Dynamic Deformation Textures

Department of Computer Science University of North Carolina at Chapel Hill April 2007

The Challenge

Dynamic contact scenarios arise when objects with rich surface geometry are rubbed against each other while they bounce, roll or slide through the scene, as shown in Fig. 1. We present an efficient algorithm for simulating contacts between deformable bodies with high-resolution surface geometry using *dynamic deformation textures*, which reformulate the 3D elastoplastic deformation and collision handling on a 2D parametric atlas to reduce the extremely high number of degrees of freedom with large contact areas and high-resolution geometry.

We simulate real-world deformable solids that can be modeled as a rigid core covered by a layer of deformable material. Examples include human bodies, furniture, tires, toys, etc. We have developed novel and efficient solutions for physically-based simulation of dynamic deformations, as well as for collision detection and robust contact response, by exploiting the layered representation of the models and decoupling the degrees of freedom between the core and the deformation layers.

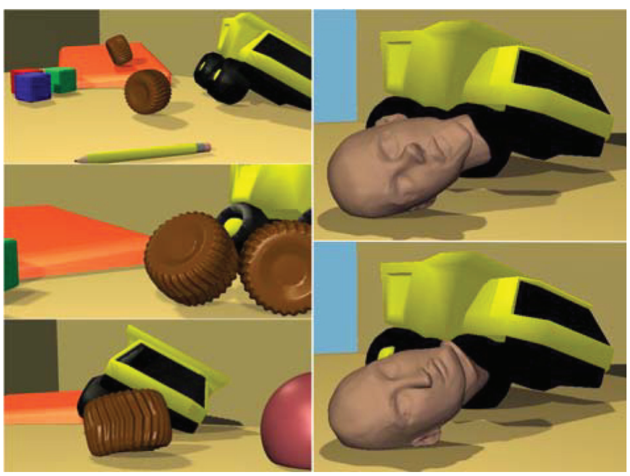

Figure 1: Soft Object Interaction in a Dynamic Scene. Deformable objects roll and collide in the playground.

Approach

Our mathematical formulation of dynamic simulation and contact processing, along with the use of dynamic deformation textures, is especially well suited for realization on commodity SIMD or parallel architectures, such as graphics processing units (GPU), Cell, and physics processing units (PPU). We map our algorithm concepts to the GPU as shown in the highlighted box.

The implicit formulation of the dynamic motion equations and collision response yields linear systems of equations with dense coupling between the core and elastic velocities. We were able to formulate the velocity update and collision response in a highly parallelizable manner, by exploiting our layered representation as shown in Fig. 2. We model real-world deformable solids as a rigid core covered by a layer of deformable material. The deformation in the outer layer is mapped to a 2D domain, which we call *Dynamic Deformation Textures*. This representation maps naturally to graphics memory textures. The updates of elastic displacements and velocities are executed by performing GPU shader operations. The updates of core velocities, on the other hand, are executed in the CPU after gathering intermediate computations performed in parallel on all nodes. Note that the communication between CPU and GPU is reduced in our algorithm.

Concepts mapped to GPU

- **Dynamic Deformation Textures: Reformulate 3-dimensional elasto-plastic deformations and collision processing on 2-dimensional dynamic deformation textures.**
- **Two-stage collision detection algorithm for parameterized layered deformable models; image-based collision detection implemented on the GPU.**
- **Parallelized constraint-based contact response method, exploited by GPU-based implementation.**

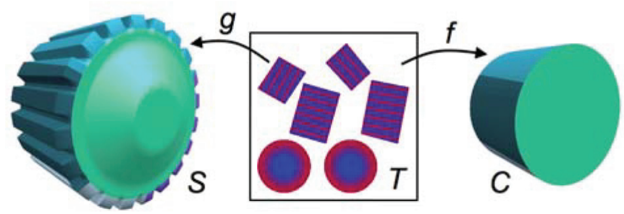

Figure 2: Deformable Object Representation. Deformable surface *S* (52K triangles) and core *C* (252 triangles) of a gear, showing the patch partitioning of its parameterization. The dynamic deformation texture *T* (256×256) stores the displacement field values on the surface. The gear contains 28K simulation nodes on the surface and 161K tetrahedra, allowing the simulation of highly detailed deformations.

