Model of Sphere-to-Edge Contact for Spherical Discontinuous Deformation Analysis

Long Wang[1]; Yu-Yong Jiao[2]; Gang-Hai Huang[3]; Fei Zheng[4]; and Qiang Zhao[5]

Abstract: Spherical element-based discontinuous deformation analysis (SDDA) usually uses simple types of contact such as sphere–sphere and sphere–plane contact for detection. However, in practical engineering simulations, there is also found sphere–edge contact in the context of interactions between spheres and fixed boundaries. This incompleteness of contact types in the original SDDA restricts its application to problems involving complex boundary shapes. To fill this gap, this paper presents a sphere–edge contact model to extend the application of SDDA to complex boundary shapes such as edges, corners, and finite and infinite planes. The contact formulation is deduced and the algorithm is implemented in SDDA source code developed by the authors. Four numerical examples are simulated for verification. The results show that the proposed model is correct and effective. **DOI: [10.1061/(ASCE)GM.1943-5622.0001021.](#)** © *2017 American Society of Civil Engineers.*

Author keywords: Spherical element-based discontinuous deformation analysis (SDDA); Sphere-to-edge contact; Improvement.

Introduction

Discontinuous deformation analysis (DDA) is a numerical method (Shi 1998). It can be used to analyze the discontinuous deformation behavior of a discrete blocky system. DDA represents rock masses as an assemblage of discrete blocks and joints as interfaces between blocks. The principal equations, which are derived from Newton's second law or the principle of minimum potential energy, are solved implicitly. Since the DDA method can simulate the large displacements of a system, it has been applied widely in many practical engineering circumstances such as tunnels (Yeung and Leong 1997; Wu et al. 2004; Chen et al. 2016), rockfalls (Wu et al. 2005; Chen et al. 2013), landslides (Sitar et al. 2005; Zhang et al. 2013; Jiao et al. 2014; Zhang et al. 2014; Zhang et al. 2015), and others (Bao et al. 2012; Jiang et al. 2013; Nie et al. 2014).

[1]Ph.D. Candidate, State Key Laboratory of Geomechanics and Geotechnical Engineering, Institute of Rock and Soil Mechanics, Chinese Academy of Sciences, Wuhan 430071, P.R. China; Univ. of Chinese Academy of Sciences, Beijing 100049, P.R. China. E-mail: 540676302@qq.com

[2]Professor, Faculty of Engineering, China Univ. of Geosciences, Wuhan 430074, P.R. China (corresponding author). ORCID: https://orcid.org/0000-0001-9081-6932. E-mail: yyjiao@whrsm.ac.cn

[3]Lecturer, School of Civil Engineering, Hunan Univ. of Science and Technology, Xiangtan 411201, P.R. China. E-mail: 307354163@qq.com

[4]Ph.D. Candidate, State Key Laboratory of Geomechanics and Geotechnical Engineering, Institute of Rock and Soil Mechanics, Chinese Academy of Sciences, Wuhan 430071, P.R. China; Univ. of Chinese Academy of Sciences, Beijing 100049, P.R. China; Visiting Student, Dept. of Civil and Environmental Engineering, Univ. of California, Berkeley, Berkeley, CA 94720. E-mail: feizhengprchina@hotmail.com

[5]Ph.D. Candidate, State Key Laboratory of Geomechanics and Geotechnical Engineering, Institute of Rock and Soil Mechanics, Chinese Academy of Sciences, Wuhan 430071, P.R. China; Univ. of Chinese Academy of Sciences, Beijing 100049, P.R. China. E-mail: zhaoqiang_201314@163.com

For any discontinuity-based numerical method, contact analysis plays an important role. Unlike the distinct element method (DEM), large penetration between blocks is not allowed in DDA. Accordingly, contact analysis in DDA is more precise and complex. However, Cheng (1998) pointed out that the process of penetration detection in DDA is extremely time-consuming, and he presented two modifications to DDA in the problems of contact constraint. Many methods and algorithms for the accurate and efficient solution of contacts have been presented for DDA. Fan and He (2015) developed a new preprocessing and angle-based method for two-dimensional DDA to determine entrance edges; this new method can not only reduce forced contacts but also maintain the movement trend of blocks and speed up open–close iterations. Jiang and Yeung (2004) and Yeung et al. (2007) proposed an edge-to-edge contact model and a point-to-face contact model for three-dimensional (3D) DDA to effectively identify the contact types between two polyhedron-based blocks. Wu et al. (2014) coupled neighbor searching with a contact-pattern identification process and presented a multishell cover algorithm for contact detection in 3D DDA. Shi (2015) introduced a new concept of an entrance block to solve the contacts between two general blocks; given a reference point, the contact computation can be simplified. To efficiently solve contacts of polyhedral blocks, Zheng et al. (2016) proposed a new angle-based contact detection approach which consists of three phases, i.e., a rough search phase, a delicate search phase, and an identification phase.

