Impulse-based rigid body interaction in SPH

By Seungtaik Oh*, Younghee Kim and Byung-Seok Roh

A new boundary force is proposed to solve rigid body interaction in SPH (Smoothed Particle Hydrodynamics). The new boundary force is impulse-based rather than potential-based. The impulse is converted to a boundary force with a fundamental observation that the impulse is a momentum change which is a result of a force acting on an object during a time interval. Our new impulse-based boundary force gives more stable and accurate results in all kinds of rigid body interactions, especially in rigid–rigid interaction since our boundary force faithfully reflects the dynamic features of rigid body. With our impulse-based boundary force, SPH fluid solver is capable of full two way interaction of fluid and rigid body and it turns to be a particle-based rigid body solver if only rigid bodies are simulated without fluid. Copyright © 2009 John Wiley & Sons, Ltd.

Received: 23 March 2009; Accepted: 24 March 2009

KEY WORDS: SPH; fluid; rigid body; boundary force; impulse

Introduction

Rigid body interaction is one of the most important issues in fluid simulations. Boundary condition describes the fluid motion at contact interface between fluid and a rigid body. It is common to consider two boundary conditions called no-stick and no-slip conditions. In no-slip condition fluid has the same velocity as that of rigid body at contact interface and in no-stick condition fluid can move freely in the tangential direction at contact interface. Both boundary conditions are for fluid motion only in some sense and are not telling how the rigid body moves properly interacting with fluid. If the rigid body is fixed, there will be no problem. Most situations in computer graphics, however, are required to be capable of two-way coupling of fluid and rigid body to increase realism.

SPH (Smoothed Particle Hydrodynamics) is a particle-based Lagrangian method to simulate fluids. It was originally invented for astrophysics and it has been successfully applied for free surface problems since. Reference [2] SPH was introduced to computer graphics community in Reference [3] to simulate deformable bodies. Now it became an alternative for the grid-based Eulerian approach in fluid simulations. SPH has some unique features. Most of all, the fluid momentum is well preserved since it is a Lagrangian method where every information is carried by moving particles, and so fine details such as splash and spray which might disappear due to grid resolution and numerical dissipation in Eulerian approaches, are well captured in SPH. The standard SPH simulates slightly compressible fluid for incompressible fluid and the pressure is given by a function of density, that is, an equation of state, instead by solving Poisson equation. Another interesting thing is that it is easy to couple SPH fluids to other physics such as rigid dynamics.

Boundary force scheme suggested by Monaghan is a popular method to solve interactions of fluid and rigid body in SPH. Monaghan made use of a repulsive force based on Lennard–Jones potential to prevent a penetration of fluid into a rigid body. Later, the repulsive force has improved to a more stable form in Reference [4]. Such a repulsive force is called boundary force and especially the repulsive force used by Monaghan is called Monaghan boundary force (MBF) here. Easy implementation and overall effectiveness are main advantages of the boundary force scheme. The MBF
depends only on the distance to rigid body because it is designed to prevent a penetration of a particle with the maximum speed in simulation. It is worth noting MBF doesn’t respect well the dynamic properties or behaviors of rigid body and so bouncing velocity of a colliding particle is hard to predict, which makes it difficult to accomplish the natural boundary conditions mentioned above. If MBF is applied to an interaction between rigid bodies, it gets harder to accomplish a proper collision solving.

Our aim in this paper is to propose a new boundary force based on the impulse of rigid bodies, which is called impulse-based boundary force (IBF). The idea of IBF is that every interaction where a rigid body is involved can be understood as a collision between a particle and a rigid body or rigid bodies. Basically our IBF enjoys the full advantages of boundary force scheme as MBF does and it works well for all rigid body interactions, that is, fluid–rigid and rigid–rigid. Especially in rigid–rigid interactions, it gives more stable and reasonable results than MBF by calculating the exact impact force at collisions. So, our IBF-based SPH solver may turn to be a particle-based rigid body solver if only rigid bodies are simulated without fluids.

