

Generalized Contributing Vertices-Based Method for Minkowski Sum Outer-Face of Two Polygons

Peng Zhang and Hong Zheng^(✉)

State Key Laboratory of Geomechanics and Geotechnical Engineering,
Institute of Rock and Soil Mechanics, Chinese Academy of Sciences, Wuhan, China
zhangpeng_50@163.com, hzheng@whrsm.ac.cn

Abstract. A new method is presented for computing the Minkowski sum outer-face of two polygons of any shape. Stemming from the contributing vertex concept, the concept of generalized contributing vertex is proposed. Based on the new concept, an efficient algorithm is developed, which starts from the construction of the superset of the Minkowski sum edges. The superset is composed of three types of edges: translated-corner edges, translated edges and corner edges. Then the Minkowski sum outer-face is extracted from the arrangement of the superset edges. The algorithm is implemented using C++ and the Computational Geometry Algorithms Library (CGAL). The experiments including very complicated polygons are conducted, suggesting that the proposed algorithm is more efficient than other existing algorithms in most cases.

Keywords: Computational geometry · Minkowski sum outer-face · Generalized contributing vertex · Translated-corner edge · Superset · Polygon

1 Introduction

The Minkowski sum of two objects A and B in an Euclidian space was defined by the German mathematician Hermann Minkowski (1864–1909) as the position vector addition of each point a in A and each point b in B

$$A \oplus B = \{a + b | a \in A, b \in B\} \quad (1)$$

where $a + b$ denotes the sum of position vector a and position vector b , corresponding to points in A and B respectively. The Minkowski sum can also be written as another form

$$A \oplus B = \bigcup_{a \in A} \{a + b | b \in B\} = \bigcup_{a \in A} \{a + b_0 + (b - b_0) | b \in B\} \quad (2)$$

where b_0 is the position vector of any selected point of B .

H. Zheng—This study is supported by the National Basic Research Program of China (973 Program), under the Grant Nos. 2011CB013505 and 2014CB047100; and the National Natural Science Foundation of China, under the Grant No. 11172313.

Since no transformation between coordinate systems is involved, in this study we do not differentiate between points and the position vectors of these points. For example, suppose point $b \in B$, by saying “vector b ” we means the vector starting at the origin O of the coordinate system in use and ending at point b . In this way, Eq. 2 suggests that the Minkowski sum can be achieved by the following operations: (1) translate A by vector b_0 , denoting by A' ; and (2) sweep all points of A' with B by letting point b_0 traverse all points of A' , the trace of sweeping is just $A \oplus B$. Since the polygons A and B are represented by their boundaries, it is sufficient to consider only the boundary points. Figure 1 illustrates the two operations.

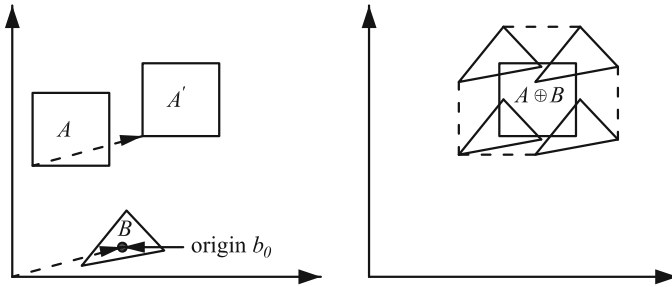


Fig. 1. The Minkowski sum of two polygons A and B is obtained by two operations. Firstly, translating A by vector b_0 , A' is obtained; secondly, sweeping A' by letting point b_0 traverse all points of A' , $A \oplus B$ is given by the trace of B .

The Minkowski sum is applied in many domains such as computer-aided design and manufacturing, image processing, motion planning in robotics. Shi [16] recently proposed a concept, the entrance block $E(A, B)$, to judge whether two blocks, represented by sets A and B , contact, which greatly simplifies the contact detection. The entrance block $E(A, B)$ is defined as

$$E(A, B) = \bigcup_{a \in A, b \in B} \{b - a + a_0\} \tag{3}$$

with a_0 an arbitrary point in A . By comparing Eqs. 1 and 3, we can immediately see that the entrance block $E(A, B)$ is actually the Minkowski sum $B \oplus \bar{A}$, where \bar{A} is obtained through translating the centrally symmetric counterpart of A by vector a_0 . So the Minkowski sum can be used to calculate the entrance block $E(A, B)$.

In most applications, the outer-face of the Minkowski sum rather than the interior is of interest. Therefore, we will propose an algorithm to compute the Minkowski sum outer-face.

