An Effects Recipe for Rolling a Dough, Cracking an Egg and Pouring a Sauce

Tolga G. Goktekin   Jon Reisch   Darwyn Peachey   Apurva Shah

Pixar Technical Memo #07-06
Pixar Animation Studios

Creating the digital effects for cooking in *Ratatouille* posed a number of unique challanges. First we had to adopt efficient methods for simulating a wide variety of material behaviours. Second we needed to direct our simulations in order to match the expressiveness of the character's animation, e.g. forming specific shapes while the character pounds a dough. Finally we had to apply shading to our simulated surfaces which underwent complex deformations and topological changes. In this sketch we will focus on materials ranging from elastoplastic solids to viscous liquids and illustrate with several shot examples from the film.

Figure 1: Dough rolling effect (top) and several stages of the egg during the cracking (bottom). ⓒDisney / Pixar. All rights reserved.

1 Directable Softbodies

In order to simulate a wide range of materials we had to use different techniques for different parts of the solid to liquids spectrum. Materials that did not undergo any topological changes such as the dough and the egg were treated as deformable solids. There are various methods for simulating deformable solids ranging from mass spring damper (MSD) systems to finite element methods. We chose MSD systems for their simplicity, efficiency and relative ease of integrating them into our existing pipeline. The main difference in our setup compared to standard Maya softbodies is that we first create a Delaunay tetrahedralization of our input surface and convert the internal nodes and edges into internal particles and springs respectively. This creates an internal structure for the softbody, resulting in an unbiased set of springs that cover the volume efficiently. In the case of two different MSD systems interacting such as the egg white and yolk we created extra springs between the interior and exterior surface particles of the objects.

Creating the dough effect presented two main challanges. First the dough had to collide physically with the characters hand and the rolling pin, leaving natural looking dents and bumps. Moreover the dough had to flatten under and buldge in front of the rolling pin. Second the dough needed to form specific shapes at specific times based on creative direction.

Interaction with prescribed animation and controlling the shape of the dough turned out to be conflicting requirements. Changing the surface directly from one shape to the other using goal forces looked too controlled. It also made accounting for the interaction with the character's hand very difficult to achieve since the collision response forces were being overwhelmed by the stiff goal forces. On the other hand relying purely on the simulation did get the interaction right but the results didn't go through any of the shapes we were creatively after. In order to solve these problems we only directed the internal particles of the dough to match the "internal" goal shapes. This made the surface move freely (still connected to internal nodes with springs) and respond to collisions and changes to the momentum naturally; and thus produced believable interaction with colliders and other secondary motions. Plastic behaviour is achieved by resetting the rest configuration of the dough to the goal shapes as it went through them.

We didn't have to specifically direct the shapes during the egg crack. However we had to constrain the egg white at several points and animate the constraints to encourage the natural draping behaviour.

2 Viscous Fluids

For viscous liquids such as our sauces and soups we chose a grid based Eulerian fluid simulator with a levelset surface tracking or a particle based fluid simulator. To control the timing, the amount and the shape of our liquid simulations we animateed the gravity, added sources and sinks and used lattice deformers. We also used extra Maya particles (either as blobs or as input to our surface extractor) to augment the regions where we ran out of resolution.

The shading applied to our sauces and soups had multiple components such as depth tinting, translucency, subsurface scattering and layered textures. In order to make these work we needed a parametrization of our surfaces that was spatially and temporally coherent. However the simulated surfaces were not consistent between frames and were undergoing topological changes. These made applying any shaders that relied on a consistent parametrization over the surface problematic. To solve this problem we first initialized a set of particles in a band around the surface on a "reference" frame and stored the initial point (u,v,w) of each particle as an attribute. Then we advected these particles with the simulated velocity field. Finally as a post-process, at each frame we tagged the vertices with these (u,v,w) values using the method in [Shen et al. 2007]. We used this per vertex (u,v,w) value as a reference point to the inital surface during shading.

In some of our chunkier sauces we needed to add suspended bits that flowed with the sauce. Simulating these bits directly with the liquid as colliders proved to be inefficient due to the their amount and their relatively small size. Therefore we added them as advected instanced geometry particles as a post-process.

Figure 2: Various sauce effects in *Ratatouille*. ⓒDisney / Pixar. All rights reserved.

References

SHEN, C., AND SHAH, A. 2007. Extracting and parametrizing temporally coherent surfaces from particles. *Submitted to Siggraph 2007 Technical Sketches.*