However, the contact detection of polyhedral blocks is complex and time consuming. Against this background, fundamental studies of the spherical element DDA (SDDA) have commenced (Zhao 2000; Beyabanaki and Bagtzoglou 2012; Jiao et al. 2015). SDDA uses relatively simple contact types like sphere–sphere and sphere–infinite plane contact to perform the contact detection. However, in practical engineering simulations, there is a third type of contact, sphere–edge contact, for dealing with the interaction between spheres and fixed boundaries. This incompleteness of contact types in the original SDDA might result in errors in simulated results.

This paper presents a sphere–edge contact model for SDDA to simulate complex sphere–boundary problems. The corresponding formulation is deduced and the contact detection algorithm is implemented into the SDDA source code. The proposed sphere–boundary contact detection algorithm consists of two main loops: sphere–plane contact search and sphere–edge contact detection. For

verification, four numerical examples are simulated, and the results show that the proposed model is correct and effective.

SDDA

Displacement and Deformation of Sphere

SDDA is a displacement-based method. Large displacement and relative movement between spheres are the accumulation of small displacement within each time step. Sphere elements are assumed to be rigid. Therefore, the movement and deformation of the sphere are defined by six independent deformation variables in the deformation matrix (D_i):

$$(D_i) = \begin{pmatrix} d_x & d_y & d_z & r_x & r_y & r_z \end{pmatrix}^T \quad (1)$$

where (d_x, d_y, d_z) are the displacements along the x-, the y- and the z-axes, respectively and (r_x, r_y, r_z) are the rotation angles around the sphere center.

The displacement function of an arbitrary point (x,y,z) of sphere i can be written as

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = [T_i(x,y,z)](D_i) \quad (2)$$

where $[T_i(x,y,z)]$ is called the displacement transformation matrix of sphere i,

$$[T_i(x,y,z)] = \begin{pmatrix} 1 & 0 & 0 & 0 & \bar{z} & -\bar{y} \\ 0 & 1 & 0 & -\bar{z} & 0 & \bar{x} \\ 0 & 0 & 1 & \bar{y} & -\bar{x} & 0 \end{pmatrix} \quad (3)$$

where $\bar{x} = x-x_c$, $\bar{y} = y-y_c$, $\bar{z} = z-z_c$ and (x_c, y_c, z_c) are the coordinates of the center of sphere i.

General Equilibrium Equations

According to the principle of minimum potential energy, the governing equation of a system of spheres can be obtained from the extremum condition of the total potential energy (including different kinds of deformation potential energy and potential energy of external forces). Assuming that there are n spherical elements in the computational model, the governing equation can be expressed as follows:

$$\begin{bmatrix} [K_{11}] & [K_{12}] & \cdots & [K_{1n}] \\ [K_{21}] & [K_{22}] & \cdots & [K_{2n}] \\ \vdots & \vdots & \ddots & \vdots \\ [K_{n1}] & [K_{n2}] & \cdots & [K_{nn}] \end{bmatrix} \begin{bmatrix} [D_1] \\ [D_2] \\ \vdots \\ [D_n] \end{bmatrix} = \begin{bmatrix} [F_1] \\ [F_2] \\ \vdots \\ [F_n] \end{bmatrix} \quad (4)$$

where $[K_{ij}]$ ($i,j = 1,2,\cdots,n$) is a 6×6 submatrix, $[K_{ii}]$ depends on the material properties of sphere i, and $[K_{ij}]$ ($i \neq j$) depends on the interactions between sphere i and sphere j; $[D_i]$ are 6×1 submatrices of the basic unknowns of sphere i; $[F_i]$ are 6×1 submatrices of the generalized forces acting on sphere i.

In the sphere system, all the contact force and displacement constraints of spheres are caused by the neighboring spheres, and all contact issues ultimately come down to the establishment of the simultaneous equilibrium equations. By using the minimum total potential energy principle, the effects of external forces and internal interaction among contacting spheres are both transferred into submatrices which are added to the governing equation. By the same token, the submatrices of the volume loading, inertial forces, point loading, displacement constraint, and directional constraint can be easily obtained.

Problem of Boundary Condition in the Original SDDA

In the original SDDA, the sphere–sphere and the sphere–infinite plane types of contact are used to perform contact detection, as shown in Figs. 1(a and b). However, in practical engineering simulations, they are not appropriate for modeling the problem of sphere–convex contact, as shown in Figs. 1(c and d). This incompleteness of the contact types in the original SDDA might result in errors in the simulated results.

Sphere–Edge Contact

In this section, to solve the sphere–boundary condition problem, a sphere–edge contact model is proposed, and the formulation is deduced. The shape of the boundaries in the modified SDDA consists of one or more fixed triangular planes. As shown in Fig. 2, any kind of sphere–convex contact can be simplified into sphere–edge contact. Assume that sphere i will contact edge **AB** in the next time step. The radius of the sphere is R_i, and the coordinate of sphere center is C_i (x_{ci}, y_{ci}, z_{ci}). Line segment **L** joints point P_2 and the center of sphere i. P_2 (x_2, y_2, z_2) is the contact point.