This paper is organized as follows. Section 2 reviews the related literature. Section 3 introduces the original contributing vertices-based Minkowski sum algorithm. In Sect. 4, we expound in detail the algorithm based on the concept of generalized contributing vertices to calculate the Minkowski sum outer-face. In

Sect. 5, we present an implementation and make comparisons with the existing algorithms. Section 6 concludes the paper.

2 Related Works

There is one category of methods [2, 8] for computing the Minkowski sum of two non-convex polygons based on convex decomposition: decomposing each non-convex polygon into convex sub-polygons, computing the Minkowski sum of each pair of sub-polygons, then performing the union of the pairwise Minkowski sums. These methods are time consuming because of the large size of the decomposition and union. Now, some other categories of methods are briefly reviewed below.

Ghost [9] presented a slope diagram algorithm to compute the Minkowski sum of polygons and polyhedrons. The polygons are represented in their slope diagram forms. The sum polygon can be obtained by merging the two slope diagrams. However, no implementations of the algorithm for non-convex polygons has been published.

Ramkumar [15] presented a method to compute the Minkowski sum outer-face of two simple polygons using convolution, which was introduced by Guibas et al. [10, 11]. The method detects self-intersections for each cycle of the convolution, and snips off the loops thus created. The outer-face is obtained by connecting the cycles using paths inside the convolution. The convolution method is also used for computing the Minkowski sum of 2D curved objects [12], the Minkowski sum of ruled surfaces [13], and the Minkowski sum of boundary surfaces of 3D-objects [14].

Wein [18] described an efficient and robust method to compute the Minkowski sum using convolution too. The method keeps the faces with non-zero winding numbers from the arrangement of the convolution segments. The method is superior to the decomposition methods in most cases.

Behar and Lien [6] proposed a fast and robust method to compute the 2D Minkowski sum using reduced convolution. The arrangement computation time reduces since the number of convolution segments decreases, which takes a large portion of the total time. Several filters were proposed to extract the Minkowski sum boundaries. The method is faster than the convolution method.

Barki et al. [3–5] proposed the contributing vertices-based Minkowski sum algorithm. The algorithm will be reviewed in the next section since the new algorithm is improved from it.

3 Original Contributing Vertices-Based Minkowski Sum Algorithm

Barki et al. [3] proposed the contributing vertices-based Minkowski sum algorithm for convex polyhedrons (polygons). They showed that the Minkowski sum polygon S of two convex polygons A and B is a convex polygon composed of two types of edges: the edges parallel to the edges of A , named the “translated edges” of S ; and the edges parallel to the edges of B , named the “corner edges” of S .

The two categories of edges are obtained from the edges of A and B by computing their contributing vertices. Here is the definition of a contributing vertex.

The **contributing vertex** $v_{k,B}$ of an edge $e_{i,A} \in A$ with an outer normal $n_{i,A}$ is a vertex of B that is farthest, in the sense of algebraic values, away from the line $e_{i,A}$, suggesting

$$v_{k,B} = \operatorname{argmax}[\langle v_{l,B}, n_{i,A} \rangle | v_{l,B} \in B] \tag{4}$$

where $\langle \cdot, \cdot \rangle$ denotes the scalar product. An example is illustrated in Fig. 2.

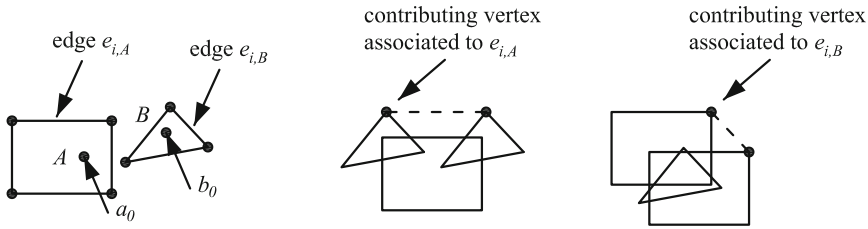


Fig. 2. The concept of contributing vertex

Interchanging A and B in Eq. 4, the contributing vertex $v_{k,A}$ of an edge $e_{i,B} \in B$ with an outer normal $n_{i,B}$ can be obtained See Fig. 2.

There is at least one contributing vertex for each edge of A or B . For some edges, there are two contributing vertices if they are equally farthest away from the line coinciding with these edges. In this case, the edge of A having two contributing vertices has the same orientation as the edge of B with the two vertices as the endpoints. Hereafter, by saying that edge $e_A \in A$ has the same orientation as edge $e_B \in B$, we mean that e_A and e_B have the same unit outward normal.

The process of contributing vertices-based Minkowski sum algorithm consists of the two main steps as follows.