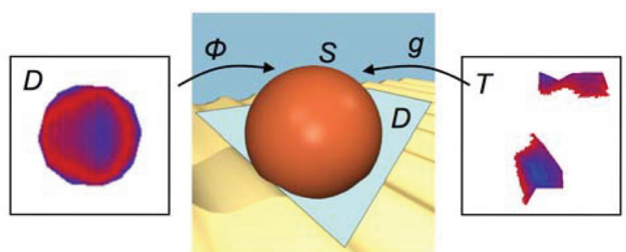

Figure 3: Texture-Based Collision Detection Process. Center: A sphere *S* collides with a textured terrain. Left: Contact plane *D* for texture-based collision detection. The contact plane *D* shows the penetration depth. Right: Dynamic deformation texture *T*.

We also exploit *image-based collision detection* on the GPU as shown in Fig. 3. The computations of per-texel penetration depth and contact normal are performed by orthonormal projection of the low-resolution core geometry onto the contact plane *D*, and by using texture mapping to map the positions of the high-resolution surfaces to *D*. Finally, we project the contact information from the contact plane *D* back to the dynamic deformation texture *T* for contact response.

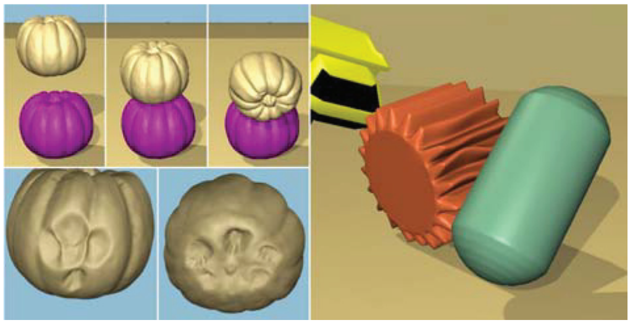

Figure 4: Rich Deformation of Detailed Geometry. Bottom-left corner: views from below the upper pumpkin as it collides with the bottom pumpkin and deforms.

Results

In Fig. 1, we show a scene where deformable tires with high-resolution features on their surfaces roll, bounce, and collide with each other. This simulation consists of 324 K tetrahedra and 62K surface simulation nodes. Such high resolution enables the simulation of rich deformations and robust contact handling. We were able to simulate this scene, processing over 15,000 contacts per second, on a 3.4 GHz P4 with NVidia GeForce 7800.

Our approach is considerably faster than other methods that enable large time steps, such as those that focus on the surface deformation and corotational methods that compute deformations within the entire volume, with more stable collision response. Our approach can also handle many more contact points than novel quasi-rigid dynamics algorithms using LCP, while producing richer deformations, between moving objects (Fig. 4).

Project Leader

Ming C. Lin, Professor

Team Members

Nico Galoppo, Graduate Research Assistant
Paul Mecklenburg, Graduate Research Assistant

Other Collaborators

Miguel Otaduy, Postdoctoral Researcher, ETH
Markus Gross, Professor, ETH

Research Sponsors

National Science Foundation
Defense Advanced Research Projects Agency
Research, Development & Engineering Command
U.S. Army Research Office
Intel Corporation
NCCR CoMe of the Swiss NSF

Selected Publications

N. Galoppo, M.A. Otaduy, P. Mecklenburg, M. Gross and M.C. Lin. "Fast Simulation of Deformable Models in Contact Using Dynamic Deformation Textures," *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2006.

N. Galoppo, M.A. Otaduy, P. Mecklenburg, M. Gross and M.C. Lin. "Fast Simulation of Detailed Layered Deformable Objects in Contact," *ACM SIGGRAPH 2006 Sketches and Applications*, Boston, MA.

N. Galoppo, M.A. Otaduy, P. Mecklenburg, M. Gross and M.C. Lin. "Accelerated Proximity Queries for Haptic Rendering of Deformable Models," *World Haptics Conference 2007*, Tsukuba, Japan.