

# Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface

Songdong Shao<sup>a</sup>, Edmond Y.M. Lo<sup>b,\*</sup>

<sup>a</sup> *Environmental Engineering Research Center, School of Civil and Environmental Engineering, Nanyang Technological University, Singapore 639798, Singapore*

<sup>b</sup> *Division of Environmental and Water Resources Engineering, School of Civil and Environmental Engineering, Nanyang Technological University, Singapore 639798, Singapore*

Received 19 February 2002; accepted 20 January 2003

## Abstract

An incompressible smoothed particle hydrodynamics (SPH) method is presented to simulate Newtonian and non-Newtonian flows with free surfaces. The basic equations solved are the incompressible mass conservation and Navier–Stokes equations. The method uses prediction–correction fractional steps with the temporal velocity field integrated forward in time without enforcing incompressibility in the prediction step. The resulting deviation of particle density is then implicitly projected onto a divergence-free space to satisfy incompressibility through a pressure Poisson equation derived from an approximate pressure projection. Various SPH formulations are employed in the discretization of the relevant gradient, divergence and Laplacian terms. Free surfaces are identified by the particles whose density is below a set point. Wall boundaries are represented by particles whose positions are fixed. The SPH formulation is also extended to non-Newtonian flows and demonstrated using the Cross rheological model. The incompressible SPH method is tested by typical 2-D dam-break problems in which both water and fluid mud are considered. The computations are in good agreement with available experimental data. The different flow features between Newtonian and non-Newtonian flows after the dam-break are discussed.

© 2003 Elsevier Science Ltd. All rights reserved.

**Keywords:** SPH; Pressure projection; Free surface; Mudflow; Dam-break

## 1. Introduction

Free surface hydrodynamic flows are of significant industrial and environmental importance but are difficult to simulate because the surface boundary conditions are specified on an arbitrarily moving surface. The MAC [6] and VOF [7] methods are two of the most flexible and robust approaches for treating such flows. The former uses marker particles to define the free surface while the latter solves a transport equation for the volume fraction of the fluid. They have been successfully applied to a wide variety of flow problems involving free surfaces such as dam-break [18] and wave breaking in the surf zone [12]. However, in both the MAC and VOF methods, the Navier–Stokes equations are solved on a fixed Eulerian grid. Problems of numerical diffusion arise due to advection terms in the N–S equations. The numerical diffusion becomes severe when the deformation of the free surface is very large, during which the treatment procedures of the surface cells for capturing the sharpness of the surface becomes complicated.

The smoothed particle hydrodynamics (SPH) is a pure Lagrangian method originally developed for astrophysical computations [13,16] and has later been extended to model a wide range of problems. Unlike the earlier particle-in-cell method, SPH does not use a grid to calculate spatial derivatives and thus the back and forth interpolations between grid and particle information are avoided. The basic concept of SPH is that a particle is fundamental in the Lagrangian description and the motion of a continuum can be represented with very high accuracy by simulating the advection of an increasing large number of such particles. Through the use of integral interpolants, the dependent field variables are expressed by integrals which are approximated by

\* Corresponding author.

E-mail addresses: [csdshao@ntu.edu.sg](mailto:csdshao@ntu.edu.sg) (S. Shao), [cymlo@ntu.edu.sg](mailto:cymlo@ntu.edu.sg) (E.Y.M. Lo).

### Nomenclature

$a$	dam width	$\vec{A}$	shear strain tensor
$E_d(t)$	time-dependent normalized density error	$\Delta t$	time increment
$E_{l_0}$	particle spacing-dependent numerical error	$\Delta \vec{u}$	changed velocity
$\vec{g}$	gravitational acceleration	$\eta$	small number to avoid singularity ( $= 0.1h$ )
$h$	kernel smoothing distance	$\mu$	constant viscosity of water
$H$	dam height	$\mu_0$ ( $\mu_\infty$ )	viscosity at very low (high) shear rates
$K(m)$	constant parameter in Cross model	$\mu_B$	Bingham viscosity
$l_0$	initial particle spacing	$\mu_{\text{eff}}$	effective viscosity
$m$	particle mass	$\rho$	fluid density
$n$	order of convergence	$\rho_0$	initial density
$N(N')$	number of total (inner) fluid particles	$\vec{\tau}$	shear stress tensor
$P$	pressure	$\tau_B$	Bingham yield stress
$q$	non-dimensional distance between particles ( $= r/h$ )	$\phi$	quantity interpolated/interpolating
$r$	distance between particles	<i>Subscripts and symbols</i>	
$\vec{r}$	position vector	$a$	reference particle
$S_0$	channel bed slope	$b$	neighboring particle
$T$	non-dimensional time ( $= t\sqrt{g/H}$ )	$ab$	values between particle $a$ and $b$
$\vec{u}$	velocity vector	$s$	surface particle
$V_{\text{max}}$	maximum velocity	$t$	time
$W$	interpolation kernel	$x(y)$	coordinates direction
$X$	non-dimensional leading edge from dam site ( $= x/H$ )	$*$	intermediate step value
$\alpha$	coefficient for numerical stability ( $= 0.1$ )	$**$	corrective value
$\dot{\gamma}$	comprehensive shear rate	$\nabla$	gradient operator
$\dot{\gamma}_c$	shear rate at yield interface	$\nabla^2$	Laplacian operator
		$\sum$	summation operator

summation interpolants over neighboring particles. Similarly, spatial derivatives at each particle location are evaluated by summation interpolants. The derivatives are evaluated based on the derivatives of the kernel function, which are obtained analytically.

In the original simulation of incompressible flows by SPH, incompressibility is realized through a stiff equation of state so that the fluid is considered slightly compressible. In this case, a large sound speed has to be assumed so that the corresponding density fluctuation can be kept small, typically to about 1%. Low Reynolds number flows through porous media such as groundwater flow [20] and moderate Reynolds number flows such as bore and wave propagation [17] have been successfully simulated by this weakly compressible SPH formulation. However, the artificial compressibility can cause problems with sound wave reflection at boundaries and the high sound speed leads to a stringent CFL time step constraint. However, the SPH scheme has proved to be very effective in tracking the free surfaces.

A similar approach is the moving particle semi-implicit (MPS) method, which also models the fluid motion using Lagrangian particles. Differential operators such as the gradient and Laplacian in the governing equations, are transformed into equivalent particle in-

teractions using a weight function. A semi-implicit algorithm is employed to implement incompressibility [4,11]. Large deformation of free surfaces can be easily and accurately tracked by the Lagrangian motion of particles in both the SPH and MPS methods.

This paper presents a strictly incompressible SPH model to simulate free surface flows. The basic approach is similar to the SPH projection method by Cummins and Rudman [3] for flows without a free surface. It employs a strict incompressible SPH formulation, and thus the CFL condition is based on the fluid velocity field rather than the speed of sound. The trade-off is that in the incompressible SPH method, the pressure is not an explicit thermodynamic variable obtained through an equation of the state, but obtained by solving a pressure Poisson equation. The advantage of the incompressible SPH presented here lies in its ease and efficiency to track the free surface using similar procedures employed in the MPS method. The versatility of the SPH method is further demonstrated by extending the formulation to rheological fluids with non-constant viscosity.

The SPH method is tested on a two-dimensional dam-break problem for two types of fluid, water and fluid mud which is modeled using the Cross rheological model. The simulations are in good agreement with

experimental data reported in the literature. The SPH calculations also reveal several distinctive flow features between these two types of flows, mainly due to their different viscosity and flow mechanisms. The findings may be of engineering interest in view of the mitigation of natural hazards in flood and debris flows.

## 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

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

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

where  $\rho$  = fluid particle density;  $t$  = time;  $\vec{u}$  = particle velocity;  $P$  = pressure;  $\vec{g}$  = gravitational acceleration and  $\vec{\tau}$  = shear stress tensor. The above equations are written in full form and applicable to both Newtonian and non-Newtonian flows.

Eq. (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 a correction step of the time integration.