Our SPH-based rigid body simulation has some advantages in comparison to the conventional methods described in References [5–9]. One of the biggest advantage is simplicity. The conventional methods often need highly sophisticated algorithms for plausible collision and contact solving, which are the most important issues in rigid body simulations. However, in our SPH-based approach, no such sophisticated treatments are needed and both of the issues can be handled in a simpler and unified way. The other advantage is that one can easily accomplish a smooth integration of rigid dynamics to fluids simply by applying a proper boundary force between fluid and rigid body.

Related Works

Physically-based fluid simulation has been one of the most attractive areas in computer graphics for recent years. To our knowledge, Reference [10] is the first work in computer graphics to make use of numerical solutions of full 3D incompressible Navier-Stokes equation for fluids animation. Later, Reference [11] introduced a semi-Lagrangian advection scheme for stable and longer time stepping, and the most current grid-based solvers are based on this together with the level set based fluid surface tracking.12,13 Adaptive spatial partitioning based on octree structure was proposed for effective simulations in Reference [14] and two-phase simulation of water and air was studied to simulate realistic bubbles in liquids in Reference [15].

While the previously mentioned works are all in grid-based Eulerian approach, particle-based Lagrangian methods to simulate fluids have been studied as well. SPH models were considered to simulate water in an interactive rate in Reference [16] and incompressible fluid simulation based on the moving particle semi-implicit method was studied in Reference [17]. A dynamic variable resolution and smooth surface extraction technique for SPH were proposed in Reference [18] and realistic small scale effects such as bubbles and foam were accomplished in the SPH framework in Reference [19]. The standard SPH formulation is applied for free surface flows with natural attraction-based surface tension algorithm in Reference [20]. Some hybrid approaches to combination of strengths of SPH and Eulerian grid methods are tried more recently in References [21,22]. Porous flow physics is integrated in SPH to simulate fluids and deformable objects in Reference [23].

A typical approach to coupling fluid and solid in the Eulerian methods is to apply the solid velocity as fluid boundary condition and to integrate pressure forces on the solid surface. Reference [24] introduced a two-way coupling scheme handling a rigid body as like a highly viscous fluid and Reference [25] accomplished a fluid-thin shell solid coupling. Reference [26] achieved a full simultaneous two-way coupling between Eulerian fluids and rigid bodies. A variational framework to obtain noise-free pressure solves on the interface of arbitrarily shaped boundary is introduced in Reference [27]. A purely pressure-based solid-fluid coupling with momentum conservation is proposed in a fully Eulerian fashion for both fluid and solid in Reference [28].

Due to its Lagrangian nature, particle-based fluids can handle rigid body interaction much more easily than the Eulerian grid fluids. Rigid body interaction is generally solved by forces between particles. Monaghan2,4 modeled and developed a repulsive force with the Lennard–Jones potential. Reference [29] also used the Lennard–Jones potential to model both repulsion and adhesion to contact surface. A unified approach to simultaneous coupling of fluid and solid was proposed in References [30,31]. A penalty force-based approach introduced in Reference [32] was applied for collision handling in porous flow modelling in Reference [23].
Rigid body simulation has a long history and lots of literature in computer graphics. Only limited works are addressed here. Reference [5] proposed an impulse-based rigid body simulation with back up in time and Reference [6] presented an analytic method to solve contact problems. Reference [8] modeled both contact and collision in a unified way. Reference [9] achieved huge stacking simulations of non-convex rigid bodies with some sophisticated processes in collision and contact. Reference [32] shows granular material simulations such as sand as well as rigid body simulation in particle-based approach. In contrast to the conventional grid-based approach, there are few works on particle-based rigid body simulation.

