Available online at www.sciencedirect.com
SCIENCE DIRECT
Computers & Graphics 30 (2006) 423–431
COMPUTERS & GRAPHICS
www.elsevier.com/locate/cag

Virtual Reality Interaction and Physical Simulation

Approximate collision response using closest feature maps

Thanh Giang*,[1], Carol O'Sullivan

Image Synthesis Group, Department of Computer Science, Trinity College, Dublin, Ireland

Abstract

In this paper we consider the novel idea of closest feature maps (CFMs) applied to refinable collision response in order to address the potential issues and problems associated with over approximation of contact information for time-critical collision detection schemes that utilise sphere-tree bounding volume hierarchies. Existing solutions to time-critical collision handling can at times suffer from over-approximation of required contacting data, which may lead to undesirable or implausible physical response. Our CFM solution essentially utilises information for contact data approximation based on the underlying geometry of the colliding objects rather than potentially problematic properties of the contacting bounding volumes. The merits of the scheme lie in its simplicity and effectiveness to handle refinable collision data in an efficient manner and could quite easily be extended to incorporate other types of bounding volume hierarchies for interruptible collision handling.

Keywords: Physically based modelling; Three dimensional graphics and realism; Time-critical dynamics; Collision detection and response

1. Introduction

Obtaining useful information for collision resolution after a potential collision has been detected is vital for good plausible dynamic response. The traditional approach in a collision detection scheme that utilises bounding volume hierarchies (BVHs) such as sphere-trees, is to have each leaf of the tree contain the information about which polygon(s) it bounds. Upon traversal down to the leaf level signifying a positive collision, more expensive and exact schemes such as polygon-polygon, face-edge or similar intersection tests are further conducted. From these tests, useful collision information such as point(s) and normal(s) of contact are obtained for collision resolution. In such collision detection approaches, the BVHs are utilised as a detection acceleration technique only.

For time-critical collision detection, such an approach may not be possible due to the uncertainty that we may ever reach the leaf level of our BVH. For such schemes, a BVH must be treated as inherently part of the entire collision handling process, whereby a quick definitive answer must be obtainable at any level of the hierarchy.

The problem of what to do during interruption in a time-critical collision detection schema still very much remains an open question. To our knowledge, very few researchers to date have considered the question of the *next step* beyond interruption of collision detection. In this paper we build upon the work of Dingliana and O'Sullivan [1]. While their proposed contact resolution scheme works quite well, it can potentially provide over-approximations. This is most noticeable during cases of continuous contact and when we are forced to resolve collisions at higher levels of the bounding volume hierarchy, resulting in a detraction from the overall plausibility of the physical simulation. This paper proposes an improved refinable collision handling scheme that rectifies the potential drawbacks of Dingliana and O'Sullivan's approximation scheme. To improve contact data approximation, we propose the use of *closest feature maps* (CFMs) to quickly approximate essential contact information at the polygonal level rather than using spherical geometry. The merits of the proposed scheme lie in its simplicity and robustness. Although we present the scheme here for spheres in a sphere-tree BVH,

*Corresponding author.
E-mail addresses: thanhg@gmail.com (T. Giang), Carol.OSullivan@cs.tcd.ie (C. O'Sullivan).
[1]This paper represents work done by the author while with the Image Synthesis Group, Trinity College Dublin as a Ph.D. Research Fellow.

0097-8493/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cag.2006.02.019

the general idea of the scheme could be easily extended to accommodate other bounding volume structures such as AABB-trees, OBB-trees or K-DOPs.

2. Background and previous work

The notion of interruptible collision detection, which allows us to arbitrarily stop collision detection processing to conform to timing restrictions was first introduced by Hubbard [2]. In Hubbard's scheme, sphere-trees were used for interruptible collision detection, accounting for time constraints. In this manner, a refinable level of detail collision detection was achieved whereby it was possible for the simulation to fall back to a lower level approximation of collisions whenever time constraints prohibited full collision processing. In Hubbard's algorithm, collision response was very roughly approximated through an elementary reversal of the colliding objects' velocities during collisions. No reasonable contact modelling was considered but rather the research was focussed on the collision detection scheme alone.

O'Sullivan and Dingliana [3,1] later adapted Hubbard's work to use perceptual metrics and a more appropriate collision response solution to Hubbard's elementary response model was also proposed; the geometry of the resultant contacting spheres was used to quickly approximate appropriate contact data for collision response. The issues of collision response in an interruptible system were also addressed to some extent in Dingliana and O'Sullivan's paper. In their discussions, however, the problems of multiple contact responses and continuous contact are identified as open issues.

Most recently, Klein and Zachmann [4] have also explored the idea of time-critical collision detection using a data structure that they refer to as *ADB-trees* (*augmented distributed trees*). Their ADB-tree is essentially a hierarchical bounding box tree but actually, any hierarchy adopted to use their *average case approach* is considered to be an ADB-tree. Traversal of the tree is guided and aborted based on the probability that a pair of bounding volumes contain intersecting polygons. The case of collision response is not considered in this work. However it is stated that the approximate response model proposed by Dingliana and O'Sullivan [1] can be incorporated into the method.