The prediction–correction computation of the incompressible SPH method is composed of two steps similar to the two step projection method of Chorin [2] for solving the Navier–Stokes equation. The first prediction step is an explicit integration in time without enforcing incompressibility. Only the stress tensor and gravitational terms are considered in the Navier–Stokes equation and an intermediate temporal particle velocity and position are obtained as

$$\Delta \vec{u}_* = \left( \vec{g} + \frac{1}{\rho} \nabla \cdot \vec{\tau} \right) \Delta t \quad (3)$$

$$\vec{u}_* = \vec{u}_t + \Delta \vec{u}_* \quad (4)$$

$$\vec{r}_* = \vec{r}_t + \vec{u}_* \Delta t \quad (5)$$

Here  $\vec{u}_t$  and  $\vec{r}_t$  = particle velocity and position at time  $t$ ;  $\vec{u}_*$  and  $\vec{r}_*$  = intermediate particle velocity and position;  $\Delta \vec{u}_*$  = changed particle velocity during the prediction process and  $\Delta t$  = time increment.

In the first step, incompressibility is not satisfied. This is demonstrated by that the fluid density  $\rho_*$  (calculated as shown later using Eq. (12)) based on the intermediate particle positions  $\vec{r}_*$  deviates from that of the constant value  $\rho_0$ . Thus a second corrective step is applied to adjust fluid densities at the particles to initial constant values prior to the time step. In the correction, the

pressure term is used to update the particle velocity obtained from the intermediate step. The relevant steps are

$$\Delta \vec{u}_{**} = -\frac{1}{\rho_*} \nabla P_{t+1} \Delta t \quad (6)$$

$$\vec{u}_{t+1} = \vec{u}_* + \Delta \vec{u}_{**} \quad (7)$$

where  $\Delta \vec{u}_{**}$  = changed particle velocity during the correction process;  $\rho_*$  = intermediate particle density between the prediction and correction; and  $P_{t+1}$  and  $\vec{u}_{t+1}$  = particle pressure and velocity of time  $t + 1$ .

Finally, the positions of particle are centered in time

$$\vec{r}_{t+1} = \vec{r}_t + \frac{(\vec{u}_t + \vec{u}_{t+1})}{2} \Delta t \quad (8)$$

where  $\vec{r}_t$  and  $\vec{r}_{t+1}$  = positions of particle in time  $t$  and  $t + 1$ .

The equation used to obtain the pressure for enforcing incompressibility, is the mass conservation Eq. (1). It is represented in the discrete form as

$$\frac{1}{\rho_0} \frac{\rho_0 - \rho_*}{\Delta t} + \nabla \cdot (\Delta \vec{u}_{**}) = 0 \quad (9)$$

where  $\rho_0$  and  $\rho_*$  = final (and initial) and temporal fluid density of each of the particles.

Combining Eqs. (6) and (9), the pressure Poisson equation is obtained as follows

$$\nabla \cdot \left( \frac{1}{\rho_*} \nabla P_{t+1} \right) = \frac{\rho_0 - \rho_*}{\rho_0 \Delta t^2} \quad (10)$$

The above equation is analogous to the formulation in the MPS method [4,11] in that the source term of the Poisson equation is the variation of particle densities, while it is usually the divergence of velocity vector in finite difference methods. In the MPS method, the particle number density appears in the source term, while Eq. (10) employs the fluid density. By employing the relevant SPH formulation for the Laplacian operator in the following section, Eq. (10) is discretized into simultaneous linear equations whose matrix is symmetric and positive definite. These equations are solved efficiently by using a preconditioned conjugate gradient method.

## 3. SPH formulations

The SPH formulations as developed by Monaghan [16] 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  $\vec{u}$ , and other properties, depending on the problem.

### 3.1. Interpolation

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

$$\phi_a(\vec{r}_a) = \sum_b m_b \frac{\phi_b(\vec{r}_b)}{\rho_b(\vec{r}_b)} W(|\vec{r}_a - \vec{r}_b|, h) \quad (11)$$

where  $a$  and  $b$  = reference particle and its neighbors;  $\phi_a$  and  $\phi_b$  = scalar or vector quantity being interpolated and interpolating;  $\vec{r}_a$  and  $\vec{r}_b$  = position of particles;  $W$  = interpolation kernel and  $h$  = smoothing distance. Thus the fluid density at particle  $a$ ,  $\rho_a$  is evaluated by

$$\rho_a = \sum_b m_b W(|\vec{r}_a - \vec{r}_b|, h) \quad (12)$$

### 3.2. Kernel

The use of different kernels is the SPH analogue of using different difference schemes in finite difference methods. While different equations can have different kernels, usually the same kernel is used throughout all formulations in one model. By balancing the computational accuracy and efficiency, the following kernel based on the spline function and normalized in 2-D is adopted [16]

$$\begin{aligned} W(r, h) &= \frac{10}{7\pi h^2} \left( 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 \right) \quad q < 1 \\ W(r, h) &= \frac{10}{28\pi h^2} (2 - q)^3 \quad 1 < q < 2 \\ W(r, h) &= 0 \quad q > 2 \end{aligned} \quad (13)$$

where  $q = r/h$  and  $r$  = separation distance between the particles. The smoothing distance or kernel range  $h$  determines the degree which a particle interacts with neighboring particles.

This kernel has the advantage of possessing compact support, the second derivative being continuous and the dominant error term in the integral interpolant being the order of  $h^2$ . The continuity of the second derivative means that the kernel is not too sensitive to particle disorder and the errors in approximating the integral interpolants by summation interpolants are small provided the particle disorder is not too large.

### 3.3. Gradient/divergence

The formulation of the gradient term in the Navier–Stokes equation has different forms depending on the derivation used [16]. The following symmetric form is widely used since it conserves linear and angular momentum exactly. For example, the gradient of the pressure is expressed as

$$\left( \frac{1}{\rho} \nabla P \right)_a = \sum_b m_b \left( \frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \nabla_a W \quad (14)$$

where the summation is over all particles other than particle  $a$  and  $\nabla_a W_{ab}$  = gradient of the kernel taken with respect to the positions of particle  $a$ . In practice only near neighbors contribute, because the kernel has a finite range.

Similarly, the divergence of a vector  $\vec{u}$  at particle  $a$  can be formulated symmetrically by

$$\nabla \cdot \vec{u}_a = \rho_a \sum_b m_b \left( \frac{\vec{u}_a}{\rho_a^2} + \frac{\vec{u}_b}{\rho_b^2} \right) \cdot \nabla_a W_{ab} \quad (15)$$

### 3.4. Laplacian

The Laplacian in the pressure Poisson equation (10) can be formulated in a standard way by SPH definitions, i.e., to formulate a Laplacian operator composed of the divergence and gradient operators of SPH formulations. However, it has been found that the resulting second derivative of the kernel is very sensitive to particle disorder and will easily lead to pressure instability and decoupling in the computation due to the co-location of the velocity and pressure. Thus the Laplacian is formulated as a hybrid of a standard SPH first derivative with a finite difference approximation for the first derivative. The same approach is used for projecting an estimate of the velocity field onto a divergence-free space [3]. Thus the following representation for the Laplacian is adopted

$$\nabla \cdot \left( \frac{1}{\rho} \nabla P \right)_a = \sum_b m_b \frac{8}{(\rho_a + \rho_b)^2} \frac{P_{ab} \vec{r}_{ab} \cdot \nabla_a W_{ab}}{|\vec{r}_{ab}|^2 + \eta^2} \quad (16)$$

where  $P_{ab} = P_a - P_b$ ,  $\vec{r}_{ab} = \vec{r}_a - \vec{r}_b$  and  $\eta$  is a small number introduced to keep the denominator non-zero during the intermediate computational steps and usually equal to  $0.1h$ . This form is very similar to the original formulation by Cummins and Rudman [3] but has the advantage of being in symmetric form. Thus after discretization, the corresponding coefficient matrix of the linear equations is symmetric and positive definite, and can be more efficiently solved by available solvers.

