EFFICIENT SIMULATION OF THE FLUID-STRUCTURE INTERFACE

Luis Gerardo de la Fraga[1], Ernesto Olguín-Díaz[2] and Fernando García-Arreguín[1]
[1]*Computer Science Department, CINVESTAV, Av. I.P.N. 2508. CP 07360, México City, México*

[2]*Robótica y Manufactura Avanzada, CINVESTAV, Carr Saltillo-Monterrey Km 13.5 CP 25900, Saltillo, México*

Keywords: Fluid-structure simulation, Haptics.

Abstract: In this work, an alternative method to simulate fluid interactions with compliant surfaces is described. The simulation of the fluid phenomena is performed by the Smooth Particle Hidrodynamics (SPH) method. In the normal SPH method, solid interfaces are modeled with sets of static particles covering the boundaries. In the proposed alternative method, a compliant-solid interface is modeled as a polygon allowing to substitute the particles that represent a surface in the normal SPH method. Therefore, by considering less particles, a simplification on simulations is achieved, and the alternative method still describes the general behavior of the phenomena. Furthermore, a deformable object, this is, a time variant object, can be modeled as a polyhedron, with a mass-spring-dashpot system in each of its edges, and with each polygon as a compliant-solid interface. Bidirectional dependence on the alternative method for fluid simulation and the deformable model gives a new method for the simulation of compliant solids partially or fully immersed in an incompressible fluid. This new method is not intended to have a high accuracy in the numerical results but to have a perceptual high qualitative behavior and fast numerical response, to be applied in visual/haptic simulators.

1 INTRODUCTION

Understanding the dynamic of fluids has been a quest for many engineers and physicists mainly because its knowledge can help in the design process of many fluid applications such as hydroelectric dams (chuan Bai et al., 2007), water supply nets (Chen et al., 1995), aerodynamics of an aircraft (Eddy et al., 2006) or the behavior of an alien corp in blood stream of a patient (Tai et al., 2007). Some of these problems have been solved with approximate solutions of the well known Navier-Stokes equations, which describe with partial equations the behavior of a fluid under specific circumstances.

When the interaction of the fluid with a solid structure needs to be computed (for the case of the container or an object inside it) the problem becomes even more complicated due to the fact that the border conditions on the partial differential equations are dependent on time. The complexity of these partial differential equations plus the geometric complexity of volume containing the fluid lead to very complex and large amount of simultaneous equations that can only be solved approximately using numerical solutions, and finite element methods. This problem is known in the literature as the Fluid-Structure problem One possible, yet very expensive, way of compute the overall solution can be that of computing the conservation of momentum for every particle in the fluid and apply Newton's second law. The big problem with this approach is that there become too many simultaneous equations to solve three dimensional variables (interaction forces between the particles) plus the time varying border conditions of the working volume. So even when this method may lead to more accurate solutions, it is impractical for online –realtime– calculations. Fluid dynamics simulations used in haptics (kinesthetic feedback to the user) need faster numerical solutions but preserving the qualitative behavior of the fluid physical system, not only within the fluid but also with the container or bodies moving along. The big problem reduces to how the Navier-Stokes equations can be simplified to still represent, with some degree of accuracy, the fluid dynamic so that a common computer would be able to calculate the algorithm fast enough to be used in real time applications.

We propose a system where the fluid dynamics is solve with the Smoothed Particle Hydrodynamics algorithm (SPH) (Monaghan, 2005), and the border effects are solved by a compliant model composed

Gerardo de la Fraga L., Olguín-Díaz E. and García-Arreguín F. (2009).
EFFICIENT SIMULATION OF THE FLUID-STRUCTURE INTERFACE.
In *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics - Robotics and Automation*, pages 268-273
DOI: 10.5220/0002211702680273

by plane surfaces. We are using a deformable model composed by simplex meshes and a simple mechanical system in every edge on the mesh (Trejo and de la Fraga, 2005). The contribution of this work lies in the fluid-structure interface modeling that is used in the SPH as forces on the particles at the borders and as pressure in every mesh face. Equivalences between forces and pressures are given by the time varying known areas of the mesh faces.