A multiresolution approach using a hierarchy of convex hulls was proposed by Otaduy and Lin [5,6] and Ehmann and Lin [7,8] that can also be applied to level of detail collision handling. The Lin–Canny [9] closest feature algorithm coupled with a *Voronoi marching* algorithm for further speed up, is used to track closest features at each level within these schemes. In fact, Ehmann and Lin [8] describe using a directional lookup table approach, similar to the CFM approach presented in this paper, in order to initialise their Voronoi marching algorithm. However, the approach presented in this paper differs from that described by Ehmann and Lin in that we do not use our algorithm to initialise any sort of Voronoi marching (or indeed consider the use of Voronoi regions at all). Rather, we attempt to quickly obtain appropriate contact data based on the underlying polygonal geometry through a direct mapping of the closest feature from a spherical sample point which is nearest to a determined point of contact.

The idea of determining appropriate contact information by finding the closest feature to a point of contact is by no means new. There has been much work done in determining closest features, mainly through the use of Voronoi regions [7,9–11]. Our closest feature approach is much simpler in the sense that no Voronoi regions are ever considered but rather a simple direct mapping from nearest spherical sample points is taken instead.

3. Approximating contact information

In Dingliana and O'Sullivan's approach, for each pair of colliding spheres detected, an approximate collision point for this sphere pair collision is obtained by considering the intersection of the line through the centres of each sphere and the intersecting plane of both spheres (the idea of using the line connecting the centroids of polytopes to find the *witness* for collision detection was first suggested by Baraff [12]). An approximate contacting normal for the collision is further determined by a simple calculation of the vector that runs through the center of both colliding spheres.

While this solution is an elegant one and works quite well for certain situations of refinable collision handling, at times the information produced may be too approximate. For objects in constant motion, the effects of overly approximate contact information may not be noticeable due to the expected chaotic nature of dynamics [13]. However, this is not always the case. Some collisions forced to be resolved at the root level (i.e. a single bounding sphere per object) can produce a response that is very noticeably implausible; for example a collision between two cubic objects. Furthermore, for situations where contact information is required for more than a small period of time, such as cases of continuous contact, over-approximation of contact information may also cause unwanted anomalous dynamic behaviour.

Consider two cases of a pair of rectangular objects in 2D colliding with each other, whereby collision detection is allowed to traverse beyond the root level as illustrated in Fig. 1. In the first case 1(a), where both objects are freely floating through space, over-approximations of contact normals do not adversely affect the plausibility of the simulation. However, when one moving object falls directly on top of another stationary one, 1(b), assuming that the second object is falling under gravity, the scenario will eventually bring the two bodies to a state of rest causing them to be in a final stacked position. In such cases, it is necessary to have as close a contact normal approximation as possible to the underlying polygonal representation of the model. Using the previously described method to

