Collision Handling

Carol O'Sullivan and John Dingliana, Image Synthesis Group, Trinity College Dublin

People are very sensitive to physical events occurring around them... We know that one solid object cannot merge into another; We make decisions about the properties of objects based on the way in which they interact with each other; We judge whether objects are animate or inanimate depending on whether we perceive them as moving of their own volition, or being "caused" to move by another object (referred to as the perception of causality [Mic68]). Many studies have shown that these perceptual mechanisms are established very early on in infancy (e.g. see [BSW85])... but how accurate are they? Research in the realm of physics education has shown that most people have erroneous, yet very robust, pre-conceptions regarding the physical behaviour of objects [Cle82]. This is obviously not a good thing if you are trying to teach introductory mechanics, but it could be very useful if you are trying to get away with fast, yet plausible, physically-based simulations for real-time applications.

Despite significant advances in technology, real-time applications are still quite far from achieving photo-realistic rendering and mathematically precise simulation. Only through a great deal of simplification in the scene or the procedural models used to generate the animation, is it possible to even come close to achieving real-time frame rates. However, in most cases, the complexity of an animated scene, and the visibility of objects in the scene tend to change significantly over the course of the animation as objects move in and out of view and user focus. Many approaches try to interactively modulate levels of detail based on visibility information, culling the behaviours of objects outside the viewing frustum [CF97] or simplifying the procedures used to generate the motions of faraway entities [CH97]. Wherever tradeoffs are required, it is desirable to achieve them in the most efficient way possible. In the Physically Based Animation domain this entails:

1. That we guarantee target frame rates for real-time performance regardless of scene complexity
2. That we handle as much complexity as possible within the allowable time limit
3. That we get the best return for any simplifications that become necessary
4. That we minimize the overhead costs for the part of the system which manages the tradeoffs between complexity and speed

The first requirement for a system which will trade-off accuracy and speed is a mechanism that can handle processing at different resolutions and return consistent results. Refinable solutions process input data and return increasingly better results as more time is spent on the solution. A time critical or interruptible mechanism needs to be able to return a result as soon as it is instructed that it has exhausted all the processing time that has been allocated to it [Hub96]. For the purposes of Collision Detection, Hubbard's approach based on Hierarchical Sphere Trees is a prime example of a time-critical mechanism which returns refinable data. The approach is based on a Hierarchical Multiresolution sphere-tree representation of an object's volume (See Figure 1). Collision detection involves intersection tests between the nodes of the tree. When such collisions are found, and if there is time remaining, then a more accurate intersection test is performed between the children of the two colliding nodes.

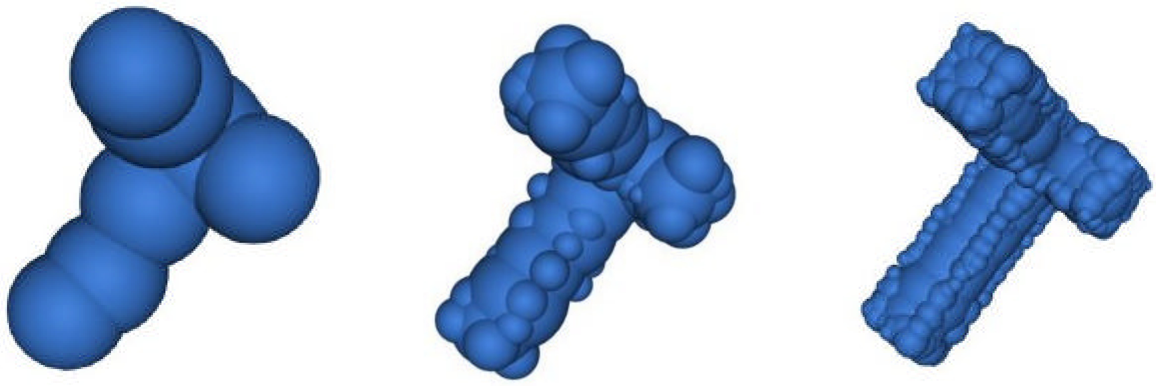

Figure 1: Representing an object with multiple levels of spheres.