Assume that point P_1 (x_1, y_1, z_1) on the surface of sphere i will contact point P_2, which is on edge **AB** in the next time step. The coordinates of P_1 are obtained by

$$P_1 : \begin{cases} x_1 = x_{ci} + V_x R_i \\ y_1 = y_{ci} + V_y R_i \\ z_1 = z_{ci} + V_z R_i \end{cases} \quad (5)$$

where (V_x, V_y, V_z) are the unit vectors of $\overrightarrow{C_i P_2}$.

Assuming that the displacement of point P_1 is (u_1, v_1, w_1), therefore the distance d between points P_1 and P_2 is obtained by

$$d^2 = \begin{bmatrix} x_2 - (x_1 + u_1) \\ y_2 - (y_1 + v_1) \\ z_2 - (z_1 + w_1) \end{bmatrix}^T \begin{bmatrix} x_2 - (x_1 + u_1) \\ y_2 - (y_1 + v_1) \\ z_2 - (z_1 + w_1) \end{bmatrix}$$
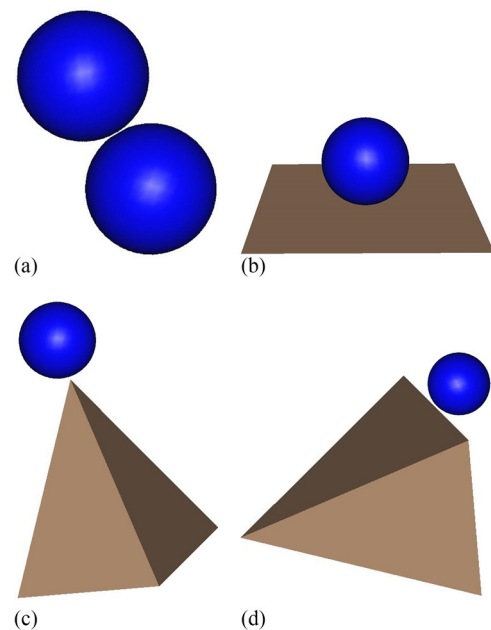

Fig. 1. Contact model: (a) sphere–sphere contact; (b) sphere–plane contact; (c) sphere–convex contact; (d) sphere–convex contact
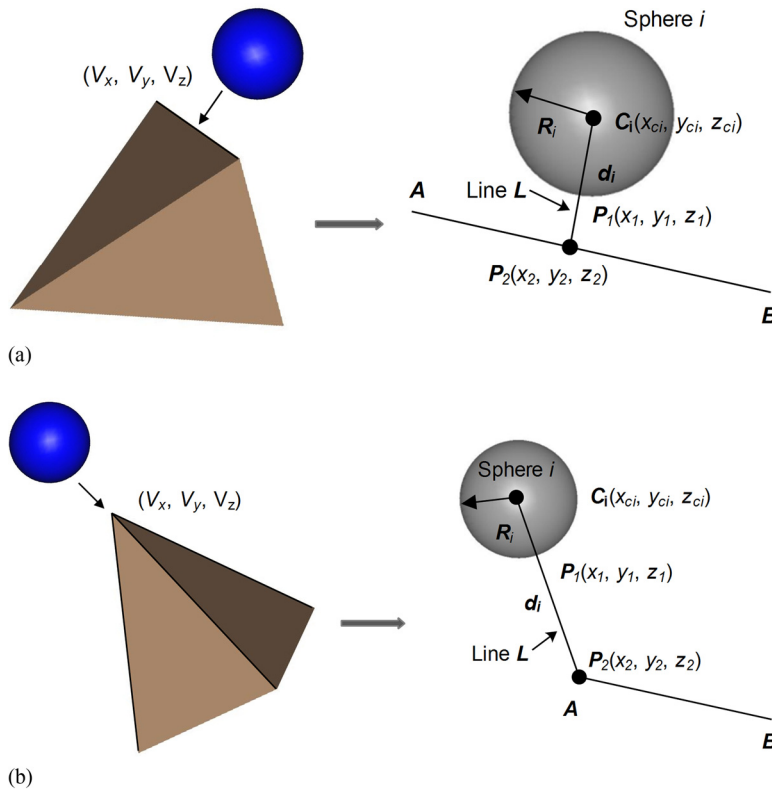
Fig. 2. Sphere–edge contact: (a) sphere–edge contact; (b) sphere–tetrahedron contact

$$
d^2 = \left\{ \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix}^T - (D_i)^T [T_i(x_1, y_1, z_1)]^T \right\}
$$
$$
\times \left\{ \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} - [T_i(x_1, y_1, z_1)](D_i) \right\}
$$

$$
d^2 = \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} T \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} - 2(D_i)^T [T_i(x_1, y_1, z_1)]^T
$$
$$
\times \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} + (D_i)^T [T_i(x_1, y_1, z_1)]^T [T_i(x_1, y_1, z_1)](D_i) \quad (6)
$$