- Determination of translated edges. For each edge $e_{i,A}$ of A , find out its contributing vertices $v_{k,B}$, deduce the translated edge corresponding to each $v_{k,B}$. If there are two contributing vertices that are two endpoints of some edge $e_{j,B}$, the translated edge is the Minkowski sum of $e_{i,A}$ and $e_{j,B}$. The edge $e_{j,B}$ will be ignored in the determination of corner edges because it has contributed once. If there is only one contributing vertex $v_{k,B}$, the translated edge is obtained through translating edge $e_{i,A}$ by vector $v_{k,B}$.
- Determination of corner edges. For each edge $e_{i,B}$ of B that has no the same orientation as any edge of A , find out its contributing vertex $v_{k,A}$, and deduce the corner edge which is obtained through translating edge $e_{i,B}$ by vector $v_{k,A}$.

An illustration of translated edges and corner edges is given in Fig. 3.

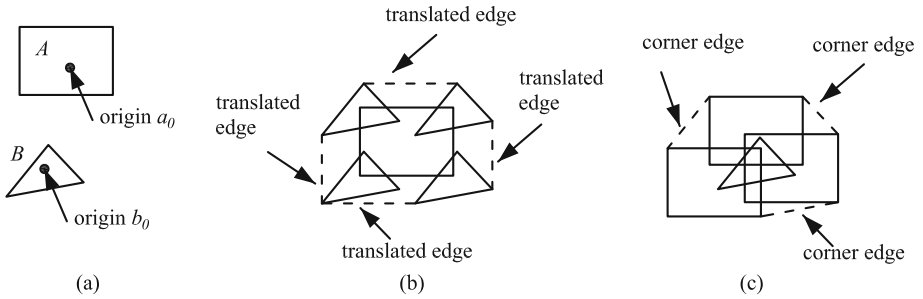


Fig. 3. (a) Two polygons A and B . (b) The translated edges. (c) The corner edges.

Select an arbitrary edge from the translated edges and corner edges as a seed edge, traverse other edges by neighborhood, then the Minkowski sum polygon is obtained.

Barki et al. adapted the contributing vertices-based Minkowski sum algorithm for a nonconvex-convex pair of polyhedra in [4]. Its implementation includes two main steps: (1) form the superset of the Minkowski sum edges through the use of the contributing vertex concept, and; (2) extract the Minkowski sum of the two polyhedra from the superset. However, it doesnt apply to the case of two non-convex polygons.

4 Generalized Contributing Vertices-Based Minkowski Sum Outer-Face Algorithm

In this section, we generalize the concept of contributing vertex to the concept of generalized contributing vertex. Then we use the new concept to develop an algorithm for computing the Minkowski sum outer-face of two polygons of any shape. The algorithm also has two main steps: first, construct the Minkowski sum superset edges; second, extract the Minkowski sum outer-face from the superset.

Lets define some notations here. For two polygons A and B in consideration, A has e_a edges and B has e_b edges; $v_{i,A}$ denotes the starting point of edge $e_{i,A}$, and $v_{i+1,A}$ the end point; similarly, $v_{j,B}$ denotes the starting point of edge $e_{j,B}$, $v_{j+1,B}$ denotes the end point.

4.1 Definition of Generalized Contributing Vertices and Translated-Corner Edge

In order to compute the Minkowski sum outer-face of two polygons of any shape, we generalize the contributing vertex concept to the generalized contributing vertex concept for the two polygons in consideration below.

Definition 1. The **generalized contributing vertex** $v_{k,B}$ of edge $e_{i,A}$ with an outer normal $n_{i,A}$ is a vertex of B whose distance away from the line $e_{i,A}$ is not less than either of its two adjacent vertices and its angle is non-reflex. The generalized contributing vertex $v_{k,B}$ of edge $e_{i,A}$ satisfies:

$$\begin{cases} \langle v_{k,B}, n_{i,A} \rangle \geq \langle v_{k-1,B}, n_{i,A} \rangle \\ \langle v_{k,B}, n_{i,A} \rangle \geq \langle v_{k+1,B}, n_{i,A} \rangle \\ \langle v_{k+1,B} - v_{k,B}, n_{k-1,B} \rangle \leq 0 \end{cases} \quad (5)$$

where $v_{k-1,B}$ denotes the previous vertex of $v_{k,B}$ and $v_{k+1,B}$ the next vertex of $v_{k,B}$; $n_{k-1,B}$ denotes the outer normal of edge $e_{k-1,B}$ starting at $v_{k-1,B}$ and ends at $v_{k,B}$.

When B is convex, generalized contributing vertices degenerate to contributing vertices. The generalized contributing vertices of the edges of B can be found out in a similar manner by interchanging A and B in Eq. 5. An illustration of the generalized contributing vertex concept is shown in Fig. 4.

