Computing 72, 235–246 (2004)
Digital Object Identifier (DOI) 10.1007/s00607-003-0060-0

Computing
Printed in Austria

Efficient Collision Detection for Moving Ellipsoids Using Separating Planes

W. Wang, **Y-K. Choi**, **B. Chan**, Hong Kong, **M-S. Kim**, Seoul, **J. Wang,** Jinan

Received March 28, 2003; revised November 4, 2003
Published online: April 8, 2004

Abstract

We present a simple, accurate and efficient algorithm for collision detection among moving ellipsoids. Its efficiency is attributed to two results: (i) a simple algebraic test for the separation of two ellipsoids, and (ii) an efficient method for constructing a separating plane between two disjoint ellipsoids. Inter-frame coherence is exploited by using the separating plane to reduce collision detection to simpler subproblems of testing for collision between the plane and each of the ellipsoids. Compared with previous algorithms (such as the GJK method) which employ polygonal approximation of ellipsoids, our algorithm demonstrates comparable computing speed and much higher accuracy.

AMS Subject Classifications: 65D17, 68U07, 68U05.

Keywords: Collision detection, ellipsoids, algebraic test, separating plane, characteristic equation, self-polar.

1. Introduction

Collision detection has many important applications in computer graphics, including the simulation of virtual environments, computer animation and, in particular, 3D computer games [5], [8], [9], [15], [18]. Ellipsoids are frequently used for exact shape representation (e.g. in molecule simulation) and also as a tight bounding shape for many natural objects and organic forms used in character modeling [2]. Thus efficient algorithms for detecting collision among ellipsoids have considerable potential.

We represent the interiors of two ellipsoids $\mathscr{A}$ and $\mathscr{B}$ by the inequalities $X^TAX < 0$ and $X^TBX < 0$, where A and B are 4×4 real symmetric matrices and $X = (x, y, z, w)^T$ represents a point in homogeneous coordinates.

A simple algebraic condition for the separation of two ellipsoids is established by Wang et al. [26]. Given two ellipsoids $\mathscr{A} : X^TAX = 0$ and $\mathscr{B} : X^TBX = 0$, the quartic equation $f(\lambda) = \det(\lambda A + B) = 0$ is called the *characteristic equation* of $\mathscr{A}$ and $\mathscr{B}$. Two ellipsoids are said to be *disjoint* if they do not have a common boundary or interior point.

Proposition 1 [26]. *Let* $\mathscr{A}$ and $\mathscr{B}$ *be two ellipsoids with the characteristic equation* $\mathrm{f}(\lambda) = 0$. *Then*

1. $\mathscr{A}$ *and* $\mathscr{B}$ *are disjoint if and only if* $\mathrm{f}(\lambda) = 0$ *has two distinct positive roots*;
2. $\mathscr{A}$ *and* $\mathscr{B}$ *touch each other externally if and only if* $\mathrm{f}(\lambda) = 0$ *has a positive double root*.

Figure 1a shows two disjoint ellipsoids. Note that their characteristic equation has two distinct positive roots. In Fig. 1b, two ellipsoids overlap and their characteristic equation has no positive root.

Combining this result with a simple method for constructing a separating plane for two disjoint ellipsoids, we devise an efficient algorithm for detecting collisions between two moving ellipsoids. An arbitrary number of moving ellipsoids can also be dealt with by repeated pairwise application of the algorithm.

It is well known that the efficiency of collision detection can be greatly improved by using a separating plane [1]. Once a plane separating two ellipsoids is found, there can be no collision between the ellipsoids until one of them collides with the separating plane. Thus the original problem is reduced to two simpler subproblems of searching for an intersection between a plane and an ellipsoid. Applying an affine transformation, an ellipsoid and a plane can be reduced to a sphere and a plane, and each of these subproblems then becomes equivalent to computing the distance between the center of the sphere and a plane.

Since ellipsoids are preserved under affine motion, our approach can be applied to ellipsoids that are moving and deforming under affine transformation. This is an important advantage over specialized algorithms that work only for simple geometric shapes such as axis-aligned boxes, spheres, cylinders, cones, or tori; such algorithms may not be generalized when affine motions are used.

The remainder of this paper is organized as follows. In Section 2, we briefly review related previous work. In Section 3, we develop a method for constructing a separating plane for two disjoint ellipsoids. In Section 4, we present our complete collision detection algorithm. In Section 5, experimental results are discussed. Section 6 concludes this paper.

2. Related Work

In the past, collision detection for ellipsoids was usually performed by faceting, and then applying a collision detection package appropriate to general convex polyhedra, such as GJK [7], I-COLLIDE [4], or V-Clip [14]. A drawback with this approach is that accuracy and efficiency are compromised by polyhedral approximation.

