INTERNATIONAL JOURNAL FOR NUMERICAL AND ANALYTICAL METHODS IN GEOMECHANICS
Int. J. Numer. Anal. Meth. Geomech., 2006; **30**:783–801
Published online 23 January 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nag.500

Shortest link method for contact detection in discrete element method

Erfan G. Nezami[§], Youssef M. A. Hashash[*,†,‡], Dawei Zhao[§]
and Jamshid Ghaboussi[¶]

Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, IL 61801, U.S.A.

SUMMARY

With the increasing demand for discrete element simulations with larger number of particles and more realistic particle geometries, the need for efficient contact detection algorithms is more evident. To date, the class of common plane (CP) methods is among the most effective and widely used contact detection algorithms in discrete element simulations of polygonal and polyhedral particles. This paper introduces a new approach to obtain the CP by employing a newly introduced concept of 'shortest link'. Among all the possible line segments that connect any point on the surface of particle A to any point on the surface of particle B, the one with the shortest length defines the shortest link between the two particles. The perpendicular bisector plane of the shortest link fulfils all the conditions of a CP, suggesting that CP can be obtained by seeking the shortest link. A new algorithm, called shortest link method (SLM), is proposed to obtain the shortest link and subsequently the CP between any two polyhedral particles. Comparison of the analysis time between SLM and previously introduced algorithms demonstrate that SLM results in a substantial speed up for polyhedral particles contact detection. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: discrete element method; contact detection; common plane; polyhedral particles

1. INTRODUCTION

Cundall [1] introduces the discrete element method (DEM) to simulate the large deformation of jointed rock masses. Cundall and Strack [2] extend the method to analyse assemblies of circular disks and spheres. During the past two decades, DEM has been extensively used to study the

[*] Correspondence to: Youssef M. A. Hashash, 2230c Newmark Civil Engineering Laboratory, 205 North Mathews Avenue, Urbana, IL 61801, U.S.A.
[†] E-mail: hashash@uiuc.edu
[‡] Associate Professor.
[§] Graduate Student.
[¶] Professor.

Contract/grant sponsor: National Science Foundation; contract/grant number: CMS-0113745
Contract/grant sponsor: Caterpillar, Inc.

Received 15 April 2005
Revised 30 November 2005
Accepted 1 December 2005

behaviour of granular materials [3–5]). DEM simulations have been used in large-scale simulations of landslides [6, 7], ice flows [8], and dragline excavation [9].

The DEM requires the detection of contacts and the calculation of contact forces between pairs of particles at every simulation time step. Contact detection constitutes up to 70–80% of the total analysis time, especially when particles of complex geometry, such as polyhedrons, are considered. As a result, the development of a fast and reasonably accurate contact detection scheme remains the major challenge in DEM simulations. This paper proposes a new approach to obtain the common plane (CP) for contact detection between two- and three-dimensional, polyhedral convex particles. Concave particles, can be modelled as a combination of several convex particles attached to each other.

2. COMMON PLANE METHOD FOR CONTACT DETECTION

Cundall [10] states that 'a common plane is a plane that, in some sense, bisects the space between the two contacting particles' as illustrated in Figure 1. If the two particles are in contact, then both intersect the CP (Figure 1(a)), and if they are not in contact, then neither intersects it (Figure 1(b)). In CP method, the particle-to-particle contact detection problem reduces to much simpler plane-to-particle contact detection problem. CP can easily handle non-trivial situations such as vertex-to-vertex, vertex-to-edge or edge-to-edge contacts in 3-D, and it provides a robust technique to obtain the direction of contact forces. The use of the CP for contact detection is advantageous over other methods as discussed in Reference [11]. The following section provides a brief description of the CP. The reader is referred to References [10, 11] for more details.

2.1. *Definition of CP*

The 'distance' d^V of any point V in the space to any arbitrary plane is defined as

$$d^V = \mathbf{n} \cdot (V_0 - V) \tag{1}$$

where $\mathbf{n}$ is the unit vector normal to the plane and V_0 is any point on the plane.

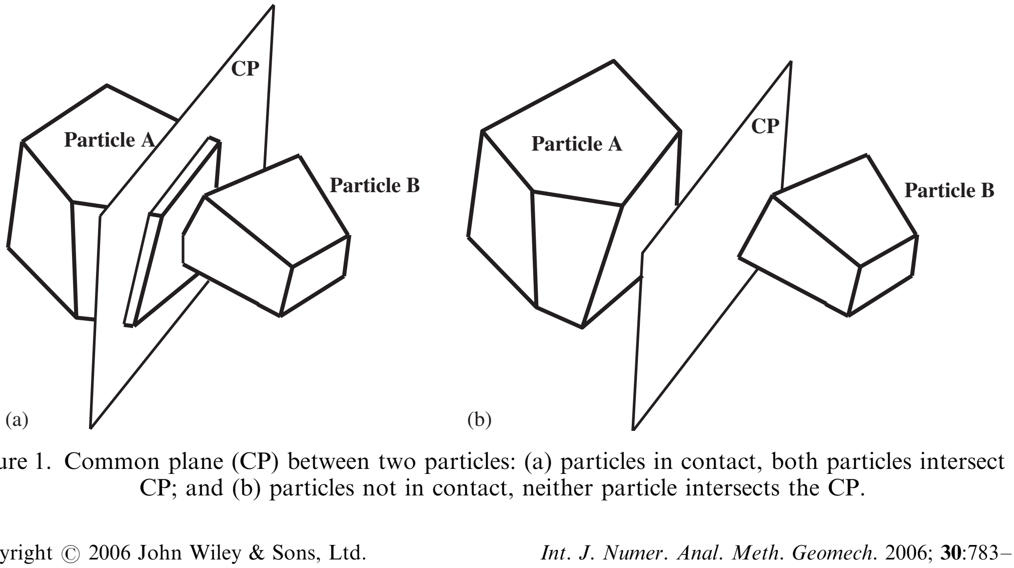

Figure 1. Common plane (CP) between two particles: (a) particles in contact, both particles intersect the CP; and (b) particles not in contact, neither particle intersects the CP.

The 'distance' d_{A} of any polygonal or polyhedral particle A to any plane in the space is defined as

$$d_{\mathrm{A}} = \begin{cases} \max(d^V) & \text{if } d^{\mathrm{C}} < 0 \\ \min(d^V) & \text{if } d^{\mathrm{C}} > 0 \end{cases} \tag{2}$$

in which d^V denotes the distance of a vertex V on the particle to the plane, and $\min\{\cdot\}$ and $\max\{\cdot\}$ are the minimum and the maximum values taken over all vertices V of particle A. d^{C} is the distance of the centroid of the particle to the plane.

Equation (2) implies that if the particle is on the left-hand side of the plane then d_{A} is defined by the right most vertex (or vertices) of the particle and *vice versa*. The distance d^{C} of the particle's centroid is used as a criterion to determine on which side of the plane the particle is located.