### 3.5. Viscosity

The viscosity term in Eq. (2) is also formulated as a mixture of a standard SPH first derivative with a finite difference approximation for the first derivative. By applying the SPH definition of divergence in Eq. (15), the viscosity term is represented by

$$\left( \frac{1}{\rho} \nabla \cdot \vec{\tau} \right)_a = \sum_b m_b \left( \frac{\vec{\tau}_a}{\rho_a^2} + \frac{\vec{\tau}_b}{\rho_b^2} \right) \cdot \nabla_a W_{ab} \quad (17)$$

where the stress tensor  $\vec{\tau}$  is related to the strain tensor  $\Delta$  in summation form as

$$\tau_{ij} = \tau_{ji} = \mu_{\text{eff}} \Delta_{ij} \quad (18)$$

where  $\mu_{\text{eff}}$  = effective viscosity which is variable in non-Newtonian flow and  $\Delta_{ij}$  is defined by

$$\Delta_{ij} = \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \quad (19)$$

Here it should be noted that while applying the above equation to SPH particles, the full derivative between two particles is first obtained using finite difference before decomposing it into  $x$  and  $y$  directions. Thus

$$\begin{aligned} \left( \frac{\partial u_i}{\partial x_j} \right)_a &= \left( \frac{\partial u_i}{\partial r_{ab}} \right) \left( \frac{\partial r_{ab}}{\partial x_j} \right)^{-1} \\ &= \frac{(u_i)_a - (u_i)_b}{r_{ab}} \frac{(x_j)_a - (x_j)_b}{r_{ab}} \end{aligned} \quad (20)$$

With regard to Newtonian fluids such as water, the viscosity coefficient  $\mu_{\text{eff}}$  has a constant value  $\mu$ . Combining Eqs. (17)–(20) and applying incompressibility, the SPH formulation of viscosity term simplifies to

$$\left( \frac{\mu}{\rho} \nabla^2 \vec{u} \right)_a = \sum_b \frac{4m_b(\mu_a + \mu_b) \vec{r}_{ab} \cdot \nabla_a W_{ab}}{(\rho_a + \rho_b)^2 (|\vec{r}_{ab}|^2 + \eta^2)} (\vec{u}_a - \vec{u}_b) \quad (21)$$

## 4. Treatment of wall boundary conditions and free surface

### 4.1. Wall boundaries

Generally, the solid walls are also simulated by particles, which balance the pressure of inner fluid particles and prevent them from penetrating the wall. Special provisions must be made at the edges of these wall particles in order for the motion of the adjacent fluid to be modeled correctly. The wall boundary conditions can be modeled either by fixed particles exerting a repulsive force on inner fluid particles through a force function [17], or by image particles that mirror the physical properties of inner fluid particles, so that boundary conditions are strictly satisfied along the solid walls (e.g. [3]).

Here we follow the treatment used by Koshizuka et al. [11] to model the wall boundaries by fixed wall particles, which are spaced according to the initial configuration. The Poisson equation (10) is solved on these wall particles to repulse the inner flow particles accumulating in the vicinity of the wall. In this sense, the wall particles are dependent on the inner flows. For example, the pressure of wall particles increases when the particle density in the vicinity of the wall increases and thus the inner fluid particles are repelled from the wall, and vice versa. In addition, several lines of dummy particles are also placed outside of solid walls in order to keep the fluid density at the wall particles to be consistent with that of the inner fluid. The thickness of dummy particles depends on the kernel range  $h$ . If we employ a kernel range  $h = 2l_0$ , where  $l_0$  is the initial particle spacing, four lines of dummy particles at spacing  $l_0$  between the lines should be used. The velocities of both wall and dummy particles are set to be zero to represent no-slip boundary conditions.

This treatment of wall boundaries is analogous to that employed in the MPS method [4,11] but differs in that the MPS method does not calculate the pressure gradient between dummy particles and inner fluid particles. However, it is required in the SPH method since most of the SPH formulations are represented in symmetric form. Specifically, the pressure of dummy particles is used in the Poisson equation (10) and included in the pressure-corrected velocity equation (6). Homogeneous Neumann conditions are applied, i.e., the pressure of a dummy particle is set to that of a wall particle in the normal direction of the solid walls. The effectiveness of this wall treatment is reflected in the following dam-break simulations, especially near the leading edge of the dam-break flow, when the fluid collapses and impacts the lower boundary as well as subsequent dry–wet changes of the bottom with the leading edge.

### 4.2. Free surfaces

Free surfaces are identified by particle densities. Since no particle exists in the outer region of the free surface, the particle density will drop on the surface. A particle is regarded as a surface particle if its density fluctuation is over 1% below that of the inner fluid. A Dirichlet boundary condition of zero pressure is given to this particle.

Special treatment is also needed when applying the pressure-corrected velocity equation (6) on surface particles to keep a symmetric particle configuration. Let us assume that  $s$  is a surface particle with zero pressure and  $i$  is an inner fluid particle with pressure  $P_i$ . In calculating the pressure gradient between these two particles, a mirror particle  $i'$  is placed on the other side of free surface. Since the deformation of free surfaces is sometimes

very large, an approximate approach of mirroring is adopted here as follows. The position of mirror particle  $i'$  is just the direct reflection of inner particle  $i$  through the surface particle  $s$ . This mirror particle assumes a pressure of  $-P_i$ . In this way, the zero pressure conditions on the free surface are satisfied. Note that the mirror particles are not used for computation of shear stresses in Eq. (17). Thus only particles below the surface can exert shear forces on surface particles and the surface boundary condition of zero shear force is approximately satisfied.

## 5. Numerical convergence and particle link list

### 5.1. Convergence analysis

Because the individual fluid particles are discrete points and cannot deform as the real fluid does, the number of particles employed in the computation must be sufficiently large to give numerical convergence and a realistic flow simulation. Convergence is achieved by repeatedly doubling the number of fluid particles into the computational domain until the numerical solutions are essentially unchanged. Meanwhile, the time step  $\Delta t$  should also be halved to be in consistency with the doubling of particle numbers.

The time step is controlled in the computation to satisfy the following Courant condition:

$$\Delta t \leq 0.1 \frac{l_0}{V_{\max}} \quad (22)$$

where  $V_{\max}$  = anticipated maximum particle velocity in the computation. The factor 0.1 ensures that the particle moves only a fraction (in this case 0.1) of the particle spacing per time step. In addition, the constraint of the time step due to viscous diffusion must also be satisfied by

$$\Delta t \leq \alpha \frac{l_0^2}{\mu_{\text{eff}}/\rho} \quad (23)$$

where  $\alpha$  = coefficient depending on the choice of the kernel type and particle arrangement.  $\alpha$  is usually the order of 0.1, which is determined from SPH numerical trials.

Eq. (23) is the usual stability condition for an explicit finite difference method simulating diffusion. For simulations having high resolution or large viscosity, it is more stringent than Eq. (22).

### 5.2. Particle link list

In the incompressible SPH method, each fluid particle needs a list of neighboring particles within a distance of kernel range ( $2h$  in this paper). The whole list, which is updated in each time step, requires the scale of  $N^2$  operations for the calculation of distances between all pairs

of particles, where  $N$  is the number of particles. This list generation can dominate the computation time in large problems involving many particles.

Various optimization procedures of the list generation algorithms to decrease the count of operations have been proposed, such as those employed by the MPS method [11].

In this paper, linked list data structures [15] are employed to identify neighboring particles within a distance of  $2h$ . The computational domain is divided into square cells having side lengths of  $2h$  and a list of particles belonging to each cell is created. A particle located within a given cell then considers interactions only with particles in neighboring cells. In this way, computational efficiency is significantly improved. The typical run times for the longer simulations discussed below require several (5–6) h on a 550 MHz Pentium PC having 512MB of memory. The scaling with particle number is slightly less than  $N^2$  per time step due to the matrix solver and link list calculation.

## 6. Rheological model

Despite the important role non-Newtonian fluids play in a large number of engineering applications, it is only in the last few decades that the study of the underlying fluid dynamics has experienced a substantial growth. It is commonly accepted that the non-Newtonian flow can be modeled effectively by the general Cross model [1]

$$\frac{\mu_0 - \mu_{\text{eff}}}{\mu_{\text{eff}} - \mu_{\infty}} = (K\dot{\gamma})^m \quad (24)$$

where  $\mu_0$  and  $\mu_{\infty}$  = viscosity at very low and very high shear rates, respectively; and  $K$  and  $m$  = constant parameters.

However, the Cross model uses four rheological parameters. For convenience we adopt an approach to define these parameters in terms of the more common Bingham fluid parameters as follows.

