# Improvements to Ellipsoidal Fit Based Collision Detection

Yu-Ren Chien and Jing-Sin Liu

Institute of Information Science

Academia Sinica

Email: `{yrchien,liu}@iis.sinica.edu.tw`

# Abstract

In the collision detection literature, use of ellipsoidal fits has been successfully made to accelerate the detection beyond polyhedron based methods [12], [4]. In this paper, we propose some improvements to the state-of-the-art work of this methodology [4]. These improvements are made in the following three respects that are crucial to the overall performance of a collision detector. First, object-modeling robustness is enhanced by adopting a recent reliable algorithm for computing the maximum-volume inscribed ellipsoid [14]. Second, detecting efficiency is furthered by avoiding reference to polyhedral models. Third, detecting accuracy is also improved by a correction of ellipsoidal overlap checking. These improvements have been verified by extensive numerical experiments using randomly generated convex polyhedra.

# 1 Introduction

In robotics and computer graphics, it is frequently required to automatically generate a path for mobile objects to move from one configuration to another. In order for the generated path to be practical, overlaps among all mobile or stationary object models have to be avoided. These overlaps are what we mean by collisions in this paper. A collision detector helps a path planner to find a collision-free path to the desired configuration. Since collision detection is to be performed for each intermediate configuration along each candidate path, the efficiency of the collision detector greatly determines that of the path planner. Therefore, collision detection should be as efficient as possible, especially in the case of real-time path planning.

Collision detection is closely related to distance computation; when a distance-computing algorithm takes into account the zero distance between overlapping objects, collision detection can be achieved by checking a zero distance reported by the algorithm. With objects modeled as unions of convex polyhedra, an efficient algorithm for distance computation without separation assumption was developed by Gilbert *et al.* [2], known as the GJK algorithm. Later, full advantage of the polyhedral convexity was taken by two distance-computing algorithms to exhibit constant-time complexity: One is a slightly modified version [1] of the GJK algorithm and thus inherits the applicability to collision detection; the other assumes the separation between convex polyhedra and uses Voronoi regions to search for the closest feature pair [8].

In the collision detection literature, most of the recent works focus on the use of *bounding volumes* or *spatial subdivisions* to further speed up collision detection. A bounding volume for an object model is a simple geometric primitive that contains the model; a spatial subdivision decomposes the space occupied by each object model into a three-dimensional array of repeating cells. The simplicity of bounding volumes always makes overlap checking among them extremely fast. Before exact collision checks are performed for objects, overlaps among their bounding volumes or cell collections are first checked, thereby saving the collision check between each pair of objects whose respective bounding volumes or cell collections are disjoint. For instance, Hubbard [3] assumes bounded acceleration and uses a space-time bound for the upcoming motion of each object to focus on the objects that are likely to collide. He also uses a hierarchy of approximating spheres to perform collision detection with scalable accuracy. Ponamgi *et al*. [10] use axis-aligned bounding boxes to prune down the number of collision checks, extend the distance calculation algorithm in [8] to collision (penetration) detection between convex hulls of nonconvex polyhedral models by introducing pseudo-internal Voronoi regions, and exploit a hierarchy of, again, axis-aligned bounding boxes to prune down the number of overlap checks involving cavity features. Klosowski *et al*. [7] proposed an algorithm for checking overlap between object boundaries. They construct for the boundary of each polyhedral model a hierarchy of convex polytopes that have face normals from a small fixed set of orientations and respectively bound some subsets of the boundary up to its component triangles, so that the number of pairs of triangles that need to be checked for contact can be reduced. For the simulation of particle systems, Kim *et al*. [6] developed an approach to collision detection and reaction among spherically modeled objects, where the space containing the moving spheres is decomposed into a hierarchy of 3D grids to localize collision checks for acceleration.