For any two particles A and B, a CP is the plane that satisfies the following three conditions:

1. Centroids of particles A and B are located on opposite sides of the CP. It is assumed that particle A is the one with its centroid in the negative side and B is the one with its centroid in the positive side of CP.
2. The gap, defined as $d_{\mathrm{B}} - d_{\mathrm{A}}$, is a maximum.
3. $d_{\mathrm{A}} = -d_{\mathrm{B}}$

where d_{A} and d_{B} are the distances of particles A and B to the CP. Whenever the particles are not in contact $d_{\mathrm{A}} < 0$ and $d_{\mathrm{B}} > 0$, and the gap is positive. Whenever the particles are in contact $d_{\mathrm{A}} > 0$ and $d_{\mathrm{B}} < 0$, and the gap is negative.

Cundall [10] proposes an iterative algorithm by which the CP associated with the maximum gap (when the particles are not in contact) or the minimum gap (when the particles are in contact) is sought. Each iteration involves translation and rotation of the CP. The number of iterations is generally large, especially for the first-time formation of the CP.

Nezami *et al.* [11] propose the fast common plane (FCP) algorithm for identifying the CP. FCP recognizes that, when particles are not in contact, the CP can be chosen among a few candidate planes. This dramatically reduces the number of iterations. Whenever the particles are in contact, an additional step is performed to temporarily separate the particles prior to the iteration process. The separation is accomplished by translating each of the two particles in the direction perpendicular to the CP, which is available from the previous time step. As long as the translation distances of the two particles are very small, the CP of particles in their original configuration can be assumed to be the same as that of the separated configuration.

3. SHORTEST LINK METHOD

Shortest link method (SLM) presents an alternative, even faster approach to find the CP for particles not in contact. For particles in contact, SLM employs the same separation scheme as FCP [11] to temporarily separate the particles.

3.1. Definition of the shortest distance

For any two particles A and B not in contact, any segment such as PQ that connects any arbitrary point P from the surface of particle A to any arbitrary point Q on the surface of particle B is called a 'link' between the two particles (Figure 2). The link with the shortest length $|PQ| = L$ is called the 'shortest link'.

3.2. Closest points

Let A be a particle which is not in contact with the CP. Any point P on particle A for which $|d_{\mathrm{A}}| = |d^P|$ is called the 'closest point' of particle A to the CP. Figure 3 also shows the set $\mathbf{S}_{\mathrm{A}}$, which is defined as the projections of all the closest points on the CP. Note that the set $\mathbf{S}_{\mathrm{A}}$ may correspond to a single vertex (Figure 3(a)), all the points on one edge (Figure 3(b)), or all the points on one face of the particle (Figure 3(c)).

3.3. Relationship between the shortest link and the CP

Statement
The perpendicular bisector plane (PBP) of the shortest link between two particles is the CP.

Proof
Let $\mathbf{S}_{\mathrm{A}}$ and $\mathbf{S}_{\mathrm{B}}$ denote the sets of the projections of all the closest points from the particles A and B on the CP, respectively (Figure 4). The following provides a stepwise proof for the statement above.

(i) $\mathbf{S}_{\mathrm{A}} \cap \mathbf{S}_{\mathrm{B}} \neq \phi$. In other words, there is at least one point on the CP which belongs to both sets $\mathbf{S}_{\mathrm{A}}$ and $\mathbf{S}_{\mathrm{B}}$.

Assume that $\mathbf{S}_{\mathrm{A}}$ and $\mathbf{S}_{\mathrm{B}}$ have no overlap (Figure 4(a)). Then a line l can be drawn on the CP such that $\mathbf{S}_{\mathrm{A}}$ and $\mathbf{S}_{\mathrm{B}}$ are located on opposite sides of l. Rotation of the CP around the axis l in the direction shown in Figure 4(a) will increase the distance of all the points of both particles to the CP and results in a larger gap between the particles. This contradicts condition 2 that the CP results in the maximum possible gap between the two particles.

Therefore, the two sets $\mathbf{S}_{\mathrm{A}}$ and $\mathbf{S}_{\mathrm{B}}$ have at least one point, such as R, in common. Point R is the projection on CP of a closest point P from particle A, and a closest point, such as Q, on particle B (Figure 4(b)).

(ii) The link PQ is the shortest link between the two particles.

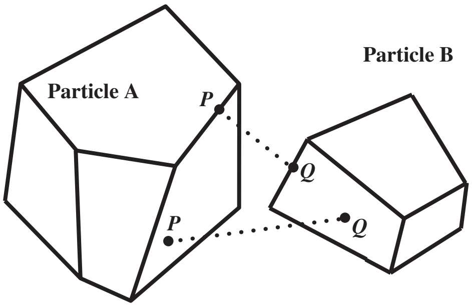

Figure 2. A link is a line segment that connects any point, P, on surface of particle A to any point, Q, on surface of particle B.

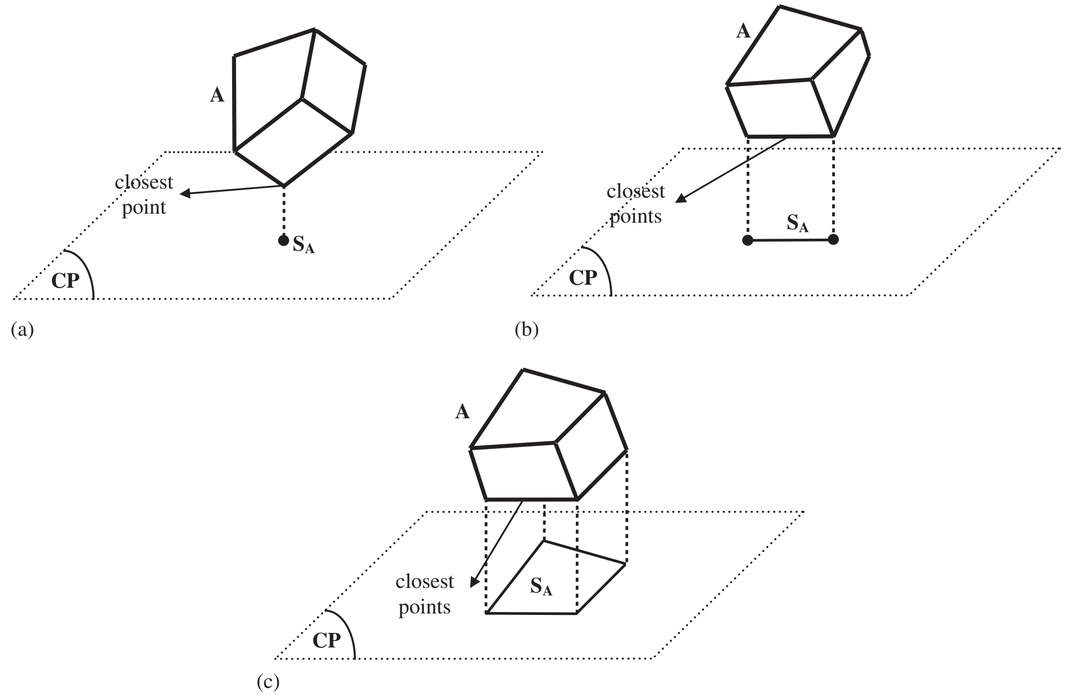

Figure 3. Closest points of particle A to the CP and the set $\mathbf{S_A}$: (a) $\mathbf{S_A}$ consists of only one point; (b) $\mathbf{S_A}$ consists of all the points of a segment line; and (c) $\mathbf{S_A}$ consists of all the points inside and on the edges of a polygon.

