

Numerical investigation of sandy beach evolution using an incompressible smoothed particle hydrodynamics method

N. Amanifard¹, S. M. Mahnama¹, S. A. L. Neshaei²
& M. A. Mehrdad²

¹*Mechanical Engineering Department, University of Guilan, Iran*

²*Civil Engineering Department, University of Guilan, Iran*

Abstract

The current work presents an incompressible smoothed particle hydrodynamics (SPH) model to simulate sandy beach evolution. The Navier-Stokes equations are solved in a Lagrangian framework using a three-step fractional method. In the first step, a temporary velocity field is provided according to the relevant body forces. This velocity field is renewed in the second step to include the viscosity effects. A Poisson equation is employed in the third step as an alternative for the equation of state in order to evaluate pressure. The present method is validated by solving a free surface problem and comparing the computational results with the experimental results, as well as numerical data that is evaluated from the standard SPH method. Then, based on an experimental model, the simulation of sandy beach evolution has been investigated by this method. Comparison of the computed results with previous studies that are reported in coastal engineering references implies the capability of the method for the simulation of such complex flows.

Keywords: smoothed particle hydrodynamics (SPH), Lagrangian method, free surface, sandy beach evolution.

1 Introduction

The best understanding of coastal processes requires a blend of analytical study on the nature of variety of shorelines and making many experimental researches. However, due to the complexity of coastal processes, a number of reliable



laboratory and field data are limited. The aim of the current research is to present a numerical method for simulating sandy beach evolution in the coastal zone. For this purpose, in this paper, the smoothed particle hydrodynamics (SPH) method is applied, which uses a purely Lagrangian approach and has been successfully employed in a wide range of applications. The meshless characteristic of SPH makes it unnecessary to use data connectivity for this method, which is not the case for the finite volume and finite element methods. This gives the method a very useful feature when dealing with complex flows that exhibit large deformations and/or free-surfaces.

This method was originally developed by Lucy [1] to solve compressible astrophysical problems. The method was later extended to incompressible flows by Monaghan [2]. Several other researchers have contributed to the method and solved various engineering problems including water wave problems, fluid-structure interaction and interfacial flows [3–5].

In this paper, based on the incompressible SPH method, a three step incompressible SPH algorithm is applied to simulate sandy beach evolution in shorelines. The proposed algorithm is similar to the three step explicit SPH algorithm proposed by Hosseini *et al.* [6] for simulation of incompressible fluid flows. In the first step of this algorithm, the momentum equation is solved in the presence of the body force neglecting all other forces. The calculated temporary velocities are renewed in the second step to include the viscosity effect. A Poisson equation is employed in the third step as an alternative of the equation of state in order to evaluate pressure by projecting the provisional velocity. This Poisson equation considers a trade-off between density and pressure, which is utilized in the third step to impose the incompressibility effect. In order to validate this algorithm, first a wave propagation problem is modeled by this method and the computational results are compared with the experimental data and standard SPH data. Then it is utilized for simulation of an arbitrary pattern of surf zone, which is very similar to the experimental model [7]. The numerical surf zone is much smaller than the experimental one, because of reduced run time and prevention of the numerical divergence.

2 Governing equations

The governing equations for simulating free surface flow in 2-D dimensions are the mass and momentum conservation equations. With regard to fluid particles, they are written in Lagrangian form as:

$$\frac{1}{\rho} \frac{D\rho}{Dt} + \nabla \cdot V = 0 \quad (1)$$

$$\frac{DV}{Dt} = -\frac{1}{\rho} \nabla P + g + \frac{1}{\rho} \nabla \cdot \vec{\tau} \quad (2)$$



where ρ is the fluid density, V is velocity and p represents the pressure of fluid. Eqn. (1) is in the form of a compressible flow. The purpose is that the deviation of fluid densities at the particle can then be used to enforce incompressibility in the correction step of time integration.

3 SPH formulation

The SPH formulations as developed by Monaghan [8] are obtained by interpolating from a set of points that may be disordered. The interpolation is based on the theory of integral interpolants using kernels that approximate a delta function. The interpolants are analytic functions that can be differentiated without the use of grids. If the points are fixed in position, the equations reduce to finite difference equations, with different forms depending on the interpolation kernel. The SPH equations describe the motion of the interpolating points, which can be thought of as particles. Each particle carries a mass m , a velocity V , and other properties, depending on the problem.

Using the above concepts, any quantity of particle i , whether scalar or vector, can be approximated by the direct summation of the relevant quantities of its neighboring particles:

$$\phi_i(r_i) = \sum_j m_j \frac{\phi_j(r_j)}{\rho_j(r_j)} W(|r_i - r_j|, h) \quad (3)$$

where i and j are the reference particle and its neighbor; ϕ_i and ϕ_j are scalar or vector quantities being interpolated; r_i and r_j are the position of particles; W represents the interpolation kernel and h is the smoothing distance. The smoothing kernel used in this literature is a cubic spline, which is most commonly used in hydrodynamic calculations.