Normal Contact Submatrices of Sphere–Edge Contact

The normal distance d_N between sphere i and the edge is the projection of d onto vector (V_x, V_y, V_z) and is obtained by

$$
d_N = \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}^T \begin{bmatrix} x_2 - (x_1 + u_1) \\ y_2 - (y_1 + v_1) \\ z_2 - (z_1 + w_1) \end{bmatrix} = \lambda - \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}^T \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix}
$$
$$
\text{where } \lambda = \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}^T \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix}. \quad (7)
$$

The potential energy of the normal spring between sphere and edge is

$$
\prod_N = \frac{p_N}{2} d_N^2 = \frac{p_N}{2} \left[\lambda - \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix}^T \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix} \right]
$$
$$
\times \left[\lambda - \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}^T \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} \right]
$$

$$
\prod_N = \frac{p_N}{2} \lambda^2 - p_N \lambda \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix}^T \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}
$$
$$
+ \frac{p_N}{2} \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix}^T \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix} \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}^T \begin{pmatrix} u_1 \\ v_1 \\ w_1 \end{pmatrix} \quad (8)
$$

where p_N is the stiffness of the normal contact spring.

After rearrangement, Eq. (8) becomes

$$
\prod_N = \frac{p_N}{2} \lambda^2 - p_N \lambda (D_i)^T (V_N)^T + \frac{p_N}{2} (D_i)^T (V_N)^T (V_N)(D_i) \quad (9)
$$

where $(V_N) = \begin{pmatrix} V_x & V_y & V_z & 0 & 0 & 0 \end{pmatrix}$.

By minimizing $\prod_N$, the normal contact submatrices of sphere–edge contact are obtained

$$
p_N (V_N)^T (V_N) \rightarrow (K_{ii}) \quad (10)
$$

$$p_N \lambda (V_N)^T \rightarrow (F_i) \tag{11}$$

Shear Contact Submatrices of the Sphere and Edge

The relative shear displacement d_s between points P_1 and P_2 is the projection of d to the edge, so $d_s^2 = d^2 - d_N^2$. Assuming that the stiffness of the shear spring between sphere i and edge is p_S, the potential energy of this shear spring is

$$
\begin{aligned}
\prod_S &= \frac{p_S}{2} d_S^2 = \frac{p_S}{2} d^2 - \frac{p_S}{2} d_N^2 + \frac{p_S}{2} \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix}^T \\
&\times \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} - p_S (D_i)^T [T_i(x_1, y_1, z_1)]^T \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} \\
&+ \frac{p_S}{2} (D_i)^T [T_i(x_1, y_1, z_1)]^T [T_i(x_1, y_1, z_1)](D_i) - \frac{p_S}{2} \lambda^2 \\
&+ p_S \lambda (D_i)^T (V_N)^T - \frac{p_S}{2} (D_i)^T (V_N)^T (V_N)(D_i)
\end{aligned}
\tag{12}
$$

By minimizing $\prod_S$, the shear contact submatrices of sphere–edge contact are obtained:

$$p_S [T_i(x_1, y_1, z_1)]^T [T_i(x_1, y_1, z_1)] - p_S (V_N)^T (V_N) \rightarrow (K_{ii}) \tag{13}$$

$$p_S [T_i(x_1, y_1, z_1)]^T \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} - p_S \lambda (V_N)^T \rightarrow (F_i) \tag{14}$$

Friction Force Submatrices of Sphere–Edge Contact

When the normal contact force between sphere and edge is compressive, and the shear contact force is large enough to cause sliding, the friction force is applied to the point P_1 of the sphere. The value of the friction force is obtained by

$$f_S = p_N d_p \tan \varphi \tag{15}$$

where φ is the sphere–edge friction angle and d_P is the penetration between sphere and edge at the end of the last time step.

Assuming that at the end of the time step, point P_1 moves to point $P_{1*}(x_{1*}, y_{1*}, z_{1*})$, and the projection vector of $\overrightarrow{P_1 P_{1*}}$ on the edge is marked as (T_x, T_y, T_z). They are obtained by

$$
\begin{cases}
T_x = x_{1*} - x_1 - V_x d_2 \\
T_y = y_{1*} - y_1 - V_y d_2 \\
T_z = z_{1*} - z_1 - V_z d_2
\end{cases}
\tag{16}
$$

where $d_2 = \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix}^T \begin{pmatrix} x_{1*} - x_1 \\ y_{1*} - y_1 \\ z_{1*} - z_1 \end{pmatrix}$ is the projection of $\overrightarrow{P_1 P_{1*}}$ to vector (V_x, V_y, V_z).