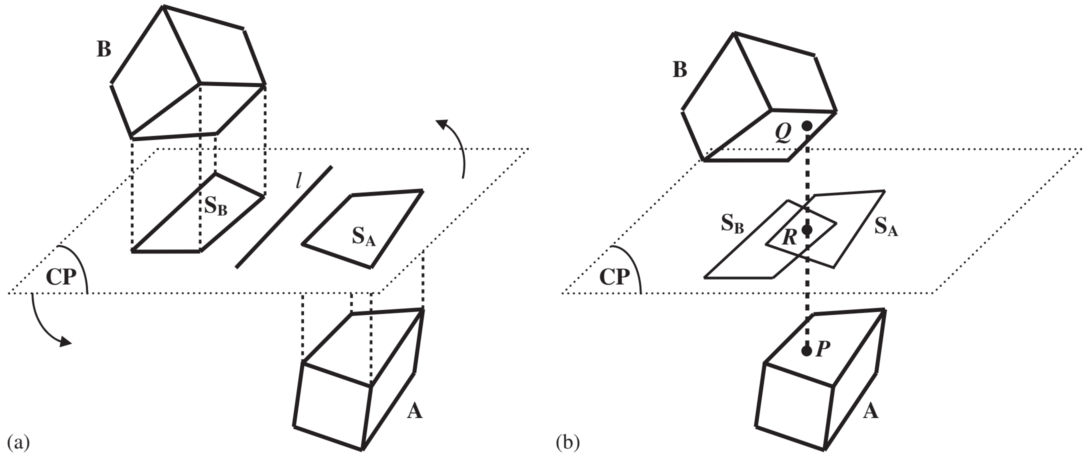

Figure 4. Relative position of $\mathbf{S_A}$ and $\mathbf{S_B}$: (a) if $\mathbf{S_A} \cap \mathbf{S_B} = \phi$, then any rotation along l increases the gap; and (b) $R \in \mathbf{S_A} \cap \mathbf{S_B}$.

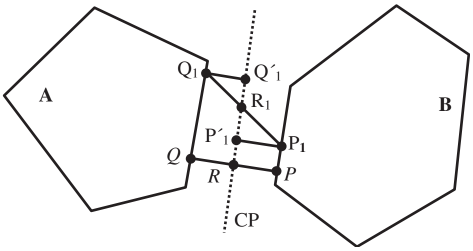

Figure 5. PQ is the shortest link between the two particles.

Let P_1Q_1 be any arbitrary link with P_1 on particle A and Q_1 on particle B (Figure 5) and R_1 be the point of intersection of segment P_1Q_1 and the CP. Let P'_1 and Q'_1 denote the projections of P_1 and Q_1 on the CP, respectively. As P is one of the closest points of particle A to the CP, $|PR|\leqslant|P_1P'_1|$. For the same reason, $|QR|\leqslant|Q_1Q'_1|$. So,

$$|P_1Q_1| = |P_1R_1| + |Q_1R_1|\geqslant|P_1P'_1| + |Q_1Q'_1|\geqslant|PR| + |QR| = |PQ|$$

or

$$|P_1Q_1|\geqslant|PQ|$$

Therefore, the link PQ is the shortest link between the two particles.

(iii) The CP is the PBP of the shortest distance PQ.

Point R is the projection of points P and Q on the CP. By definition, PR and QR are perpendicular to the CP.

Therefore, PQ is perpendicular to the CP.

Also,

$$|PR| = -d_{\mathrm{A}} \quad \text{and} \quad |QR| = d_{\mathrm{B}}$$

From condition 3 of the definition of the CP,

$$-d_{\mathrm{A}} = d_{\mathrm{B}} \Rightarrow |PR| = |QR|$$

Therefore, CP is the bisector of segment PQ. □

Therefore, the CP is PBP of segment PQ.

3.4. Identifying the shortest link between a point and a particle

The shortest link between a point and a particle is defined as the shortest possible segment that connects that point to any point from the particle. SLM frequently uses the shortest link between a point in space and a particle to find the CP. This section is solely devoted to describe how such a link is found in SLM.

Let Q be a point outside particle A, as shown in Figure 6(a). Find a destination point P on the surface of A such that PQ has the shortest possible length. Starting from any arbitrary point P_0 on the surface of A, SLM obtains a series of intermediate points P_i $(i = 1, 2, \ldots, n)$ on the surface of A, with $|P_iQ|<|P_{i-1}Q|$. The last point $P_n = P$ is the closest point to Q.

P_i is determined based on the location of P_{i-1} on the particle and its relative position with respect to Q.

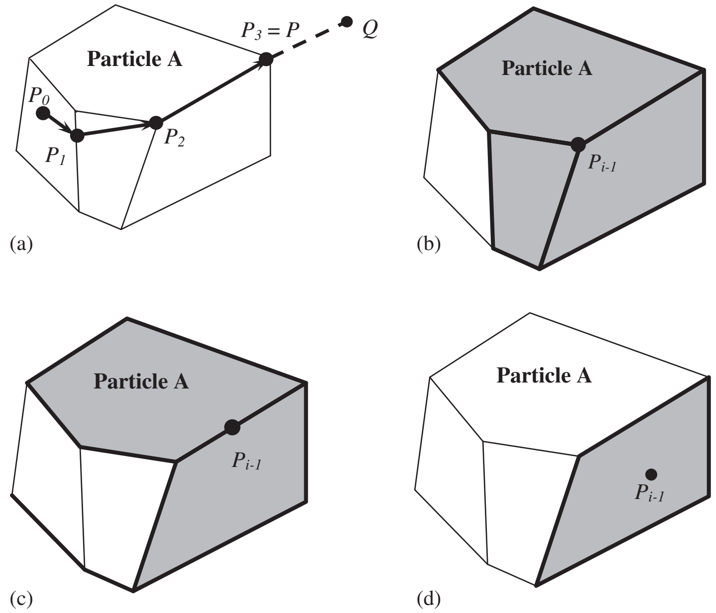

Figure 6. Shortest link between a point and a particle: (a) SLM iterative process to find point P from P_0; (b) P_{i-1} is a vertex of particle A; (c) P_{i-1} is on an edge; and (d) P_{i-1} is on a face.

Case 1: P_{i-1} is a vertex of particle A (Figure 6(b)). In this case P_i is sought only on the neighbouring faces of P_{i-1} (neighbouring faces of a vertex are those that share that vertex).

Case 2: P_{i-1} is located on an edge of the particle but not on the vertices connected to it (Figure 6(c)). In this case point P_i is sought only on the two neighbouring faces that share that edge.

Case 3: P_{i-1} is located on a face of the particle but not on any edges or vertices of that face, Figure 6(d). In this case point P_i is sought only on the same face (including its edges and vertices).

The process continues until $P_i = P_{i-1}$. The number of iterations depends on the relative position of the starting point P_0 and point Q, as well as the geometry of the particle. The closer the starting point P_0 to destination point P, the faster the algorithm converges and less number of iterations required.