4 Three-step incompressible SPH algorithm

In this section, an algorithm is presented to show the sequence of computation of each term in the governing equations. In this paper, a fully explicit three-step algorithm is used. In the first step of this algorithm, the momentum equation is solved in the presence of the body forces neglecting all other forces. As a result, an intermediate velocity is computed as:

$$V^* = V_i + g\Delta t \quad (4)$$

where $g = (g_x, g_y)$ represents the gravity acceleration and $V^* = (u^*, v^*)$ is the first intermediate velocity. Our experience has shown that it is important to impose the body forces in the first step of the solution algorithm, especially in highly viscous fluids. In the second step, the calculated intermediate velocities are employed to compute the divergence of the stress tensor.



In this work, the divergence of the stress tensor in the momentum equation is obtained as:

$$\left(\frac{1}{\rho} \nabla \cdot \vec{\tau} \right)_i = \sum_j m_j \left(\frac{\vec{\tau}_i}{\rho_i^2} + \frac{\vec{\tau}_j}{\rho_j^2} \right) \cdot \nabla_i W(r_{ij}, h) \quad (5)$$

At the end of the second step, the velocity components of each particle are updated according to:

$$V^{**} = V^* + \left(\frac{1}{\rho} \nabla \cdot \vec{\tau} \right) \Delta t \quad (6)$$

At this stage, each particle is moved according to its second intermediate velocity $V^{**} = (u^{**}, v^{**})$.

Finally, the velocity of each particle at the end of time-step will be obtained as:

$$\hat{V}_i = -\Delta t \sum_j m_j \left(\frac{P_i}{\rho_i^{*2}} + \frac{P_j}{\rho_j^2} \right) \nabla_i W_{ij} \quad (7)$$

$$V_{t+\Delta t} = V^{**} + \hat{V} \quad (8)$$

and the final position of particles is calculated using a central difference scheme in time:

$$r_{t+\Delta t} = r_t + \frac{\Delta t}{2} (V_{t+\Delta t} + V_t) \quad (9)$$

This completes the computations required for one time-step. The procedure should be repeated for every other time-step until a desired time is reached.

5 Simulation of sandy beach evolution

In this section, deformation of a sandy beach is simulated using the present numerical model. The pattern of this problem is an experimental work that is performed by Mehrdad and Neshaei [7]. The scale of the numerical beach is smaller than the experimental one and so no quantitative comparison is done between the numerical results and the experimental data. In fact, quantitative investigation of this simulation is not our purpose; we are only going to show that this numerical work can be applicable to model these complex problems.

In this paper, two shapes of beaches are considered: an open beach and a beach that is in the vicinity of a seawall.

A major problem in simulating a sandy beach profile is detecting a suitable model for sand. It is here assumed that sand particles behave as a non-Newtonian fluid. There are many rheological models to simulate these fluids. The Bingham model is the simplest and the most known and it is expressed as:

$$\mu_{eff} = \mu_B + \frac{\tau_B}{\dot{\gamma}} \quad (10)$$

where μ_B , τ_B and $\dot{\gamma}$ are viscosity, Bingham yield stress and shear rate, respectively, to calculate the effective viscosity.

In this model, the fluid behaves like a rigid body at shear rates below the yield stress, while it behaves like a Newtonian fluid at shear rates greater than the yield stress. The general Cross model is another rheological model that effectively simulates non-Newtonian fluids:

$$\frac{\mu_0 - \mu_{eff}}{\mu_{eff} - \mu_\infty} = (K\dot{\gamma})^m \quad (11)$$

where μ_0 , μ_∞ are viscosity at very low and very high shear rates, respectively; K and m are constant parameters.

By combining eqns. (10) and (11) and taking m as unity, the effective viscosity in the Cross model is defined as:

$$\mu_{eff} = \frac{1000 \left(\mu_B + \frac{\mu_B^2}{\tau_B} \dot{\gamma} \right)}{1 + \frac{1000 \mu_B}{\tau_B} \dot{\gamma}} \quad (12)$$

In order to avoid numerical instability, μ_0 is frozen at a fixed high value of $1000 \mu_\infty$. It should be noted that under this condition, the Cross model, unlike the Bingham model, is a continuous variable.