Rimon and Boyd [17] present an efficient numerical technique for computing a quasi-distance, which they call the 'margin', between two disjoint ellipsoids, using an incremental approach. Based on line geometry, Sohn et al. [20] devised a

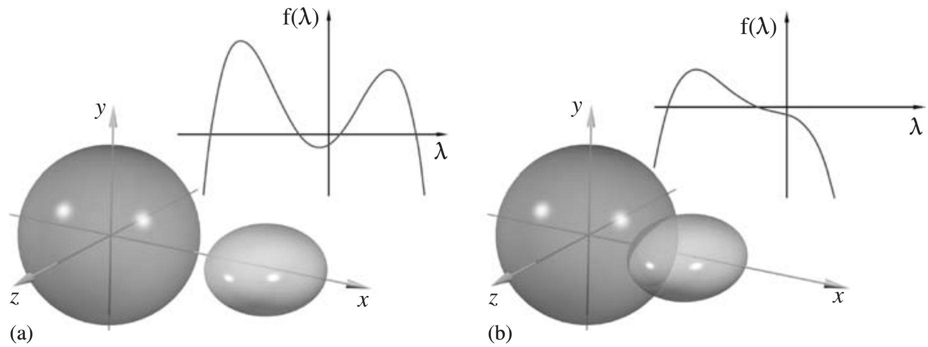

Fig. 1. Two (a) disjoint; (b) overlapping ellipsoids and the corresponding f(λ)

distance computation method for two ellipsoids by solving a system of two equations in two variables. Lennerz and Schömer [10] compute the distance between two general quadrics using Lagrange multipliers. Other schemes have been developed in related areas, such as molecule simulation in computational physics [16] and geomechanics [12], based on numerical iterations; but they leave much to be desired for efficiency. The algorithms described by Baraff [1] and Gilbert and Foo [6] also belong to this category, but are applicable to a wider class of objects bounded by smooth surfaces.

The relationship between two quadric surfaces, including ellipsoids as a special case, has been studied in classical geometry and CAGD [23]. The Segre characteristics, defined by the elementary divisors of the matrix $\lambda A + B$, are used in algebraic geometry [3, 21] to classify a degenerate intersection curve between two quadric surfaces in complex projective space. Similar work in real projective space is presented by Tu et al. [22]. These results are not applicable to our collision detection problem since we are concerned with the relationship between two ellipsoids in real affine space: two ellipsoids always intersect in complex projective space, but this does not mean that they share any common points in real affine space. Various algorithms have also been studied in CAGD for computing the intersection curve of two quadric surfaces (e.g., [11], [13], [24], [25], [27]). The objectives of these algorithms are to classify the topological or geometric structure of the intersection curve and to derive its parametric representation; efficient collision detection is not their primary purpose.

3. Constructing a Separating Plane

In this section we show how to construct a separating plane of two disjoint ellipsoids.

Theorem 1. *Let* $\mathscr{A} : X^T AX = 0$ *and* $\mathscr{B} : X^T BX = 0$ *be two disjoint ellipsoids. Let* V_i *denote the four eigenvectors of* $-A^{-1}B$ *associated with the eigenvalues* λ_i, $i = 0, 1, 2, 3$. *Then their endpoints* V_i *form the vertices of a tetrahedron, denoted by*

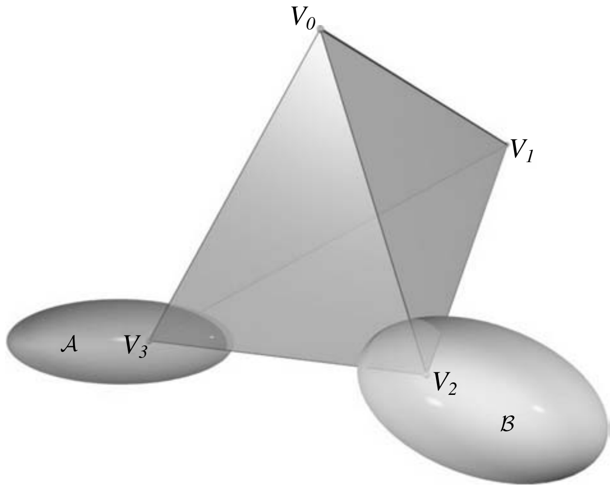

Fig. 2. The tetrahedron $[V_0V_1V_2V_3]$

$[V_0V_1V_2V_3]$, *that is self-polar for the ellipsoids* $\mathscr{A}$ *and* $\mathscr{B}$ [1]. *Furthermore*, V_0 *and* V_1 *are outside* $\mathscr{A}$ *and* $\mathscr{B}$, V_2 *is inside* $\mathscr{B}$, *and* V_3 *is inside* $\mathscr{A}$.