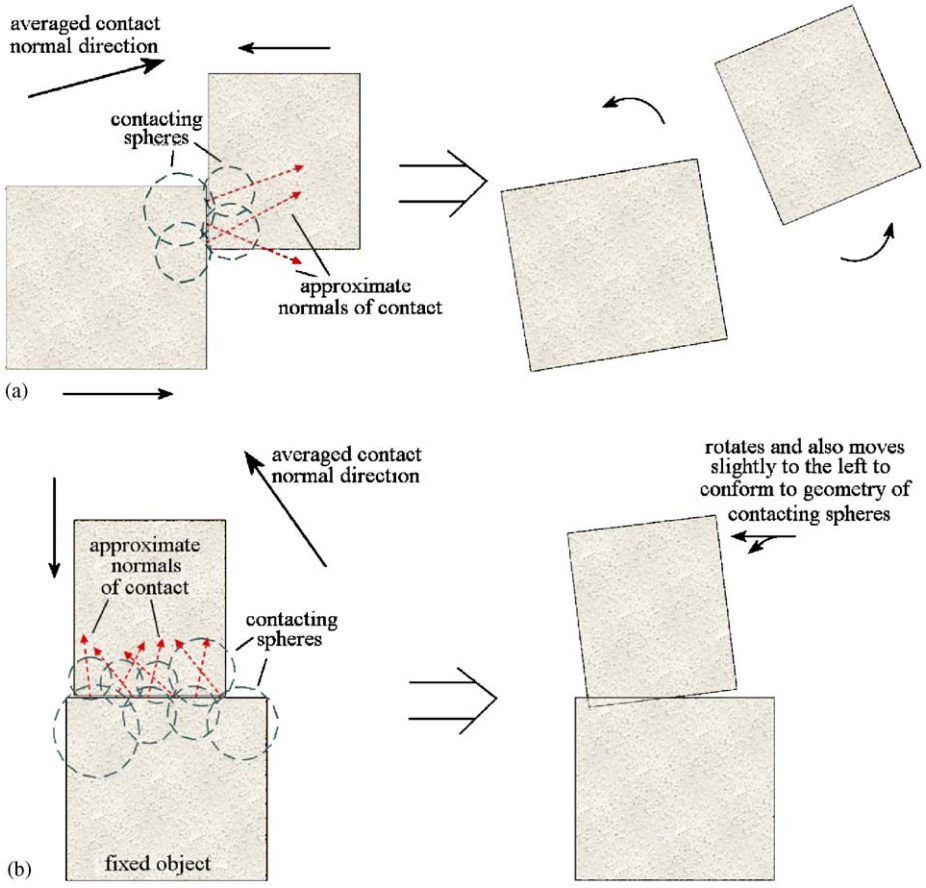

Fig. 1. Cases where over-approximation of contact data is appropriate and inappropriate: (a) normal approximation does not affect overall plausibility of object response; (b) inaccurate contact normals cause unwanted rotation and shifting during response.

approximate contacting normals may result in unwanted rotations and shifting during collisions, due to bad normal approximations, thus detracting from the overall plausibility of the simulation.

3.1. Improved approximations using closest feature maps

To address the problems mentioned above and to provide better contact approximations, we propose the novel concept of CFMs for improving refinable collision handling. Essentially, a CFM is a mapping of contact information based on the closest features from a set of sample points around a sphere that bounds the model or part of the model. It not only maintains the advantage that the further down the sphere-tree we manage to traverse, the more refined the contact information will be, but all contact information is essentially obtained at the polygonal level rather than through rougher approximations made on the geometry of the contacting spheres, no matter where in the sphere-tree we are approximating from.

We illustrate a CFM for four sphere-tree levels in Fig. 2.

The CFM is pre-calculated and loaded in with the sphere-tree information during system startup. A set of sample points is defined around each sphere within the sphere-tree. During the pre-calculation stage it is possible

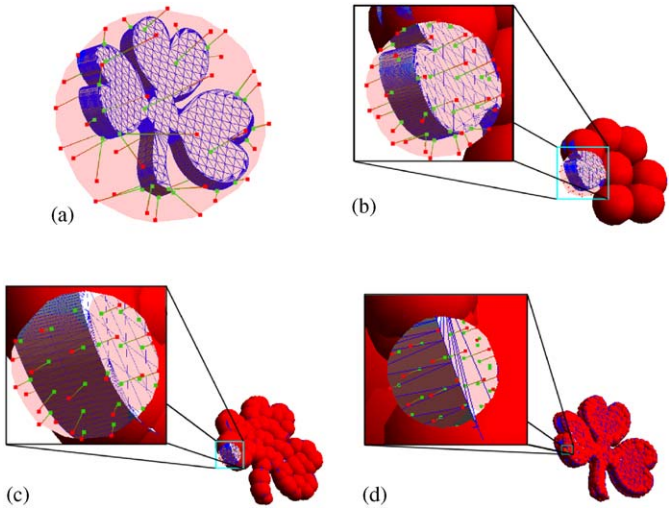

Fig. 2. Closest feature map of four sphere-tree levels for a shamrock model: (a) top level; (b) level 2; (c) level 3; (d) level 4.

for us to determine for each sample point on the corresponding sphere:

(1) the closest feature on the model to each sample point,
(2) the closest point on the surface of the model to the sample point in question on the bounding sphere, and

(3) the outward pointing normal from the closest feature to the sample point.

If a sphere within the sphere-tree is involved in a collision we simply approximate the point of contact in the same manner as that proposed by Dingliana and O'Sullivan. We then determine the closest sampling point on the sphere from this approximate contact point and from that sample point we may obtain the desired closest feature information for collision resolution. Of course, it is not necessary to store all the suggested closest feature information within each sphere of the sphere-tree, only what we deem necessary. In practice, to reduce the memory requirements we only retain the normals of the closest features to each sample point on the sphere (perhaps a more appropriate name for the closest feature map if utilised in this manner would then be a *Normal Map*). We use these normals as the contact normals during collision detection rather than approximating normals in the manner previously described. Such normals are closer approximations to the real normals needed for contact resolution.

4. CFM construction

Before we can utilise the CFM scheme for collision handling within our simulations, we must first construct them. Construction of CFMs involves several steps. The first is to determine how we wish to distribute the sampling points for each sphere within the sphere-tree. This is important as the distribution and density of sampling points around each sphere strongly affects our approximations of closest feature information during collision handling.

For assigning sample points onto the spherical surface we utilise a simple polar sampling scheme:

$$x = \sin(\phi) * \cos(\theta), \tag{1}$$

$$y = \sin(\phi) * \sin(\theta), \tag{2}$$

$$z = \cos(\phi). \tag{3}$$

Such a scheme was chosen as it affords us a very quick and intuitive retrieval strategy later on when determining the closest spherical sample point to our point of contact.