Under the Bingham model, the fluid at rest is capable of resisting any shear stress below the yield. When the yield stress is exceeded, the flow structure changes and the material behaves like a Newtonian fluid driven by the excess of the shear stress beyond the yield stress. When the shear stress falls below the yield stress, the fluid structure changes again and there is either plug flow or no flow at all. Accordingly for numerical computation, the effective viscosity of a Bingham fluid can be represented as

$$\mu_{\text{eff}} = \mu_B + \frac{\tau_B}{\dot{\gamma}} \quad (25)$$

where  $\tau_B$  and  $\mu_B$  = Bingham yield stress and viscosity, respectively.  $\dot{\gamma}$  is the shear rate and is defined in general by the second invariant of the deformation strain  $\Delta_{ij}$  as

$$\dot{\gamma} = \sqrt{\frac{1}{2} \sum_i \sum_j A_{ij} A_{ji}} \quad (26)$$

In 2-D, the shear rate  $\dot{\gamma}$  simplifies to

$$\dot{\gamma} = \sqrt{2 \left( \frac{\partial u}{\partial x} \right)^2 + 2 \left( \frac{\partial v}{\partial y} \right)^2 + \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2} \quad (27)$$

We take  $m$  as unity in Eq. (24) so that the effective viscosity in a Cross model can be represented as

$$\mu_{\text{eff}} = \frac{\mu_0 + K\mu_\infty\dot{\gamma}}{1 + K\dot{\gamma}} \quad (28)$$

By comparing the above equation with the Bingham model (25) under the condition of  $K\dot{\gamma} \gg 1$ , two other parameters in the Cross model are defined as

$$K = \frac{\mu_0}{\tau_B} \quad (29)$$

and

$$\mu_\infty = \mu_B \quad (30)$$

Thus the remaining unknown parameter in the Cross model is  $\mu_0$  the viscosity at low shear rate  $\dot{\gamma}$ . In the Bingham model (25), the effective viscosity  $\mu_{\text{eff}}$  attains an infinite value when  $\dot{\gamma}$  tends to be infinitesimal such as in the core region of a Bingham flow. In numerical computation, such large values of  $\mu_{\text{eff}}$  will create numerical divergence. Thus in numerical computation using the Bingham model, the effective viscosity  $\mu_{\text{eff}}$  is usually frozen at a fixed high value to ensure convergence. Using this approach, Hammad and Vradis [5] concluded that the accuracy of their numerical solution is insensitive to the choice of this high value, once it is  $10^3$  times larger than  $\mu_B$ . We adopt this approach by setting  $\mu_0 = 10^3\mu_\infty$  for the Cross model.

Notice that the advantage of the Cross model (28) over the Bingham model (25) is that the effective viscosity is a continuous variable and numerical instability is avoided.

## 7. Model application

### 7.1. Broken-dam analysis for a Newtonian fluid

In this example, a rectangular column of water, a Newtonian fluid, in hydrostatic equilibrium is confined between two vertical walls. The water column is  $a = 0.1$  m wide and  $H = 0.2$  m high and gravity acts vertically downwards. At the beginning of the computation, the right wall (dam) is instantaneously removed and the water is allowed to flow out along a dry horizontal bed. Experimental data [14] are available for comparison with the SPH calculation as well as previous numerical calculations using MAC [18] and VOF methods [7].

During the simulation the total number of fluid particles is  $N = 20 \times 40$ , corresponding to a particle spacing of 0.005 m in the initial configuration. The fluid particles are initially arranged in a regular, equally-spaced grid, with boundary particles added to form the left-hand wall and bed. In the computation a constant time step of 0.0005 s is employed. In order to account for the influence of the turbulent nature of the broken-dam flow, the viscosity in simulation is taken to be 1000 times larger than that of the constant laminar value, i.e.,  $\nu = 10^{-3}$  m<sup>2</sup>/s.

The particle configurations for the flow at different times  $t = 0.05, 0.1, 0.15$  and  $0.18$  s, are shown in Fig. 1(a)–(d). Similar computational results by the VOF method [7] are given for comparison in Fig. 1(b) and (c), while corresponding VOF data is unavailable for Fig. 1(a) and (d). The capability of the incompressible SPH method to capture the shock is obvious. The simulated flow patterns are very similar to those found in the previous computations, e.g., Harlow and Welch [6] using MAC, Hirt and Nichols [7] using VOF, and Koshizuka et al [10] using MPS. Fig. 2 indicates the comparison of the time variation for the position of the leading edge from the dam site with experiments from Martin and Moyce [14], as well as with MAC computations from Pan et al. [18] and VOF computations from Hirt and Nichols [7]. The relations between the normalized time  $T = t\sqrt{g/H}$  and leading edge  $X = x/H$  (note that the origin of  $X$  is defined at the dam site) are in good agreement.

The velocity fields and pressure isobars of the dam-break flow at time  $t = 0.05$  and  $0.18$  s, are given in Fig. 3(a) and (b) and Fig. 4(a) and (b), respectively. It is seen that the instantaneous removal of the dam causes an abrupt adjustment of the pressure field since the fluid is incompressible with infinite sound speed. This adjustment of the pressure causes the resulting unsteady motion, eventually producing a propagating bore. Fig. 1(a) and (b) clearly show that a jet is formed at the lower right corner on the free surface, which is similar to that observed experimentally by Stansby et al. [19]. Meanwhile, the velocity changes rapidly and Fig. 3(a) shows that the velocity on the free surface assumes larger values while the fluid at large distances from the surface remains almost stationary. The pressure distribution inside the fluid deviates significantly from hydrostatic and the maximum pressure is about half of the initial hydrostatic value. The pressure isobars are sparse at the early time  $t = 0.05$  s after dam-break, as shown in Fig. 4(a).

At larger time, the initial large acceleration decreases and the velocity gradually becomes stable, until a bore develops downstream as seen at time  $t = 0.18$  s. During this period, the magnitude of the pressure also gradually increases until the pressure distribution is nearly hydrostatic as seen in Fig. 4(b). This phenomenon is more prominent near the left wall but becomes less obvious

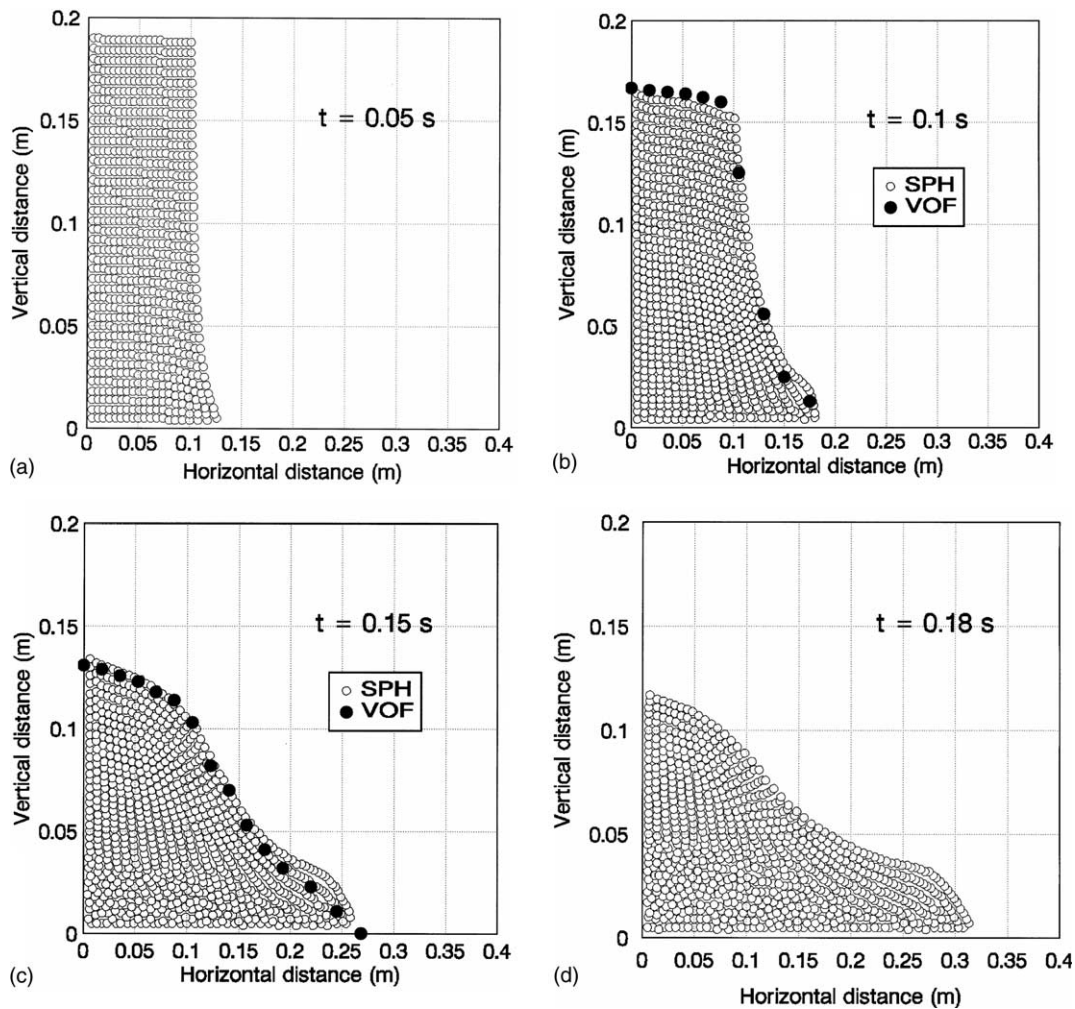


Fig. 1. Particle configurations after collapse of dam at time (a)  $t = 0.05$  s, (b)  $t = 0.1$  s, (c)  $t = 0.15$  s and (d)  $t = 0.18$  s. Black dots correspond to computations by VOF method.