Assuming that the unit vector of (T_x, T_y, T_z) is (l, m, n), the potential energy of the friction force, f_S is obtained:

$$
\begin{aligned}
\prod_f &= f_S \begin{pmatrix} u_1 & v_1 & w_1 \end{pmatrix} \begin{pmatrix} l & m & n \end{pmatrix}^T \\
&= f_S (D_i)^T [T_i(x_1, y_1, z_1)]^T \begin{pmatrix} l & m & n \end{pmatrix}^T
\end{aligned}
\tag{17}
$$

If $\prod_f$ is minimized, the friction submatrices of the sphere and edge are obtained:

$$-f [T_i(x_1, y_1, z_1)]^T \begin{pmatrix} l & m & n \end{pmatrix}^T \rightarrow (F_i) \tag{18}$$

Implementation of Sphere–Edge Contact Detection

Contact Detection

As shown in Fig. 3, the convex consists of fixed planes F_1 and F_2, and F is the line segment where fixed plane F_1 intersects fixed plane F_2. Assume that sphere i will contact the convex in the next time step, and point P_1 is the contact point on the surface of the sphere. Point P_2, the contact point, is on the plane, and it is the projection of the center of the sphere onto fixed plane.

In contact detection, it is first necessary to detect whether the contact type is sphere–plane or sphere–edge. If point P_2 is within in the plane, this indicates that the contact type is sphere–plane contact; otherwise, the sphere contacts edge F.
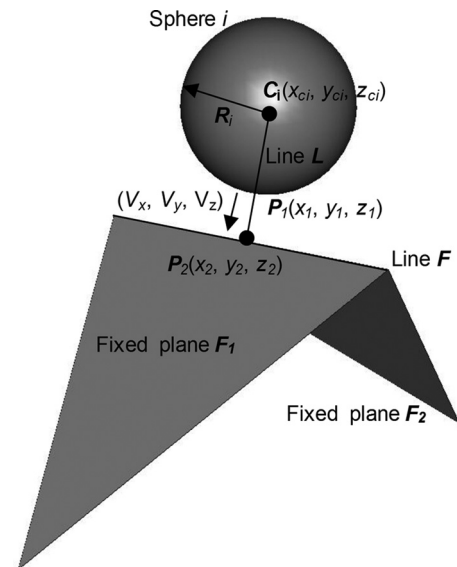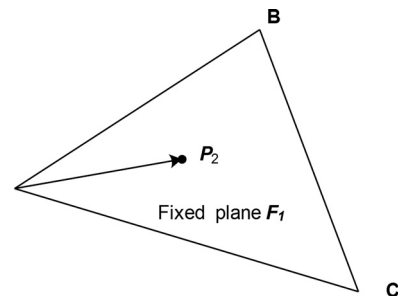

Fig. 3. Sphere–convex contact

Fig. 4. Relation between point and line segment

To detect whether point P_2 is in the fixed plane or not, the vector method is adopted in this study. As shown in Fig. 4, along the route A–B–C–A, point P_2 is always on the right side of line segments AB, BC, and CA, or, in other words, if point P_2 has the same direction as
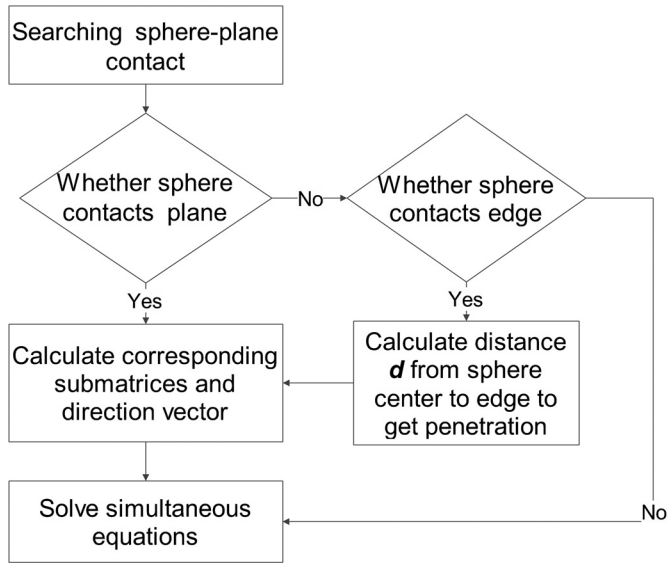