Proof: Since $\mathscr{A}$ and $\mathscr{B}$ are disjoint, by Proposition 1, $\det(\lambda A + B) = 0$ has two negative roots and two distinct positive roots: $\lambda_0 \leq \lambda_1 < 0 < \lambda_2 < \lambda_3$. When $\lambda_0 < \lambda_1$, the four eigenvalues of $-A^{-1}B$, which are equal to the roots of the quartic equation $f(\lambda) \equiv \det(\lambda A + B) = 0$, are distinct. Thus their corresponding eigenvectors V_i are linearly independent. From the equalities $(\lambda_i A + B)V_i = 0$ and $(\lambda_j A + B)V_j = 0$, $0 \leq i < j \leq 3$, it follows that $\lambda_i V_i^T A V_j + V_i^T B V_j = 0$ and $\lambda_j V_i^T A V_j + V_i^T B V_j = 0$, respectively. Since $\lambda_i \neq \lambda_j$, we have $V_i^T A V_j = V_i^T B V_j = 0$. When $\lambda_0 = \lambda_1$, it can easily be shown [26] that the eigenspace of $-A^{-1}B$ has dimension 2. Therefore, two linearly independent vectors V_0 and V_1 can be selected to be the eigenvectors associated with λ_0 such that $V_i^T A V_j = V_i^T B V_j = 0$ for $0 \leq i < j \leq 3$. Hence, the tetrahedron $[V_0V_1V_2V_3]$ is self-polar with respect to both $\mathscr{A}$ and $\mathscr{B}$ [19, 21]. This means that, for both ellipsoids, each of the points V_i is the pole of a plane which passes through the other three vertices of the tetrahedron $[V_0V_1V_2V_3]$ (see Fig. 2).

It follows from $(\lambda_0 A + B)V_0 = 0$ that $\lambda_0 V_0^T A V_0 + V_0^T B V_0 = 0$, since $\lambda_0 < 0$, $V_0^T A V_0$ and $V_0^T B V_0$ have the same sign. If $V_0^T A V_0 < 0$ and $V_0^T B V_0 < 0$, then V_0 would be inside both $\mathscr{A}$ and $\mathscr{B}$, and hence $\mathscr{A}$ and $\mathscr{B}$ would overlap. We deduce that $V_0^T A V_0 > 0$ and $V_0^T B V_0 > 0$: i.e. V_0 is outside $\mathscr{A}$ and $\mathscr{B}$. Similarly, V_1 is outside $\mathscr{A}$ and $\mathscr{B}$.

It follows from $(\lambda_3 A + B)V_3 = 0$ that $\lambda_3 V_3^T A V_3 + V_3^T B V_3 = 0$. Since $\lambda_3 > 0$, $V_3^T A V_3$ and $V_3^T B V_3$ have different signs. We are going to show that $V_3^T A V_3 < 0$, i.e. that V_3 is inside $\mathscr{A}$.

[1] $[V_0V_1V_2V_3]$ is a self-polar tetrahedron for a quadric $X^T A X = 0$, if $V_i^T A V_j = 0$ for $i \neq j$. See [19], page 272.

Clearly, there is a point P on the line V_2V_3 which is outside both $\mathscr{A}$ and $\mathscr{B}$. Thus $P = aV_2 + bV_3$ for some constants $a \neq 0$ and $b \neq 0$. Then $P^TAP = a^2V_2^TAV_2 + b^2V_3^TAV_3 > 0$, since P is outside $\mathscr{A}$. Similarly, $P^TBP = a^2V_2^TBV_2 + b^2V_3^TBV_3 > 0$. Since $(\lambda_2 A + B)V_2 = 0$ and $(\lambda_3 A + B)V_3 = 0$, we have $BV_2 = -\lambda_2 AV_2$ and $BV_3 = -\lambda_3 AV_3$. Substituting for BV_2 and BV_3 yields

$$\begin{aligned} P^TBP &= a^2V_2^TBV_2 + b^2V_3^TBV_3 \\ &= a^2V_2^T(-\lambda_2 AV_2) + b^2V_3^T(-\lambda_3 AV_3) \\ &= -\lambda_2(a^2V_2^TAV_2 + b^2V_3^TAV_3) + b^2(\lambda_2 - \lambda_3)V_3^TAV_3 \\ &= -\lambda_2 P^TAP + b^2(\lambda_2 - \lambda_3)V_3^TAV_3. \end{aligned}$$

Thus

$$V_3^TAV_3 = b^{-2}(\lambda_2 - \lambda_3)^{-1}(P^TBP + \lambda_2 P^TAP) < 0.$$

Hence, V_3 is inside $\mathscr{A}$. Similarly, it can be shown that V_2 is inside $\mathscr{B}$. This completes the proof.

