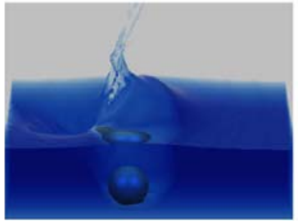
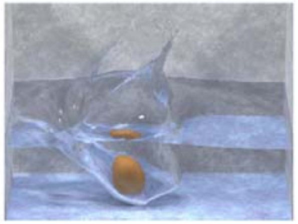
Physically Based Animation of Liquids



- Nick Foster, PDI/Dreamworks
- Ron Fedkiw, Stanford University



- Douglas Enright, Stanford/ILM
- Stephen Marschner, Stanford
- Ron Fedkiw, Stanford/ILM

Goals / Challenges

- A physically plausible liquid surface
- Interaction with surrounding scene, including dynamic objects
- Efficient enough to be practical for real production

The Usual Equations...

- Incompressible Navier-Stokes:

Conservation of momentum

$$\frac{\partial \mathbf{u}}{\partial t} = \underbrace{\nu \nabla \cdot (\nabla \mathbf{u})}_{\text{viscosity}} - \underbrace{(\mathbf{u} \cdot \nabla)\mathbf{u}}_{\text{convection}} - \underbrace{\frac{1}{\rho}\nabla p}_{\text{pressure}} + \mathbf{g}$$

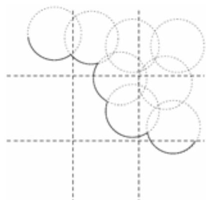
Conservation of mass

$$\nabla \cdot \mathbf{u} = 0$$

$$\nabla = \{\partial / \partial x, \partial / \partial y, \partial / \partial z\}$$

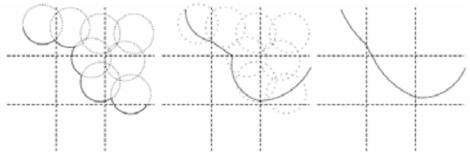
Surface Tracking, Take 1

- Particles:
- Populate computational grid with particles
- Update particle positions at each time step using simple convection, by tri-linear interpolation of grid velocities.
- The particles define an implicit surface



Surface Tracking, Take 1

- Surface smoothing:



- Drawbacks:
  - High particle density is required near the surface, even in well-behaved areas.
  - Some particles do not contribute to the surface.

Surface Tracking, Take 2

- Dynamic level set:
- The liquid-air interface is modeled as an implicit surface, $\phi(x,y,z) = 0$.
  - $\phi < 0$ inside the liquid
  - $\phi > 0$ outside the liquid
- The implicit surface evolves according to

$$\frac{\partial \phi}{\partial t} = -\mathbf{u} \cdot \nabla \phi$$

Surface Tracking, Take 2

- Drawbacks:
  - Too smooth (on coarse computational grids), so unable to resolve fine features.
  - Loss of volume occurs.

Hybrid Surface Model

- Use the level set *together* with particles!
- At each time step, both the level set and the particles are updated.
- Delete particles away from the surface.
- Introduce more particles near the surface, where necessary.
- **Summary:** overall smooth surface tracked using level set, detail is added using particles.

Hybrid Surface Model

- For each particle near the surface estimate curvature as: $\kappa = \nabla \cdot \left(\nabla \phi / \|\nabla \phi\| \right)$
- Low curvature (smooth region): ignore particles, use only level set.
- High curvature (detail region): allow particles to modify the value of ϕ.
- Treat "escaped" particles as individual liquid drops.

Boundary Conditions

- Non-liquid cells:
  - Empty cells: air dynamics is ignored.
    - The pressure inside an empty cell is atmospheric
    - The velocity between empty cells is zero.
  - Object cells.
- Liquid surface cells:
  - Must make sure that the velocities satisfy the incompressibility equation $\nabla \cdot \mathbf{u} = 0$