In contrast with most of the aforementioned works, our research has been focused on the way how the underlying collision detector can be accelerated through approximation of object shapes. In other words, we are interested in *approximate collision detection*, which is intended to be faster than exact methods. This focus is actually shared with Hubbard [3], who used a hierarchy of approximating spheres to trade detection accuracy for efficiency. Our goal has been to approximate a convex polyhedron with an ellipsoid, which is described by a single quadratic inequality and specified by invariably nine parameters—three for its center, three for its axis lengths, and the other three for its orientation. When overlap checking is done for approximating ellipsoids instead of polyhedra, the analytic nature of this approximating volume can save the collision detector from searching over such numerous features as done for polyhedra. This promises dramatic enhancement in the efficiency of collision detection. Use of this approximating volume was first made by Rimon and Boyd in [11], where the minimum-volume circumscribed ellipsoids (MVCEs) are computed for convex polyhedra. Although they dealt with distance estimation assuming separation between the polyhedra, their work did inspire our research on ellipsoidal fit based collision detection.

Besides the MVCE, a closely-related ellipsoidal fit to a convex polyhedron is the maximum-volume inscribed ellipsoid (MVIE) [5]. These two fits, considered together, provide inner as well as outer bounds for the polyhedral boundary and can serve as observations for the unbiased detection of polyhedral overlaps. This idea was fulfilled first by Shiang *et al*. in [12] and later by Ju *et al*. in [4]; however, the online computation in [4] involves a search over polyhedral faces, as is opposed to the spirit of approximate collision detection and introduces a great degradation to the efficiency.

In this paper, we propose some improvements, both in accuracy and in efficiency, to the detector proposed in [4] and reviewed in Section 2. A similar object model is given in Section 3, with ellipsoidal elements computed by innovated methods from some recent advances in optimization. Reference to the polyhedral model is completely prevented in the new design of the collision detector described in Section 4, as makes the main contribution to the improved efficiency. On the other hand, the improved accuracy results from a correction of the ellipsoidal overlap checking in [4]. Some implementation details and experimental results are documented in Section 5, where two variants of the new detector, obtained from replacing the exact ellipsoidal distance with its estimate, are also shown to be much more efficient with almost the same accuracy. We conclude this paper by summarizing the modifications made in this work for improving the overall performance of ellipsoidal fit based collision detection.

# 2 Review of Previous Work

Consider the overlap detection between two convex polyhedra, $\mathcal{P}_1$ and $\mathcal{P}_2$. Denote the enclosing ellipsoid and the enclosed ellipsoid of $\mathcal{P}_1$ and those of $\mathcal{P}_2$ by $\mathcal{E}_1^c$, $\mathcal{E}_1^i$, $\mathcal{E}_2^c$, and $\mathcal{E}_2^i$, respectively. Let the near points, respectively, computed [11] for $\mathcal{E}_1^i$ and $\mathcal{E}_2^i$ be $\mathbf{v}_1^i$ and $\mathbf{v}_2^i$. The point at which the ray emanating from $\mathbf{v}_1^i$ [resp. $\mathbf{v}_2^i$] toward $\mathbf{v}_2^i$ [resp. $\mathbf{v}_1^i$] intersects the boundary of $\mathcal{E}_1^c$ [resp. $\mathcal{E}_2^c$], denoted $\mathbf{v}_1^c$ [resp. $\mathbf{v}_2^c$], is computed by solving a quadratic equation. The point at which the ray emanating from $\mathbf{v}_1^i$ [resp. $\mathbf{v}_2^i$] toward $\mathbf{v}_2^i$ [resp. $\mathbf{v}_1^i$] intersects the boundary of $\mathcal{P}_1$ [resp. $\mathcal{P}_2$], denoted $\mathbf{v}_1$ [resp. $\mathbf{v}_2$], is computed