We have extended this technique [DO00][DOB01] to return collision data, which can be used in computing physically based responses to object collisions. As with the collision detection process the accuracy of the data (such as details of contact points) is improved as the mechanism traverses deeper into the sphere tree. As we deal with higher resolutions of the volume model, the approximated contact points become increasingly accurate (see Figure 2). If the mechanism is interrupted, e.g. when it has used up its allocated time quota, then it immediately returns the most accurate approximation it has so far computed.

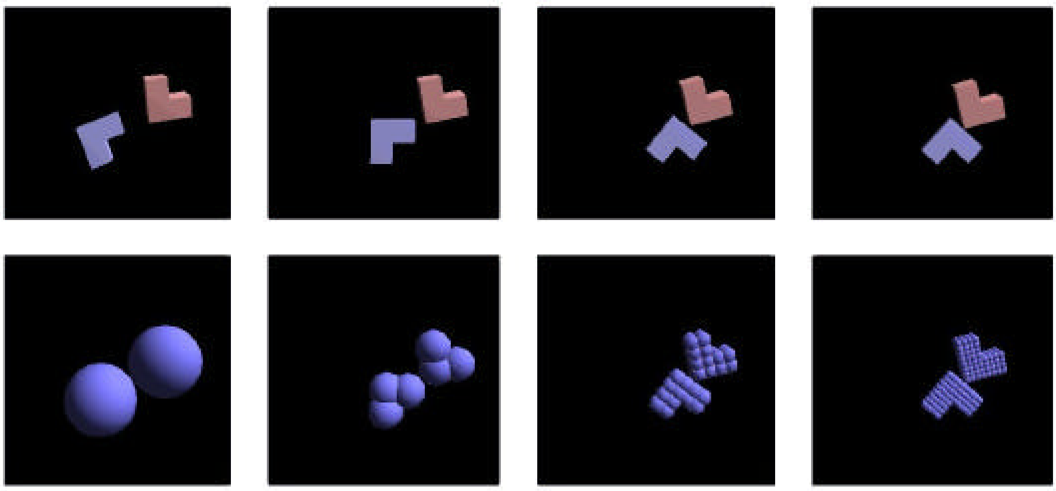

Figure 2: Multi-resolution collisions between objects The bottom row of images show the volumes used to perform collision detection at increasing levels of detail, while the top row shows what is actually seen by the viewer at the moment of impact.

We must remember that we are dealing with a viewer-centric model and that what we are trying to optimise is the perception of the animation rather than it's mathematical accuracy [BHW96][CF00]. In the viewer-centric model, the first thing we should note is that objects (and collisions between objects) further away from the user, due to occlusion or perspective fore-shortening become more difficult to judge and, as a result, errors and approximations in their states and behaviours become more difficult to notice. Furthermore, even a casual study will show that users' primary awareness of events in a scene focuses in a small radius around the point of fixation (see Figure 3). Other factors, which might influence the user's ability to judge an event on the scene, are properties of the objects e.g. size, shape, velocity; or properties of the scene and surroundings e.g. crowdedness, lighting, etc. Since the user's ability to notice error is non-uniformly spread across the scene, this suggests that processing time (and as a result simulation accuracy) for different events across the scene should also not be uniform. Instead, the solution we advocate is to prioritise events across the scene and distribute processing time based on this prioritisation.

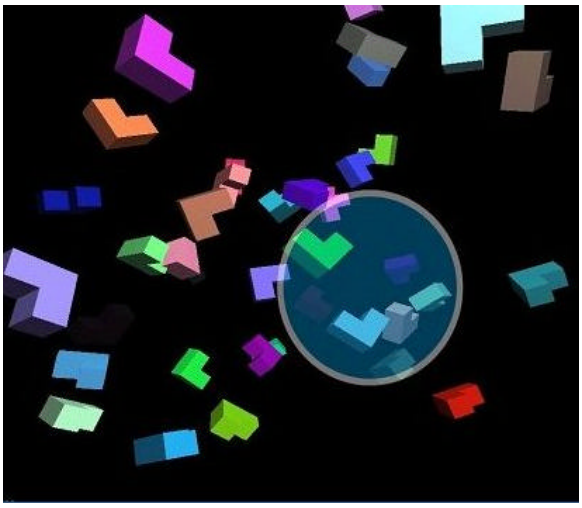

Figure 3: Important collisions, e.g. those close to the viewer's fixation position, should be processed first.