3.5. *CP identification in SLM*

SLM indirectly identifies the CP by seeking the shortest link between the two particles. The algorithm seeks two points $P \in \mathrm{A}$ and $P \in \mathrm{B}$ such that PQ is the shortest link between particles A and B. Starting from any two arbitrary points $P^0 \in A$ and $Q^0 \in B$, SLM uses an iterative process, in which a series of intermediate points $P^i \in A$ and $Q^i \in B$ $(i = 1, \ldots, n)$ are found, such that $P^n = P$ and $Q^n = Q$ (Figure 7). The ith iteration consists of two steps as follows.

Step 1. For the point Q^{i-1}, use the algorithm in Section 3.4 to find the closest point P^i on particle A. Use P^{i-1} as the starting point on particle A.

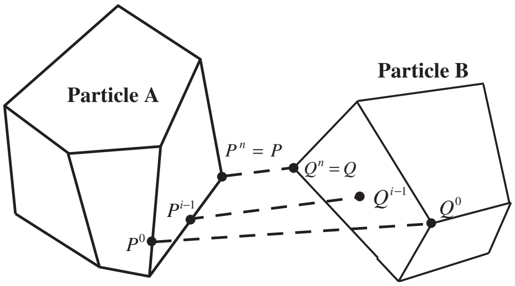

Figure 7. Shortest link method algorithm.

Step 2. For the newly found point P^i, use the same algorithm to find the closest point Q^i on particle B. Use Q^{i-1} as the starting point.

The ith iteration results in a pair of (P^i, Q^i) such that $|P^{i-1}Q^{i-1}| > |P^iQ^i|$. The iterative process continues until $P^i = P^{i-1}$ and $Q^i = Q^{i-1}$. The number of iterations depends on the starting points P^0Q^0 as well as relative displacement of particles from the previous time step. In DEM applications, the change in particles' positions between two time steps is very small. As a result, the average number of iterations required in SLM is rarely larger than 3 (see Section 5).

3.5.1. Initialization. The first iteration for finding the shortest link requires two starting points P^0 and Q^0 on the surface of particles. If a shortest link is available from the previous DEM time step, then the endpoints of that link are used as P^0 and Q^0. If there is no shortest link available from the previous DEM time step (that is, no CP has been detected so far between the two particles), then three steps are performed to obtain the starting points P^0 and Q^0:

1. The PBP of the line that connects the centroids of the two particles is drawn.
2. Among all the vertices of particle A, the one with the minimum absolute distance from that plane is chosen as P^0.
3. Among all the vertices of particle B, the one with the minimum absolute distance from that plane is chosen as Q^0.

3.5.2. Special cases for identifying the shortest link. There are a few cases in which the SLM iterative scheme results in large number of iterations or potentially an infinite loop. One possible example is shown in Figure 8(a), in which both points P^0 and Q^0 are located on edges of particles. Applying the iterative SLM process results in a series of points P^i and Q^i, all remaining on the same two edges. Such large/infinite loops may occur only if both points P^0 and Q^0 are located on faces or edges of particles and are limited to those shown in Figure 8(a–c). In Figure 8(b) both points P^0 and Q^0 are located on faces of particles, and in Figure 8(c) one point is on an edge and the other one is on a face.

The above iterative algorithm is modified to prevent any potential large/infinite loop. At the beginning of each iteration step, points P^{i-1} and Q^{i-1} are examined. If they fall in one of the

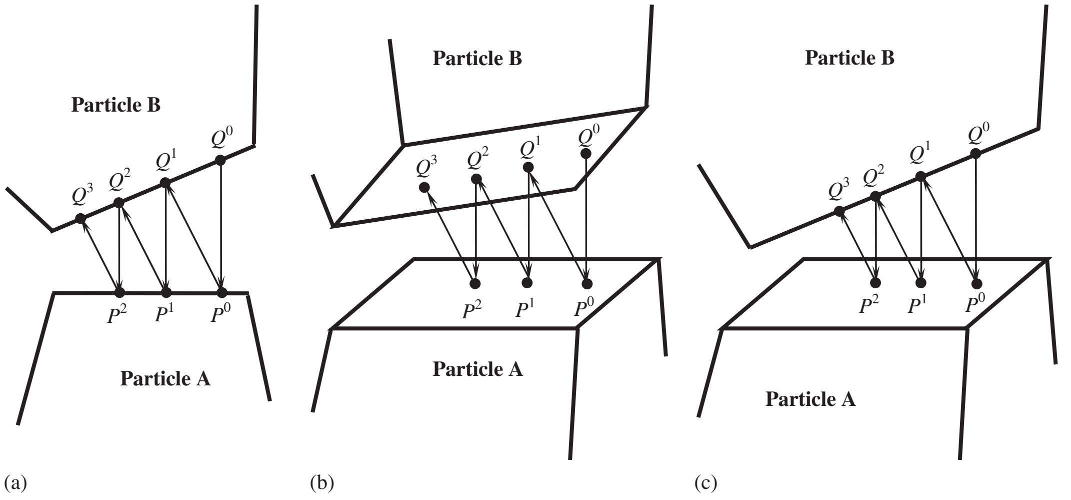

Figure 8. Cases with non-unique shortest link: (a) both points are on edges; (b) both points are on faces; and (c) one point is on a face and the other on an edge.

following three categories, then, instead of SLM iterative scheme, a direct geometrical approach is used to find the points P^i and Q^i:

Case 1: If P^{i-1} and Q^{i-1} are both located on edges (but not vertices) of particles, then points P^i and Q^i are found on the same edges, such that segment $|P^iQ^i|$ has the shortest possible length between the two edges. This is done in three steps as follows:

i. Let e_1 and e_2 denote the edges. Let Π be a plane parallel to both e_1 and e_2. The projections AB and CD of the two edges on plane Π are obtained (Figure 9).
ii. Points $P' \in AB$ and $Q' \in CD$ are found such that $P'Q'$ is the shortest link between segments AB and CD (see Appendix A for details).
iii. The points $P^i \in e_1$ and $Q^i \in e_2$ corresponding to P' and Q' are determined.

Case 2: If both P^{i-1} and Q^{i-1} are located on the faces of particles (but not edges or vertices), then the following steps are performed:

i. Let f_1 and f_2 denote the planes containing the faces (Figure 10). Let Π be the plane normal to both f_1 and f_2 that passes through P^{i-1}. The intersections of Π and the faces define two line segments (AB and CD in Figure 10).
ii. The endpoints of the shortest link between the two segments AB and CD (found as explained in Appendix A) define P^i and Q^i.

Case 3: If one of P^{i-1} or Q^{i-1} is located on an edge (not a face or a vertex) and the other one on a face (not an edge or a vertex), then the following steps are performed to find P^i and Q^i.

i. Let Π denote the plane passing through the edge (e_1 in Figure 11) normal to the face (f_1). Such a plane specifies a line segment AB on the face.
ii. The endpoints of the shortest link between the edge and the segment AB (obtained as explained in Appendix A) define P^i and Q^i.