Suppose that $\mathscr{A} : X^TAX = 0$ and $\mathscr{B} : X^TBX = 0$ are two disjoint ellipsoids. Since V_3 is inside $\mathscr{A}$, its polar plane $V_0V_1V_2$ does not intersect $\mathscr{A}$ [19]; and, since V_2 is inside $\mathscr{B}$, its polar plane $V_0V_1V_3$ does not intersect $\mathscr{B}$. Thus, the line V_0V_1, which is the intersection of the planes $V_0V_1V_2$ and $V_0V_1V_3$, does not intersect either $\mathscr{A}$ or $\mathscr{B}$. So there are two planes, $\mathscr{T}_L^A$ and $\mathscr{T}_R^A$, tangent to $\mathscr{A}$ that pass through V_0V_1 (see Figure 3(a)).

Let P_L^A and P_R^A denote the points at which $\mathscr{T}_L^A$ and $\mathscr{T}_R^A$ touch $\mathscr{A}$. Clearly, $\mathscr{A}$ is contained entirely between the planes $\mathscr{T}_L^A$ and $\mathscr{T}_R^A$. Since the tetrahedron $[V_0V_1V_2V_3]$ is self-polar with respect to $\mathscr{A}$, the line V_0V_1 is conjugate with the line V_2V_3. Hence, the points P_L^A and P_R^A are on the line V_2V_3, i.e. the line segment $[P_L^AP_R^A]$ is contained in V_2V_3. Let P_L^B and P_R^B denote the points where the two tangent planes $\mathscr{T}_L^B$ and $\mathscr{T}_R^B$ that pass through V_0V_1 touch the ellipsoid $\mathscr{B}$. By a similar argument to that just used for ellipsoid $\mathscr{A}$, we can show that $\mathscr{B}$ is contained between the planes $\mathscr{T}_L^B$ and $\mathscr{T}_R^B$, and the line segment $[P_L^BP_R^B]$ is contained in the line V_2V_3.

Since $\mathscr{A}$ and $\mathscr{B}$ are disjoint, the line segments $[P_L^AP_R^A]$ and $[P_L^BP_R^B]$ are disjoint. Let the four tangent points P_L^A, P_R^A, P_L^B, and P_R^B be labeled such that the open line segment $(P_R^AP_L^B)$ is outside $\mathscr{A}$ and $\mathscr{B}$ (see Figure 3). Clearly, any plane passing through V_0V_1 and intersecting $(P_R^AP_L^B)$ does not intersect $\mathscr{A}$ or $\mathscr{B}$ and lies between $\mathscr{A}$ and $\mathscr{B}$; hence, this plane separates $\mathscr{A}$ and $\mathscr{B}$.

Therefore, we have proved

Theorem 2. *Any plane passing through the line V_0V_1 and intersecting the open line segment $(P_R^AP_L^B)$ is a separating plane of $\mathscr{A}$ and $\mathscr{B}$* (*see Figure* 3(b)). *In particular*,

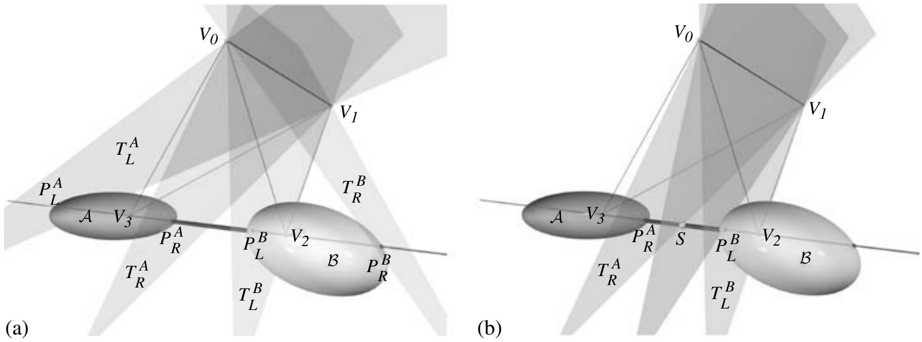

Fig. 3. (a) Four planes passing through V_0 and V_1: $\mathscr{T}_L^A$ and $\mathscr{T}_R^A$ are tangent to ellipsoid $\mathscr{A}$; $\mathscr{T}_L^B$ and $\mathscr{T}_R^B$ are tangent to ellipsoid $\mathscr{B}$; (b) A separating plane is one that passes through V_0, V_1 and S, where S can be any point on the open line segment $(P_R^A P_L^B)$

the plane passing through V_0, V_1, *and* P_R^A *touches* $\mathscr{A}$ *at* P_R^A, *and the plane passing through* V_0, V_1, *and* P_L^B *touches* $\mathscr{B}$ *at* P_L^B.