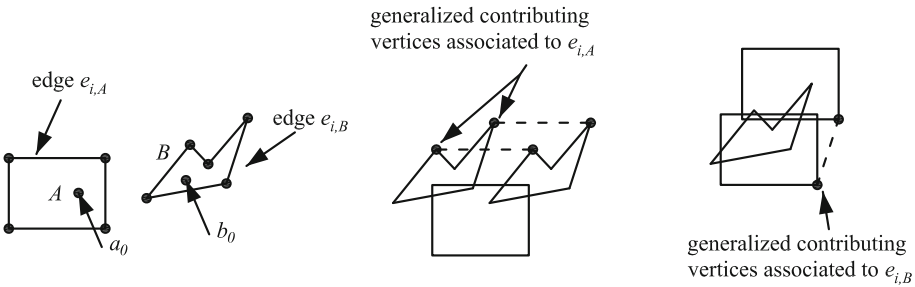


Fig. 4. The concept of generalized contributing vertex

When edge $e_{i,A}$ and edge $e_{j,B}$ have the same orientation, and at least one endpoint of each edge is the generalized contributing convex of the other, the translated edge and corner edge are collinear, have same orientation and will be joined or overlap, and should be merged into one edge, which will be called “**translated-corner edge**” subsequently. Three cases exist that are stated below to reduce the number of edges in the superset and decrease the complexity of the extraction of the Minkowski sum outer-face from the superset.

Case 1. If $e_{i,A}$ and $e_{j,B}$ have the same orientation, $v_{i,A}$ is the generalized contributing vertex of $e_{j,B}$, and $v_{j+1,B}$ is the generalized contributing vertex of $e_{i,A}$, then there is one translated-corner edge that starts at $v_{i,A} \oplus v_{j,B}$ and ends at $v_{i+1,A} \oplus v_{j+1,B}$.

Proof. Because $v_{j+1,B}$ is the generalized contributing vertex of $e_{i,A}$, there is the translated edge, $e_{i,A} \oplus v_{j+1,B}$. It starts at point $v_{i,A} \oplus v_{j+1,B}$, ends at point $v_{i+1,A} \oplus v_{j+1,B}$, and has the same orientation as $e_{i,A}$.

Because $v_{i,A}$ is the generalized contributing vertex of $e_{j,B}$, there is the corner edge, $v_{i,A} \oplus e_{j,B}$. It starts at point $v_{i,A} \oplus v_{j,B}$, ends at point $v_{i,A} \oplus v_{j+1,B}$, and has the same orientation as $e_{j,B}$.

The starting point of the translated edge $e_{i,A} \oplus v_{j+1,B}$ and the end point of the corner edge $v_{i,A} \oplus e_{j,B}$ are both $v_{i,A} \oplus v_{j+1,B}$. The translated edge $e_{i,A} \oplus v_{j+1,B}$ and the corner edge $v_{i,A} \oplus e_{j,B}$ are linked at $v_{i,A} \oplus v_{j+1,B}$, and accordingly merged into one translated-corner edge that starts at point $v_{i,A} \oplus v_{j,B}$, ends at point $v_{i+1,A} \oplus v_{j+1,B}$, and has the same orientation as $e_{i,A}$ or $e_{j,B}$. A case is illustrated in Fig. 5.

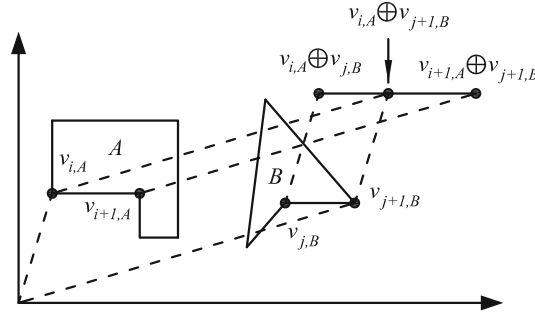


Fig. 5. An illustration of translated-corner edge case 1

Case 2. If $e_{i,A}$ and $e_{j,B}$ have the same orientation, $v_{i,A}$ is the generalized contributing vertex of $e_{j,B}$ but $v_{i+1,A}$ is not, and $v_{j,B}$ is the generalized contributing vertex of $e_{i,A}$ but $v_{j+1,B}$ is not, then there is the translated-corner edge that starts at $v_{i,A} \oplus v_{j,B}$, and ends at one of point $v_{i+1,A} \oplus v_{j,B}$ and point $v_{i,A} \oplus v_{j+1,B}$ that is farther away from the starting point $v_{i,A} \oplus v_{j,B}$.

