualified.

In future work, as well as additional validation, the model will be extended to three spatial dimensions with CPU GPU parallelisation. Two-way fluid structure coupling and a sediment phase will also be introduced so that complex scour around structures can be investigated.

## ACKNOWLEDGEMENTS

The author would like to thank both Tim Chesher, group manager Coasts and Estuaries division, and Ian Townend, research director, at HR Wallingford for lending their support to this work.

## References

- Aslam, T. (2004). "A partial differential equation approach to multi-dimensional extrapolation." *J. Comp. Physics*, 193, 349–355.
- Brackbill, J. and Ruppel, H. (1986). "FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flow in two dimensions." *J. Comput Phys.*, 65, 314–343.
- Camfield, F. and Street, R. (1969). "Shoaling of solitary waves on small slopes." *J. Waterways and Harbours Div.*, 95, 1–22.
- Chan, R. K. C. and Street, R. L. (1970). "A computer study of finite-amplitude water waves." *J. Comp. Phys.*, 6, 68–94.
- Chorin, A. J. (1968). "Numerical solution of the Navier–Stokes equations." *Math. Comput.*, 22, 745–762.
- Chorin, A. J. and Marsden, J. E. (1993). *A mathematical introduction to fluid mechanics*. Springer, New York.
- Cooker, M. J., Peregrine, D. H., Vidal, C., and Dold, J. W. (1990). "The interaction between a solitary wave and a submerged semi-circular cylinder." *J. Fluid Mech.*, 125, 51–55.
- Dalrymple, R. and Rogers, B. D. (2006). "Numerical modeling of water waves with the SPH method." *Coastal Eng.*, 53, 141–147.
- Gibou, F., Fedkiw, R. P., Cheng, L. T., and Kang, M. (2002). "A second-order accurate discretisation of the Poisson equation on irregular domains." *J. Comp. Physics*, 176, 205–227.
- Gray, J. P. P. and Monaghan, J. J. (1998). "A study of waves developed by caldera collapse." *Proceedings of 13th Australian Fluid Mechanics Conference*. 279–282.
- Harlow, F. (1964). *Methods in Computational Physics*. Academic Press, London.
- Harlow, F. H. and Welch, J. E. (1965). "Numerical computation of time-dependent viscous incompressible flow of fluid with free surface." *Phy. Fluids*, 8, 3–325.
- Jiang, C., Chen, J., Tang, H., and Z., C. Y. (2011). "Hydrodynamic processes on a beach: Wave breaking, up-rush and backwash." *Comm. Nonlin. Sci and Num. Sim.*, 16, 3126–3139.

- Karambas, T. and Tozer, N. (2003). "Breaking waves in the surf and swash zone." *J. Coast Res.*, 19, 514–528.
- Kelly, D. M. (2012). "PICIN: A particle-in-cell solver for incompressible free surface flows in arbitrary domains." *In Preparation*.
- Monaghan, J. J. and Kos, A. (1999). "Solitary waves on a cretan beach." *J. Waterways and Harbours Div.*, 125, 145–154.
- Ng, Y. T., Min, C., and Gibou, F. (2009). "An efficient fluid–solid coupling algorithm for single–phase flows." *J. Comp. Physics*, 228, 8807–8829.
- Noh, W. F. (1964). *Methods in Computational Physics*. Academic Press, London.
- Osher, J. and Fedkiw, R. P. (2002). *Level set methods and dynamic implicit surfaces*. Springer, New York.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 3rd edition.
- Rogers, B., Dalrymple, R., and Stansby, P. (2008). "SPH modelling of floating bodies in the surf zone." *Proc. 30<sup>th</sup> ASCE Int. Conf. Coast. Eng.* ASCE, 965–972.
- Seiichi, K., Nobe, A., and Oka, Y. (1998). "Numerical analysis of breaking waves using the moving particle semi-implicit method." *Int. J. Numer. Meth. Fluids*, 26, 751–769.
- Tonelli, M. and Petti, M. (2009). "Hybrid finite volume – finite difference scheme for 2DH improved Boussinesq equations." *Coast. Engng.*, 56, 609–620.
- Viecelli, J. A. (1969). "A method for including arbitrary external boundaries in the MAC incompressible fluid computing technique." *J. Comput. Physics*, 4, 543–551.
- Zhu, Y. and Bridson, R. (2005). "Animating sand as a fluid." *Proceedings ACM SIGGRAPH*. SIGGRAPH, 965–972.