Now we will consider the computational procedure for obtaining a separating plane for two disjoint ellipsoids $\mathscr{A}$ and $\mathscr{B}$. Using Proposition 1, we first compute the four real roots of the characteristic equation $\mathrm{f}(\lambda) = \det(\lambda A + B) = 0$: two of the roots are negative and the other two are distinct positive roots, and can be labeled according to the inequalities $\lambda_0 \le \lambda_1 < 0 < \lambda_2 < \lambda_3$. Since $\det(\lambda A + B) = \det(A)\det(\lambda I + A^{-1}B)$, these roots λ_i are also the eigenvalues of the matrix $-A^{-1}B$. The eigenvectors V_2 and V_3 are obtained by solving the equations $(\lambda_i I + A^{-1}B)X = 0$, $i = 2, 3$. Next we compute the points P_R^A and P_L^B where the line V_2V_3 intersects the ellipsoids $\mathscr{A}$ and $\mathscr{B}$ (see Figure 3). Then we obtain the tangent plane $\mathscr{T}_R^A$ of $\mathscr{A}$ at P_R^A and the tangent plane $\mathscr{T}_L^B$ of $\mathscr{B}$ at P_L^B. Since, by Theorem 2, $\mathscr{T}_R^A$ and $\mathscr{T}_L^B$ intersect on the line V_0V_1, a separating plane can be found by taking an appropriate linear combination of the equations of $\mathscr{T}_R^A$ and $\mathscr{T}_L^B$.

4. The Complete Algorithm

Based on the results discussed in the previous sections, we present below an algorithm for detecting collision between two moving ellipsoids. Figure 4 gives a schematic description of the algorithm.

Suppose that two ellipsoids $\mathscr{A}_{i-1}$ and $\mathscr{B}_{i-1}$ at frame $i-1$ are disjoint. The separating plane is computed and becomes the *candidate separating plane* at the next frame, i. Under continuous motion, the positions and orientations of the moving ellipsoids may be expected to change little between frame $i-1$ and frame i. Thus, in most cases, the candidate separating plane may still separate $\mathscr{A}_i$ and $\mathscr{B}_i$, and this can be verified efficiently. If the candidate separating plane at frame i does indeed separate $\mathscr{A}_i$ and $\mathscr{B}_i$, then a great deal of time is saved by avoiding solving the characteristic equation of $\mathscr{A}_i$ and $\mathscr{B}_i$, which is a relatively expensive procedure; otherwise, we need to compute the roots of the

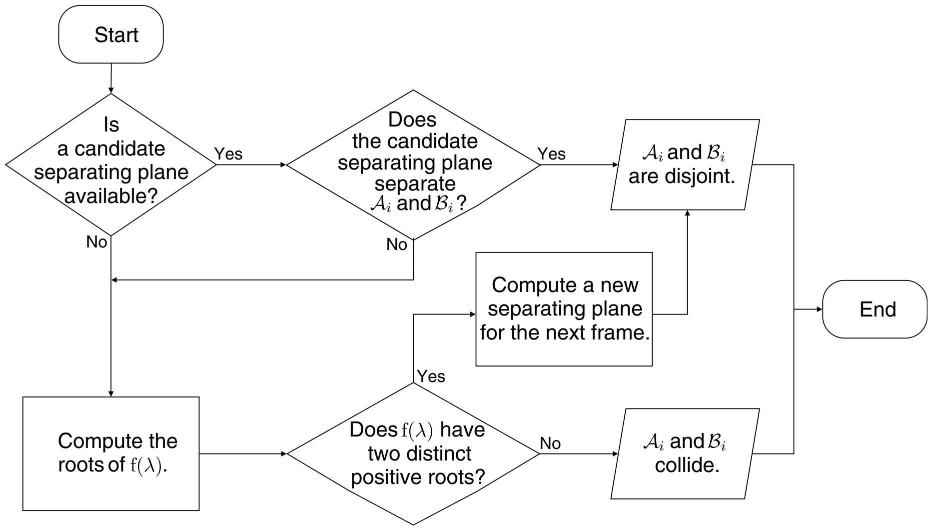

Fig. 4. Algorithm for collision detection between two ellipsoids $\mathscr{A}_i$ and $\mathscr{B}_i$ at frame i

characteristic equation, either to find that the ellipsoids collide, or to find that they are disjoint and then to compute a new separating plane. The algorithm is efficient because the non-collision case occurs much more frequently than the collision case, due to inter-frame coherence. It is also possible to maintain more than one candidate separating plane, so as to increase the likelihood of detecting disjoint ellipsoids.

A plane separates two ellipsoids if and only if (i) the plane does not intersect either of the ellipsoids; and (ii) the centers of the two ellipsoids are on the opposite sides of the plane. Since it is relatively easy to test the second condition, the problem essentially reduces to that of testing whether a plane intersects an ellipsoid. Note that an ellipsoid can be transformed to the unit sphere by an affine transformation, which is normally available as the inverse of the motion matrix. Applying this transformation, the problem is simplified to one of detecting the intersection between a plane and a unit sphere centered at the origin. The problem can be further reduced to one of finding the distance between the origin and a plane.