**Boundary Force Scheme**

We follow the standard SPH formulations in Reference [2] to simulate fluids. So, the density $\rho$ of fluid is determined by the continuity equation and the pressure $p$ is given by a function of density, that is, $p(\rho) = B(\rho/\rho_0)^\gamma - 1$ where $\gamma = 7$. This kind of SPH simulates slightly compressible fluid for incompressible fluid. It is known that the volume variation is determined by the ratio of the maximum speed to the sound speed, more precisely, $d\rho/\rho \approx (v_{\text{max}}/c_s)^2$, where $c_s$ is the sound speed. In our all SPH simulations here, we set $v_{\text{max}} = 0.1c_s$ and have a volume variation less than 0.01. Denote the smoothing length by $h$ and the time stepping is given by $\Delta t = 0.5h/c_s$.

Boundary force scheme needs a particle representation of a rigid body and so boundary particles are generated on the surface of rigid body. The spacing between particles is set to the fluid particle spacing. A particle representation can be obtained by an almost uniform mesh representation in practice, where each edge length is close to a constant. Rigid body particles are managed by a proper neighbor searching algorithm based on a uniform grid or K-d tree together with the fluid particles. To solve rigid body interactions, a repulsive force or boundary force is applied to each pair of particles which are close enough and meet some additional conditions as author's end. If a fluid particle $a$ exerts on a boundary particle $b$ a boundary force $f_{ba}$, then the reaction force $f_{ab} = -f_{ba}$ is also acted on the fluid particle $a$ by the law of reciprocal actions. Of course, we have the same for a rigid–rigid particle pair. In boundary force scheme, the linear and angular momentum of the entire system are conserved in the absence of an external force and an interaction of fluid to a fixed rigid body.

**Monaghan Boundary Force (MBF)**

MBF is now explained in detail. Suppose that a particle $a$ is interacting with a rigid particle $b$. Let $n$ be the normal at $b$. Denote by $x$ and $y$ the tangential and the vertical coordinates, respectively, of $a$ with respect to $b$ and $n$. Then MBF $F_M$ is given by

$$F_M(x, y) = \frac{0.02c_s^2}{y} \frac{m_am_b}{m_a + m_b} \Gamma(y)\chi(x)n$$  \hspace{1cm} (1)

In Reference [4], where $m_i$ is the mass of particle and for $q = y/h$

$$\Gamma(y) = \begin{cases} \frac{2}{3} & : 0 < q < \frac{2}{3} \\ 2q - \frac{3}{2}q^2 & : \frac{2}{3} < q < 1 \\ \frac{1}{2}(2 - q)^2 & : 1 < q < 2 \\ 0 & : \text{otherwise} \end{cases}$$  \hspace{1cm} (2)

and for the particle spacing $\Delta \rho$

$$\chi(x) = \begin{cases} 1 - x/\Delta \rho & : 0 < x < \Delta \rho \\ 0 & : \text{otherwise} \end{cases}$$  \hspace{1cm} (3)

In Equation (1), the term $\frac{0.02c_s^2}{y} \frac{m_am_b}{m_a + m_b}$ is the main part of MBF called here magnitude of MBF $\|F_M\|$, and $\Gamma(y)$ and $\chi(x)$ can be understood as scaling factors along the two coordinate directions. $\Gamma(y)$ comes from the gradient of the smoothing kernel, and $\chi(x)$ gives proper weight factors for multiple near rigid particles so that a particle moving along the tangential direction experiences a constant boundary force. From a simple observation on $\|F_M\|$ it is seen that MBF is designed to prevent a penetration of a particle with the maximum velocity into a rigid body, and so for a slower particle, MBF might be too strong since it depends only upon the distance to rigid body. In addition, the resulting velocity of a particle is not predictable after the boundary force is applied. This is a potential instability in simulation and the problem could be more serious in rigid–rigid interaction.
Impulse-based Boundary Force (IBF)

Derivation of IBF

In this section we propose a new boundary force to overcome the defects of MBF mentioned in the last section. To begin with, we recall the impulse equation at a collision of two rigid bodies.\(^5,34\) Suppose that two rigid bodies \(R_A\) and \(R_B\) are about to collide with each other at a point \(p\). Put \(r_i = p - c_i\) for the center of mass \(c_i\). Let \(v_i\) be the velocity at \(p\) for each rigid body and denote the relative velocity by \(v_{AB} = v_A - v_B\). The collision normal at \(p\) of \(R_B\) is denoted by \(n\). Then the impulse \(J = j n\) is given by

\[
j = -\left(1 + e\right) v_{AB} \cdot n \sum_{i=A,B} \left(M_i^{-1} + (I_i^{-1}(r_i \times n) \times r_i) \cdot n\right)
\]  