This paper is organized as follow: in Sec. 2 the SPH method to model a fluid is described. In Sec. 3 the model of a compliant body and the way how it is integrated with the SPH to build the complete simulator are presented. In Sec. 4 the implementation of the simulator is discussed and some results presented. Finally, Sec. 5 presents the conclusions of this work.

2 FLUID DYNAMIC MODELING USING SPH

SPH use a Lagrangian approximation to the fluid mechanics in the context that it calculates the evolution of variables associated with the particles within the fluid, instead of inertial referenced positions (as done in the Euler method). This is the strongest characteristic of the method.

In order to explain the SPH method, we can start with the continuity equation, which refers to the particle's mass in the form of density (Streeter and Wylie, 1988):

$$\frac{d}{dt}\rho = -\nabla \cdot \rho \mathbf{v} \tag{1}$$

The particle's acceleration is given by the pressure's gradient as $\frac{d}{dt}\mathbf{v} = -\frac{1}{\rho}\nabla p$, where $\mathbf{v}$ is the velocity if particle ($\mathbf{v} = \frac{d}{dt}\mathbf{r} = \dot{\mathbf{r}}$) and $\mathbf{r}$ stands for the position of the fluid particle. By using the chain rule of $\nabla\left(\frac{p}{\rho}\right) = \frac{\nabla p}{\rho} - \frac{p}{\rho^2}\nabla\rho$, particle's acceleration can be written as

$$\frac{d}{dt}\mathbf{v} = -\nabla\left(\frac{p}{\rho}\right) - \frac{p}{\rho^2}\nabla\rho, \tag{2}$$

On the other hand, the interpolation function

$$\langle f(r)\rangle = \int_D f(r')W_{r,h}(r')dr' \cong \sum_{j=1}^{N} f(r_j)W_{r,h}(r_j)V_j \tag{3}$$

is the Monte Carlo numerical integration of equation $\langle f(r)\rangle = \int_D f(r')W_{r,h}(r')dr'$ in order to obtain discrete points from the information of finite set of known points r_j. The function $\langle f(r)\rangle$ is the average of function $f(r)$ around point r, where D is the domain of this function, $W(r)$ is the *kernel* and V_j is the volume of point r_j.

Using (3) in (2), a particle's movement equation is obtained as (Pérea et al., 2005):

$$\frac{d}{dt}\mathbf{v} = -\sum_j m_j\left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{i,j}\right)\nabla_i W_{i,j} + \bar{\mathbf{f}}_i \tag{4}$$

Two terms are added in (4): the first is an artificial viscosity $\Pi_{i,j}$, which is a dissipation term that should tend to decrease the kinetic energy of the particles when no external forces are acting on the set of particles. The second term is indeed the external force, $\bar{\mathbf{f}}_i$, applied to every particle i.

Densities are calculated as:

$$\frac{d}{dt}\rho_i = \sum_j m_j \mathbf{v}_{i,j}\cdot\nabla_i W_{i,j} \tag{5}$$

where $\mathbf{v}_{i,j} \triangleq \mathbf{v}_i - \mathbf{v}_j$ is the relative velocity of particles i and j.

Kernel. The kernel function determines the interaction among the different particles of the fluid. Each particle interacts only with its nearest neighbor particles, inside its influence zone. We use B-splines (Monaghan and Lattanzio, 1985) defined as:

$$W_{i,j}(\mathbf{r}_i,\mathbf{r}_j,h) \triangleq \begin{cases} C\left(1-\frac{3}{2}q^2+\frac{3}{4}q^3\right), & \text{if } 0\le q<1 \\ C\left(\frac{1}{4}(2-q)^3\right) & \text{if } 1\le q<2 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where h is a half of the influence radio, $q = r_{i,j}/h$ is the normalized distance $r_{i,j}$, and this last one is the magnitude of the relative position of particles i and j as $r_{i,j} = \|\mathbf{r}_{i,j}\|$, where $\|\cdot\|$ stands for the norm of any vector, i.e: $\|\mathbf{a}\| = \sqrt{a_x^2+a_y^2+a_z^2}$ and $\mathbf{r}_{i,j} = \mathbf{r}_i - \mathbf{r}_j$. The constant parameter C is equal to $2/(3h)$ for one dimension, $10/(7h^2)$ for 2D and $1/(\pi h^3)$ for 3D.