Unfortunately, while this is an elementary method to use for generating spherical sample points, the points that are thus generated are not evenly distributed across the surface of the sphere, as can be seen from Fig. 3(a). This results in very poor sampling of closest features as sample points in close proximity to each other will usually determine the same closest feature information. An adaptive value of θ is used during sample point creation depending on proximity to the spherical poles to give better sampling distribution. Since we are assuming a unit sphere during sample point creation, this is easily done if we sample θ with an interval proportional to $\sin(\phi)$. The results of adopting such a strategy are illustrated in Fig. 3(b). As can be seen, such a strategy gives a much more favourable distribution of points around the sphere.

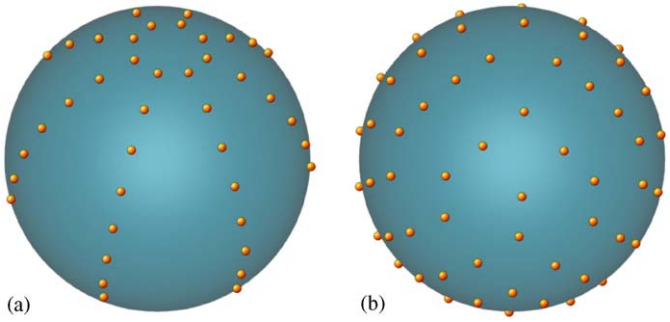

Fig. 3. Difference in point distribution for non-adaptive and adaptively created polar sample points: (a) normal polar sampling; (b) adaptive polar sampling.

4.1. Determining closest feature information from sample points

For determining closest feature information to each sample point we adopt the same closest point algorithm as used by Hubbard [14] and Bradshaw [15] within their sphere-tree generation approaches. Unlike Hubbard and Bradshaw, we are not concerned about whether a point lies inside or outside the polygonal mesh just simply the closest feature to the point.

In checking for the closest point to a triangle on the mesh surface, the algorithm perpendicularly projects the point p on the sphere's surface to a point p' on the mesh. Three cases are then considered:

Face:

- p' lies behind every edge of the triangle (Fig. 4(a)).

Vertex:

- p' lies outside of the triangle but is in front of two edges (Fig. 4(b)).
- p' lies in front of one edge and the perpendicular projection of p' (i.e. p'') lies outside of the triangle (Fig. 4(d)).

Edge:

- p' lies in front of one edge and the perpendicular projection of p' (i.e. p'') lies inside of the triangle (i.e. behind two edges, Fig. 4(c)).

In all cases, once the closest feature has been determined, the calculation of the corresponding normal to the discovered closest feature is a trivial matter.

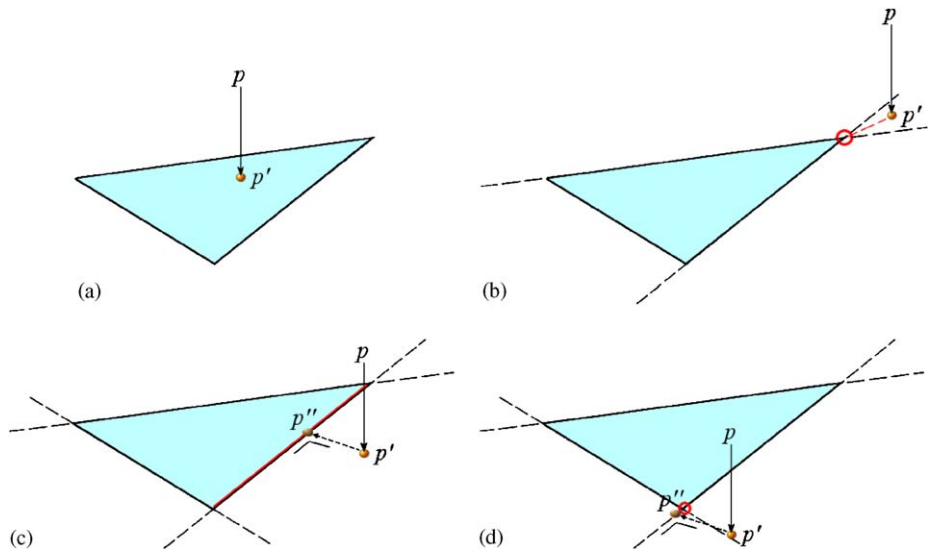

Fig. 4. Different cases that can be encountered during closest point searching: (a) point inside triangle; (b) outside triangle in front of two edges; (c) in front of one edge and inside triangle; (d) in front of one edge and outside triangle.

5. Efficient nearest sample point determination

During simulation, it is important that we can obtain appropriate contact information as quickly as possible once a collision has been detected. Fortunately, the polar sampling schema provides us with an elementary and constant time retrieval solution, provided that we set up the data structure as appropriate. The appropriate closest feature information can be quickly retrieved by determining the ϕ and θ angle of the point of contact from the contacting sphere's centre. These values can then be used as index values to quickly determine the nearest spherical sample point and hence the closest feature. Since this is essentially a lookup process, the retrieval strategy is conducted in a constant time manner, regardless of number of sample points per sphere.