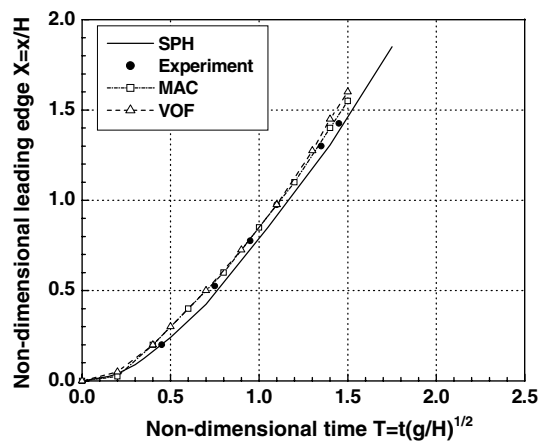


Fig. 2. Relationship between the non-dimensional leading edge and time after dam-break.

near the shock front. The flow pattern is in qualitative agreement with the analytical solutions of the non-linear shallow-water equations (e.g. [19]). In their analytical

solutions, the water height at the dam site is about  $4/9$  that of the initial height  $H$  and the downstream surface profile near the front is represented by a parabola. In our numerical simulations, the size of the upstream reservoir is finite. Thus the computed profile of the shock front and shock velocity slightly deviate from the analytical solutions, which also ignore the friction of bed and vertical variations of flow properties. The computed bore velocity at  $t = 0.18$  s ( $T = 1.26$ ) is  $1.95$  m/s from Fig. 2, which can be represented by the relationship of  $\sqrt{2gH}$ . This is also in agreement with the results by Pan et al. [18], who employed the MAC method to compute flows of broken-dam of different dam height  $H$ . In contrast, the analytical shallow-water solution by Stansby et al. [19] for an infinite size of reservoir gives a value of  $2\sqrt{gH}$  for the bore velocity.

In order to estimate the time of bore formation, various numerical tests are run by changing the width and height of the dam from 10 to 40 cm. For consistency, the starting particle spacing is constant for all



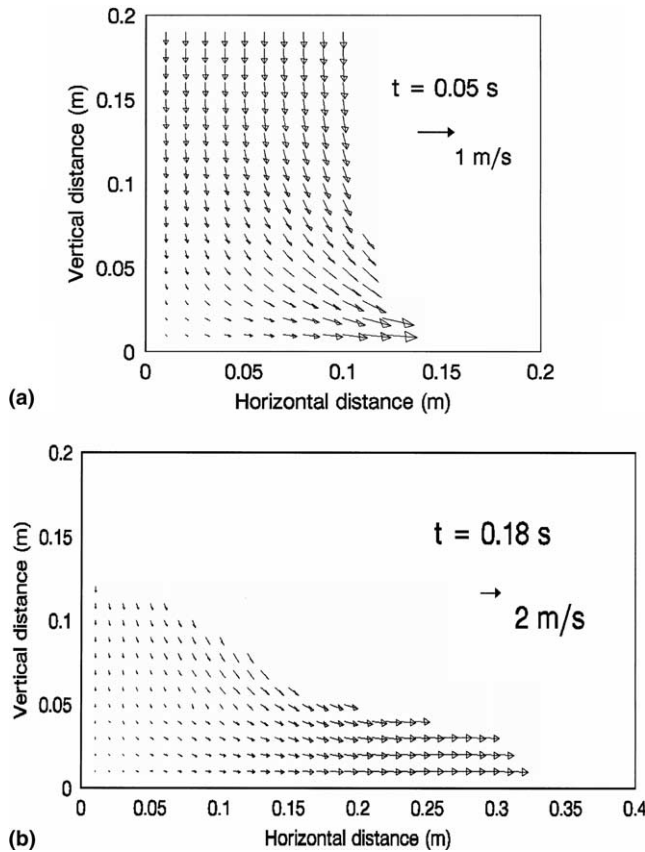


Fig. 3. Velocity distribution after collapse of dam at time (a)  $t = 0.05$  s and (b)  $t = 0.18$  s.

runs at 0.005 m as that used in the previous calculation of Fig. 1 ( $N = 20 \times 40$ ). Thus, the total number of particles differ from run to run. In the computation, the dam width is constrained to be greater than half of the dam height in order to diminish the influence of the negative wave reflecting from the left wall. The relationship between the time normalized by  $\sqrt{2H/g}$  and leading edge from the dam site normalized by dam height  $H$  is shown in Fig. 5. The legend in the figure denotes the size of the dam width and height in cm. From the figure it is seen that the formation of the bore is around time  $t/\sqrt{2H/g} = 1$ , from which point the slope of curves tends to be almost constant. Another phenomenon observed is that all groups of data overlap each other very well at the earlier time but scatter afterwards. This difference is due to the finite size of the upstream reservoir. The reflecting negative wave from the left wall (with an estimated celerity of  $\sqrt{gH}$ ) strongly influences the flow behavior at longer time. If the size of the reservoir is increased, we can expect more overlapping of the computed data. A useful conclusion can be drawn for practical purposes from Fig. 5. The bore develops later with increasing dam height and the time of bore formation after dam-break can be approximately evaluated by  $\sqrt{2H/g}$ . The shallow-water equations,

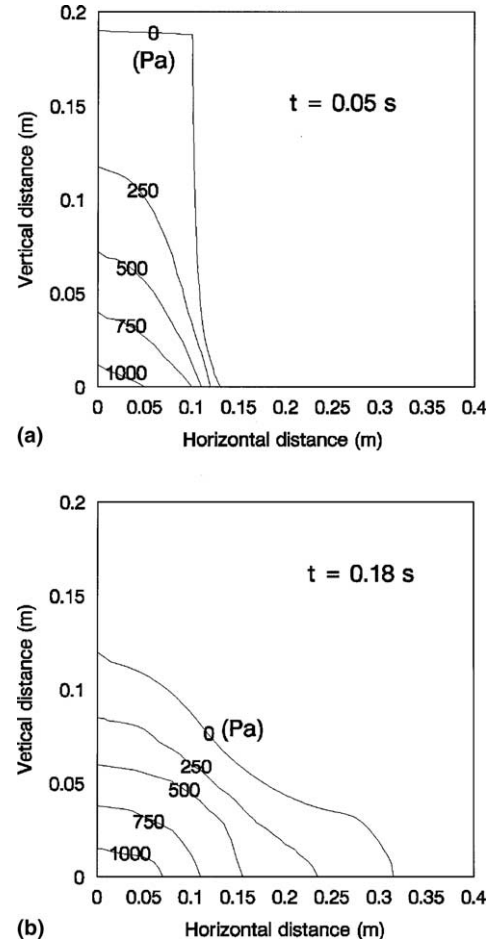


Fig. 4. Pressure distribution after collapse of dam at time (a)  $t = 0.05$  s and (b)  $t = 0.18$  s.