5.1 Simulation of sandy open beach evolution

The numerical open beach is shown in Fig. 1. The overall length of the surf zone is 18m and the maximum depth of water is 0.6m. An inclined beach with a uniform slope of 6%, which is covered by sand particles, is placed at the end of the numerical surf zone. Sinusoidal waves are generated by a wave-maker, which is at the right-hand side of the surf zone. The velocity of the wave-maker particles is computed as:

$$u(t) = \sin(2\pi t) \quad (13)$$

During the simulation the total number of particles is 6471, corresponding to a particle spacing of 0.04m. In the computation a constant time step of 0.00002s is employed. The particle configuration during the evolution of the beach by the effect of waves on sand particles is illustrated in Fig. 2. In this simulation, the following approach of the yield stress and viscosity of sand is adopted by setting $\tau_B = 250 \text{ pa}$ and $\mu_B = 0.1 \text{ pas}$.

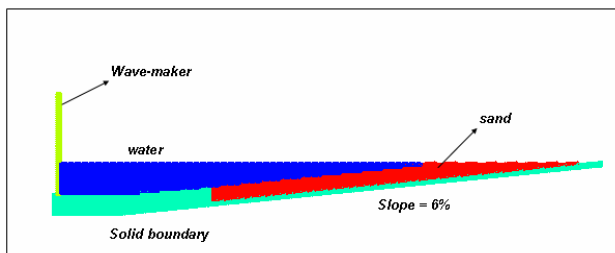


Figure 1: Numerical open beach.

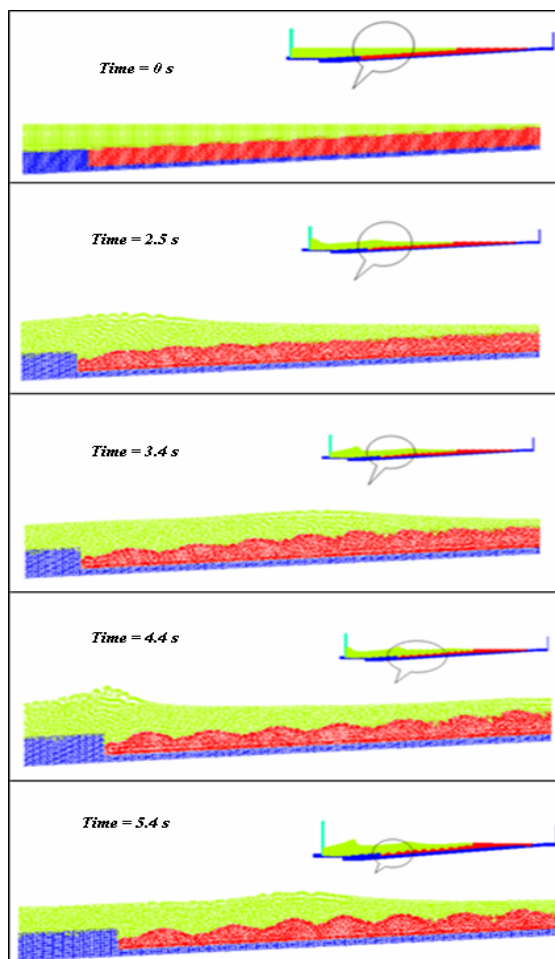


Figure 2: Simulation of sandy open beach at different times.

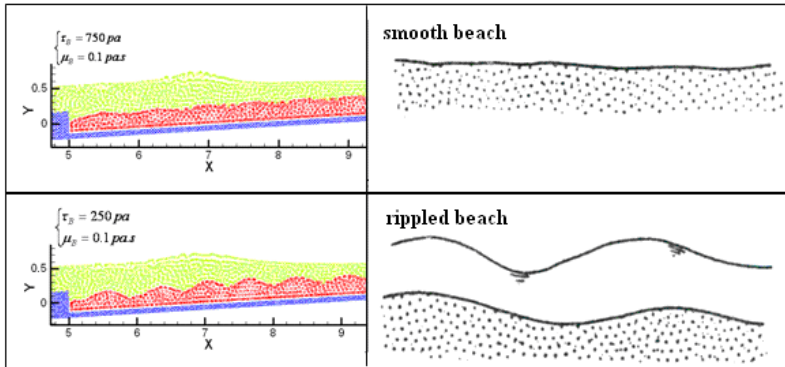


Figure 3: Comparison of beach shapes for different yield stresses of sand.

This shape of the beach is similar to the *rippled beach* that is reported in coastal engineering references. In order to obtain the other shapes of beach, we change τ_B as a changeable value. For this purpose, by setting $\tau_B = 750 \text{ pa}$, the expected resistance of sand particles against the deformation increases, thus a *smooth beach* is formed. Comparison of these two shapes of beach is shown in Fig. 3 (where X is the distance from the shoreline and Y is the water depth).