To account for non-constant polar sampling intervals of θ for each ring at angle ϕ, the equations for data retrieval can be expressed as

$$Index_{\phi} = \frac{\phi * (ListSize_{\phi})}{180}, \tag{4}$$

$$Index_{\theta} = \frac{\theta * (ListSize_{\theta})}{360}, \tag{5}$$

whereby $ListSize_{\phi/\theta}$ are the respective sizes of the ϕ and θ lists.

6. Discussion

For a time-critical dynamics system, the approximation of the required contacting data for collision resolution should not further hinder the imposed timing constraints of the entire collision handling process. The polar sampling scheme retrieves appropriate contact information in constant time, as it is essentially a look up process, so it conforms well to this required time criterion.

An alternative to the polar sampling approach is to employ a geodesic sampling approach. This has the advantage that all sample points created are equidistant to each other. However, closest feature data retrieval is not as intuitive as that of the polar sampling scheme. We have also implemented a geodesic sampling algorithm for comparison to the polar sampling scheme using the vertices of an icosohedron for sample point generation. Timing tests were conducted for both the polar and geodesic sampling schemes for comparison. For the geodesic sampling scheme we adopted an elementary divide and conquer approach for closest feature data retrieval. The quadrant of the hemisphere in which the nearest neighbouring sample point resided was determined and then an exhaustive nearest neighbour search of the remaining points was conducted in order to obtain the appropriate closest feature data. This scheme proved to be four times faster than an elementary linear search scheme.

A constant time retrieval of contact data is obviously most favourable. As such, the polar sampling scheme is preferred due to its elementary but yet guaranteed constant time retrieval solution. However, if a geodesic sampling scheme is used or any other similar point distribution scheme for bounding volumes other than spheres, then a voxel based search scheme is recommended. Other temporal coherence schemes may be similarly adopted to further speed up closest feature retrieval.

It is quite possible that, depending on the adopted sphere generation algorithm, at lower levels of the sphere-tree some bounding spheres may only bound a single polygon. Such a case should be checked for during the closest feature determination and sample point creation process. It may be that, in such a case, the sphere simply has the same closest feature information for all

sample points distributed over the sphere. In such an instance it is not necessary to store or even maintain excessive sample points for the sphere in question but merely the associated bounded closest feature. Thus, when such a sphere is involved in a collision, the appropriate closest feature information is trivially obtained. Such a check during the preprocessing stage invariably leads to further speedup of closest feature data retrieval if a geodesic scheme is used.

We may further optimise our CFM schema by enhancing how we deal with the point distribution around the bounding spheres of the sphere-tree if we adopt a progressive refinement strategy for the point distribution scheme as well. In other words, the further down the tree we progress, the more sparse the point sampling on each sphere need be, since the possibility of catching duplicate closest feature information to associated sample points on each sphere increases as sphere size decreases. We adopt such a strategy during sample point creation. It was found that using between 10 and 16 stacks for sample point creation gave, in general, a good sampling resolution. It should be noted that we did not use very high detailed polygonal models in our animations; for example, the Stanford bunny model only had 1500 triangles. Table 1 shows the maximum (i.e. at the root) and minimum (i.e. at the leaf nodes) sampling resolution along with actual CFM file size for sample points created using 10 and 16 stacks for a cube and bunny model. It should be noted that the CFM file also contained information on the makeup of the spherical bounding volume hierarchy for the corresponding model.

The question arises why we did not simply use an averaging of the normals of the closest features bounded by the corresponding sphere. While this in itself may seem like a sensible idea and does potentially give us a very quick data retrieval strategy, there are cases whereby this averaging scheme once again may provide us with undesirable over-approximation of contact data for collision resolution. Such an instance is best illustrated if one takes the case of a sphere bounding the corner of a box object. In such a case the sphere bounds at least two faces of the box that are perpendicular to each other. If we take the case of Fig. 1(b) where we have one box falling directly on top of each other, the averaging of normal data in such an instance will produce normal information that causes unwanted rotation during initial contact which will detract from the overall plausibility of the resultant animation.

Table 1
Sample point resolution and file size for cube and Stanford bunny models

Model	No. of stacks	Root samples	Leaf samples	File size
Cube	10	65	3	112k
Bunny	10	65	3	130k
Cube	16	167	11	345k
Bunny	16	167	11	415k

7. Results

In order to better gauge the sampling distribution of the adaptive polar sampling scheme, the standard deviation of distances between neighbouring sample points was taken. The ideal standard deviation that we would prefer is obviously zero as this suggests equidistant points (as is the case for a geodesic sampling scheme), which in turn reduces the possibility of unnecessary duplicate closest feature information. Standard deviations of distances were measured and compared for both the full and adaptive polar sampling scheme for points created with 5, 10 and 20 stacks. As the results shown in Table 2 illustrate, adaptive polar sampling greatly improves the *quality* of point distribution as the number of contacting points are reduced. As point distribution increases, neighbouring points start to become fairly close to equidistant from each other, so much so that the adaptive scheme only provides minor improvements. However, it must be noted that as the number of sampling points increases, so too does the possibility of producing too many unnecessary duplicate closest feature data. The ideal situation is to have as much unique closest feature information as possible utilising the least number of sample points per sphere.