(4)

where \(e\) is the coefficient of restitution, \(M_i\) is the total mass, \(I_i\) is the moment of inertia. We may assume that the impulse is the result of a constant impact force in a time interval \(\Delta t\). In computations, \(\Delta t\) would be the simulation time step. Then the impact forces \(F_A\) and \(F_B\) exerted on \(R_A\) and \(R_B\) are given by \(F_A = \frac{j}{M_A}\) and \(F_B = -F_A\), respectively. This impact force changes the linear and angular momentums of rigid body so that the normal component of \(v_{AB}\) at the next time step becomes \(-e(v_{AB} \cdot n)\). Our natural idea is to make use of the impact force \(F_i\) as a new boundary force. A fluid–rigid interaction is interpreted as a collision of a point mass and a rigid body, and the impulse is simplified by setting \(I_A^{-1} = 0\). When the rigid body is static, the impulse for fluid–rigid can be further simplified as \(j = -(1 + e)M_A v_{AB} \cdot n\). When \(v_{AB} = v_A\), in case of rigid–rigid interaction, the same impulse as Equation (4) is used, of course. Now we define the primitive IBF \(F_i\) by the above impact force.

In fluid–rigid interaction, there might be a penetration of fluid into a rigid body with the primitive IBF \(F_i\) only. This is because \(F_i\) is not strong enough to overcome the fluid pressure. So an auxiliary force \(F_{aux}\) is considered and it should be the sum of all forces, including pressure gradient, acting on fluid particle not by boundary interaction. In our implementation, we take the auxiliary force computed in the previous time step for the current time step since we are not able to know the exact resulting sum in the loop of particle interaction computation. Adding the tangential scaling \(\chi(x)\) (Equation (3)), we complete the IBF finally as follows:

\[
F_{f-r} = \chi(x)(F_i + F_{aux})
\]  

(5)

Note that \(\chi(x)\) is quite essential in the sense that it provides a sufficient condition for angular momentum conservation.\(^35\) Water pouring result with the complete IBF for fluid–rigid interaction is shown in Figure 1.
In rigid–rigid interaction, we complete our IBF by multiplying both of the scalings on \( F_I \), that is, vertical and tangential scalings (Equations (2) and (3)). The vertical scaling reduces the speed of approaching object at impact by an amount. This increases the stability of simulation in relation to multiple points collision that will be mentioned in the next section. The complete IBF for rigid–rigid is given by \( F_{I,r} = \Gamma(y)\chi(x)F_I \).

For any particle pair \((p, q)\) of fluid–rigid or rigid–rigid, IBF is applied as an external force if the two particles are close enough and approaching to each other, more precisely, \(\| p - q \| < 2h \) and \( v_{AB} \cdot n < 0 \). Damping-based kinetic friction \( F_{fr} \) is also applied for non-vanishing tangential component of \( v_{AB} : F_{fr} = -\mu v_{AB} \) where \( v_{AB} = v_{AB} - (v_{AB} \cdot n)n \). The kinetic friction can arise for any pair of particles close enough, that is, \( d < 2h \) although they are not approaching to each other.

### Collision Normal Determination

Normal determination is important in solving rigid body interaction in SPH. A rigid body may have a corner where there exists a big normal change. Reference [4] adopted a variable normal setup at a corner of rigid body (Figure 2(a)). The authors choose a normal by the radial vector in a conical section of a corner so that the normal varies smoothly with respect to outside points. This method can be extended to the 3D case, but it would be too expensive. For 3D simulations, the following normal setup is simple and practical: Let \( n_i \) be the normal of a face attached to a corner and let \( r \) be the relative position of fluid particle to the corner. We also denote by \( \bar{n} \) an average normal of the corner, called the default normal. Then a proper normal is given by

\[
\tilde{n} = \frac{\sum w_i n_i + \bar{w}\bar{n}}{\| \sum w_i n_i + \bar{w}\bar{n} \|}
\]

where the weight \( w_i \) is \( n_i \cdot r \) if \( n_i \cdot r > 0 \) or zero otherwise. This is very effective to fluid–rigid interaction with a sharp corner.