Proof. Because $v_{j,B}$ is the generalized contributing vertex of $e_{i,A}$ but $v_{j+1,B}$ is not, there is the translated edge, $e_{i,A} \oplus v_{j,B}$. It starts at point $v_{i,A} \oplus v_{j,B}$, ends at point $v_{i+1,A} \oplus v_{j,B}$, and has the same orientation as $e_{i,A}$.

Because $v_{i,A}$ is the generalized contributing vertex of $e_{j,B}$ but $v_{i+1,A}$ is not, there is the corner edge, $v_{i,A} \oplus e_{j,B}$. It starts at point $v_{i,A} \oplus v_{j,B}$, ends at point $v_{i,A} \oplus v_{j+1,B}$, and has the same orientation as $e_{j,B}$.

As a result, the translated edge $e_{i,A} \oplus v_{j,B}$ and the corner edge $v_{i,A} \oplus e_{j,B}$ overlap since they start at the same point $v_{i,A} \oplus v_{j,B}$ and have the same orientation. The translated-corner edge should be the longer one, and accordingly ends at one of point $v_{i+1,A} \oplus v_{j,B}$ and point $v_{i,A} \oplus v_{j+1,B}$ that is farther away from the starting point $v_{i,A} \oplus v_{j,B}$. Figure 6 illustrates such a case.

Case 3. If $e_{i,A}$ and $e_{j,B}$ have the same orientation, and $v_{i,A}$ is not the generalized contributing vertex of $e_{j,B}$ but $v_{i+1,A}$ is; and $v_{j,B}$ is not the generalized contributing vertex of $e_{i,A}$ but $v_{j+1,B}$ is, then there is the translated-corner edge that ends at $v_{i+1,A} \oplus v_{j+1,B}$, and starts at one of $v_{i,A} \oplus v_{j+1,B}$ and $v_{i+1,A} \oplus v_{j,B}$ that is farther from the end point $v_{i+1,A} \oplus v_{j+1,B}$.

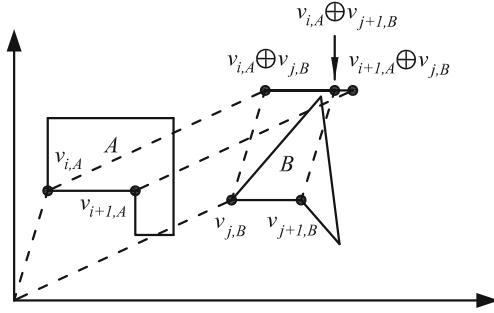


Fig. 6. An illustration of translated-corner edge case 2

Proof. Because $v_{j,B}$ is not the generalized contributing vertex of $e_{i,A}$ but $v_{j+1,B}$ is, we have the translated edge $e_{i,A} \oplus v_{j+1,B}$. It starts at point $v_{i,A} \oplus v_{j+1,B}$ and ends at point $v_{i+1,A} \oplus v_{j+1,B}$, with the same orientation as $e_{i,A}$.

Because $v_{i,A}$ is not the generalized contributing vertex of $e_{j,B}$ but $v_{i+1,A}$ is, we have the corner edge, $v_{i+1,A} \oplus e_{j,B}$. It starts at point $v_{i+1,A} \oplus v_{j,B}$ and ends at point $v_{i+1,A} \oplus v_{j+1,B}$, with the same orientation as $e_{j,B}$.

As a result, the translated edge $e_{i,A} \oplus v_{j+1,B}$ and the corner edge $v_{i+1,A} \oplus e_{j,B}$ overlap since they end at the same point $v_{i+1,A} \oplus v_{j+1,B}$ and have the same orientation. The translated-corner edge should be the longer one, and accordingly starts at one of point $v_{i,A} \oplus v_{j+1,B}$ and point $v_{i+1,A} \oplus v_{j,B}$ that is farther away from the ending point $v_{i+1,A} \oplus v_{j+1,B}$. Figure 7 illustrates such a case.

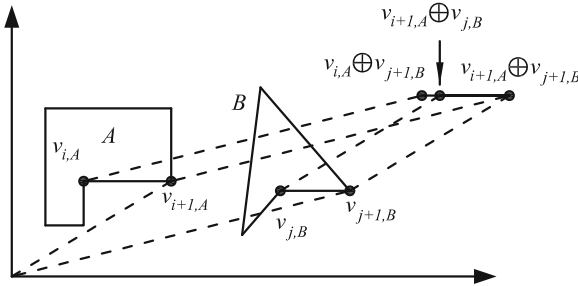


Fig. 7. An illustration of translated-corner edge case 3

4.2 Construction of the Minkowski Sum Edge Superset

The construction of the Minkowski sum edge superset starts with determining the translated-corner edges, translated edges and corner edges by finding out the generalized contributing vertices associated with all edges of A and B respectively. Four steps are involved in this operation.