Artificial Viscosity. The main purpose of the artificial viscosity term in equation (4) is to avoid the presence of oscillations or nonphysical collisions in the simulation procedure. It is calculated as:

$$\Pi_{i,j} \triangleq \begin{cases} \frac{-\alpha\bar{c}_{i,j}\mu_{i,j}+\beta\mu_{i,j}^2}{\bar{\rho}_{i,j}} & \text{for } \mathbf{v}_{i,j}\cdot\mathbf{r}_{i,j}<0; \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $\mu_{i,j} \triangleq \frac{h\mathbf{v}_{i,j}\cdot\mathbf{r}_{i,j}}{r_{i,j}{}^2+0.001h^2}$ is a normalized version of the relative velocity between two particles, with an artificial term in order to avoid numerical ill conditioning; $\bar{\rho}_{i,j} \triangleq (\rho_i+\rho_j)/2$ is the average of densities of the two particles; and $\bar{c}_{i,j} \triangleq (c_i+c_j)/2$ is the average of sound's velocity due to their different density. The constants α and β stand for the laminar and turbulent

flow dissipation and according to (Monaghan, 1994) they take values of $0.1 - 0.01$ and 0, respectively.

To use SPH with incompressible fluids, *slightly compressibility* must be considered to allow dynamic simulation of these fluids. This is achieved considering the atmospheric pressure negligible as:

$$p = B\left(\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right) \tag{8}$$

where ρ_0 is a reference density (e.g. for water is 1000). This relationship is known as the *state equation* (which in this context refers to the physical phase of the fluid and not to the dynamic *state* as usually referred in control engineering jargon).

If constants γ and B are high enough (for example, take 7 and 3000 respectively), state equation (8) computes constrained variation on pressures (under 10^5 atmospheres for water in the example). In this case, sound's velocity is sufficiently high and variations in the relative density are small:

$$\frac{|\delta\rho|}{\rho} \approx \frac{v^2}{c_s^2} \tag{9}$$

where v is the maximum fluid velocity. Moreover, if $v/c_s < 0.1$, it can be assumed that $|\delta\rho|/\rho \approx 0.01$. In this case, sound's velocity can be calculated as $c_s^2 = (\gamma B)/\rho_0$. Thus, if $B = 100\rho_0 v^2/\gamma$, then the variations on the relative velocity can take values of the order of 0.01. The calculations finish by approximating the maximum fluid velocity by $v^2 = 2gH$, where g is the gravity constant and H is the fluid's working area (Monaghan, 1994).

3 COMPLIANT SOLID MODELING

In contrast with rigid bodies, compliant solids can not be represented by dynamical lumped equations. This means that the order of the time varying dynamical equation should be infinite. Reduction of the order of this type of equations, for practical simulation purposes, yields to the so called Finite Element Methods (FEM). These methods consist basically in discretisize the body on a finite number of small simple mechanical models. Then a set of simultaneous but not so complex equations may be solve by different numerical methods. The new problem is then determined by the border or boundary conditions that exist in the new compliant solid.

In the next paragraphs, we propose a new method to calculate these boundary conditions.

3.1 Modeling with Simplex Meshes

Simplex meshes are used to represent surfaces in the three-dimensional space. These meshes have similarities with triangulations, in fact a 2-simplex mesh is the topological dual of the triangulation, but they are not geometric duals. This means that, we can not build a geometric transformation between triangulations and simplex meshes. An important property of simplex meshes is their constant vertex connectivity: each vertex in the mesh has three and only three neighboring vertices. This condition allows to use the same deformation engine to solve four differential equations for the four mass-spring-dashpot systems attached on each simplex. In addition, it has the advantage that allows smooth deformations in a simple and efficient manner. In this work we used the model of a sphere built as is presented in (Flores and de la Fraga, 2004).