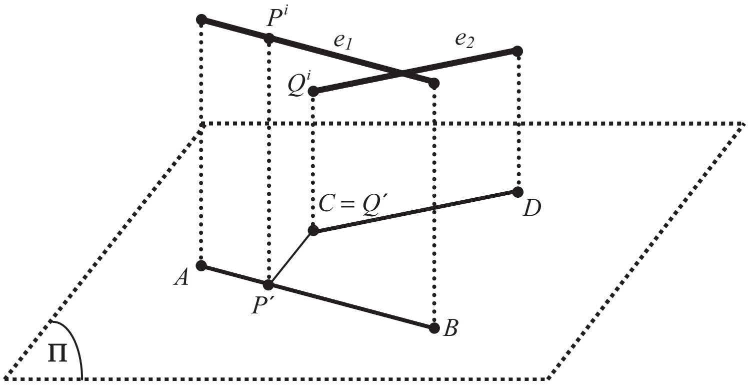

Figure 9. $P'Q'$ is the shortest link between segments AB and CD. P^i is the point corresponding to P'. Q^i is the point corresponding to Q'.

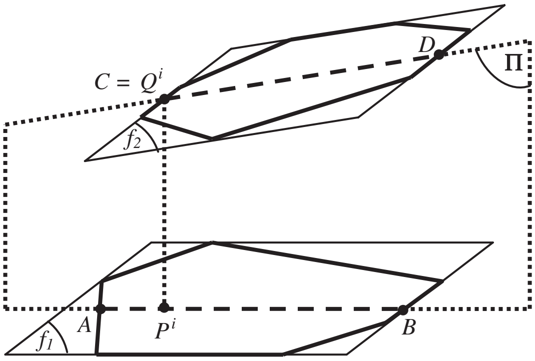

Figure 10. Π is the plane normal to both faces. The shortest link between segments AB and CD defines P^iQ^i.

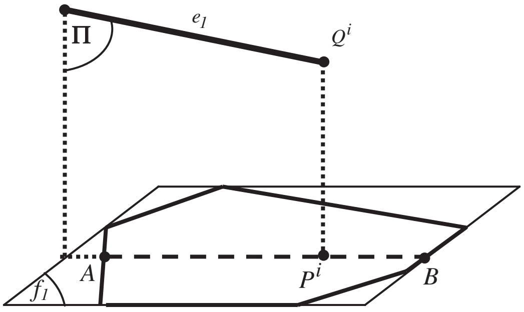

Figure 11. Π is the plane normal to the face passing through the edge. The shortest link between segment AB and edge e_1 defines P^iQ^i.

3.6. 2-D implementation of SLM

In 2-D, the CP is a line rather than a plane. However, definitions of the shortest link and the shortest distance as well as the statement of Section 3.3 still hold. The algorithm to obtain the shortest link between a point and a particle in Section 3.4 is simplified as follows:

Case 1: P_{i-1} is a vertex of particle A. In this case P_i is sought only on the two edges that share P_{i-1}.

Case 2: P_{i-1} is located on an edge of the particle but not on any vertices of that edge. In this case point P_i is sought only on the same edge (including its vertices).

In 2-D, CP identification algorithm of Section 3.5 is as follows. Starting from initial points $P^0 \in A$ and $Q^0 \in B$, a series of intermediate points $P^i \in A$ and $Q^i \in B$ $(i = 1, \ldots, n)$ are found, in which the ith iteration consists of the two following steps.

Step 1: For the point Q^{i-1}, find the closest point P^i on particle A.

Step 2: For the newly found point P^i, find the closest point Q^i on particle B.

In the beginning of each iteration, if P^{i-1} and Q^{i-1} are both located on edges of particles then, instead of the iterative process explained above, points P^i and Q^i are found on the same edges, such that segment $|P^iQ^i|$ has the shortest possible length between the two edges.

4. UNIQUENESS OF CP

Statement

For any two convex particles the CP defined in Section 2.1 is unique.

Proof

Assume that for two convex particles A and B, two different CPs exist, both satisfying conditions 1–3 of the definition of the CP, and each of them being the PBP of a shortest link between the two particles. Let PQ and $P'Q'$ be the associated shortest links, where P and P' belong to particle A, and Q and Q' belong to particle B (Figure 12).

The following proves that the PBP of these two shortest link are identical.

(i) $P_1Q_1Q_2P_2$ is a parallelogram

Any point $p = (p_1, p_2, p_3)$ on segment PP' (with p_i being the ith Cartesian component in the global Cartesian co-ordinate system 1-2-3) can be described as

$$p_i = P_i + m_i r, \quad 0 \leqslant r \leqslant 1;\ i = 1, 2, 3 \tag{3a}$$

where $P = (P_1, P_2, P_3)$ and subscript i represents the ith Cartesian component.

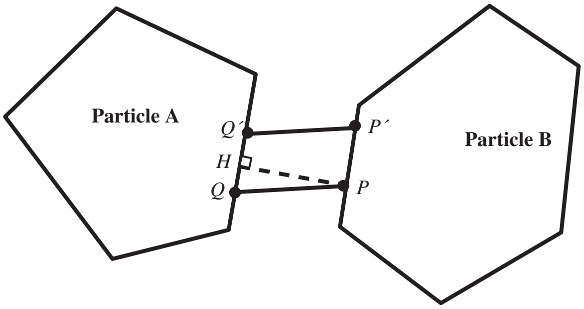

Figure 12. Uniqueness of the CP.

$\vec{m} = (m_1, m_2, m_3)$ is the vector pointing from P to P':

$$m_i = P_i - P'_i,\ i = 1, 2, 3$$

Likewise, any point $q = (q_1, q_2, q_3)$ on segment QQ' can be represented as

$$q_i = Q_i + n_i s, \quad 0 \leqslant s \leqslant 1;\ i = 1, 2, 3 \tag{3b}$$

where $Q = (Q_1, Q_2, Q_3)$ and $n_i = Q_i - Q'_i$, $i = 1, 2, 3$.

The distance from a point p on PP' to a point q on segment QQ' is a function of two variables r and s and is given by

$$L(r, s) = \sqrt{\sum\nolimits_{i=1}^{3} (p_i - q_i)^2}, \quad 0 \leqslant r,\ \ s \leqslant 1 \tag{4}$$

Substituting from (3a) in (4) results in

$$L(r, s) = \sqrt{\sum\nolimits_{i=1}^{3} (P_i - Q_i + m_i r - n_i s)^2}, \quad 0 \leqslant r,\ \ s \leqslant 1 \tag{5}$$

Note that

$$|PQ| = L(0, 0) = \sqrt{\sum\nolimits_{i=1}^{3} (P_i - Q)^2} = L \tag{6a}$$

and

$$|P'Q'| = L(1, 1) = \sqrt{\sum\nolimits_{i=1}^{3} (P_i - Q + m_i - n_i)^2} = L \tag{6b}$$

Combining (6a) and (6b)

$$\sqrt{\sum\nolimits_{i=1}^{3} (P_i - Q)^2} = \sqrt{\sum\nolimits_{i=1}^{3} (P_i - Q + m_i - n_i)^2}$$

or

$$2 \sum_{i=1}^{3} (P_i - Q_i)(m_i - n_i) = -\sum_{i=1}^{3} (m_i - n_i)^2 \tag{7}$$

