Contact Trees: Adaptive Contact Sampling for Robust Dynamics

Danny M. Kaufman*‡ Shinjiro Sueda* Dinesh K. Pai*‡
*University of British Columbia ‡Rutgers University

Introduction

Algorithms for rigid body dynamics with contact are well known, but challenging to implement due to the interplay between large time steps, general purpose collision detection packages and pragmatic approximations of the underlying inequality constrained contact problems. While research on rigid body simulation has focused heavily both on contact resolution and collision detection, contact generation has largely been ignored. Most contact resolution algorithms presume that an ideal set of contacts, fully characterizing system constraints, are available, while collision detection methods generally presume that their task is finished once a set of intersecting primitives has been identified. Bridging the gap between these domains, by generating representative contact samples, contact point locations and their associated normals, is crucial for the accuracy, robustness and speed of simulation. This is highlighted by two observations:

- A contact resolution method, no matter how robust and accurate, is restricted by the quality of constraints it has been provided with.
- Narrow phase collision detection algorithms, although in some senses highly optimized, often perform *too much* and the *wrong kind* of work because they do not consider the form of input desired by the physical solver.

We address these issues by developing an adaptive contact generation approach that tightly integrates hierarchical collision detection with the generation of well sampled contact constraints.

Overview

We augment an adaptively sampled distance field [Frisken et al. 2000], stored on an octree grid, with a Bounding-Volume Hierarchy (BVH), local error terms, and surface cell half-spaces. This generates an efficient contact tree structure for both narrow-phase collision detection, *and* distance field evaluation.

We precompute a contact tree starting with an input mesh, a maximum error tolerance, ε_s, and a maximum allowed tree depth. The error term guides the distance field's refinement, while the maximum depth term provides a hard limit for recursion depth. Given an exact signed distance function on our mesh [Baerentzen 2005], we refine our octree until each cell's local trilinear interpolation of the distance field has an error less than ε_s.

Adaptive sampling allows asperities, such as sharp corners, and other high frequency features to be well represented by the octree, both on the surface of, and inside each body up to the pre-determined requirements given by ε_s. Likewise, simple trilinear interpolation is sufficient for evaluating the distance field within each cell, since the refinement has already considered the second order error terms.

For collision detection we construct a BVH on the octree by wrapping each cell associated with an octree node using a tight bounding sphere. Leaf nodes in the hierarchy then correspond to the most refined cells with the least interpolation error, while branch nodes moving towards the root correspond to coarser cells with increasing error. We store the error associated with each cell at its corresponding node to guide graceful degradation during narrow phase recursion.

Each surface spanning cell is also assigned a pre-computed half-space whose boundary spans the locally defined zero-level set. Half-spaces are efficiently stored in local coordinates by normal vectors and their offsets. Within the boundary of the cell, the bounding plane of this half-space is acceptable as a first order surface proxy up to the locally stored error term, and so is suitable for contact point sampling.

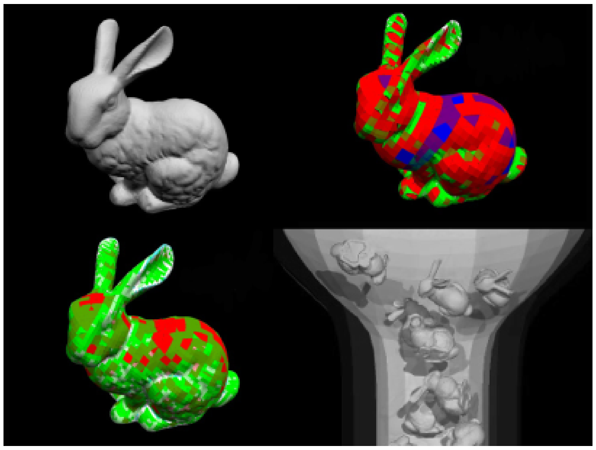

Figure 1: Base mesh, two levels of contact trees, and a simulation.

Because surface spanning cells and their associated wrapping spheres may not closely conform to the actual geometry in large flat regions, we also use the stored half-spaces for collision detection. Whenever a leaf cell is reached by tree traversal during a collision check, we use its half-space to continue checking spheres against it with a half-plane test.

Once completed, collision detection generates a list of intersecting sphere pairs from two bodies. For each sphere wrapping a surface spanning cell, we compute a distribution of sample points along the cell's half-plane boundary. Each such sample point is tested against the cell wrapped by the other sphere and, if inside, against the cell's distance field. If both tests indicate contact or interpenetration, a final trilinear interpolation returns the constraint normal approximated by the gradient at that point.

Conclusion

Our approach allows us to produce representative contact samples, online, from arbitrary triangulated meshes. We efficiently sample both low and high frequency regions of geometries, thus avoiding redundant contacts that lead to poorly conditioned constraint systems. Additionally, by extending our sampling away from the mesh, we enable constraint generation for interpenetrating objects. Finally, our system allows for graceful degradation with local feature awareness.

In simulations, this generates an accurate depiction of contact constraints on a wide range of scales. We can simulate contacts between large flat polygonal regions in a time and memory efficient manner, while simultaneously reproducing the non-linear effects caused by small-scale surface features during collisions [Barzel et al. 1996].

References

BAERENTZEN, J. A. 2005. Signed distance computation using the angle weighted pseudonormal. *IEEE Transactions on Visualization and Computer Graphics 11*, 3, 243–253.

BARZEL, R., HUGHES, J. F., AND WOOD, D. N. 1996. Plausible motion simulation for computer graphics animation. In *Computer Animation and Simulation 96*, 183–197.

FRISKEN, S. F., PERRY, R. N., ROCKWOOD, A. P., AND JONES, T. R. 2000. Adaptively sampled distance fields: a general representation of shape for computer graphics. In *Proceedings of ACM SIGGRAPH 00*, 249–254.