A surface made with simplex meshes can be visualize it as a mesh of hexagons, and it is easy to represent each hexagon with four triangles. Then, each triangle can be modeled as a single compliant solid surface. The result is a compliant solid body with arbitrary three-dimensional geometry.

3.2 Fluid Particles Interaction

The interaction of a fluid particle and a surface can be modeled by the interaction forces or pressures. The same force, in opposite sense, is exerted by the surface to the particle.

The contact force that the surface exert to the particle can be modeled in two orthogonal components as $\mathbf{f}_c = \mathbf{f}_d + \mathbf{f}_f \in \mathbb{R}^3$, where $\mathbf{f}_d$ is the deformation force due to the elastic stresses and mechanical deformation of the body surface and in this work is considered to be strictly perpendicular (normal) to the surface. The friction component $\mathbf{f}_f$ is considered to be strictly tangential to the surface.

The Normal Operator. Lets be $s(x,y,z) = 0$ the function in the Euclidean space defining the surface with whom the particle is contact at point $\mathbf{r}_c = (x_c, y_c, z_c)^T$, where the variables x_c ,y_c and z_c are the Cartesian coordinates of the contact point. The unit vector $\lambda_N \in \mathbb{R}^3$ is defined as the normalized gradient of the surface at point $\mathbf{p}_c$:

$$\lambda_N \triangleq \frac{\nabla s(\mathbf{p}_c)}{\|\nabla s(\mathbf{p}_c)\|} \tag{10}$$

The deformation can be calculated as the normal component at the relative position of the particle $\mathbf{r}(x,y,z)$ with respect to the contact point at the surface $\mathbf{x} \triangleq \mathbf{r} - \mathbf{p}_c \in \mathbb{R}^3$. The normal component $\mathbf{x}_N$ is

obtained from the next equation

$$\mathbf{x}_N = N\mathbf{x} \tag{11}$$

where the Normal Operator N is defined as

$$N \triangleq \lambda_N \lambda_N^T \tag{12}$$

The normal component of the relative velocity can be calculated using the Normal Operator N as

$$\dot{\mathbf{x}}_N = \lambda_N \lambda_N^T \dot{\mathbf{x}} \tag{13}$$

where $\dot{\mathbf{x}} = \mathbf{v} - \dot{\mathbf{p}}_c$, is the relative velocity between the particle and the surface, $\mathbf{v}$ and $\dot{\mathbf{p}}_c$ are the linear velocities of the particle and the contact point of a rigid surface, respectively [1].

The Tangential Operator. As well as for the normal component of the deformation or velocity to the surface, a linear operator $T : \mathbb{R}^3 \mapsto \mathbb{R}^3$ shall exist that gives the tangential projection of these vectors. This Tangential Operator must be of the form

$$\dot{\mathbf{x}}_T = T\dot{\mathbf{x}} \tag{14}$$

In the same way that Normal Operator has been defined in (12), The Tangential Operator is :

$$T \triangleq \lambda_T \lambda_T^T \tag{15}$$

where λ_T stands for any unit vector tangent to the surface s at point $\mathbf{p}_c$. One possible way is defining it in the direction of the tangent velocity as:

$$\lambda_T \triangleq \frac{\dot{\mathbf{x}}_T}{\|\dot{\mathbf{x}}_T\|} \tag{16}$$

By construction, operators N and T shall be orthogonal complements and fulfill next properties

$$NT = TN = [0], \text{ and } N + T = I \tag{17}$$

Then the Tangential Operator defined as

$$T = -[\lambda_N \times]^2 \tag{18}$$

fulfill properties (17), where the operator $[\mathbf{a}\times]$ stands for the matrix representation of the cross product: $\mathbf{a} \times \mathbf{b} = [\mathbf{a}\times]\mathbf{b}$.