Convexity of particles guarantees that any point on segments PP' and QQ' is located inside or on the surface of particles A and B, respectively. As a result, the distance between any point p on PP' and any point q on QQ' is greater than the shortest distance L:

$$L(r, s) \geqslant L, \quad 0 \leqslant r,\ \ s \leqslant 1$$

or

$$L(r, s)^2 - L^2 \geqslant 0, \quad 0 \leqslant r,\ \ s \leqslant 1 \tag{8}$$

Substituting (5) in (8)

$$\sum_{i=0}^{3} (P_i - Q_i + m_i r - n_i s)^2 - (P_i - Q_i)^2 \geqslant 0, \quad 0 \leqslant r,\ \ s \leqslant 1$$

or

$$\sum_{i=0}^{3} 2(P_i - Q_i)(m_i r - n_i s) + \sum_{i=0}^{3} (m_i r - n_i s)^2 \geqslant 0, \quad 0 \leqslant r,\ \ s \leqslant 1$$

For $(r = s)$

$$2r\sum_{i=0}^{3}(P_i - Q_i)(m_i - n_i) + r^2\sum_{i=0}^{3}(m_i - n_i)^2 \geqslant 0, \quad 0 \leqslant r \leqslant 1 \tag{9}$$

Substituting (7) in (9) gives

$$-r\sum_{i=0}^{3}(m_i - n_i)^2 + r^2\sum_{i=0}^{3}(m_i - n_i)^2 \geqslant 0, \quad 0 \leqslant r \leqslant 1$$

or

$$(r^2 - r)\sum_{i=0}^{3}(m_i - n_i)^2 \geqslant 0, \quad 0 \leqslant r \leqslant 1 \tag{10}$$

On the other hand, $\sum_{i=0}^{3}(m_i - n_i)^2$ is always zero or positive and for any $0 < r < 1$, $(r^2 - r)$ is negative; that is,

$$(r^2 - r)\sum_{i=0}^{3}(m_i - n_i)^2 \leqslant 0, \quad 0 \leqslant r \leqslant 1 \tag{11}$$

Combining (10) and (11) results in

$$(r^2 - r)\sum_{i=0}^{3}(m_i - n_i)^2 = 0, \quad 0 \leqslant r \leqslant 1$$

or

$$\sum_{i=0}^{3}(m_i - n_i)^2 = 0$$

which leads to

$$m_i = n_i, \quad i = 1, 2, 3 \tag{12}$$

This proves that $PQ = P'Q'$ and $PQ \| P'Q'$.

Therefore, $P_1Q_1Q_2P_2$ is a parallelogram.

(ii) $P_1Q_1Q_2P_2$ is a rectangle

If PP' is not perpendicular to PQ (Figure 12), then a point such as H can be found on QQ', such that $|PH| < |PQ|$, which is a contradiction. As a result, $P_1P_2 \perp P_1Q_1$.

Therefore, $P_1Q_1Q_2P_2$ is a rectangle.

(iii) CP is unique

From (ii) PBP's of P_1Q_1 and P_2Q_2 are identical.

Therefore, the two CPs are identical. □

5. IMPLEMENTATION AND EXAMPLES

The performance of the SLM algorithm is demonstrated through a series of examples for 2-D and 3-D particles. The computation time is compared with that using the FCP algorithm proposed by Nezami *et al.* [11] and the original iterative algorithm used by Cundall [10].

5.1. Contact detection in 2-D

Figure 13 depicts nine pairs of particles in various configurations in 2-D. For each configuration, no previous CP is available. The CPs are obtained with both SLM, FCP and the conventional CP method and are shown in the figure, along with the shortest link for each configuration. The resulting CPs from all three algorithms are identical. The speed up ratio *R*, defined as the ratio of the CPU run time for the conventional algorithm to that for SLM or FCP, is shown for each configuration and varies from 5 to 16 for SLM compared to 1–5 for FCP.

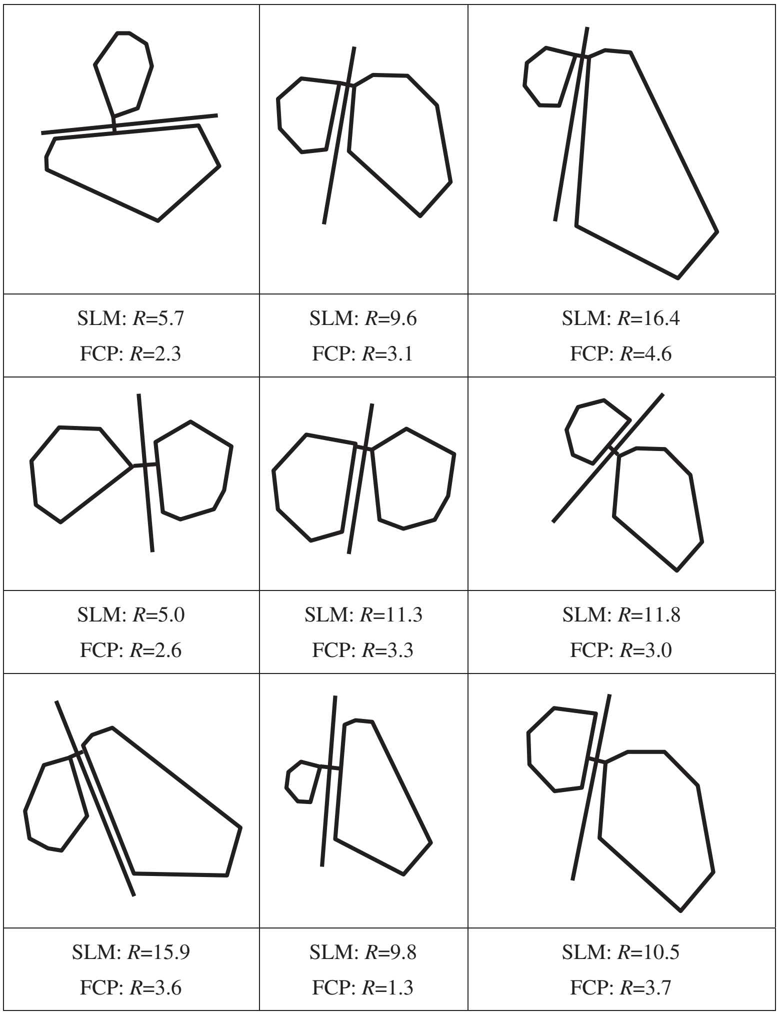

Figure 13. SLM performance in 2-D relative to iterative method.

5.2. *Contact detection in 3-D*

SLM algorithm is implemented in the 3-D computer program BLOCKS3D version 2.0, developed by the authors. The program simulates polyhedral particles of any shape and size distribution. The performance of SLM is demonstrated through a series of examples and compared to FCP and conventional common plane method [10] and FCP [11].

