Non-Reflective Boundary Conditions For Incompressible Free Surface Fluids

Andreas Söderström[1] and Ken Museth[1,2,3]

[1]*Linköping University,* [2]*Digital Domain, Inc. and* [3]*DreamWorks Animation, Inc.*
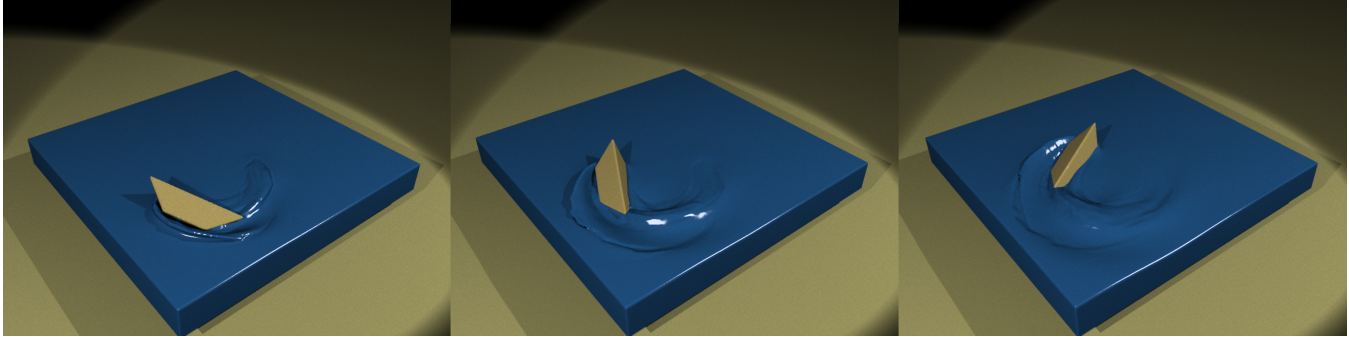
Figure 1: Frames from a free surface fluid animation employing our novel non-reflective (*i.e.* open) boundary conditions. Note how the waves from the moving object (*e.g.* boat) are *not* reflected when they hit the boundary of the fluid simulation domain, thus emulating an open ocean.

Abstract: We have developed a novel approach to open-boundaries for fluid animations. More specifically we present a highly efficient energy absorbing boundary condition for the incompressible Navier-Stokes equations in the prescence of a free surface. Our work extends and adapts a Perfectly Matched Layer (PML) approach [Berenger 1994; Johnson 2007], recently developed for the Navier-Stokes equations, to free surfaces in the context fluid animations. We show how our PML boundary condition is able to effectively eliminate reflections generated by the presence of solid boundaries in the simulation domain, and that our method is far superior to simpler approaches for reducing wave reflection. Furthermore, we have adapted our theoretical PML model to work with the Stable-Fluids Eulerian Navier-Stokes solver commonly used in computer graphics. Finally, we show that the cost of deploying our method in terms of memory and additional computations is small, and for a given quality significantly less than other known methods.

Overview: Imagine a ship moving rapidly across the surface of a wide ocean. The water breaking around its bow as it cuts trough the waves and the powerful swell left in its wake as the massive vessel presses on towards its destination. Now imagine that you want to make an accurate and visually pleasing simulation of this scenario. This requires the use of a high quality simulation engine capable of resolving non-linear behaviour such as breaking waves. Due to the high computational cost the volume of fluid that can be simulated will typically be restricted to a close neighbourhood around the hull of the ship. Walls (*i.e.* boundaries) must then be added in order to contain the water within this region. The issue is now that if the simulation is allowed to run long enough the waves generated by the ship will travel outwards and eventually reflect back towards the ship. The resulting wave pattern (*i.e.* inference) will be far from what is to be expected on an open body of water, thus breaking the illusion of a wide ocean. However, by deploying our energy dampening boundaries at the edge of the simulation domain these wave reflections can be automatically cancelled out.

Making the Finite Appear Infinite: The core of our problem is essentially that we attempt to make a finite simulation domain behave like it is infinitely large. In order to realize this we focus on the most obvious difference between these scenarios - wave reflection from the walls containing our simulated body of water. We want to create boundaries that contain the fluid, but at the same time eliminate the reflection of waves entering the boundary region. We address this

problem by modifying the governing equations of the fluid close to the actual boundaries. More specifically we transform the Navier-Stokes equations in a neighborhood close to the actual walls such that they dampen waves moving trough this boundary region in a given direction. The governing equations in the boundary region become

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{f} + \mu \nabla^2 \mathbf{u} - (\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{\nabla p}{\rho} - \sigma_x \mathbf{A} \mathbf{q}_1 - \sigma_y \mathbf{A} \mathbf{q}_2 - \sigma_z \mathbf{A} \mathbf{q}_3 \quad (1)$$

$$\nabla \mathbf{u} = -\sigma_x \mathbf{B} \mathbf{q}_1 - \sigma_y \mathbf{B} \mathbf{q}_2 - \sigma_z \mathbf{B} \mathbf{q}_3 \quad (2)$$

where we introduce the constants $\mathbf{A}$ and $\mathbf{B}$ together with the auxillary vector fields $\mathbf{q}_1$, $\mathbf{q}_2$ and $\mathbf{q}_3$. These auxillary fields only exist in the boundary region and will act to counter and dampen any waves that enter. As can be seen in equation (2) the auxillary variables also enter into the continuity equation, allowing for sources and sinks in our incompressible medium.

The functions σ_x, σ_y and σ_z describe the magnitude of the dampening and are applied trough a transfer function such that full dampenig is achieved close to the physical walls of the simulation. The dampening is then reduced towards zero as we approach virtual internal boundaries inside the simulation domain.

Performance impact: Our method requires solving a modified Navier-Stokes equation with four vector fields instead of one. The computations on each of these fields are similar to those for solving the regular Navier-Stokes equations. Thus, a conservative upper bound for the computational and memory cost will be four times what is used without our boundaries. However, the additional equations only need to be solved in the boundary region which is typically very small - in our tests only 5 to 10 percent of the width of the simulation domain. Thus the typical impact on memory and time will be significantly less than the conservative case.

References

BERENGER, J.-P. 1994. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics 114*, 2, 185 – 200.

JOHNSON, S. G., 2007. Notes on perfectly matched layers. online MIT course notes (Aug. 2007).