A different consideration on proper normal determination is required for rigid–rigid interactions. There exist three cases according to whether the point of collision is a corner or not. A point which is not a corner is called regular. The first case is that the collision point is regular in each rigid body (Figure 3(a)). Here, we may assume that the two rigid bodies meet tangentially, and so either normal can be chosen. The second case is that the collision point is a corner in each rigid body (Figure 3(b)). We again use either default normal for this case. The last case is that the collision point is regular in one rigid body but a corner in the other rigid body (Figure 3(c)). For this case, we choose the normal from the regular point. Note that the variable normal setup doesn’t need to be applied to rigid–rigid case. Distinction of corner points and default normals are enough to choose a proper normal. This normal determination makes it possible for us to have more exact impulse direction and accordingly more realistic results in rigid–rigid interactions.

In order to simulate thin shell rigid bodies, some extra consideration is needed. A thin shell rigid body might have a 1-dimensional boundary, for instance a boundary edge. Obviously, there exists a big normal change along the perpendicular direction of a boundary edge. This gives rise to a similar variable normal setup as the previous corner case. To do this, we need some extra normals on a boundary edge. Given a boundary edge, one can define the unique normal on the edge, called edge normal, which is parallel to the polygon containing the edge (see Figure 2(b)). The extra normals at a boundary edge particle come from the edge normals. At a boundary edge particle, the average of the extra normals becomes the default normal, and the extra normals and the existing polygon normals are considered in the variable normal setup just as the corner case at a boundary edge particle. Now we assume that
a corner includes a boundary edge vertex or point. A two ring simulation of thin-shell type is shown in Figure 4.

**Additional Issues**

Generally, rigid bodies collide at multiple points at the same time. While a single point collision leads to a simple algebraic equation, multiple points collision gives us a complex linear system since the velocity change at a collision point is determined by the net force and the net torque which are the accumulated sum of the impact force and the associated torque at all collision points. Unfortunately, it is not possible to implement this in our solver because the impulse at individual collision point cannot be identified before solving a coupled linear system invoked by all collision points. From this reason our method treats a case of multiple points collision as several cases of single point collision ignoring the effects from other collision points. With this treatment, one may face an undesirable situation where too strong force is generated. The situations can be avoided by dividing the resulting impulse by the number of multiple collision points as described in Reference [37]. Although the method doesn’t follow the correct physics, the results are acceptable and physically-feasible enough in all simulations we have done. Most conventional collision solving algorithms try to avoid multiple collision points. For example, the deepest point of penetration is considered and going back to the previous time step, a single point collision for the point is solved. This might change the collision state and so need several iterations until no collision is detected.

There is no distinction between collision and contact in our method. In other words the same impulse with the same coefficient of restitution as collision is used for contact solving and there is no special treatment to solve contacts. In velocity integration, external forces including gravitational force are applied before computing the boundary forces. Thus an object with no motion can experience a boundary force as a contact force and stays in rest when the external force and the boundary force are balanced. This balance is possible because our boundary force has the vertical scaling factor increasing gradually as the distance to boundary gets shorter. The conventional methods are taking a number of special treatments to properly solve contacts. Some authors construct a contact graph identifying which bodies are resting on the top of others and apply the contact process on one by one from the bottom. To avoid sinking in a stacking example, shock propagation is taken in Reference [9]. The shock propagation forces the mass of object to be set to infinity. As mentioned, our method does not need these kind of special consideration. We have obtained plausible results with our simple approach for examples in which the conventional methods require such auxiliary processes (Figure 5).