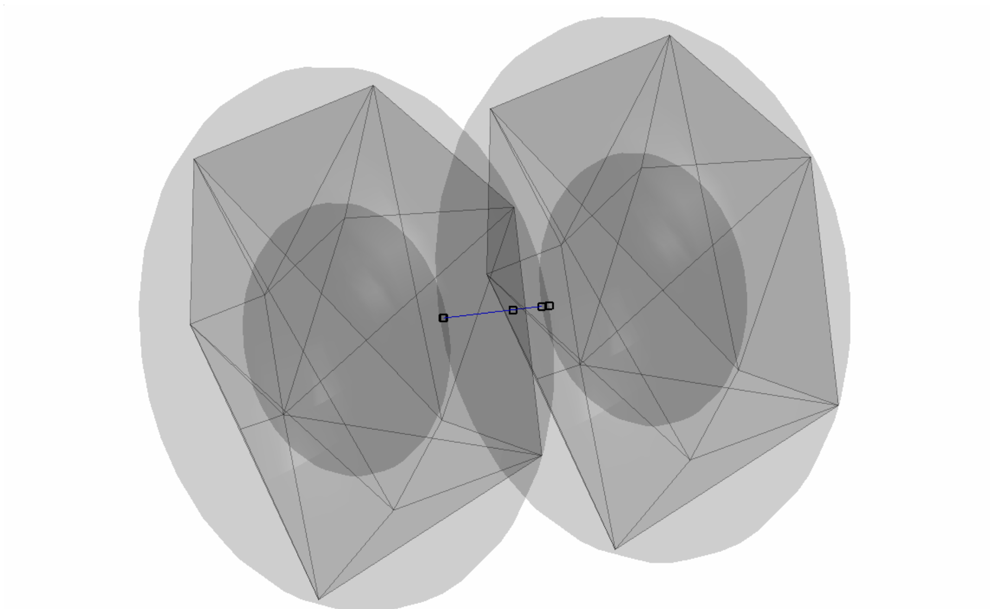

Figure 1: Illustrating the points defined for our previous work. The ellipsoids on the left are $\mathcal{E}_1^c$ and $\mathcal{E}_1^i$, and those on the right are $\mathcal{E}_2^c$ and $\mathcal{E}_2^i$. Visually from left to right, the first marker is shared by $\mathbf{v}_2^c$ and $\mathbf{v}_1^i$, the second by $\mathbf{v}_2$ and $\mathbf{v}_1$, the third represents $\mathbf{v}_2^i$, and the fourth $\mathbf{v}_1^c$.

by solving a linear equation for each face of $\mathcal{P}_1$ [resp. $\mathcal{P}_2$]. See Figure 1 for an illustration of the points defined above. An overlap is reported if and only if

$$\left(\mathbf{v}_1^c - \mathbf{v}_2^c\right) \cdot \left(\mathbf{v}_1 - \mathbf{v}_2\right) \leq 0$$

and

$$\left\|\mathbf{v}_1 - \mathbf{v}_2\right\| \leq \min\left(\left\|\mathbf{v}_1^c - \mathbf{v}_1\right\|, \left\|\mathbf{v}_2^c - \mathbf{v}_2\right\|\right).$$

# 3 The Object Model

Suppose that each object has already been modeled as a union of convex polyhedra. We append to the object model two ellipsoidal fits to each component polyhedron, i.e., the MVIE and the MVCE. As the configuration for mobile objects changes, these fits can be directly updated by corresponding rotations and translations, without recomputing from the updated polyhedra. As stated in [5], computation of the MVIE is the more fundamental problem because a problem of computing the MVCE can be reduced to one of computing the MVIE for a polyhedron in four-dimensional space (see Appendix B). A numerically reliable algorithm developed in [14] is adopted here for this key problem. Note that the enclosed ellipsoid heuristically computed in [4] is far from optimal. Also notice that the ellipsoid algorithm adopted in [11] for computing the minimum-volume enclosing ellipsoid fails to guarantee an answer with tolerable error, for the algorithm is numerically unreliable and always diverges after a large number of iterations.

# 4 The Collision Detector

The collision between two objects is detected by checking the overlap between any pair of respective component polyhedra of them. In turn, in the following two cases, the overlap between two polyhedra can be exactly detected by checking that between their MVCEs or between their MVIEs. One is that the MVCEs are separate, where the polyhedra are certainly non-overlapping; the other is that the MVIEs overlap, where the polyhedra collide for sure. Ellipsoidal overlap

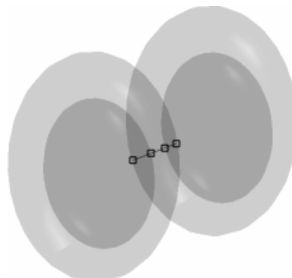

Figure 2: Illustrating the penetration between MVCEs. The ellipsoids on the left are $\mathcal{E}_1^c$ and $\mathcal{E}_1^i$, and those on the right are $\mathcal{E}_2^c$ and $\mathcal{E}_2^i$. The four marked points are, from left to right, $\mathbf{v}_2^c$, $\mathbf{v}_1^i$, $\mathbf{v}_2^i$, and $\mathbf{v}_1^c$.

checking can be performed by a straightforward extension of the ellipsoidal distance estimation results in [11] (see Appendix A), or alternatively by detecting a zero distance between two ellipsoids [9].