1. Determination of generalized contributing vertices: For each edge $e_{i,A}$, find out its generalized contributing vertices. For each edge $e_{i,B}$, find out its generalized contributing vertices.
2. Determination of translated-corner edges: For each edge $e_{i,A}$, find out the edges $e_{j,B}$ having the same orientation. If such a pair of $e_{i,A}$ and $e_{j,B}$ falls into one of the three cases in the above, we can determine the translated-corner edge. The generalized vertices belonging to $e_{j,B}$ of $e_{i,A}$ and the generalized vertices belonging to $e_{i,A}$ of $e_{j,B}$ will not be considered in the rest, since they have been considered here.
3. Determination of translated edges: For each edge $e_{i,A}$, for each of the rest generalized contributing vertices $v_{k,B}$ which have not been considered, we have a translated edge, $e_{i,A} \oplus v_{k,B}$, which is obtained through translating it by vector $v_{k,B}$, and accordingly has the same orientation as $e_{i,A}$. Here, $v_{k,B}$ is a generalized contributing vertex.
4. Determination of corner edges: For each edge $e_{i,B}$, for each of the rest generalized contributing vertices $v_{k,A}$ which have not been considered, we have a corner edge, $e_{i,B} \oplus v_{k,A}$, which is obtained through translating it by vector $v_{k,A}$, and accordingly has the same orientation as $e_{i,B}$. Here, $v_{k,A}$ is a generalized contributing vertex.

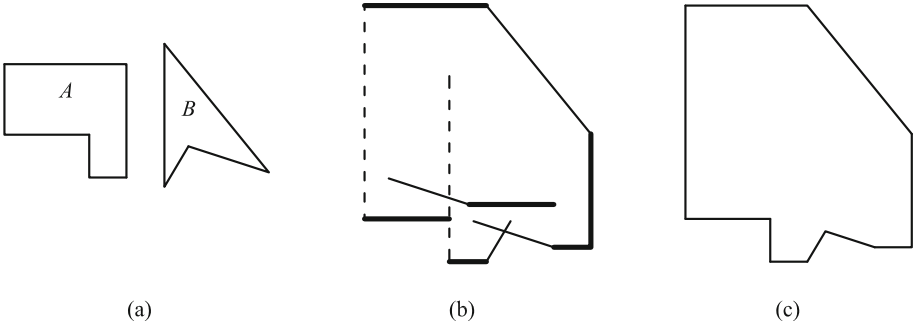


Fig. 8. (a) Two polygons A and B . (b) The Minkowski sum edges superset. The dashed lines are the translated-corner edges. The bold lines are the translated edges. The thin lines are the corner edges. (c) The Minkowski sum outer-face.

4.3 Extraction of the Minkowski Sum Outer-Face from the Superset Edges

There are two ways to extract the Minkowski sum outer-face from the superset: (1) compute the intersections of the superset edges and break the edges down at the intersections, from a seed edge containing the lexicographically smallest (or greatest) point among the superset vertices, traverse other outer-face edges by neighborhood, when there are multiple compatible edges adjacent to the end

point of the previous edge, we pick the one that makes a largest inner angle at the point, and (2) compute the arrangement [7, 17] of the superset edges, the Minkowski sum outer-face is the resulted face with no boundary.

An example of a non-convex polygon A and a non-convex polygon B is depicted in Fig. 8(a). The entire superset of the Minkowski sum edges is depicted in Fig. 8(b). The Minkowski sum outer-face is depicted in Fig. 8(c).

5 Implementation and Comparison

In this section, we describe the implementation of the generalized contributing vertex-based Minkowski sum outer-face (GCVMSOF) algorithm. Then, we compare the performance of this algorithm with other methods in the literature.

5.1 Implementation

The proposed algorithm has been implemented using C++ and the Computational Geometry Algorithms Library (CGAL) [1]. The extraction of the Minkowski sum outer-face from the superset is the arrangement method, which is done by the 2D arrangement package of CGAL, the outer-face is the unbounded face. We select nine models, the first five from [18]: chain, stars, comb, fork and knife; and the last four from [6]: g1g2, monkey, hand, bird. All data sets are shown in Table 1.

Table 1. Input sets

Input sets	A		B	
	Vertices	Concave vertices	Vertices	Concave vertices
Chain	58	26	16	8
Starts	40	14	40	14
Comb	53	24	22	0
Fork	34	19	31	18
Knife	64	40	12	5
g1g2	30	13	34	19
Monkey	1204	577	24	11
Hand	57	15	84	16
Bird	275	133	57	15

All experiments were performed on a personal computer with 4GB RAM and 3.40 GHZ Intel Core i7 CPU. Figure 9 illustrates the input polygons and their Minkowski sum outer-faces.