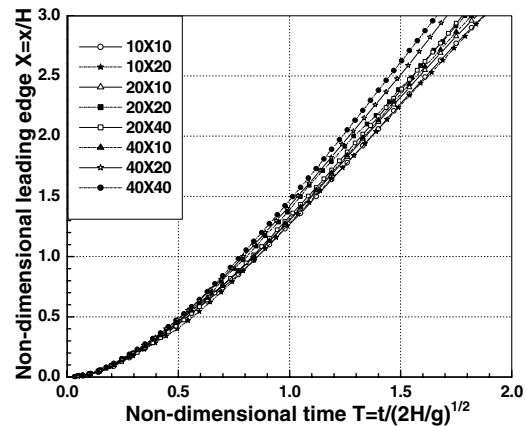


Fig. 5. Relations between the non-dimensional leading edge and time after dam-break. The legend denotes the size of dam width and height in cm.

which are based on the hydrostatic assumption and uniform velocity over the depth, are only applicable after a certain time from the initial dam-break (when a bore fully develops downstream). The problem is more dominant in case of the collapse of high dams.

## 7.2. Evaluation of numerical convergence

To investigate the convergence of the SPH method, an additional three runs with coarser and finer particle spacing for the case of Fig. 1 have been carried out. These runs have particle numbers of  $N = 10 \times 20$ ,  $40 \times 80$  and  $80 \times 160$ , respectively, corresponding to particle spacing of 0.01, 0.0025 and 0.00125 m. The time step used is 0.001, 0.00025 and 0.000125 s respectively, consistent with the previous convergence analysis. The normalized leading edge versus time for the additional runs and original run ( $N = 20 \times 40$ ) are plotted in Fig. 6(a) for comparison. It is clearly seen that the curves converge very rapidly as the particle spacing becomes smaller, i.e., particle numbers become larger. To estimate the order of the convergence, we assume that the numerical error  $E_{l_0}$  of any run with particle spacing  $l_0$  is proportional to  $l_0^n$ , where  $n$  is the order of convergence. Letting  $E_{10 \times 20}$ ,  $E_{20 \times 40}$ ,  $E_{40 \times 80}$  and  $E_{80 \times 160}$  denote the numerical error with particle numbers of  $N = 10 \times 20$ ,  $20 \times 40$ ,  $40 \times 80$  and  $80 \times 160$ , respectively, the relationship relating the error with particle spacing can be established as

$$\frac{E_{10 \times 20} - E_{20 \times 40}}{E_{20 \times 40} - E_{40 \times 80}} \approx \frac{(l_{0,10 \times 20})^n - (l_{0,20 \times 40})^n}{(l_{0,20 \times 40})^n - (l_{0,40 \times 80})^n} \quad (31)$$

A similar relationship is also obtained using  $N$  of  $20 \times 40$ ,  $40 \times 80$  and  $80 \times 160$ . Since by design,

$$l_{0,10 \times 20} = 2l_{0,20 \times 40} = 4l_{0,40 \times 80} = 8l_{0,80 \times 160} \quad (32)$$

Eq. (31) can be simplified and extended to incorporate all the runs as

$$\frac{E_{10 \times 20} - E_{20 \times 40}}{E_{20 \times 40} - E_{40 \times 80}} \approx \frac{E_{20 \times 40} - E_{40 \times 80}}{E_{40 \times 80} - E_{80 \times 160}} \approx 2^n \quad (33)$$

To determine  $n$ , we assume that the differences in numerical error between the runs at any time into the simulation can be estimated using differences in the corresponding leading edge locations from Fig. 6(a), for example,

$$E_{10 \times 20} - E_{20 \times 40} \approx |X_{10 \times 20} - X_{20 \times 40}| \quad (34)$$

where  $X_{10 \times 20}$  and  $X_{20 \times 40}$  are positions of the leading edge  $X$  for those runs. The same approach is also applied to runs of  $N = 40 \times 80$  and  $80 \times 160$ . Using this and Eq. (33), the index  $n$  can be estimated at any time into the run. An averaged value over time from  $T = 0.35$  onwards is 1.11 for the three finer runs and 1.12 for the three coarser runs. The average is not started earlier as the differences between the leading edge locations are very small and thus subjected to round-off errors. Based on this, the current SPH numerical method is estimated to be first-order accurate globally. This is consistent with error in the kernel functions being of second-order locally, and with the time integration and formulation of the Laplacian (Eq. (16)) reducing the order further. A similar conclusion has been also reached by Monaghan and Kos [17] using more qualitative arguments.

The degree to which incompressibility is achieved is another measurement of the overall accuracy of the projection scheme. This is quantitatively evaluated through the normalized density error between the computed densities and the initial constant value as

$$E_d(t) = \frac{1}{N'} \sum_{i=1}^{N'} |\rho_i(t) - \rho_0| / \rho_0 \quad (35)$$

where the summation is taken over the inner fluid particles only.

The normalized density error  $E_d(t)$  for all runs in Fig. 6(a) is shown in Fig. 6(b). It is clearly seen that the overall accuracy based on satisfying incompressibility is quite high, with  $E_d(t)$  decreases rapidly with increasing particle number. Even though a peak in the density error occurs around time  $T \sim 1$ , the error decreases thereafter and thus not accumulate. An analysis using a similar approach for the leading edge indicates that the density error decreases as  $l_0^n$  with  $n$  of 1.10 (three coarser runs) and 1.08 (three finer runs). Thus the incompressibility

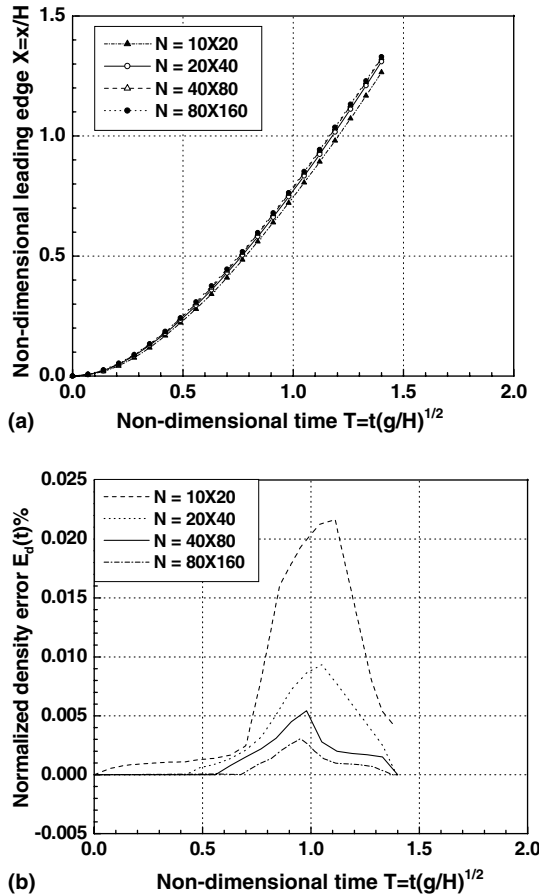


Fig. 6. (a) Leading edge at different particle spacing for convergence analysis. (b) Normalized density error at different particle spacing.

error also decreases with a similar rate as convergence in the leading edge.

### 7.3. Broken-dam analysis for a non-Newtonian fluid

To test the capability of the incompressible SPH method for non-Newtonian fluids, a 2-D unsteady mudflow released from a reservoir of finite size onto a steep channel is simulated. The reservoir is introduced by having vertical wall in a sloping channel. The problem is based on the water–clay mixture experiments carried out by Komatina and Jovanovic [9], to which the reader is referred to for a detailed description of the experiment and material properties.

A total number of 69 unsteady flow experiments were made by Komatina and Jovanovic [9], with a range of initial reservoir depth of 10–30 cm, channel bed slope of 0.0–0.1% and volume concentration of mixtures of 0.0–36.1%. The computation presented below reproduces two of the experiments, in which the initial mixture length  $a = 2.0$  m, initial height  $H = 0.1$  m and channel bed slope  $S_0 = 0.1\%$  is considered. The volume concentration of the

mixtures is taken as 0.0% and 27.4% respectively, corresponding to pure water and mud of density  $1200 \text{ kg/m}^3$ . The mud has Bingham yield stress  $\tau_B = 25.0 \text{ Pa}$  and viscosity  $\mu_B = 0.07 \text{ N s/m}^2$  as reported in the experiment. The purpose of computing water flow is for comparing the effect of viscosity on the flow characteristics.