Early results indicate that the overhead from a full prioritisation and sorting of events in the scene on a per-frame basis becomes too high. A more fruitful approach is to use a small number of different priority groups into which events are interactively distributed. Each priority group is then allocated its share of processing time by the scheduler, with more processing being spent on higher priority groups. This method, whilst preserving a prioritisation scheme, bears considerably less overhead expense than a full continuous sort and in practice delivers good results even with very small numbers of priority groups [ORC99].

The level of optimisation depends a great deal on the quality of the metrics used to perform the prioritisation of objects in the scene. Although good results have been achieved by using "obvious" metrics such as object velocities, projected screen distances and distance from the user's fixation point (determined with the use of an interactive eye-tracker, see Figure 4) more extensive studies need to be performed to identify the most important factors which affect user perception of events and to determine how the influence of all such factors varies across a simulation scene.

Figure 4: An eye-tracker is used to determine the viewer's point of fixation

We use psychophysical experiments to determine the factors that influence people's perception of dynamic events such as collisions and physical behaviours [OD01], with the purpose of developing dynamically-calculated metrics to drive the perceptual scheduling of our real-time adaptive physical simulations. Such experiments are very difficult to design, due to the high number of variables which need to be taken into account. Either the experiment needs to be reduced down to such a restrictive level of conditions, that the task is no longer representative of the real world (which is actually the case in most psychophysical investigations in the vision literature), or more natural tasks are devised, thereby introducing an inevitable subjectivity or "fuzziness". How can we examine the effect of "bad" physical events (e.g. objects not touching before they bounce) without actually directing people's attention to them? We are also currently involved in designing "real vs. simulated" psychological experiments, where participants compare "real" dynamic scenes of multiple interacting objects with corresponding simulations at varying levels of detail.

References

[BHW96] Barzel R., Hughes J.F. and Wood D.N. - Plausible Simulation for Computer Graphics. Animation and Simulation ’96. pp. 183-197. (1996)

[BSW85] Baillargeon, R. Spelke, E. and Wasserman, S. - Object Permanence in Five-month-old Infants. Cognition, 20:191-208. (1985)

[CH97] Carlson D.A. and Hodgins J.K. – Simulation Levels of Detail for Real-Time Animation. Proc. Of Graphics Interface ’97. pp. 1-8. (1997)

[CF97] Chenny S. and Forsyth D. – View Dependant Culling of Dynamic Systems in Virtual Environments. In Proc. 1997 Symposium on Interactive 3D Graphics. (1997)

[CF00] Chenny S. and Forsyth D. – Sampling Plausible Solutions to Multi-Body Constraint Problems. In Proc. SIGGRAPH 2000 pp 219-228. (2000)

[Cle82] Clement, J. - Students’ Preconceptions in Introductory Mechanics. American Journal of Physics, Vol 50. No. 1. 66-71. (1982).

[DO00] Dingliana, J. O'Sullivan, C - Graceful Degradation of Collision Handling in Physically Based Animation. - Computer Graphics Forum. Vol 19, Number 3 (Proc. Eurographics 2000), pp 239-247 (2000)

[DOB01] Dingliana, J. O'Sullivan, C. Bradshaw, G. - Collisions and Adaptive Levels of Detail - SIGGRAPH 2001 Sketches Program, LA (To appear) (2001)

[Hub96] Hubbard, P. - Approximating Polyhedra with Spheres for Time-Critical Collision Detection. - ACM Transactions on Graphics, Vol. 15, No. 3, July 1996, pp. 179-210 (1996)

[Mic63] Michotte, A. - The Perception of Causality - New York: Basic Books. (1963).

[OD01] O'Sullivan, C. Dingliana, J. - Real vs. Approximate Collisions: When can we tell the difference - SIGGRAPH 2001 Sketches Program, LA (To appear) (2001)

[ORC99] O'Sullivan C. Radach, R. Collins, S. - A Model of Collision Perception for Real-Time Animation - Animation and Simulation '99, pp. 67-76 (1999)