The availability of a separating plane not only simplifies the collision test in many frames due to inter-frame coherence, but also provides useful geometric information about the relative positions of the ellipsoids. For example, the tangent points P_R^A and P_L^B serve as good approximations to the pair of mutually closest points on the two ellipsoids; in fact, if the ellipsoids are actually spheres, then P_R^A and P_L^B are exactly the closest points. Furthermore, when two ellipsoids touch each other externally (i.e. $\lambda_2 = \lambda_3$), P_R^A and P_L^B merge into the contact point, which provides very useful information for computing collision impulses.

5. Experimental Results

In this section we shall demonstrate the effectiveness of the separating plane in speeding up our method, and also compare our method with the GJK method [7]. For brevity, our new algorithm will be referred to as EECD (for Exact Ellipsoid Collision Detection). To demonstrate the improved performance of EECD achieved by using the separating plane, two ellipsoids of the same size (with principal axes of half-lengths 3, 3 and 5) are made to rotate continuously about their centers for 10,000 frames. The distance between the two centers is 8. Three sets of experiments are carried out in which the ellipsoids rotate about a random axis with low, medium and high angular speeds (angular increments of 0.1, 0.5 and 1 radians per frame). The simulation was run on a PC equipped with a Pentium 4 2.26GHz processor and the results are shown in Table 1.

When the ellipsoids rotate at the low angular speed, the orientations of the ellipsoids between successive frames differ only by a little and we see that the use of separating planes allows the algorithm to identify as many as 8,310 separations out of the overall 8,798 separations. As a consequence, the time taken for collision detection is significantly reduced. As the angular speed increases, there are fewer cases where the candidate separating plane reports separation, but a considerable amount of time is still saved. EECD takes on average less than 5 microseconds per frame to detect collisions between two ellipsoids. It is also obvious that, when the distance between the two ellipsoids increases, more candidate separating planes will remain valid and therefore the benefit of using separating planes becomes more remarkable. Optimal performance is achieved when the candidate separating plane remains valid for all frames.

The same sets of experiments are performed using a collision detection scheme that solves the characteristic equation in every frame without using separating planes. It takes about 8 microseconds to detect collision between two ellipsoids, longer than that needed by EECD. We note that the running time is insensitive to the number of collisions and separations in each test. This is because the same amount of computation is required no matter whether the ellipsoids are disjoint or overlapping.

Another experiment was conducted to compare EECD with the enhanced GJK method, which has been described [14] as one of the most robust and efficient collision detection methods for convex polyhedra currently available. We have

Table 1. Experimental results demonstrating the effect of using the separating plane in EECD

Angular velocity	Low	Medium	High
Number of collisions (out of 10,000 frames)	1,202	1,554	1,622
Number of separations (out of 10,000 frames)	8,798	8,446	8,378
Number of separations reported by separating plane tests	8,310	6,029	4,502
Time per frame (μs) using the separating plane, averaged over 10,000 frames (EECD)	2.474	3.985	4.982
Time per frame (μs) without using the separating plane, averaged over 10,000 frames	7.906	7.951	7.899

adapted the routines for the enhanced GJK method[2] by removing the code for distance computation, thus improving their efficiency for collision detection.

The experiment is set up as follows (see Fig. 5). A sphere $\mathscr{A}(t)$ of radius 1.0 orbits along a circular path around an ellipsoid $\mathscr{B}$. The half-lengths of the three principal axes of $\mathscr{B}$ are 4.5, 4, and 2, respectively. The path of $\mathscr{A}(t)$ lies on the plane determined by the two longer principal axes of $\mathscr{B}$. $\mathscr{A}(t)$ and $\mathscr{B}$ collide only near the two ends of the longest principal axis of $\mathscr{B}$, as shown by bright spheres in Fig. 5. The central angle subtended by the displacement of $\mathscr{A}(t)$ between two consecutive frames is 0.01 radians. For each frame, the bounding boxes of $\mathscr{A}(t)$ and $\mathscr{B}$ are first checked. If they do not intersect, separation is reported, shown by framed boxes bounding $\mathscr{A}(t)$ in the figure; otherwise, a collision detection procedure, either EECD or enhanced GJK, is called.

Out of a total of 13,746 frames, the bounding boxes of $\mathscr{A}(t)$ and $\mathscr{B}$ intersect in 10,000 frames, and $\mathscr{A}(t)$ collides with $\mathscr{B}$ in 1,157 of these frames. Collision detection for the 10,000 frames using EECD takes 0.0212 seconds. Among the 8,843 frames in which $\mathscr{A}(t)$ and $\mathscr{B}$ are disjoint, 8,020 frames are detected using