To further determine the overall improvement in plausibility of animations created with our CFM schema, we took the plausibility metric proposed by O'Sullivan et al. [16] and applied it to animations produced for four different bounding sphere levels for collisions between two simple objects (i.e. two cubes) and then two relatively complex objects (i.e. two Stanford bunnies). The animations consisted of two objects in a zero gravity environment with one object moving towards another initially stationary object. The objects eventually collide providing us with a physical response whereby the appropriate response parameters were then recorded.

We did this for animations using our CFM scheme and then for animations using Dingliana and O'Sullivan's approximation scheme so that we could determine the difference in visual fidelity for each schema along with the merits of our scheme.

In order to validate our data with the proposed plausibility metric we needed some "referent to reality" in order to compare this data with. In this case, it was necessary to also simulate and determine the physical response from an exact polygonal level detected collision. This data was then used as our reference data during fidelity probability calculations. For the exact level

Table 2
Standard deviation of distances between neighbouring sample points in a polar sampling scheme

No. of stacks	Full polar	Adaptive polar
5	0.371169	0.17718
10	0.144332	0.119856
20	0.0717484	0.0661784

collisions, we used a very high detailed sphere-tree hierarchy (i.e. a sphere-tree with many levels such that leaf spheres are quite small and abundant) to first narrow in on the region of interest. Then the normal of contact is determined by the polygon that is bounded by the leaf sphere.

For each animation in both the CFM and non-CFM cases we measured the magnitude of the linear and angular velocity responses, the angle of the outgoing linear momentum and the nearest distance between the surfaces of the colliding bodies. From these results, we evaluated the angular distortion probability $P_{angular}$ for each object, x and y in the animation, along with the linear and angular velocity distortion probabilities, P_{LV} and P_{AV}, respectively. Also, the probabilities for the gap between each object, P_{gap}, for each level of the hierarchy was determined. For the case of evaluating the angular distortion probability $P_{angular}$, the probability of looking P_L for the moving box (i.e. object x) was determined to be 1 and the probability of not looking P_{NL} for this moving box was 0. This was the opposite way around for object y, the initial stationary box (i.e. $P_L = 0$ and $P_{NL} = 1$). We did not consider any delay distortion probabilities P_{delay} as there was no delay for collision response within these animations.

We present the evaluated probabilities for both the CFM and non-CFM schemas in Fig. 5 (i.e. a rating of 1 is most plausible with 0 being least plausible). The results presented by Figs. 5(a) and (c), suggest that our CFM method in general exhibits quite a high visual fidelity. This was especially true for the case of the cube experiments. This result can be attributed to the fact that with the CFM method, data for collision response is approximated at the polygonal level rather than through a rougher approximation through the bounding sphere geometry as is the case with Dingliana and O'Sullivan's non-CFM method.

These encouraging results further lend weight to validating the advantages of our CFM schema for use within time critical dynamics. For animations of simple geometric objects such as a cube, these results suggest that the CFM method preserves the overall physical behaviour for all bounding sphere levels.

To illustrate the merits of the CFM approach, we present frames from two worse case scenario animations in Fig. 6. It can be seen that the approach that utilises the approximation scheme proposed by Dingliana and O'Sullivan can potentially either cause unwanted rotations (case of continuous contact) or not enough rotation (case or resolving at highest spherical level), thus leading to noticeable anomalous physical behaviour. The CFM scheme when utilised within such scenarios however, improved the responses for such situations. For our