5.2 Simulation of sandy beach evolution in the vicinity of a seawall

In order to show the effect of a wall and a reflective wave on beach shape, a vertical wall is placed on the right-hand side of the numerical surf zone, which is explained in the previous section. The length of the surf zone is shorter than the previous model because of the formation of a reflective wave. The velocity of the wave-maker is adopted as eqn. (13). The results of this simulation are illustrated in Fig. 4. It can be seen that the reflective waves cause the water to penetrate the lower part of the wall and thus sediments (sands) are pushed back. This problem is very important in coastal engineering in order to design safe structures, which are responsible for resisting violent waves and protecting coastal zones.

6 Conclusion

In this paper, an incompressible SPH method is employed for numerical simulation of some examples of free surface flows. The Navier-Stokes equations are solved in a Lagrangian framework using a three-step fractional method. The main advantages of the proposed algorithm is simulation of free surface flows with large deformation naturally (without imposing any special condition) and more accurately than the previous Standard SPH method. Moreover, the computational time spent to reach the final results is not very long.

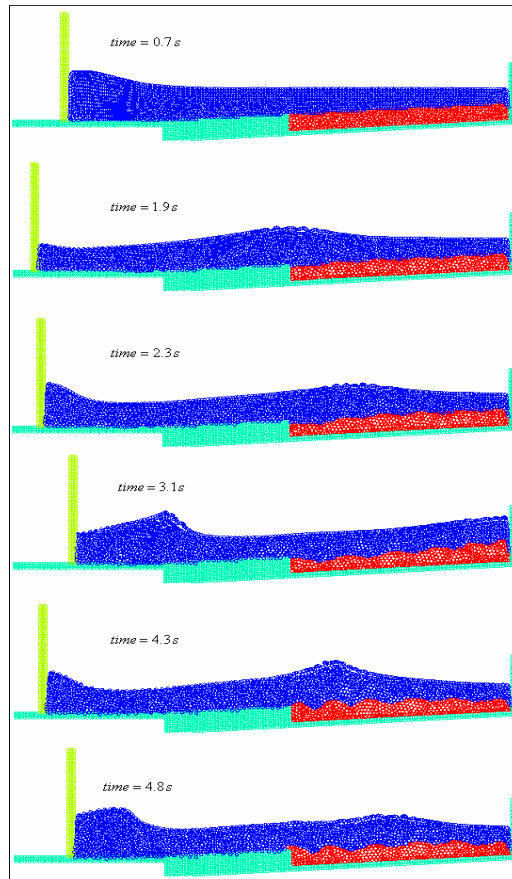


Figure 4: Simulation of a sandy beach in the vicinity of a seawall.

According to existing information, the SPH method has not already been applied to the simulation of sandy beach evolution, so the current SPH method is employed for the simulation of sandy beach deformation in various conditions. Following the previous studies about beach shapes that are reported in the coastal engineering references, the shape of computational beaches is very similar to the real ones qualitatively. This result demonstrates the capability of the method for the simulation of such flows.

In summary, in this paper, following an experimental model, the simulation of sandy beach evolution by an incompressible SPH method has been investigated. Comparison of the computed results with previous studies that are reported in coastal engineering references implies the capability of the method for simulation of such conditions. It is believed that the present work can be regarded as a basis for future researches into this problem and other similar applications.

References

- [1] Lucy, L.B., 1977. A numerical approach to the testing of the fission hypothesis. *Astron. J.* 82, 1013–1024.
- [2] Monaghan, J.J., 1994. Simulating free surface flows with SPH, *Computational Physics* 110, 399–406.
- [3] Dalrymple, R.A., Rogers, B.D., 2006. Numerical modeling of water waves with the SPH method, *Coastal engineering* 53, 141–147.
- [4] Farahani, M.H., Amanifard, N., Pouryoussefi, Gh., 2008(b). Numerical simulation of a pulsatory flow moving through flexible walls using smoothed particle hydrodynamics. *Proceeding of 2008 International Conference of Mechanical Engineering (ICME 2008), 2008 World Congress of Engineering*, London, UK, pp 1337-1341, July, 2008.
- [5] Hosseini, S.M., Amanifard, N., 2007. Presenting a modified SPH algorithm for numerical studies of fluid-structure interaction problems, *IJE Trans B: Applications* 20, 167-178.
- [6] Hosseini, S.M., Manzari, M.T., Hannani, S.K., 2007. A fully explicit three step SPH algorithm for simulation of non-Newtonian fluid flow. *Numerical Methods for Heat and Fluid Flow* 17, 715– 735.
- [7] Mehrdad, M.A., Neshaei, M.A.L., 2004. Hydrodynamics of the surf zone in the vicinity of a partially reflective seawall. *Civil Engineering* 2, No. 3.
- [8] Monaghan, J.J., 1992. Smoothed particle hydrodynamics. *Annu. Rev. Astron. Astrophys* 30, 543–574.