The initial spacing of particles is  $0.01 \text{ m}$ , with 2000 particles used in the computation and the time step is dynamically adjusted throughout the whole computation to improve efficiency while satisfying the stability conditions of Eqs. (22) and (23). The particles are initially put in a uniform square grid form with the  $x$ -axis parallel to the sloping channel bed. Two vertical walls perpendicular to the channel bottom confine the mud reservoir. Since this introduces an artificial initial slope at the free surface, the computation starts by first letting the particles in the reservoir region of the channel re-adjust until the computed pressure distribution is hydrostatic. The wall constraint is then instantaneously released corresponding to time  $t = 0$ .

The particle configurations for the flow at different times after dam-break  $t = 0.1, 0.3, 0.6, 1.0, 3.0$  and

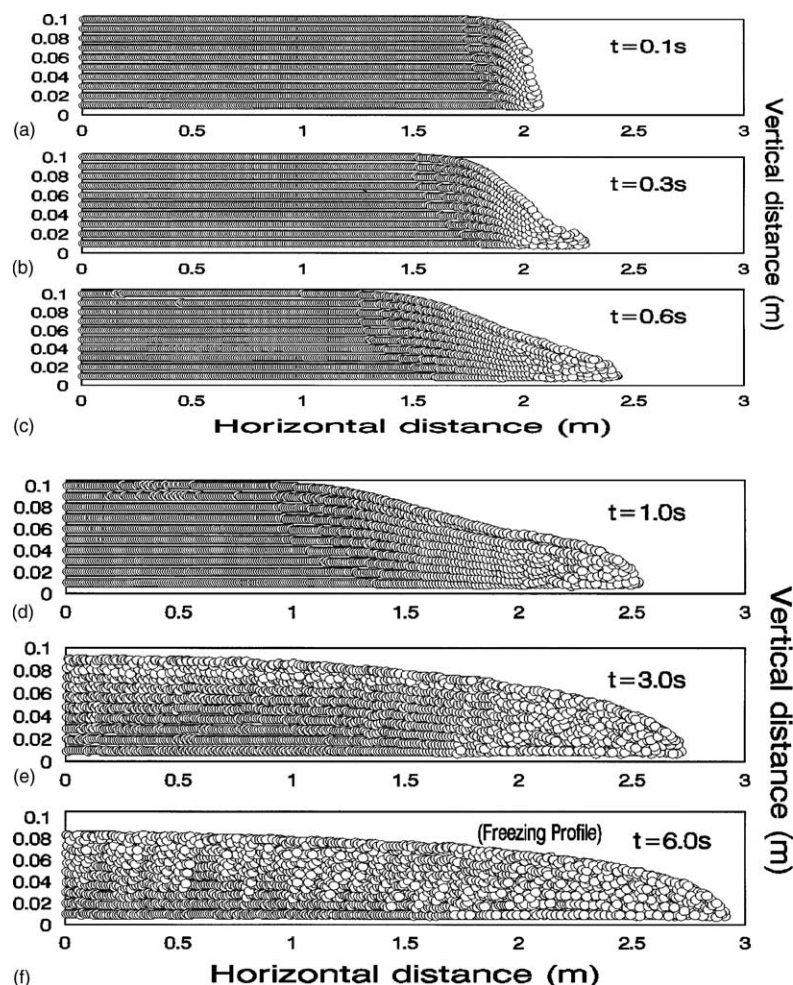


Fig. 7. (a)–(f) Particle configurations after collapse of dam at time  $t = 0.1, 0.3, 0.6, 1.0, 3.0$  and  $6.0 \text{ s}$ . (f) “Freezing” profile of mudflow.

6.0 s, are shown in Fig. 7(a)–(f), in which Fig. 7(f) corresponds to the “freezing” profile of mudflow. The simulated flow patterns are very similar to the mudflow profiles observed experimentally by Komatina and Jovanovic [9]. They are also very similar to the analytical profiles of Jeyapalan et al. [8], who employed the perturbation solutions of shallow-water equations to predict the downstream inundation processes of liquefied fluids released from a upstream reservoir of infinite size.

The computed non-dimensional leading ledge of the flow  $X = x/H$  referenced from the dam site with non-dimensional time  $T = t\sqrt{g/H}$  is plotted in Fig. 8 for the water and mudflows, and the computed surface profiles after the dam-break are shown in Fig. 9 at  $t = 0.37$  and  $0.6$  s. The experimental data of Komatina and Jovanovic [9] are given for comparison and the good agreement is obvious in both cases.

From Fig. 7–9, several distinctive flow features are observed as follows. Starting from the initial hydrostatic state, the collapsing mudflow is first characterized by a rapid drop of the flow depth near the dam site under the effect of gravity. In this computation, the flow depth is reduced to about 3/4 of the initial value  $H$  at the dam site  $X = 0$  (Fig. 7(a)–(c)). Later, the main features of fully developed mudflow are a slow decrease of the maximum flow depth and an almost constant velocity for the front propagation. Finally the stopping processes of the mudflow are described by a rapid decrease of both the maximum flow depth and front velocity down to zero. It is also shown that the final stage of the mudflow is similar to a creeping motion and the shape of surface profiles is almost unchanged. Compared with the water flow under same conditions, the mudflow has its shock depth and shock velocity decaying more rapidly with time and the mudflow propagates more slowly due to its

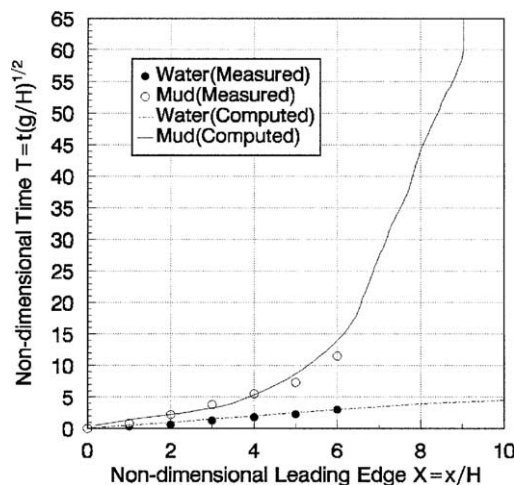


Fig. 8. Relationship between the non-dimensional leading edge and time after dam-break for water and mudflow.

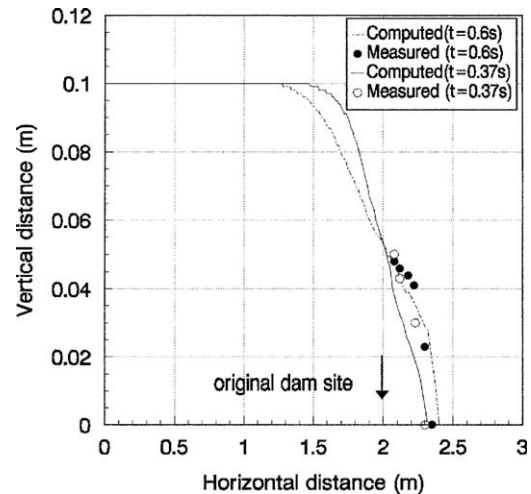


Fig. 9. Surface profiles of mudflow after collapse of dam at time  $t = 0.37$  and  $0.6$  s.

more viscous nature. For example, the computed front velocity of water from Fig. 8 is 1.8 m/s at early times, while the corresponding velocity of mudflows is only 30% of that value. This also implies that the mudflow is mainly in the state of low shear rate regime. Thus according to Eq. (28) its effective shear viscosity in the computation is much higher than its high shear rate viscosity  $\mu_\infty$  ( $\mu_B$ ). Another distinctive flow patterns of the mudflow as different from the water flow is its “freezing” features as shown in Fig. 7(f) and Fig. 8, in which the mudflow stops at  $T = 60.0$  and  $X = 9.1$  from the dam site. However, the experimental profiles for  $t > 0.6$  s are unavailable for comparison. The mudflow only propagates a finite distance downstream and the points located further downstream will be unaffected because of the “freezing” behavior of the mudflows when submitted to a shear stress much lower than its yield stress. In contrast, a Newtonian flow will propagate an infinitely long distance downstream from its source, with both its shock depth and shock velocity decaying continuously with time.