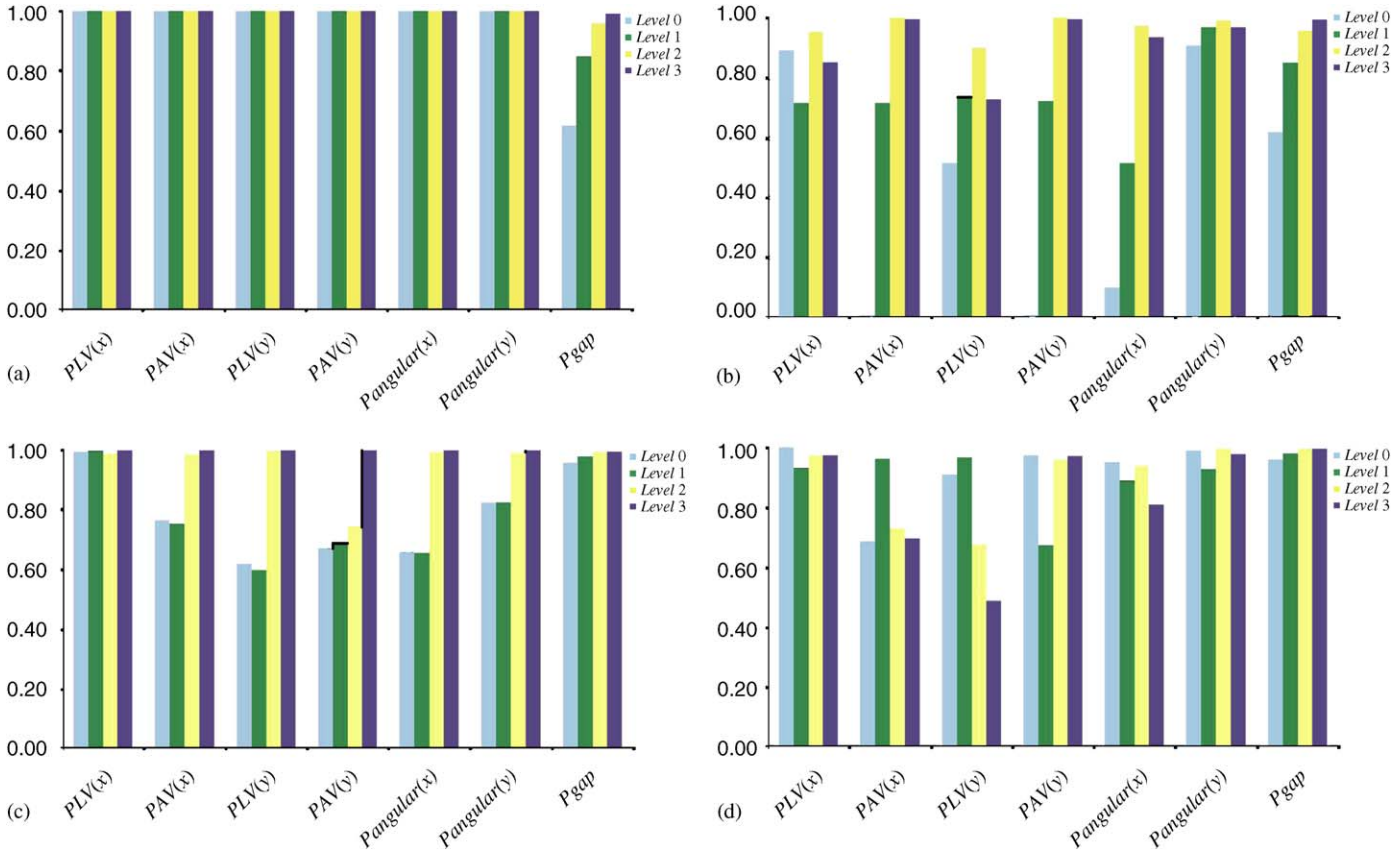

Fig. 5. Results of the visual fidelity probabilities of CFMs in an animation: (a) probabilities for CFM animations—Cubes; (b) probabilities for non-CFM animations—Cubes; (c) probabilities for CFM animations—Bunnies; (d) probabilities for non-CFM animations—Bunnies.

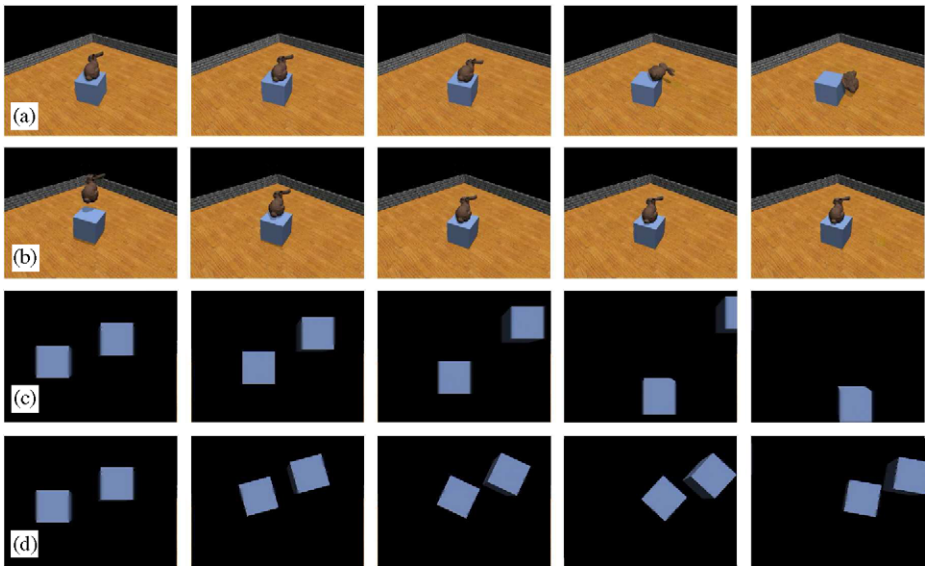

Fig. 6. Two example scenes illustrating CFM approach improving contact response for worse case scenarios: (a) non-CFM approach—bunny landing on top of another box; (b) CFM approach—bunny landing on top of another box; (c) non-CFM approach—resolving collisions at highest spherical level; (d) CFM approach—resolving collisions at highest spherical level.

simulations, an impulse based approach similar to that of Mirtich [17] is used for cases of continuous contact.