Apart from the aforementioned two cases, the other case is that the MVCEs overlap while the MVIEs are separate, where polyhedral overlap is ambiguous. However, if the distance (possibly negative) between the two polyhedra can be estimated by observing the proximity/penetration between the two MVIEs/MVCEs, the overlap can still be reasonably detected. Denote the MVCE and the MVIE of one polyhedron and those of the other polyhedron by $\mathcal{E}_1^c$, $\mathcal{E}_1^i$, $\mathcal{E}_2^c$, and $\mathcal{E}_2^i$, respectively. Let the nearest points, respectively, computed [9] for $\mathcal{E}_1^i$ and $\mathcal{E}_2^i$ be $\mathbf{v}_1^i$ and $\mathbf{v}_2^i$. To roughly measure the penetration depth between $\mathcal{E}_1^c$ and $\mathcal{E}_2^c$, the point at which the ray emanating from $\mathbf{v}_1^i$ [resp. $\mathbf{v}_2^i$] toward $\mathbf{v}_2^i$ [resp. $\mathbf{v}_1^i$] intersects the boundary of $\mathcal{E}_1^c$ [resp. $\mathcal{E}_2^c$], denoted $\mathbf{v}_1^c$ [resp. $\mathbf{v}_2^c$], is computed by solving a quadratic equation. When the intersection among $\mathcal{E}_1^c$, $\mathcal{E}_2^c$, and the line passing through $\mathbf{v}_1^i$ and $\mathbf{v}_2^i$, is nonempty, the distance between $\mathbf{v}_1^c$ and $\mathbf{v}_2^c$ serves as an estimate for the depth (see Figure 2), and a *negative distance estimate* with magnitude equal to this depth estimate can be assigned to $\mathcal{E}_1^c$ and $\mathcal{E}_2^c$. As a rule of thumb, the distance between the two polyhedra is estimated by

$$f(\mathbf{v}_1^i,\mathbf{v}_2^i,\mathbf{v}_1^c,\mathbf{v}_2^c)=\begin{cases}\frac{\|\mathbf{v}_1^i-\mathbf{v}_2^i\|+\|\mathbf{v}_1^c-\mathbf{v}_2^c\|}{2}, & (\mathbf{v}_1^i-\mathbf{v}_2^i)\cdot(\mathbf{v}_1^c-\mathbf{v}_2^c)\geq 0\\ \frac{\|\mathbf{v}_1^i-\mathbf{v}_2^i\|-\|\mathbf{v}_1^c-\mathbf{v}_2^c\|}{2}, & (\mathbf{v}_1^i-\mathbf{v}_2^i)\cdot(\mathbf{v}_1^c-\mathbf{v}_2^c)< 0\end{cases},$$

where "$\cdot$" denotes the inner product of two vectors. The resulting overall design of the polyhedral overlap detector is depicted in Figure 3.

# 5 Implementation and Results

To evaluate the performance of the improved collision detector, we consider a testing dataset with one thousand pairs of objects, for each of which collision detection is to be performed. Of each pair of objects, one is a convex polyhedron with vertices generated by randomly sampling twenty points from a unit ball centered at the origin and discarding those not in any subset that determines a plane having all the twenty points on the same side; the other is generated in the same way, except that the unit ball is shifted in the direction $[1,1,1]$ by a random number in the interval $[0,2]$. As for the ellipsoidal model of each object, the MVIE problem is solved using the package MVE [13], with the resulting volume no less than $e^{-0.0001}$ times the maximum. With this dataset, experimental results are documented below for four distinct detectors, which are implemented in MATLAB 6.5 on an Intel P4 1800-MHz PC running Redhat Linux 7.1 with 1-GB memory.

| Detector Output | Collision | No Collision |
|---|---|---|
| Collision | 375 | 32 |
| No Collision | 167 | 426 |

(a)

| Detector Output | Collision | No Collision |
|---|---|---|
| Collision | 50 | 32 |
| No Collision | 5 | 150 |

(b)

Table 1: Confusion matrices for the first collision detector: (a) showing all the 1000 results (average computation time: 100 ms per check); (b) focusing on those in the ambiguous case (average computation time: 146 ms per check).