Deformation Normal Force. This force, normal to the surface can be modeled as a simple second order, hence continuous system, from the compliant model of the surface. That is, the normal force exerted by a particle of mass m_p, when colliding with a surface s at point p_c with velocity $\dot{\mathbf{x}}$ is given by the following equation

$$\mathbf{f}_p = k_s \mathbf{x}_N + b_s \dot{\mathbf{x}}_N \tag{19}$$

[1]Note: If the surface is considered uncompliant and with no movement, velocity at the contact point is null, $\dot{p}_c = 0$

where the constant k_s is the elastic modulus of the surface material, b_s is an artificial damping coefficient and variables $\mathbf{x}_N$ and $\dot{\mathbf{x}}_N$ are the normal projections of the relative position and velocity respectively of the particle in the contact point.

By Newton's laws, the reaction force, i.e. the one felt by the particle and induced by the surface is

$$\mathbf{f}_d = -\left[\lambda_N \lambda_N{}^T\right](k_s \mathbf{x} + b_s \dot{\mathbf{x}}) \tag{20}$$

In order to simulate a non absorbent material, the artificial damping coefficient b_s should be chosen sufficiently small with the restriction $b_s << 2\sqrt{k_s m_p}$.

Friction Tangential Force. This force can be modeled by a simple Coulomb friction as

$$\mathbf{f}_f = \mu_k \|\mathbf{f}_d\| (-\lambda_T) \tag{21}$$

where μ_k is the dynamic friction coefficient between the particle and the contact surface. The $\mathbf{f}_f$ direction is given by the unit tangent vector if it is defined as in (16). Then, equation (21), can be rewritten as:

$$\mathbf{f}_f = -\mu_k \|\mathbf{f}_d\| \frac{\dot{\mathbf{x}}_T}{\|\dot{\mathbf{x}}_T\|} \tag{22}$$

where $\dot{\mathbf{x}}_T$ is calculated by (14).

3.3 Integration

The sum of the deformation force (20) and the friction one (22) is the total contact force $\mathbf{f}_c$ that each particle feels when colliding with a surface. It shall be normalized by the particle's mass in order to be included in equation (4).

To be included in the model of the compliant body made up of simplex meshes the forces of all the particles that collide with a specific planar surface (a triangle in our case) are added and then normalized by the area of that triangle in order to include these interface forces as border pressure. The resulted force is applied at the three triangle's vertices.

4 IMPLEMENTATION DETAILS

The next algorithm shows our implemented SPH method:

1: Set initial time, $t \leftarrow t_0$,
2: Set time increment Δ,
3: Set stop time, $T \leftarrow t_f$,
4: **while** $t \leq T$ **do**
5: **for all** p_i particle **do**
6: Find the neighbor particles to p_i, using (6).
7: Calculate the acceleration with (4), and

8: Calculate the density change for p_i using continuity expression (5).
9: For each particle, update its position, velocity and density, using leapfrog numerical method.
10: For each particle, calculate its pressure (8).
11: $t+=\Delta$

The most time consuming step is finding the neighbor particles to the given one in line 6 of the SPH algorithm, which is the calculation of equation (6). For n particles, $(n)(n-1)/2$ calculations of distance among each particle pair must be performed. Therefore, to reduce the execution time in the 3D simulations, we create a grid of cubes of size $2h \times 2h \times 2h$. In each cube there is a list of belonging particles. When a particle moves, its indexes are recalculated to a new position, and the particle is moved to the corresponding list. In this manner, we reduce the search of neighbors of the whole set of particles into a neighborhood of 26 cubes around the given one. However, the neighbor finding algorithm is still of complexity $O(n^2)$; this part of the algorithm could be improved with more complex data structures.

For locating if a particle it is inside the kernel, the workplace, or in a side of a plane, we use a simple collision detector which consists in testing the sign of the value given by the implicit equation of the interface.