In order to further evaluate the effect of the rheological model employed in the computation, the relationship between the shear stress  $\tau$  and strain rate  $\dot{\gamma}$  is shown in Fig. 10, for both the Cross and Bingham models at the rheological parameters used. It is seen that both models approximate each other quite well in most shear conditions. In the region of low strain rates the Cross model keeps the relationship between the shear stress and strain rate continuous, thus improving numerical stability in the computation. However, it could be seen that a yield interface could be defined at a critical shear rate  $\dot{\gamma}_c$ , where the shear stress equals the yield stress  $\tau_B$  of the Bingham model. For shear rate greater than  $\dot{\gamma}_c$ , the shear stress exceeds the yield stress and flow behaves as a shear flow with varying viscosity and velocity. For shear rate less than  $\dot{\gamma}_c$ , the viscosity rapidly

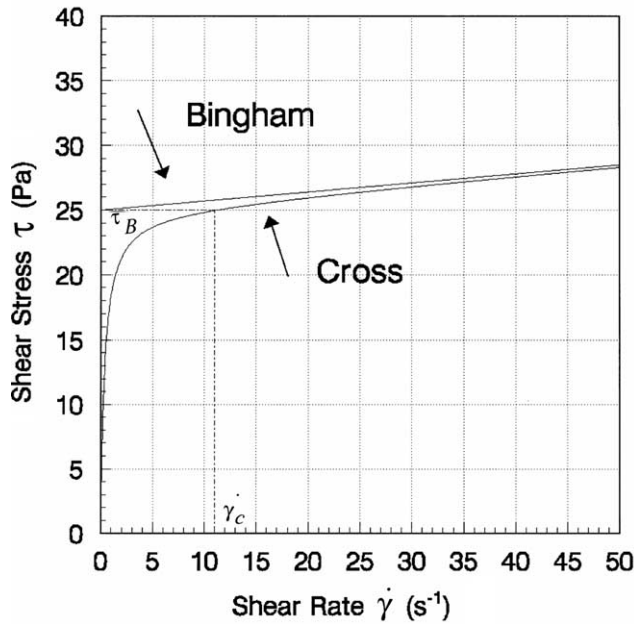


Fig. 10. Relationship between shear stress and strain rate of Bingham and Cross model for rheological parameters used in computation.

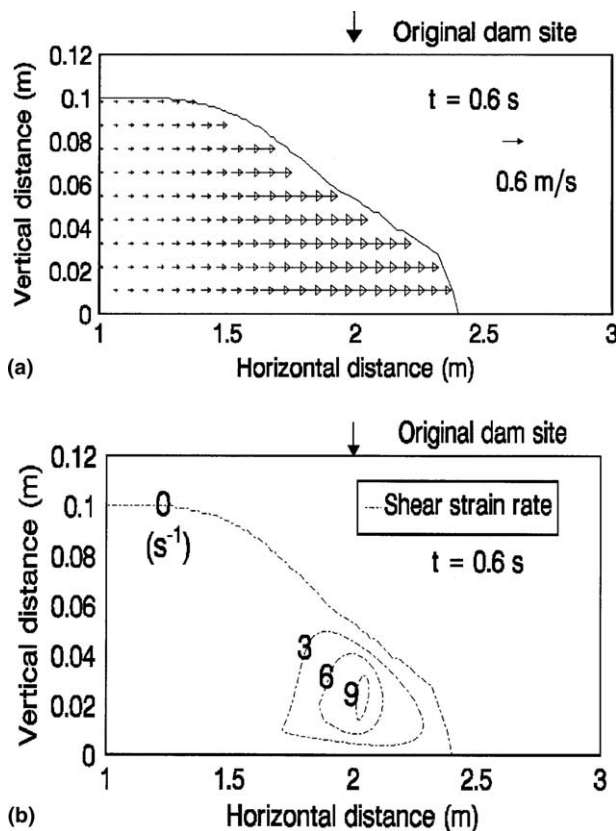


Fig. 11. (a) Velocity distribution of mudflow after collapse of dam at time  $t = 0.6$  s. (b) Contour lines of computed shear rate of mudflow after collapse of dam at time  $t = 0.6$  s.

increases and the flow moves more like a plug with uniform velocity.

The above behavior is seen by the velocity distribution and contour lines of shear strain rate for the region at the front of the flow near the dam site at time  $t = 0.6$  s, as shown in Fig. 11(a) and (b) respectively. Two phenomena are observed from this figure. One is that the maximum shear rate inside the mudflow is below the critical shear rate  $\dot{\gamma}_c$ , which is around the value of 11/s from Fig. 10. This suggests that the whole mudflow is in the state of low shear rate and plug-like with almost uniform velocity. Another distinctive phenomenon observed is that the shear rate assumes larger values near the shock front due to the large deformation of free surfaces as compared with the static wall boundary. This means that the viscosity near this region is relatively smaller and the mudflow is more Newtonian-like with non-negligible velocity variations.

## 8. Conclusion

The paper presents an incompressible SPH method to simulate Newtonian and non-Newtonian flows with free surfaces. The method employs particles to discretize the Navier–Stokes equations and the interactions among particles simulate the flows. The numerical model is very effective in dealing with large deformation of free surfaces. A dam-break problem is considered, involving both Newtonian and non-Newtonian fluids. The computations are in good agreement with available experimental data. It shows that the propagation of mudflows is much slower than the water flows and the front profile of mudflows is much steeper, with the corresponding shock depth and shock velocity decaying more rapidly. One of the distinctive features of the mudflow is that it only travels within a limited distance downstream due to the existence of the yield stress and high viscosity at low shear rate. Under the computational conditions in the paper, the mudflow is “frozen” downstream after traveling a distance of about nine times of the initial reservoir height. Furthermore, the leading edge of the flow occurs in the low shear rate regime with large variation in the viscosity.

## References

- [1] Barnes HA, Hutton JF, Walters K. An introduction to rheology. Amsterdam: Elsevier; 1989.
- [2] Chorin AJ. Numerical solution of the Navier–Stokes equations. Math Comput 1968;22:745–62.
- [3] Cummins SJ, Rudman M. An SPH projection method. J Comput Phys 1999;152:584–607.
- [4] Gotoh H, Sakai T. Lagrangian simulation of breaking waves using particle method. Coastal Eng J 1999;41(3–4):303–26.
- [5] Hammad K, Vradis GC. Flow of a non-Newtonian Bingham plastic through an axisymmetric sudden contraction: effects of Reynolds and yield numbers. Numer Meth Non-Newtonian Fluid Dyn, ASME 1994;179:63–9.

- [6] Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Phys Fluids* 1965;8(12):322–9.
- [7] Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *J Comput Phys* 1981;39:201–25.
- [8] Jeyapalan JK, Duncan JM, Seed HB. Analysis of flow failures of mine tailings dams. *J Geotech Engr, ASCE* 1983;109(2):150–89.
- [9] Komatina D, Jovanovic M. Experimental study of steady and unsteady free surface flows with water–clay mixtures. *J Hydraul Res* 1997;35(5):579–90.
- [10] Koshizuka S, Tamako H, Oka Y. A particle method for incompressible viscous flow with fluid fragmentation. *Comput Fluid Dyn J* 1995;4(1):29–46.
- [11] Koshizuka S, Nobe A, Oka Y. Numerical analysis of breaking waves using the moving particle semi-implicit method. *Int J Numer Meth Fluids* 1998;26:751–69.
- [12] Lin PZ, Liu Philip LF. A numerical study of breaking waves in the surf zone. *J Fluid Mech* 1998;359:239–64.
- [13] Lucy LB. A numerical approach to the testing of the fission hypothesis. *Astron J* 1977;82(12):1013–24.
- [14] Martin JC, Moyce WJ. An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philos Trans R Soc London, Ser A* 1952;244:312–24.
- [15] Monaghan JJ, Lattanzio JC. A refined particle method for astrophysical problems. *Astron Astrophys* 1985;149:135–43.
- [16] Monaghan JJ. Smoothed particle hydrodynamics. *Annu Rev Astron Astrophys* 1992;30:543–74.
- [17] Monaghan JJ, Kos A. Solitary waves on a Cretan beach. *J Wtrwy, Port, Coastal and Ocean Eng* 1999;125(3):145–54.
- [18] Pan CH, Xu XZ, Lin BY. Simulating free surface flows by MAC method. *Estuar Coastal Eng* 1993;1–2:51–8 (in Chinese).
- [19] Stansby PK, Chegini A, Barnes TCD. Initial stages of dam-break flow. *J Fluid Mech* 1998;370:203–20.
- [20] Zhu Y, Fox PJ, Morris JP. A pore-scale numerical model for flow through porous media. *Int J Numer Anal Meth Geomech* 1999;23:881–904.