The first detector to be evaluated is the one proposed in [4]. Table 1(a) and Table 1(b) show, respectively, the confusion matrix for all 1000 detection results and that focusing on the case where polyhedral overlap is ambiguous in terms of ellipsoidal overlap, i.e., where the enclosing ellipsoids overlap while the enclosed ones are separate. Since ellipsoidal overlap checking is supposed to be exact, all the detection errors are supposed to occur in the ambiguous case. Comparison between the respective $(2,1)$ entries in the two matrices reveals that the criterion used in [4] to check the overlap between enclosing ellipsoids is incorrect. The average computation time of this detector is 100 milliseconds per object pair.

The second detector is designed as described in Section 4. The algorithm in [9] is applied to ellipsoidal overlap checking and computation of $\mathbf{v}_1^i$ and $\mathbf{v}_2^i$, tolerating a nonzero angle less than 0.01 radians between the outward-directing

| Detector Output | Collision | No Collision |
|---|---|---|
| Collision | 515 | 23 |
| No Collision | 27 | 435 |

(a)

| Detector Output | Collision | No Collision |
|---|---|---|
| Collision | 134 | 23 |
| No Collision | 24 | 136 |

(b)

Table 2: Confusion matrices for the second collision detector: (a) showing all the 1000 results (average computation time: 46 ms per check); (b) focusing on those in the ambiguous case (samples plotted in Figure 4; average computation time: 75 ms per check).

tangent-plane normal at each near point and the vector from this point to the other near point. Two confusion matrices are similarly shown in Table 2, where only three errors are caused by the finite precision of ellipsoidal overlap checking. Figure 4 visualizes respective samples from the four kinds of results in Table 2(b). The average computation time of this detector is 46 milliseconds per pair.

The third detector is obtained from two modifications to the second detector. One is the replacement of $\mathbf{v}_1^i$ [resp. $\mathbf{v}_2^i$] with the near point at which the ellipsoidal level surfaces surrounding $\mathcal{E}_2^i$ [resp. $\mathcal{E}_1^i$] first touch $\mathcal{E}_1^i$ [resp. $\mathcal{E}_2^i$] [11]. The other is the replacement of the ellipsoidal overlap checker with the one described in Appendix A. Two confusion matrices for this detector are shown in Table 3. The average computation time of this detector is 10 milliseconds per pair. Notice here that the performance of this detector is comparable to that of the second one, as reveals that this substituted pair of near points is equally effective in measuring the proximity between the MVIEs for the purpose of polyhedral overlap detection.

The fourth detector is obtained from another two modifications to the second detector. One is the replacement of $\mathbf{v}_1^i$ [resp. $\mathbf{v}_2^i$] with the near point at which

| Detector Output | Collision | No Collision |
|---|---|---|
| Collision | 524 | 28 |
| No Collision | 18 | 430 |

(a)

| Detector Output | Collision | No Collision |
|---|---|---|
| Collision | 128 | 28 |
| No Collision | 18 | 160 |

(b)

Table 3: Confusion matrices for the third collision detector: (a) showing all the 1000 results (average computation time: 10 ms per check); (b) focusing on those in the ambiguous case (average computation time: 19 ms per check).

| Detector Output | Collision | No Collision |
| --- | --- | --- |
| Collision | 515 | 23 |
| No Collision | 27 | 435 |

(a)

| Detector Output | Collision | No Collision |
| --- | --- | --- |
| Collision | 134 | 23 |
| No Collision | 24 | 136 |

(b)

Table 4: Confusion matrices for the fourth collision detector: (a) showing all the 1000 results (average computation time: 10 ms per check); (b) focusing on those in the ambiguous case (average computation time: 15 ms per check).

the line segment joining the respective centers of $\mathcal{E}_1^i$ and $\mathcal{E}_2^i$ intersects the boundary of $\mathcal{E}_1^i$ [resp. $\mathcal{E}_2^i$]. The other is the replacement of the ellipsoidal overlap checker with a checker for the non-emptiness of the intersection among the two ellipsoids and the line passing through their respective centers. Two confusion matrices for this detector are shown in Table 4. The average computation time of this detector is 10 milliseconds per pair. Again, notice here that the performance of this detector is comparable to that of the second one, as reveals not only that this substituted pair of near points is equally effective in measuring the proximity between the MVIEs for the purpose of polyhedral overlap detection, but also the nontrivial fact that given an empty intersection among the MVCEs and the line passing through their respective centers, the separation between two polyhedra is almost certain.