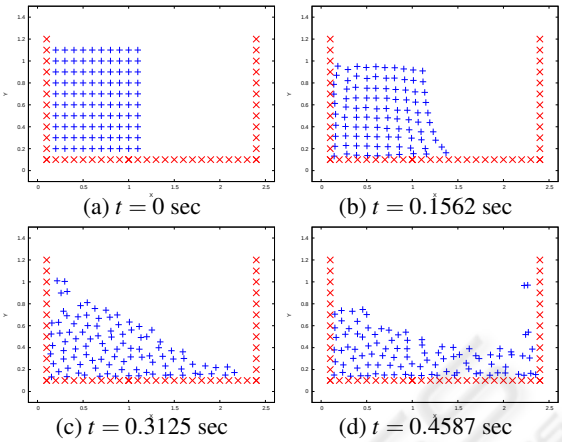
For the deformable object we use SOLID collision detector (van den Bergen, 2004), which it is optimized to calculated the collision among a ray and a set of triangles.

4.1 Results

We perform two experiments to test our simulator. The first experiment is a classical simulation of rupture of a dam. The second experiment is a deformable sphere inside a tube in which is circulating an incompressible fluid. We are going to describe in detail each experiment.

We performed the first experiment with border particles in order to test our SPH implementation. In Fig. 1(a), we can see the initial state of the dam. The working space is made up with the static particles marked as "×". The set of particles marked with "+" form the incompressible fluid. A border to the right of the fluid (the dam's border) is removed at time t_0. All fluid's particles suffer a gravity force. In Fig. 1(b)-(d) we can see the carried simulation. The simulation works as expected, and it is similar to the showed in (Monaghan, 1994).

The second experiment uses compliant simplex meshes surfaces to model a tube where an incompressible fluid is running and the surface of a deformable sphere is placed inside the moving fluid.



(a) $t = 0$ sec (b) $t = 0.1562$ sec (c) $t = 0.3125$ sec (d) $t = 0.4587$ sec

Figure 1: Simulation of a dam's rupture.

The fluid moves by the action of an horizontal force equal to the gravity force. It is difficult to check by visualization that the simulation is running ok in this scenario. Thus, values along the fluid are taken: the average particle velocity through a transverse tubes' section, and the average pressure through longitudinal tubes' sections. These values are shown on Fig. 2, and we can see that simulation's results are correct.

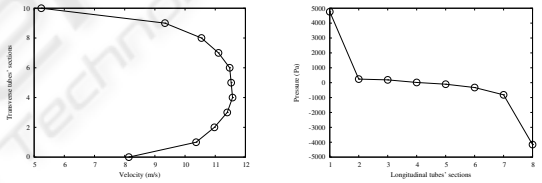


Figure 2: Left curve shows the variation of the mean velocity (horizontal axis) taken at a transverse section of the tube (vertical axis). Right curve shows the average pressure (horizontal axis) along the tube (horizontal axis).

Finally, in Fig. 3, we see several views of the simulation of the deformable sphere inside the fluid. The deformable surface takes the form of an ovoid, as it is expected.

5 CONCLUSIONS

We developed a fluid dynamics simulator based in the Smooth Particle Hydrodynamics method plus the use of compliant bodies modeled with simplex meshes. The interface interaction has been modeled by orthogonal forces produced by each particle colliding with the surface and simple models of 2^{nd} order compliant deformation and Coulomb friction.

We can simulated containers or deformable objects contact with slightly compressible fluids. Our