Fig. 5. Proposed algorithm

Table 1. Mechanical Properties Used in the Numerical Examples

Parameter	Value
Density (kg/m³)	2,700
Gravitational acceleration (m/s²)	9.8
Normal spring stiffness (GN/m)	1
Shear spring stiffness (GN/m)	0.25
Time step (s)	1×10^{-5}
Tensile strength (MPa)	0
Cohesion (MPa)	0
Friction angle (degrees)	30
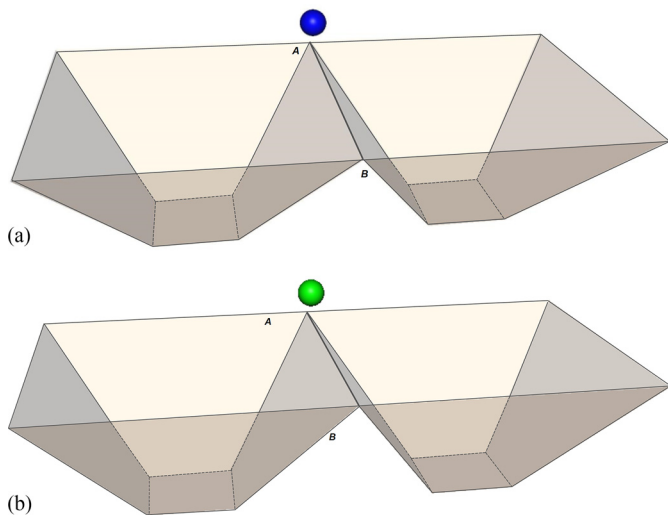
Fig. 6. Computed model of free falling body: (a) asymmetrical model; (b) symmetrical model
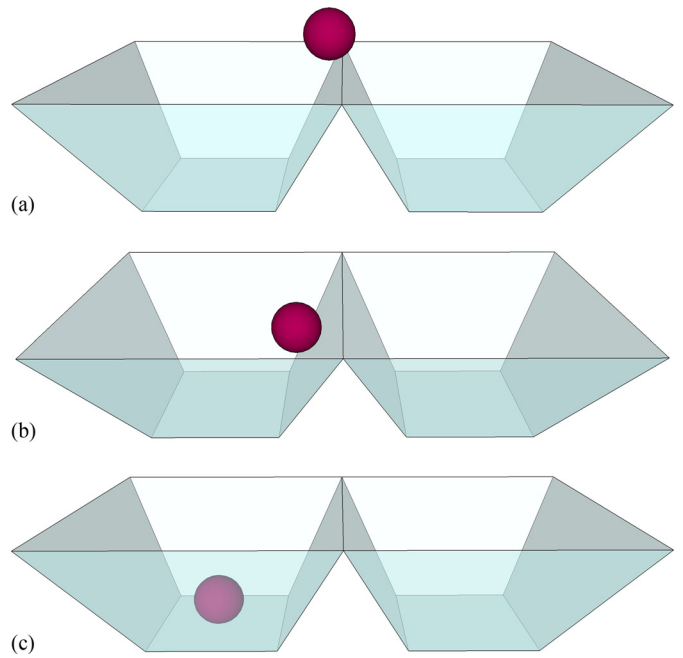
Fig. 7. Simulation results for the asymmetrical example: (a) step = 20,000; (b) step = 235,000; (c) step = 1,200,000
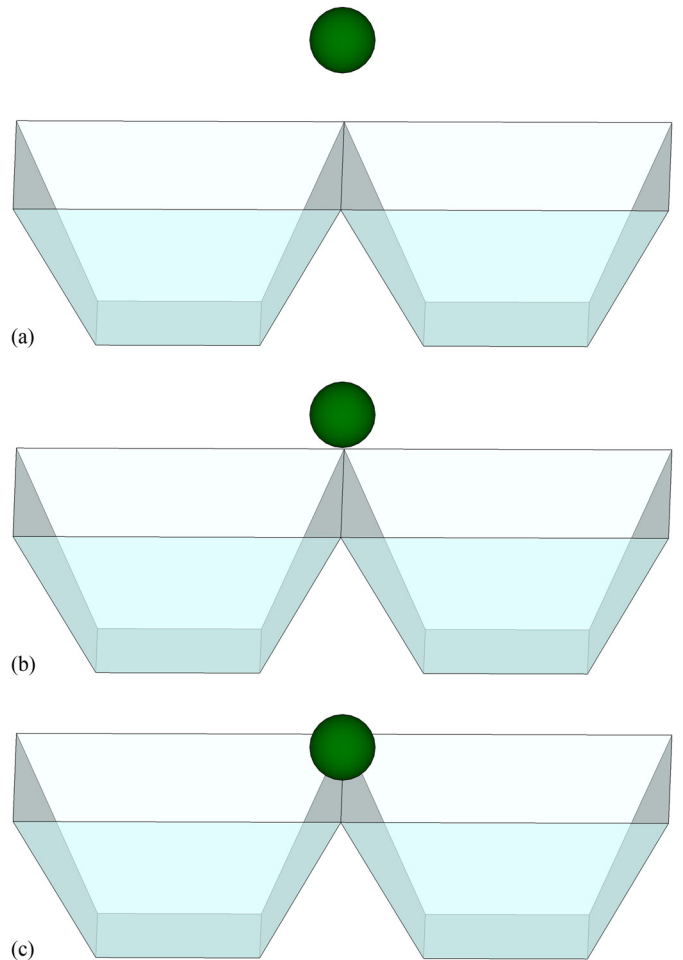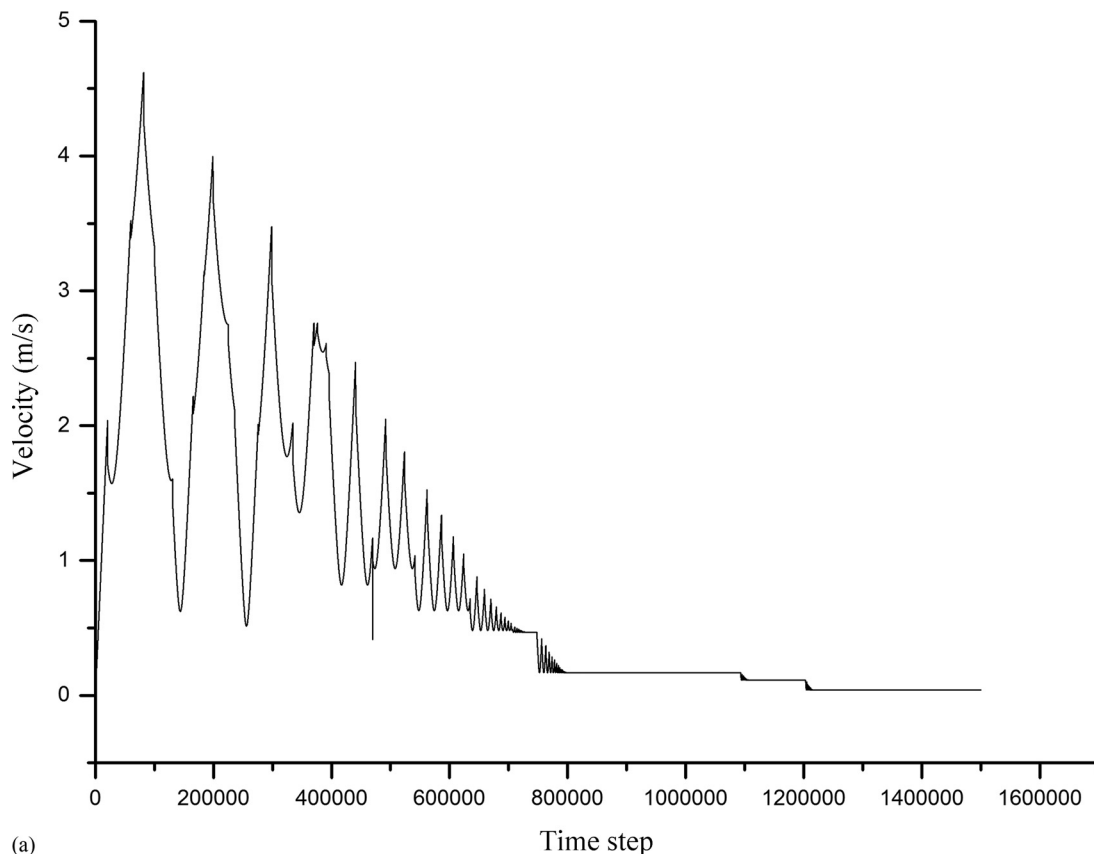