**Simulation Results and Discussions**

All simulations are performed on Intel Core2 Quad 2.66 GHz. We set \( c_s = 100 \) and \( v_{\text{max}} = 10 \) for 1% density variation. In fluid–rigid interaction, the coefficient of restitution is set to zero for no-stick condition and in rigid–rigid interaction, it is set to 0.2–0.5. For sophisticated rigid–rigid interactions, dominoes and a large number of falling glasses of non-convex type are simulated (Figures 6 and 7). The results prove that our particle-based unified approach to collision and contact solving is really effective. We also present an example of sophisticated fluid–rigid interaction (Figure 8) by filling...
water in the tank in the falling glasses example. An initial fluid setting may be useful in some cases and our IBF gives more stable results than MBF for the initial fluid setting (Figure 8). With MBF, some undesirable splashes on boundaries are detected. It is also interesting to see the natural buoyancy effect due to density difference in the example. A timing information is given in Table 1. In the dominoes example, we use an efficient rigid particle generation to speed up the simulation by eliminating rigid particles on the parts, i.e. the bottom and the steps, where no interaction occurs. Rigid body only simulation takes a shorter time than a simulation fluid is involved in since a fluid particle has more neighbors and the expensive Navier-Stokes equations
Figure 8. Wooden glasses falling into water: fluid and glasses are coupled in a fully two-way manner. The water is created by an initial particle setup, which is more stable with our impulse-based boundary force.

We have proposed a new boundary force for rigid body interaction in SPH fluid simulations. The new boundary force is impulse-based rather than the conventional potential-based. With the new boundary force we have achieved stable and plausible results for simulations that complex rigid body interactions are involved in. This particle-based approach provides a simple and unified solution to two-way coupling of fluid and rigid bodies. Although our results are all for rigid bodies, we think that this new boundary force can be extended to all kinds of solids including deformable bodies such as rubber and clothes, which will be one of our future works. The stability in a large number of thin-shell object simulations needs to be improved. We are also planning to extend our solver so that it will be capable of extremely large scale simulations based on distributed parallel computing.

Table 1. Timing for presented simulations

<table>
<thead>
<tr>
<th>Simulation</th>
<th>Number of particles</th>
<th>Time for a frame</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glasses into water</td>
<td>350 005</td>
<td>9.08 min</td>
</tr>
<tr>
<td>Water pouring</td>
<td>684 756</td>
<td>17 min</td>
</tr>
<tr>
<td>See-saw</td>
<td>95 044</td>
<td>4–5 sec</td>
</tr>
<tr>
<td>Stacking</td>
<td>22 865</td>
<td>1–2 sec</td>
</tr>
<tr>
<td>Dominoes</td>
<td>42 328</td>
<td>7.2 sec</td>
</tr>
<tr>
<td>Glasses</td>
<td>177 133</td>
<td>62.8 sec</td>
</tr>
</tbody>
</table>

Table 1. Timing for presented simulations

We thank Dr. Paul Cleary for the useful discussions and comments for overall SPH simulations. This work was supported by the R&D program of ETRI (2009: Distributed fluid simulation technology development for massive visual effects).

References


Authors’ biographies:

Seungtaik Oh obtained his PhD in mathematics from Korea Advanced Institute of Science and Technology (KAIST) in 2004. He is currently a senior member of engineering staff of ETRI, Korea. His research interests are physically-based animation such as fluid/rigid body simulation and geometric methods in computer graphics.

Younghhee Kim is a senior researcher at ETRI (Electronics and Telecommunications Research Institute), Korea. She received the BS degree in Mathematics from Pusan National University, in 1995, and the MS and PhD in Computer Science from Kyungpook National University, Korea, in 2001, and 2006, respectively. Her research interests include computer graphics, fluid simulation, and GPU programming.
Byung-Seok Roh is a researcher in Electronics and Telecommunications Research Institute (ETRI), Korea. He received BS degree (2005) and MS degree (2007) in Computer Science from Korea University, Korea. His research interests include computer graphics, computational fluid dynamics, and physically based simulation.