# 6 Conclusion

In this work, the following modifications have been made to ellipsoidal fit based collision detection to effectively improve both the efficiency and the accuracy:

1. using the maximum-volume inscribed ellipsoids for the enclosed fits to polyhedra;
2. adopting a reliable algorithm for computing the minimum-volume circumscribed ellipsoids;
3. correcting the ellipsoidal overlap checker; and
4. preventing reference to polyhedra in polyhedral overlap detection.

In addition, the enclosed ellipsoids considered in this paper are not assumed to be disjoint as in [4], and the performance of the proposed detector has been extensively evaluated with randomly generated convex polyhedra.

# A Overlap Checking between Two Ellipsoids

Denote by $\mathcal{E}_1$ and $\mathcal{E}_2$ the ellipsoids between which overlap is to be checked. We start by checking if the center of any one is in the other. If so, they overlap (for their intersection contains at least the center); if not, we compute by solving an eigenvalue problem [11] the single point at which a certain scaled (i.e., shrunk or dilated from the center) version of $\mathcal{E}_1$ touches $\mathcal{E}_2$, and $\mathcal{E}_1$ and $\mathcal{E}_2$ overlap if and only if this point is in $\mathcal{E}_1$ (the "if" part due to that their intersection contains at least the touching point, and the "only if" part due to that $\mathcal{E}_1$ is a subset of its scaled, touching version and thus cannot intersect $\mathcal{E}_2$ if the touching point is not in $\mathcal{E}_1$).

In the case where the center of any ellipsoid is not in the other ellipsoid, let $\mathcal{E}_1$ and $\mathcal{E}_2$ be respectively described by the set

$$\left\{\mathbf{x} \in \mathbb{R}^3 : (\mathbf{x}-\mathbf{a})^T \mathbf{A}(\mathbf{x}-\mathbf{a}) \le 1\right\}$$

and the set

$$\left\{\mathbf{x} \in \mathbb{R}^3 : (\mathbf{x}-\mathbf{b})^T \mathbf{B}(\mathbf{x}-\mathbf{b}) \le 1\right\},$$

where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ are their respective centers and $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{3\times 3}$ are symmetric and positive definite. Also define

$$\overline{\mathbf{B}} = \mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}$$

and

$$\overline{\mathbf{b}} = \mathbf{A}^{1/2}\left(\mathbf{b}-\mathbf{a}\right),$$

where $\mathbf{A}^{1/2}$ is the square-root matrix of $\mathbf{A}$. Then the single point at which a certain scaled version of $\mathcal{E}_1$ touches $\mathcal{E}_2$ is given by [11]

$$\mathbf{a} + \lambda\mathbf{A}^{-1/2}\left(\lambda\mathbf{I}-\tilde{\mathbf{B}}\right)^{-1}\overline{\mathbf{b}},$$

where $\tilde{\mathbf{B}} = \overline{\mathbf{B}}^{-1}$ and $\lambda$ is the minimal-real-part (necessarily negative real in this case) eigenvalue of the $6\times 6$ matrix

$$\begin{bmatrix} \tilde{\mathbf{B}} & -\mathbf{I} \\ -\tilde{\mathbf{b}}\tilde{\mathbf{b}}^T & \tilde{\mathbf{B}} \end{bmatrix}, \tilde{\mathbf{b}} = \overline{\mathbf{B}}^{-1/2}\overline{\mathbf{b}}.$$