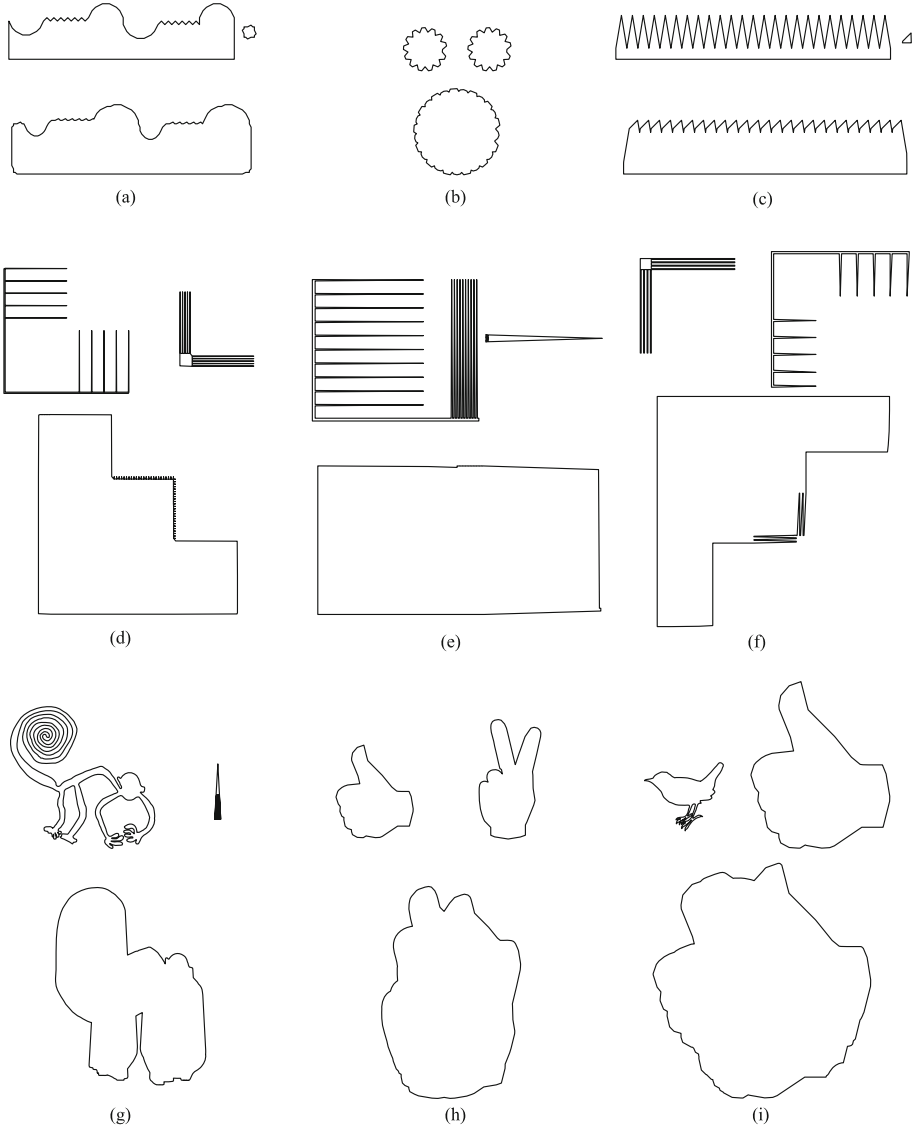


Fig. 9. Samples of input polygons (top) and their Minkowski sum outer-faces (bottom): (a) chain; (b) stars; (c) comb; (d) fork; (e) knife; (f) g1g2; (g) monkey; (h) hand; (i) bird.

5.2 Comparison

According to [18], the convolution-based Minkowski sum (CMS) algorithm is superior to the decomposition method in most cases. According to [6], the reduced convolution-based Minkowski sum algorithm is superior to the CMS algorithm. So the comparisons in the computation time are made between the

proposed algorithm (GCVMSOF) and the reduced convolution-based Minkowski sum outer-face algorithm (RCMSOF). The RCMSOF algorithm is adapted from the 2D Minkowski sum package of CGAL, which implements the convolution operation. It constructs the reduced convolution superset first; then extracts the Minkowski sum outer-face from the superset arrangement. The running times and superset edges numbers are shown in Table 2.