8. Conclusions and future work

In this paper, we have presented the novel idea of *CFM* to better approximate contact data during interruption of collision detection. The major issues and stages involved in constructing and using CFMs for more accurate refinable contact approximation were discussed.

In practice, in order to reduce the memory footprint of CFMs, it is only necessary to store closest surface normal information of the feature in question for collision resolution, as poor normal approximation invariably produces implausible collision response. Of course, other closest feature information could potentially be kept as well if more information is needed.

For sphere-tree BVHs, we advocate the use of the adaptive polar sampling scheme over a geodesic sampling scheme. Our results have shown that the standard deviation of the distances between neighbouring sample points in the adaptive polar scheme is relatively low, meaning that the sample point distribution is comparable to a scheme that produces a distribution of equidistant points.

It should be noted that CFM approximations are only as accurate as the sphere sample point distribution will allow. Of course, too sparse point sampling may produce too approximate contact data. However, it is also possible to have an over-abundance of unnecessary point samples, whereby closest feature determination becomes overly pedantic, to such a degree that memory is needlessly wasted. In our determination of how many sample points to produce for each bounding sphere, we used the current spherical level as a guide. However, probably a better strategy to employ would be to determine sampling resolution based on some heuristic that takes into account the number of faces being bound by the bounding volume along with the number of sharp edges exhibited.

We would like to expand the scheme in the future for bounding volume hierarchies other than sphere-trees for interruptible collision handling. A comparison of a nearest neighbour voxel based retrieval scheme for other bounding volumes with generic point distribution schemes to that of the constant time retrieval scheme of the polar sampling approach is also planned. This is in order to further gauge the merits and disadvantages of the current approach. Currently, the polar sampled CFM scheme has been successfully integrated into a full time adaptive dynamics system and has performed quite well with no noticeable anomalous physical behaviour during collision processing. Further investigation into good optimum levels of sample point distribution is also warranted. Perhaps a correlation between smallest polygon and bounding sphere size can be used to determine the number of sample points around each sphere or maybe even a hybrid scheme involving the heuristic mentioned in the last paragraph could be further investigated.

References

[1] Dingliana J, O'Sullivan C. Graceful degradation of collision handling for physically based animation. Computer Graphics Forum 2000;19(3):239–47.

[2] Hubbard P. Real-time collision detection and time-critical computing. In: Proceedings of the 1st workshop on simulation and interaction in virtual environments; 1995. p. 92–6.

[3] O'Sullivan C, Dingliana J. Real-time collision detection and response using sphere-trees. In: Proceedings spring conference in computer graphics; 1999. p. 83–92.
[4] Klein J, Zachmann G. Time-critical collision detection using an average-case approach. In: Proceedings of the ACM symposium on virtual reality software and technology (VRST).
[5] Otaduy MA, Lin MC. Sensation preserving simplification for haptic rendering. ACM SIGGRAPH 2003;22(3):543–53.
[6] Otaduy MA, Lin MC. CLODs: dual hierarchies for multiresolution collision detection. Proceedings of the eurographics/ACM SIGGRAPH symposium on geometry processing; 2003. p. 94–101.
[7] Ehmann SA, Lin MC. Accelerated proximity queries between convex polyhedra by multi-level Voronoi marching. In: Proceedings international conference on intelligent robots and systems.
[8] Ehmann SA, Lin MC. SWIFT: accelerated proximity queries between convex polyhedra by multi-level Voronoi marching. Technical report TR00-026, Department of Computer Science, The University of North Carolina; 2000.
[9] Lin MC. Efficient collision detection for animation and robotics. PhD thesis, University of California at Berkeley; 1993.
[10] Mirtich B. V-clip: fast and robust polyhedral collision detection. ACM Transactions on Graphics 1998;17(3):177–208.
[11] Johnson DE, Cohen E. Spatialized normal cone hierarchies. In: Proceedings of ACM symposium on interactive 3D graphics; 2001. p. 129–34.
[12] Baraff D. Curved surface and coherence for non-penetrating rigid body simulation. ACM SIGGRAPH 1990;24(4):19–28.
[13] O'Sullivan C, Dingliana J. Collisions and perception. ACM Transactions on Graphics 2001;20(3):151–68.
[14] Hubbard P. Collision detection for interactive graphics applications. PhD thesis, Brown University; 1995.
[15] Bradshaw G. Bounding volume hierarchies for level-of-detail collision handling. PhD thesis, Trinity College, Dublin; 2002.
[16] O'Sullivan C, Dingliana J, Giang T, Kaiser M. Evaluating the visual fidelity of physically based animations. ACM SIGGRAPH 2003; 22:527–36.
[17] Mirtich B. Impulse-based dynamic simulation of rigid body systems. PhD thesis, University of California at Berkeley; 1996.