# B Reduction of an MVCE Problem to an MVIE Problem

Without loss of generality, suppose that the convex polyhedron whose MVCE is to be computed contains the origin as an interior point and has $m$ vertices, $\mathbf{v}_1,...,\mathbf{v}_m \in \mathbb{R}^3$. Then it suffices [5] to compute the MVIE of the convex polyhedron

$$\left\{ \mathbf{x} \in \mathbb{R}^4 : \mathbf{A}\mathbf{x} \le \underbrace{[1,...,1]}_{2m}^T \right\},$$

where $\mathbf{x} \le \mathbf{y}$ ($\mathbf{x}, \mathbf{y} \in \mathbb{R}^{2m}$) holds if and only if $x_i \le y_i \forall i \in \{1,...,2m\}$ and

$$\mathbf{A} = \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_m & -\mathbf{v}_1 & \cdots & -\mathbf{v}_m \\ 1 & \cdots & 1 & -1 & \cdots & -1 \end{bmatrix}^T,$$

giving, say, the ellipsoid

$$\left\{ \mathbf{x} \in \mathbb{R}^4 : \mathbf{x}^T \mathbf{B} \mathbf{x} \le 1 \right\},$$

where $\mathbf{B} \in \mathbb{R}^{4\times 4}$ is symmetric and positive definite. The desired MVCE is given by the intersection [11] of the ellipsoid

$$\left\{ \mathbf{x} \in \mathbb{R}^4 : \mathbf{x}^T \mathbf{B}^{-1} \mathbf{x} \le 1 \right\}$$

with the hyperplane

$$\left\{ \mathbf{x} \in \mathbb{R}^4 : \mathbf{x}_4 = 1 \right\}.$$

# References


[1] S. Cameron. A comparison of two fast algorithms for computing the distance between convex polyhedra. *IEEE Transactions on Robotics and Automation*, 13(6):915–920, Dec. 1997.

[2] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi. A fast procedure for com-

puting the distance between complex objects in three-dimensional space. *IEEE Journal of Robotics and Automation*, 4(2):193–203, April 1988.

[3] P. M. Hubbard. Collision detection for interactive graphics applications. *IEEE Transactions on Visualization and Computer Graphics*, 1(3):218–230, Sep. 1995.

[4] M. Ju, J. Liu, S. Shiang, Y. Chien, K. Hwang, and W. Lee. Fast and accurate collision detection based on enclosed ellipsoid. *Robotica*, 19:381–394, 2001.

[5] L. Khachiyan and M. Todd. On the complexity of approximating the maximal inscribed ellipsoid for a polytope. *Mathematical Programming*, 61:137–159, 1993.

[6] D. Kim, L. J. Guibas, and S. Shin. Fast collision detection among multiple moving spheres. *IEEE Transactions on Visualization and Computer Graphics*, 4(3):230–242, July–Sep. 1998.

[7] J. T. Klosowski, M. Held, J. S. B. Mitchell, H. Sowizral, and K. Zikan. Efficient collision detection using bounding volume hierarchies of *k*-dops. *IEEE Transactions on Visualization and Computer Graphics*, 4(1):21–36, Jan.–March, 1998.

[8] M. C. Lin and J. F. Canny. A fast algorithm for incremental distance calculation. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 1008–1014, April 1991.

[9] A. Lin and S. Han. On the distance between two ellipsoids. *SIAM J. Optim.*, 13(1):298–308, 2002.

[10] M. K. Ponamgi, D. Manocha, and M. C. Lin. Incremental algorithms for collision detection between polygonal models. *IEEE Transactions on Visualization and Computer Graphics*, 3(1):51–64, Jan.–March, 1997.

[11] E. Rimon and S. P. Boyd. Obstacle collision detection using best ellipsoid fit. *Journal of Intelligent and Robotic Systems*, 18:105–126, 1997.

[12] S. Shiang, J. Liu, and Y. Chien. Estimate of minimum distance between convex polyhedra based on enclosed ellipsoids. In *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 739–744, 2000.

[13] Y. Zhang. MVE: Maximum volume ellipsoid problem. `http://www.caam.rice.edu/~zhang/maxvep/`, 2002.

[14] Y. Zhang and L. Gao. On numerical solution of the maximum volume ellipsoid problem. Technical Report TR01-15, CAAM, August 2001.