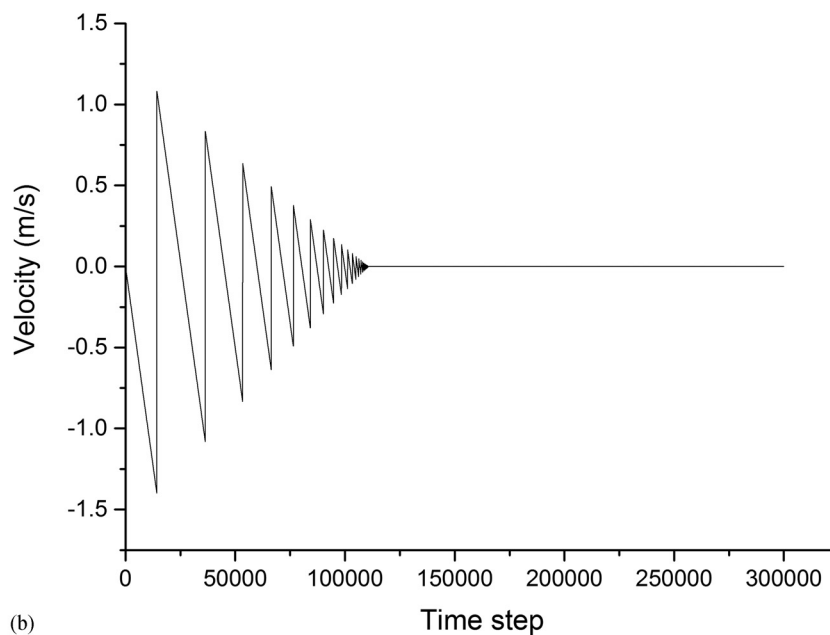**Fig. 8.** Simulation results for the symmetrical example: (a) step = 3,800; (b) step = 39,900; (c) step = 109,881

© ASCE

04017110-5

Int. J. Geomech.

Fig. 9. Velocity and time-step curve: (a) results of the asymmetrical example; (b) results of the symmetrical example

the vertex of the plane, point P_2 is in the fixed plane. Take line segment AB as an example. Assuming that vectors $\overrightarrow{V_1} = \overrightarrow{P_2A} \times \overrightarrow{AB}$ and $\overrightarrow{V_2} = \overrightarrow{CA} \times \overrightarrow{AB}$, if $\overrightarrow{V_1} \cdot \overrightarrow{V_2} > 0$, vectors $\overrightarrow{V_1}$ and $\overrightarrow{V_2}$ are in the same direction, and points C and P_2 are at the same side of AB. Otherwise, points C and P_2 are on different sides of AB.

