Accelerated Proximity Queries for Haptic Rendering of Deformable Models

Nico Galoppo* Miguel A. Otaduy† Paul Mecklenburg* Markus Gross† Ming C. Lin*
(*) University of North Carolina at Chapel Hill, USA
(†) ETH Zurich, Switzerland
E-mail: {nico,prm,lin}@cs.unc.edu, {otaduy,grossm}@inf.ethz.ch

Abstract

We present a fast proximity query algorithm for haptic display of complex deformable models using a layered representation. Assuming that each solid model can be represented as a rigid core covered by a layer of deformable material, the deformation field of the surface can be expressed as a function in the parametric domain of the rigid core. Our 2-stage collision query algorithm starts by performing an approximate object-space collision detection between low-resolution polygonal proxies. We then refine the query result by computing a directional penetration depth field using a local height-field representation of the deformable layers to detect the interference between the high-resolution surface geometry. We have developed a proof-of-concept demonstration using commodity graphics processors and been able to perform fast proximity queries between two highly complex deformable models in less than 2 msecs.

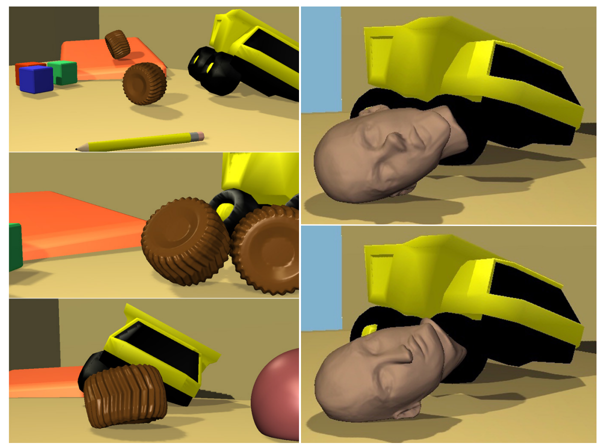

Figure 1: Soft Object Interaction in a Dynamic Scene. *Deformable objects roll and collide in the playground.*

1 Introduction

Haptic rendering of forces and torques between interacting objects, also known as 6 degree-of-freedom (DoF) haptics, has been demonstrated to improve task performance in applications such as molecular docking, nanomanipulation, medical training, and mechanical assembly in virtual prototyping. Haptic display of complex interaction between two deformable models is considered especially challenging, due to the computational complexity involved in computing contact response and performing proximity queries, including collision detection, separation distance, and penetration depth, between two deformable models at force update rates.

In this short paper, we will focus on the problem of proximity queries between two highly complex deformable models. We assume that real-world deformable solids can be modeled as a rigid core covered by a layer of deformable material [2] and that the deformation field of the surface can be expressed as a function in the parametric domain of the rigid core. Examples include animated characters, furniture, toys, tires, etc. We reformulate the problem of collision detection on a 2D parametric atlas to reduce the extremely high geometric complexity due to contacts between high-resolution deformable surfaces.

We exploit our layered representation in a scalable and output-sensitive two-stage collision detection algorithm. This novel formulation of the problem is especially well suited for realization on commodity single-instruction multiple-data (SIMD) or parallel architectures, such as multi-core architecture, graphics processor units (GPUs), Cell processors, and physics processing units (PPUs). We show a proof-of-concept demonstration using GPUs (See Fig. 1).

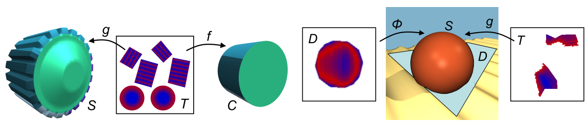

Figure 2: *Mapping from 3D deformation domain to 2D computational domain for simulation (left) and collision detection (right).*

2 Overview

We reformulate the problem of collision queries on a 2-dimensional atlas. This mapping is illustrated in Figure 2, with the 2D computational domains indicated by T and D. Using a two-stage collision detection algorithm for parameterized layered deformable models, our proximity queries are scalable and output-sensitive, i.e. the performance of the queries does not directly depend on the complexity of the surface meshes.

Our accelerated proximity query algorithm starts by performing object-space collision detection between low-resolution polygonal proxies. We identify potentially intersecting surface patches and a penetration direction for each contact region. We then refine the query result by considering a localized height-field representation of the deformable geometry parameterized on a 2D domain. This second stage computes the penetration depth field on the high-frequency surface. We have designed an image-space algorithm and developed a parallel implementation on GPUs that achieves fast computation at haptic update rates. Haptic display can be computed using a penalty-based approach to render the net forces and torques back to the user.

3 Proximity Queries on GPUs

We assume that, within regions of contact, the surfaces can be described as height fields. The directional penetration depth can then be defined as the height field difference between the intersecting patches, in the local direction of penetration. As a preprocess, we parameterize the low-resolution surfaces used in object-space collision detection, and create texture atlases that store the positions of the full-resolution deformable surfaces.

At runtime, we render the intersecting low-resolution surface patches into the contact domain D using an orthographic projection along the local penetration direction. At each fragment, we can obtain the original surface position from by looking up the position in the dynamic deformation field T stored in texture memory. We then perform a second pass over the intersecting region, where we subtract the local height fields of both deformable surfaces. Finally, we transfer the collision information from the contact domain D to the deformation domain T for contact response and force computation, using a texture coordinate transformation technique also used in perspective shadow mapping.

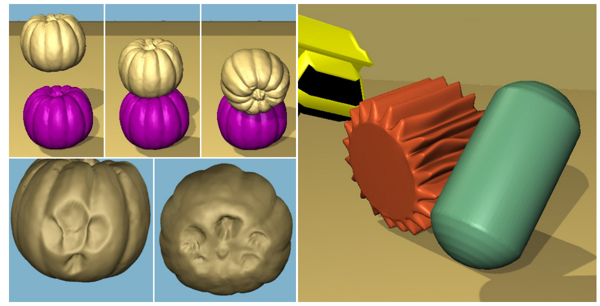

Figure 3: Rich Deformation of High-Resolution Geometry. *In the bottom-left corner, observe views from below of the top pumpkin as it collides with the bottom pumpkin and deforms.*

4 Results

We have tested our novel proximity query algorithm on deformable models of high complexity (consisting of hundreds of thousands of surface elements) with rich surface deformation, as shown in Fig. 1. The low-resolution proxies are simplified down to a few hundred of triangles, which is roughly the size that can be handled by existing collision detection techniques [1]. In the case of the head model, which has $44,000$ deformable vertices, we were able to obtain per-vertex penetration depth information within 2 ms. These timings include the transfer of the contact information to the deformation domain, where it is directly available for dynamics computation.

Using our scalable and output-sensitive collision detection algorithm, we compute object penetration depth that captures the original high-frequency geometry, and we then display dynamic effects due to surface deformation that would otherwise be missed, such as the deformation on the bottom of the pumpkins in Fig. 3 and the dynamic rolling behavior of the gears due to the deformation of its teeth.

References

[1] M. A. Otaduy and M. C. Lin. Sensation preserving simplification for haptic rendering. *ACM Trans. on Graphics (Proc. of ACM SIGGRAPH)*, pages 543–553, 2003.

[2] Terzopoulos D., Witkin A.: Physically based models with rigid and deformable components. *IEEE Computer Graphics and Applications 8*, 6 (1988).