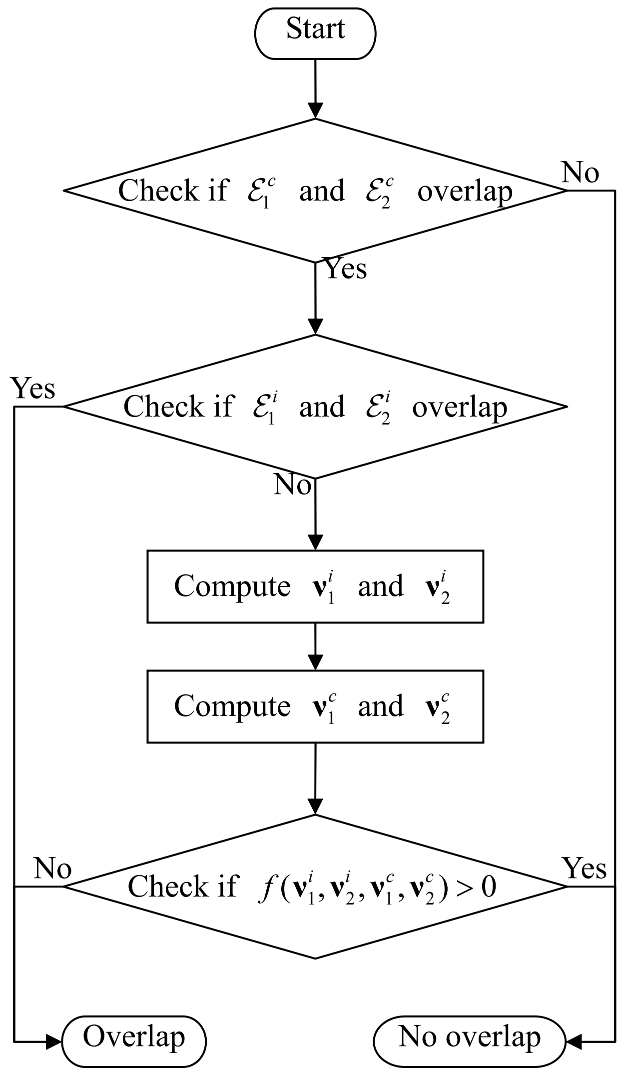


Figure 3: The polyhedral overlap detector

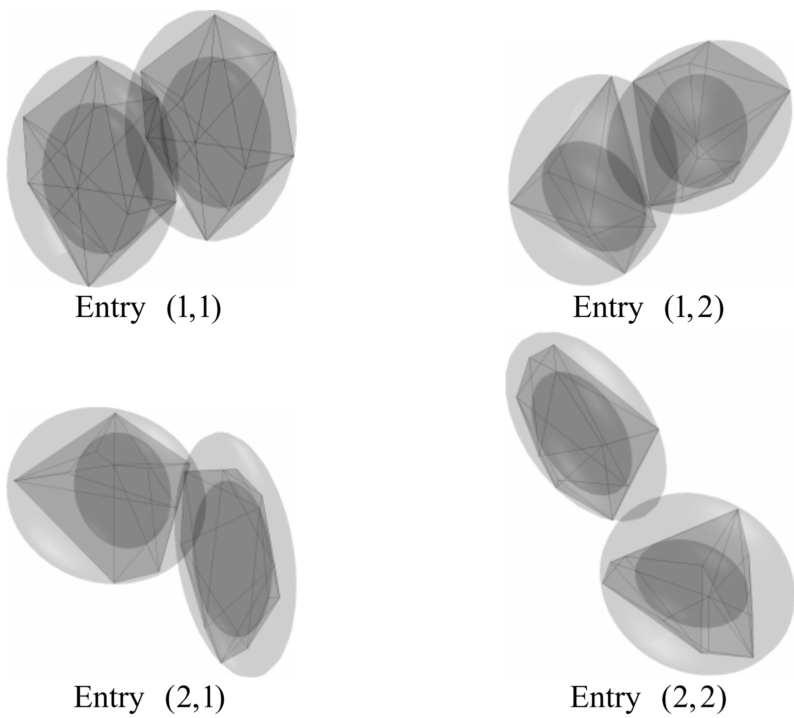


Figure 4: Visualizing a sample from each entry in Table 2(b).