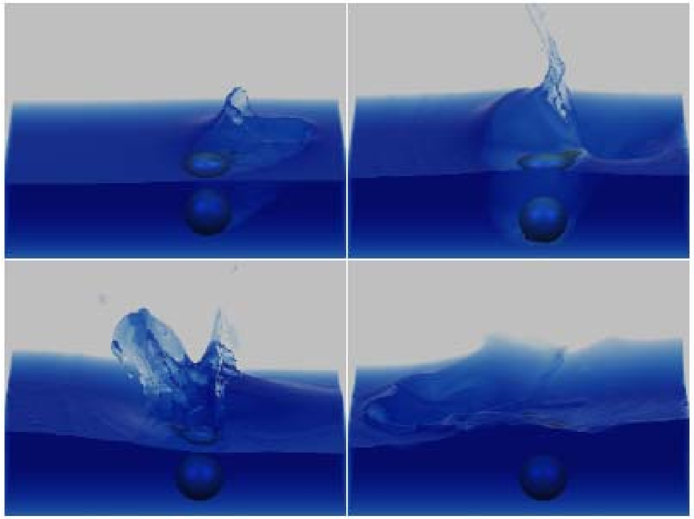
Moving Objects

- Liquid cannot flow into the object: relative normal velocity between liquid and object should be >= 0.
- Tangential velocity is unconstrained.
- Neither particles, nor the level set should interpenetrate the object's surface.

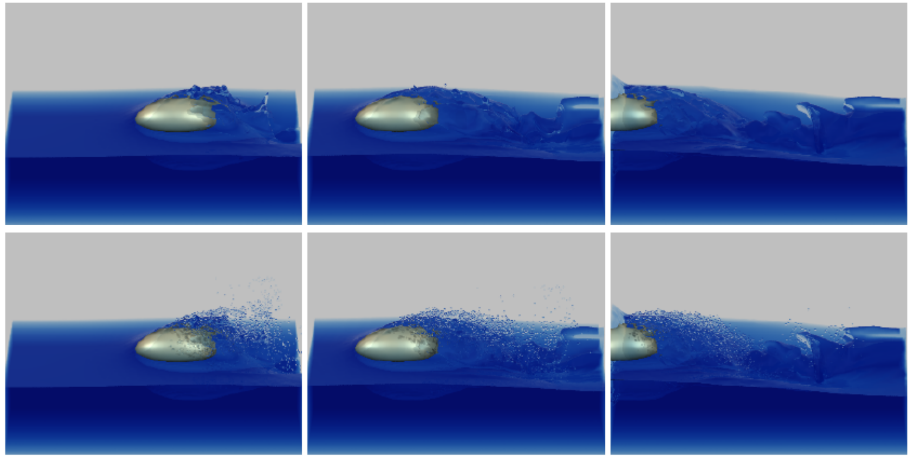
Moving Objects

- A cell inside a moving object has its velocities set to the velocity of the object.
- Simulation step is performed ignoring moving objects.
- In each cell intersected by object surface the normal velocity is modified to make sure no liquid flows inside the object.
- Velocity of cells inside the object are restored.

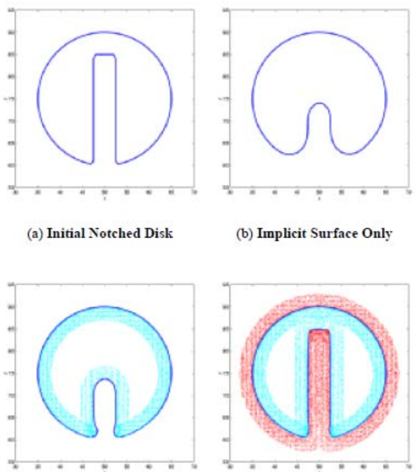
Results



Results



Problem Solved?



(a) Initial Notched Disk (b) Implicit Surface Only

(c) Particles Inside Only (d) Our New Method

Particle Level Set Method

- Two sets of marker particles: positive (outside) and negative (inside).
- Particles placed only in the vicinity of the surface (and re-seeded once in a while).
- Each particle has a dynamically changing radius r_p:

$$r_p = \begin{cases} r_{\max} & \text{if } s_p\phi(x_p) > r_{\max} \\ s_p\phi(x_p) & \text{if } r_{\min} \le s_p\phi(x_p) \le r_{\max} \\ r_{\min} & \text{if } s_p\phi(x_p) \le r_{\min} \end{cases}$$

Particle Level Set Method

- In each time step, update both particles and the level set.
- Role of particles is to detect errors in the level set: when a particle is on the wrong side of the interface (by more than its radius).
- Escaped particles are used to adjust the values of the level set function ϕ.

Error Correction

- Each particle is associated with an implicit function of its own:

$$\phi_p(\vec{x}) = s_p\left(r_p - \left|\vec{x} - \vec{x}_p\right|\right)$$

- For each escaped particle, the value of ϕ at each corner of the containing cell may be corrected. For example, for a negative particle p:

$$\phi^{\text{corrected}} = \min\left\{\left|\phi\right|, \left|\phi_p\right|\right\}$$

Results