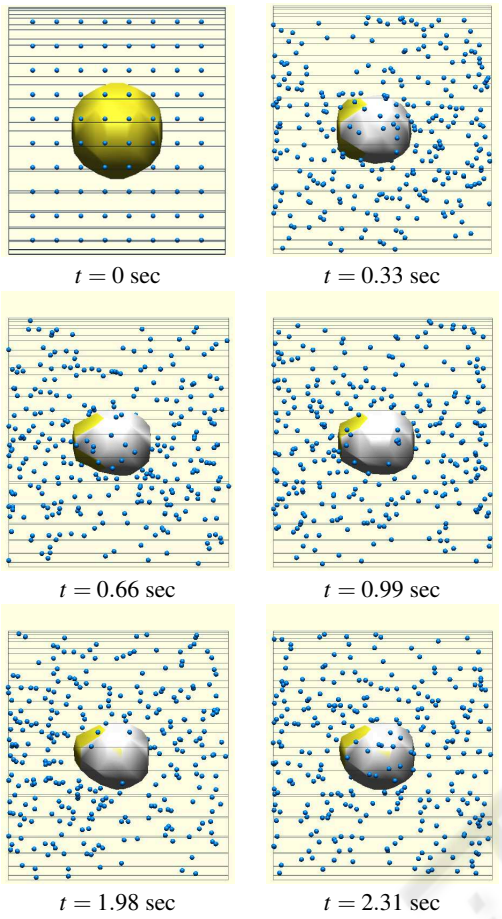


Figure 3: Simulation of a deformable sphere immerse in a fluid contained within a tube.

simulator uses a simplified model of a compliant body that allows fast computations. Our simulator can be used in visualization of a real phenomena like to cut a vein and to show how the blood fluids.

We showed bidimensional simulations of the rupture of a dam and a three-dimensional simulation of a compliant sphere completely immersed in a running incompressible fluid inside a tube. The results simulate qualitative behaviors of the proved systems.

We are convinced that our method can be used as a first approximation, or to show graphically the behavior, of partial immersed bodies on compressible or incompressible fluids, like ships or submarines at the see surface.

Also, the forces produced in immersed complex mechanism, like robot arms attached to a ROV (Remotely Operated Vehicle) can be better understood if more accurate hydrodynamical damping models can be obtained. This method can be a good approach to this end.

Finally, some work has to be done yet to reduce the computational cost when a complex deformable model is used, in order to obtain real time simulations.

ACKNOWLEDGEMENTS

This work has been partially supported by project SEP-CONACyT 80965, México.

REFERENCES

Chen, S., Johnson, D. B., and Raad, P. (1995). Velocity boundary conditions for the simulation of free surface fluid flow. *Journal of Computational Physics*, Volume 116(Issue 2, February):262–276.

chuan Bai, Y., Xu, D., and q. LU, D. (2007). Numerical simulation of two-dimensional dam-break flows in curved channels. *Journal of Hydrodynamics*, Ser. B, Volume 19(6):726–735.

Eddy, P., Natarajan, A., and Dhaniyala, S. (2006). Subisokinetic sampling characteristics of high speed aircraft inlets: A new cfd-based correlation considering inlet geometries. *Journal of Aerosol Science*, Volume 37(Issue 12, December):1853–1870.

Flores, J. R. and de la Fraga, L. (2004). Basic three-dimensional objects constructed with simplex meshes. In *First International Conference on Electrical and Electronics Engineering*, pages 166–171. IEEE Pres.

Monaghan, J. and Lattanzio, J. (1985). A refined method for astrophysical problems. *Astron. Astrophysics*, 149:135–143.

Monaghan, J. J. (1994). Simulating free surface flows with sph. *Journal of Computational Physics*, 110:399–406.

Monaghan, J. J. (2005). Smoothed particle hydrodynamics. *Rep. Prog. Phys.*, 68:1703–1759.

Pérea, S. A., González, F. F., and Iglesias, A. S. (2005). Utilización del método sph en la simulación del sloshing. Technical Report No. 822, Universidad de la Rioja.

Streeter, V. L. and Wylie, E. B. (1988). *Fluid Mechanics*. Mc. Graw Hill.

Tai, C. H., Liew, K. M., and Zhao, Y. (2007). Numerical simulation of 3d fluid-structure interaction flow using an immersed object method with overlapping grids. *Computers & Structures*, Volume 85(Issues 11-14, June-July):749–762.

Trejo, C. R. and de la Fraga, L. G. (2005). Animation of deformable objects built with simplex meshes. In *Second International Conference on Electrical and Electronics Engineering*, pages 36–39. IEEE Pres.

van den Bergen, G. (2004). *Collision Detection in Interactive 3D Environments*. Morgan Kaufmann.