Table 2. Comparisons of the proposed GCVMSOF and RCMSOF

Input sets	Algorithms	Superset size	Running time (ms)		
			Superset construction	Outer-face extraction	Total time
Chain	GCVMSOF	170	48	85	133
	RCMAOF	63	74	68	142
Starts	GCVMSOF	371	85	310	395
	RCMAOF	372	103	296	399
Comb	GCVMSOF	80	19	46	65
	RCMAOF	81	21	44	65
Fork	GCVMSOF	331	58	1965	2023
	RCMAOF	342	93	1881	1974
Knife	GCVMSOF	211	47	884	931
	RCMAOF	211	73	879	952
g1g2	GCVMSOF	389	89	875	964
	RCMAOF	346	104	724	828
Monkey	GCVMSOF	8189	1617	9371	10988
	RCMAOF	8189	2978	8922	11900
Hand	GCVMSOF	369	203	200	403
	RCMAOF	369	203	179	382
Bird	GCVMSOF	1019	661	702	1363
	RCMAOF	1017	947	651	1598

In most examples, the proposed GCVMSOF is faster than RCMSOF. This is due to the fact: though the outer-face extraction times are longer in GCVMSOF than those in RCMSOF, the superset construction times are much shorter. These experiments also suggest that the superset edges numbers of the two algorithms are almost equal, and the outer-face extraction time usually takes a large portion of the total time.

6 Conclusions

In this study, a new algorithm is developed to compute the Minkowski sum outer-face of two polygons of any shape by generalizing the contributing vertex concept. The new concept is straightforward and the algorithm is easy to

implement. Three translated-corner edge cases associated with the new concept are found and proved. The experiments show that the new algorithm is superior to other methods for most input sets. The next work to develop the algorithm for calculating the Minkowski sum outer-face of two polyhedra based on the generalized contributing vertex concept.

References

1. The CGAL project homepage. <http://www.cgal.org>
2. Agarwal, P.K., Flato, E., Halperin, D.: Polygon decomposition for efficient construction of minkowski sums. *Comput. Geom.* **21**(1), 39–61 (2002)
3. Barki, H., Denis, F., Dupont, F.: Contributing vertices-based minkowski sum computation of convex polyhedra. *Comput. Aided Des.* **41**(7), 525–538 (2009)
4. Barki, H., Denis, F., Dupont, F.: Contributing vertices-based minkowski sum of a non-convex polyhedron without fold and a convex polyhedron. In: *IEEE International Conference on Shape Modeling and Applications*, 2009, SMI 2009, pp. 73–80. IEEE (2009)
5. Barki, H., Denis, F., Dupont, F.: Contributing vertices-based minkowski sum of a nonconvex-convex pair of polyhedra. *ACM Trans. Graphics (TOG)* **30**(1), 3 (2011)
6. Behar, E., Lien, J.M.: Fast and robust 2d minkowski sum using reduced convolution. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 1573–1578. IEEE (2011)
7. De Berg, M., Van Kreveld, M., Overmars, M., Schwarzkopf, O.C.: *Computational Geometry*. Springer, Heidelberg (2000)
8. Flato, E.: Robust and efficient construction of planar minkowski sums. Master's thesis, Department of Computer Science, Tel-Aviv University (2000)
9. Ghosh, P.K.: A unified computational framework for minkowski operations. *Comput. Graphics* **17**(4), 357–378 (1993)
10. Guibas, L., Ramshaw, L., Stolfi, J.: A kinetic framework for computational geometry. In: *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 100–111. IEEE (1983)
11. Guibas, L.J., Seidel, R.: Computing convolutions by reciprocal search. *Discrete Comput. Geom.* **2**(1), 175–193 (1987)
12. Lee, I.K., Kim, M.S., Elber, G.: The minkowski sum of 2D curved objects. *Proceedings of Israel-Korea Bi-National Conference on New Themes in Computerized Geometrical Modeling*, vol. 5, pp. 155–164 (1998)
13. Mühlthaler, H., Pottmann, H.: Computing the minkowski sum of ruled surfaces. *Graph. Models* **65**(6), 369–384 (2003)
14. Peternell, M., Steiner, T.: Minkowski sum boundary surfaces of 3d-objects. *Graph. Models* **69**(3), 180–190 (2007)
15. Ramkumar, G.: An algorithm to compute the minkowski sum outer-face of two simple polygons. In: *Proceedings of the Twelfth Annual Symposium on Computational Geometry*, pp. 234–241. ACM (1996)
16. Shi, G.h.: Basic equations of two dimensional and three dimensional contacts. In: *47th US Rock Mechanics/Geomechanics Symposium*, American Rock Mechanics Association (2013)

17. Toth, C.D., O'Rourke, J., Goodman, J.E.: Handbook of Discrete and Computational Geometry. CRC Press, Boca Raton (2004)
18. Wein, R.: Exact and efficient construction of planar Minkowski sums using the convolution method. In: Azar, Y., Erlebach, T. (eds.) ESA 2006. LNCS, vol. 4168, pp. 829–840. Springer, Heidelberg (2006)