5.2.1. Simulation of particle flow. This set of examples consists of ten simulations. For each simulation a number of particles are dropped into a cubic box. For the first two simulations (500 and 4000 particles) a 0.30 m × 0.30 m box is used while the other eight simulations are done using a 1.0 m × 1.0 m box. The average particle size in all simulations is $D_{50} = 3.8$ cm, with minimum and maximum particle sizes of 2.0 and 5.0 cm, respectively. The time step is 1.3×10^{-4} s. The particles' geometries are chosen evenly from those shown in Figure 14, and have the following contact properties:

Normal stiffness K_n: 130 KN/m
Shear stiffness K_s: 102 KN/m
Contact friction angle ϕ_μ: 35°

Small stiffness values are chosen deliberately to increase the time step and reduce the computational effort. Once the particles are accumulated inside the box, two adjacent walls of the box are removed, allowing the particles to flow in two directions.

Figure 15 plots three snapshots of the simulation for 45 000 particles. The test is continued until the final stable configuration is achieved.

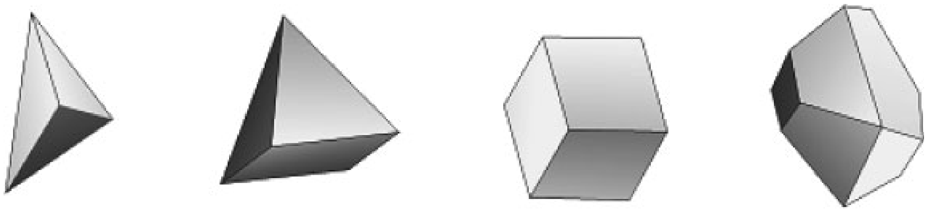

Figure 14. Particles with 4 vertices (tetrahedron), 5 vertices (pyramid), 8 vertices (cube) and 14 vertices.

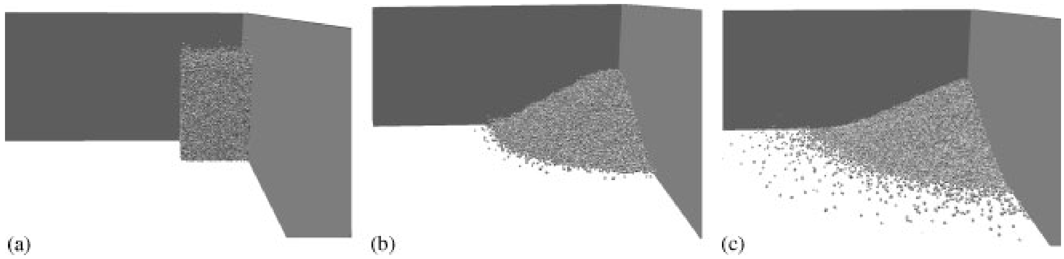

Figure 15. Particle flow: simulation with 45 000 particles: (a) $t = 0.0$ s; (b) $t = 1.4$ s; and (c) $t = 2.0$ s.

For each simulation, the speed up ratio R (defined as the ratio of the CPU run time spent in the CP detection subroutines for conventional CP method to that for the FCP and SLM) is evaluated during the first 1000 time steps of the flow phase and is shown in Figure 16 as a function of number of particles. The speed up ratio ranges for FCP algorithm from 13 to 10 and for SLM algorithm from 21 to 17. The speed up ratio decreases with increasing number of particles; but, nevertheless, SLM provides consistently significant speed up. For SLM simulations, the number of iterations required to calculate the CP at each time step is recorded. It is observed that around 95% of the contacts require two or less iterations in SLM.

5.2.2. Simulation of soil–tool interaction. This simulation involves pushing a loader bucket into a pile of soil and lifting a load. The soil pile is generated by dropping 21 952 particles into a soil bin, composed of two side plates 0.60 m apart and two end plates 0.4 m apart. Then one of the end walls is removed allowing the particles to flow along the bin. The particles geometries are chosen evenly from those shown in Figure 14, with an average particle size of $D_{50} = 2.5$ cm with minimum and maximum particle sizes of 1.0 and 4.0 cm, respectively. Contact properties of the particles are the same as those in the prior simulations. Time step is chosen to be 1.2×10^{-4} s.

The bucket is a scaled, simplified prototype of a typical commercial loader bucket, modelled with five plates, as shown in Figure 17. It is initially positioned at an elevation 0.04 m above the ground. The displacement of the bucket consists of a horizontal translation phase in which the

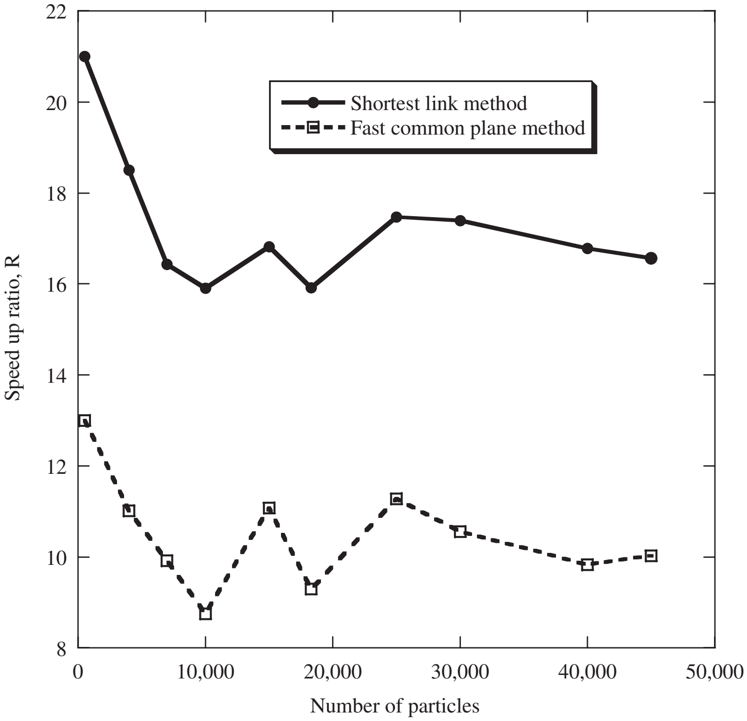

Figure 16. Speed up ratio R of SLM and FCP methods compared to conventional CP method.

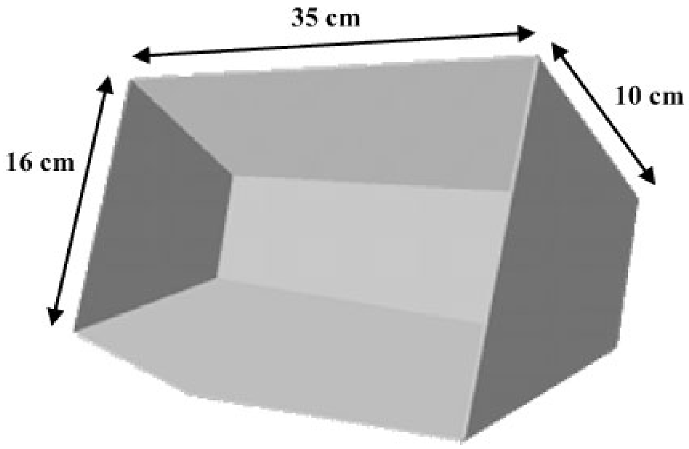

Figure 17. Loader bucket, simulated by five plates.

Table I. Soil–tool interaction, speed up of SLM and FCP relative to iterative CP procedure.