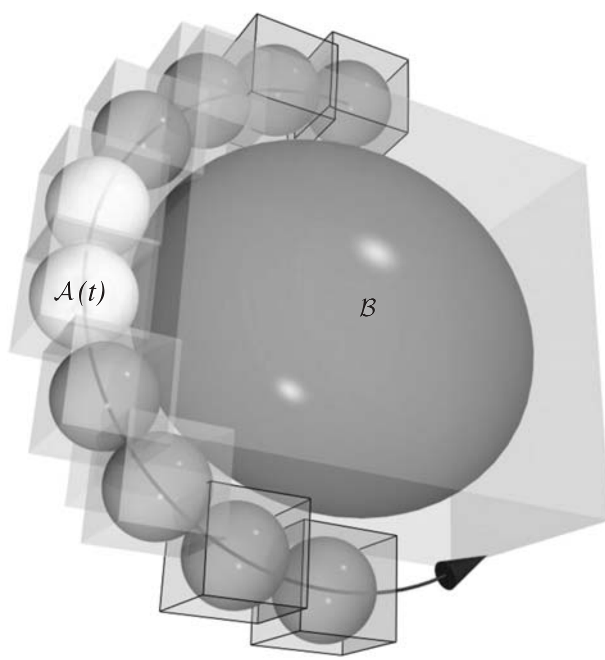

Fig. 5. Experimental set-up corresponding to the results in Table 2: a sphere $\mathscr{A}(t)$ moves along a circular orbit around an ellipsoid $\mathscr{B}$. Bright spheres indicate collision. Framed boxes bounding $\mathscr{A}(t)$ are disjoint from the bounding box of $\mathscr{B}$

[2]The enhanced GJK routines are due to Stephen Cameron and are available at http://users.comlab.ox.ac.uk/stephen.cameron/distances.html.

Table 2. Computation time for our algorithm, EECD, and for the enhanced GJK method, with a varying number of vertices (n) used in the latter for polyhedral approximation of the ellipsoids

	EECD	Enhanced GJK			
		$n = 488$	$n = 1{,}460$	$n = 4{,}376$	$n = 13{,}124$
Collision detected (frames)	1,157	0	435	971	1,111
Separation detected (frames)	8,843	10,000	9,565	9,029	8,889
Time per frame (μs), averaged over 10,000 frames	2.12	15.2	23.6	36.8	54.3

separating planes. As a comparison, without using separating planes, it takes 0.0763 seconds for the same set-up. Thus the use of separating planes reduces the computational time by a factor of more than 3 in this example. As explained earlier, this improvement is specific in this particular example, and varies with the orbit taken by $\mathscr{A}(t)$. In an extreme case, the two ellipsoids would collide in all frames and the separating planes may offer no benefit.

The enhanced GJK method was also run on the same set-up, with different numbers of vertices (n) used for approximating $\mathscr{A}(t)$ and $\mathscr{B}$. The results are summarized in Table 2.

The results show that EECD is about an order of magnitude quicker than the enhanced GJK method. When the ellipsoids are coarsely faceted, the enhanced GJK method, is also fast; however when $n = 488$, it incorrectly reports collision as separation in all frames. The number of misses reduces with increased resolution; when 13,124 vertices are used for polyhedral approximation, there are errors in only 46 frames, but the computation time per frame is now increased from 15.2 microseconds to 54.3 microseconds. This trade-off between accuracy and efficiency is intrinsic to any collision detection methods based on polyhedral approximation.

6. Conclusions

We have presented an efficient algorithm for detecting collision among ellipsoids. It is based on a simple algebraic condition on the separation of two ellipsoids, and uses a separating plane to exploit frame coherence between moving ellipsoids. Our algorithm is simple to implement and has high accuracy, since it does not rely on polyhedral approximations. The central idea behind this new algorithm can be extended to other types of quadric surfaces or objects modeled as unions of ellipsoids in a straightforward manner. A related problem for future research is to devise efficient methods for detecting collisions among CSG models of quadric primitives.

Acknowledgement

We would like to thank anonymous referees for their invaluable comments. This work is supported in part by research grants from the Research Grant Council of the Hong Kong SAR (HKU 7033/98E, HKU 7032/00E, HKU 7153/03E), and in part by grant No. R01-2002-000-00512-0 from the Basic Research Program of the Korea Science and Engineering Foundation (KOSEF).