				Speed up ratio	
Displacement phase	Displacement	Start time (s)	End time (s)	SLM	FCP
Horizontal translation	0.25 m	0	8	18.31	12.54
Rotation	65°	5	16	18.14	11.12
Vertical translation	0.18 m	13	20	18.05	11.68

bucket is pushed into the soil pile, a rotational phase, and a vertical translation phase. Each phase is performed at a constant velocity.

Table I shows the duration of each phase along with the corresponding total displacement or rotation of the bucket. Figure 18 plots the initial configuration and a cross section of the pile (passing through the centreline of the bucket) at three different time steps.

The speed up ratio of both SLM and FCP compared to the conventional CP method are calculated for the first 1000 time steps of each phase of bucket motion and shown in Table I. A speed up ratio of about 18 is observed for SLM and 12 for FCP. More than 98% of the contacts require two or less iterations in SLM.

6. CONCLUSION

It is shown that the common plane can be uniquely defined as the perpendicular bisector plane of the shortest link between the two particles. Based on this observation, a new algorithm is proposed in which the shortest link is sought. The algorithm is developed for convex 2-D and 3-D polyhedral particles.

The performance of the new algorithm is examined through several examples. It is observed that for large number of particles, the current algorithm runs up to 17 times faster than conventional common plane.

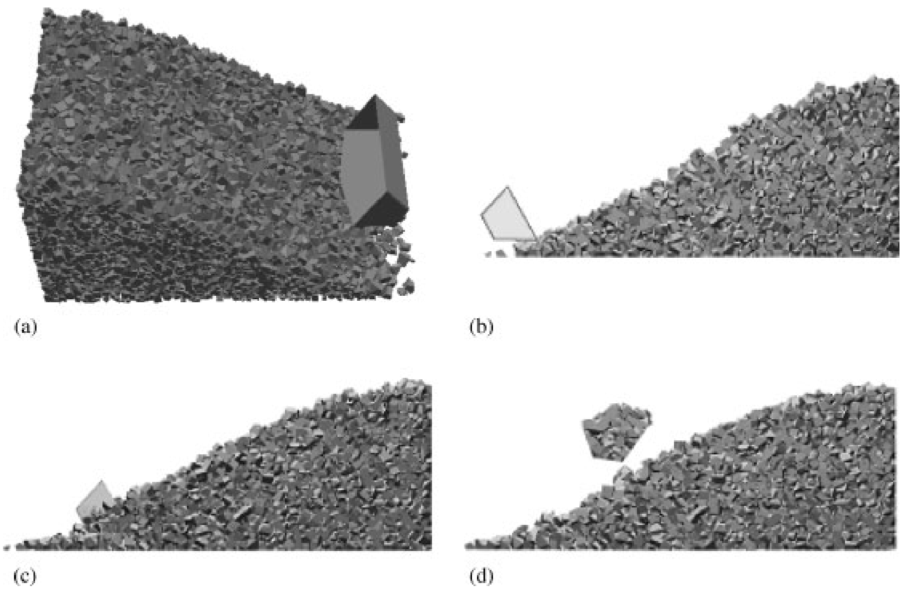

Figure 18. Soil–tool interaction simulation: (a) Initial configuration; (b) $t = 0.0$ s; (c) $t = 5.0$ s; and (d) $t = 20.0$ s.

APPENDIX A

This section provides an algorithm to obtain the shortest link between any two co-planar line segments. The algorithm is frequently used in SLM scheme (Section 3.5.2). Let AB and CD denote two co-planar segments as shown in Figure A1.

If AB and CD intersect, then the intersection point is the shortest link. In this case the shortest link has a length of zero and is reduced to a point, Figure 19(a).

If AB and CD do not intersect, then the shortest link is the shortest segment among the following, Figure A1(b).

i. AC, BC, AD, CD (Figure A1(b), left).
ii. AA_1, BB_1, CC_1 and DD_1 where A_1 and B_1 are the projection of A and B on CD and C_1 and D_1 are the projections of C and D on AB (Figure A1(b), right). Note that among AA_1, BB_1, CC_1 and DD_1, only those with both ends inside the line segments are considered.

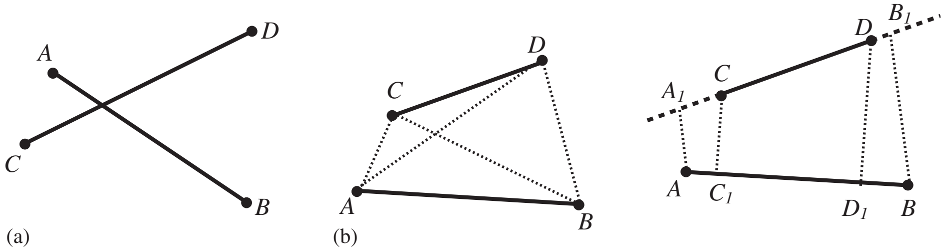

Figure A1. The shortest link between two co-planar segments AB and CD: (a) AB and CD intersect; and (b) AB and CD do not intersect; CC_1 is the shortest link.

ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant No. CMS-0113745 and Caterpillar, Inc. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or Caterpillar, Inc. This support is gratefully acknowledged. The authors would also like to thank Mr Ibrahim Mohammad for preparing the visualization code VisDEM3D.

REFERENCES

1. Cundall PA. A computer model for simulating progressive large-scale movements in block rock mechanics. *Proceedings of the Symposium of International Society of Rock Mechanics*, vol. 8. Nancy, 1971.
2. Cundall PA, Strack ODL. A discrete numerical model for granular assemblies. *Geotechnique* 1979; **29**(1):47–65.
3. Bardet JP. Numerical simulation of the incremental responses of idealized granular materials. *International Journal of Plasticity* 1994; **10**(8):879–908.
4. Thornton C, Barnes DJ. Computer simulated deformation of compact granular assemblies. *Acta Mechanica* 1986; **64**:45–61.
5. Bagi K. Stress and strain in granular assemblies. *Mechanics of Materials* 1996; **22**:165–177.
6. Cleary PW, Campbell CS. Self-lubrication for long run-out landslides: examination by computer simulation. *Journal of Geophysical Research—Solid Earth* 1993; **98**(B12):21911–21924.
7. Campbell CS, Cleary PW, Hopkins MA. Large-scale landslide simulations: global deformation, velocities, and basal friction. *Journal of Geophysical Research—Solid Earth* 1995; **100**(B5):8267–8283.
8. Hopkins MA, Hibler WD, Flato GM. On the numerical simulation of the sea ice ridging process. *Journal of Geophysical Research—Oceans* 1991; **96**(C3):4809–4820.
9. Cleary PW. DEM simulation of industrial particle flows: case studies of dragline excavators, mixing in tumblers and centrifugal mills. *Powder Technology* 2000; **209**(1–3):83–104.
10. Cundall PA. Formulation of a three-dimensional distinct element model. Part I: a scheme to detect and represent contacts in a system composed of many polyhedral blocks. *International Journal of Rock Mechanics, and Mining Sciences & Geomechanics Abstracts* 1988; **25**(3):107–116.
11. Nezami E, Hashash YMA, Zhao D, Ghaboussi J *et al.* A fast contact detection algorithm for 3-D discrete element method. *Computers and Geotechnics* 2004; **31**(7):575–587.