References

[1] Baraff, D.: Curved surface and coherence for non-penetrating rigid body simulation. Proc. SIGGRAPH' 90, pp. 19–28, 1990.
[2] Bischoff, S., Kobbelt, L.: Ellipsoid decomposition of 3D-models. Proc. 1st Symp. on 3D data processing, visualization and transmission, pp. 480–488, IEEE Press, 2002.
[3] Bromwich, T. J.: Quadratic forms and their classification by means of invariant-factors. Cambridge Tracts in Mathematics and Mathematical Physics 3, 1906.
[4] Cohen, J., Lin, M., Manocha, D., Ponamgi, M. K.: I-COLLIDE: An interactive and exact collision detection system for large-scale environments. Proc. SIGGRAPH Symp on Interactive 3D Graphics, pp. 189–196, 1995.
[5] Eberly, D.: 3D game engine design. San Francisco: Morgan Kaufmann 2001.
[6] Gilbert, E. G., Foo, C. P.: Computing the distance between general convex objects in three dimensional space. IEEE Trans. on Robotics and Automation *6*, 53–61 (1990)
[7] Gilbert, E. G., Johnson, D. W., Keerthi, S. S.: A fast procedure for computing the distance between complex objects in three dimensional space. IEEE Trans. on Robotics and Automation *4*, 193–203 (1988).
[8] Gottschalk, S., Lin, M. C., Manocha, D.: OBBTree: A hierarchical structure for rapid interference detection. Proc. of SIGGRAPH '96. pp. 171–180, 1996.
[9] Hubbard, P. M.: Approximating polyhedra with spheres for time critical collision detection. ACM Transactions on Graphics. *15*, 179–210 (1996).
[10] Lennerz, C., Schömer, E.: Efficient distance computation for quadratic curves and surfaces. Proc. of geometric modeling and processing, pp. 60–69. Saitama, Japan, July 10–12, 2002.
[11] Levin, J. Z.: Mathematical models for determining the intersections of quadric surfaces. Comp. Graph. Image Process *1*, 73–87 (1979).
[12] Lin, X., Ng, T. T.: Contact detection algorithms for three-dimensional ellipsoids in discrete element modeling. Int. J. Num. Anal. Meth. Geomech. *19*, 653–659 (1995).
[13] Miller, J. R.: Geometric approaches to nonplanar quadric surface intersection curves. ACM Trans. on Graphics *6*, 274–307 (1987).
[14] Mirtich, B.: V-Clip: fast and robust polyhedral collision detection. ACM Trans. on Graphics *17* (3), 177–208 (1998).
[15] Palmer, I., Grimsdale, R.: Collision detection for animation using sphere-trees. Comp. Graphics Forum *14* (2), 105–116 (1995).
[16] Perram, J. W., Wertheim, M. W.: Statistical mechanical of hard ellipsoids. I. Overlap algorithm and the contact function. J. Comput. Phys. *58*, 409–416 (1985).
[17] Rimon, E., Boyd, S.P.: Obstacle collision detection using best ellipsoid fit. J. Intelligent and Robotic Sys. *18*, 105–126 (1997).
[18] Schneider, P., Eberly, D.: Geometric tools for computer graphics. San Francisco: Morgan Kaufmann 2003.
[19] Semple, G. J., Kneebone, G. K.: Algebraic projective geometry. London: Oxford University Press 1952.
[20] Sohn, K.-A., Jüttler, B., Kim, M.-S., Wang, W.: Computing distances between surfaces using line geometry. Proc. of Pacific Graphics, pp. 236–245. Beijing, China, October 9–11, 2002.
[21] Sommerville, D. M. Y.: Analytical geometry of three dimensions. Cambridge University Press 1947.
[22] Tu, C. H., Wang, W., Wang, J. Y.: Classifying the morphology of the nonsingular intersection curves of two quadric surfaces. Proc. Geometric Modeling and Processing, pp. 23–32, 2002.
[23] Wang, W.: Modelling and processing with quadric surfaces. Handbook of computer aided geometric design, pp. 777–795. Elsevier 2002.
[24] Wang, W., Joe, B., Goldman, R.: Computing quadric surface intersections based on an analysis of plane cubic curves. Graphical Models *64*, 335–367 (2002).
[25] Wang, W., Goldman, R., Tu, C. H.: Enhancing Levin's method for computing quadric-surface intersections. Computer Aided Geometric Design *20* (7), 401–422 (2003).

[26] Wang, W., Wang, J. Y., Kim, M.-S.: An algebraic condition on the separation of two ellipsoids. Computer Aided Geometric Design *18* (6), 531–539 (2001).
[27] Wilf, I., Manor, Y.: Quadric surface intersection: shape and structure. Computer Aided Design *25* (10), 633–643 (1993).

Wenping Wang
Yi-King Choi
Bin Chan
Department of Computer Science
and Information Systems
The University of Hong Kong
Hong Kong
China

Myung-Soo Kim
School of Computer Science
and Engineering
Institute of Computer Technology
Seoul National University
Seoul
Korea

Jiaye Wang
Department of Computer Science
Shandong Uinversity
Jinan
China

Verleger: Springer-Verlag KG, Sachsenplatz 4–6, A-1201 Wien. – Herausgeber: Prof. Dr. Wolfgang Hackbusch, Max-Planck-Institut für Mathematik in den Naturwissenschaften, Inselstraße 22–26, D-04103 Leipzig – Satz und Umbruch: Scientific Publishing Services (P) Ltd., Madras; Offsetdruck: Manz Crossmedia. 1051 Wien. – Verlagsort: Wien. – Herstellungsort: Wien. – Printed in Austria.