After the detection mentioned above, corresponding submatrices will be obtained and added to the global equations, and the direction

vectors will be calculated to determine the position where the springs are added to.

Calculation Process

The flowchart of the proposed algorithm is shown in Fig. 5.
The main calculation process for one time step is as follows:

1. Determine sphere–plane contact.
2. Check whether the projection point of sphere center to plane is in the plane or not. If not, check whether the contact type is sphere–edge contact or not. If not, go to step (4).
3. Obtain the corresponding submatrices, calculating the direction vectors from the contact point to the center of the sphere, and establishing the simultaneous equations.
4. Solve the simultaneous equations.
5. Check whether the contact modes of all contact pairs remain the same or not; if not, go to step (4) after revising the simultaneous equations.
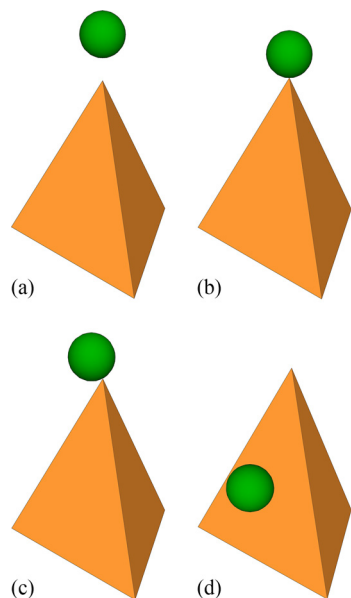6. Update the model data.

Fig. 10. Simulation results of sphere–tetrahedron contact at selected calculated steps: (a) step = 0; (b) step = 107,004; (c) step = 151,004; (d) step = 230,004

Verification Examples

The sphere–edge contact detection algorithm has been incorporated into the original SDDA source code developed by Professor Yu-Yong Jiao. This section presents four numerical examples calculated by the modified SDDA program to demonstrate the reasonableness and the accuracy of the proposed algorithm. Apart from the spheres, all the boundaries in these examples are fixed. The common mechanical properties used in these examples are listed in Table 1.

Example 1: Sphere–Edge Contact

This example intends to show the correctness of the proposed algorithm in simulating a simple physical problem. As shown in Fig. 6, the sphere above the convex is allowed to fall under gravity and to contact edge **AB** of the convex. The computing model in Fig. 6(a) is asymmetrical and the computing model in Fig. 6(b) is symmetrical. When sphere and convex contact each other, the new contact algorithm is expected to identify the contact type.

The simulated results of the asymmetrical example and the symmetrical one at different calculating steps are shown in Figs. 7 and 8, respectively, and the velocity and time-step curve of both examples is plotted in Fig. 9. As can be seen from the results, when the time step $t = 1,230,000$, the sphere system of the asymmetrical example can converge onto a stable state in the left convex, and when time step t = 109,881, the sphere of the symmetrical example can temporarily be stable on edge AB. These results indicate that the proposed algorithm is correct.

Example 2: Sphere–Three-Dimensional Corner Contact

To further demonstrate what capability the proposed algorithm has to deal with the complex sphere–edge contact problem, the sphere–three-dimensional corner contact is simulated in this section. As indicated in Fig. 10(a), the sphere is 0.2 m above the tetrahedron, which is made up of four equilateral triangles, and the sphere is allowed to fall under gravity. If the sphere contacts the tetrahedron, the sphere-edge contact type is determined by the new algorithm.

The simulated results for different calculating steps are shown in Fig. 10. The velocity and time-step curve and the displacement and
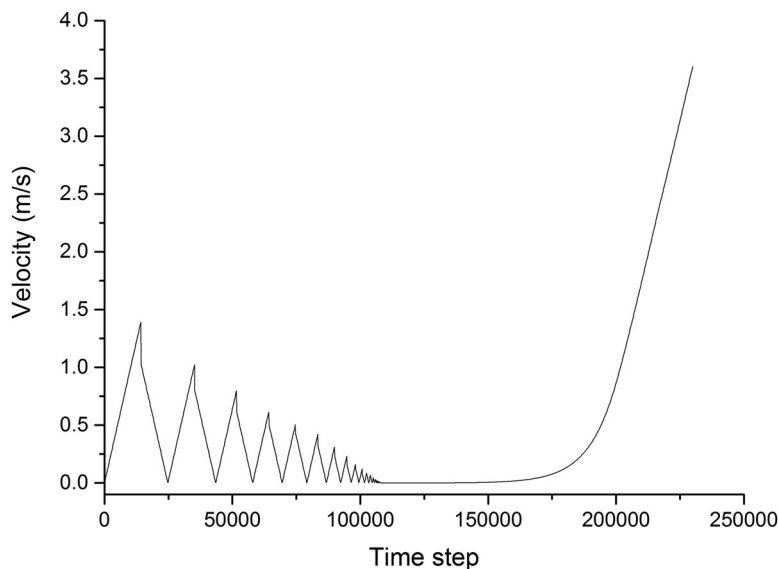

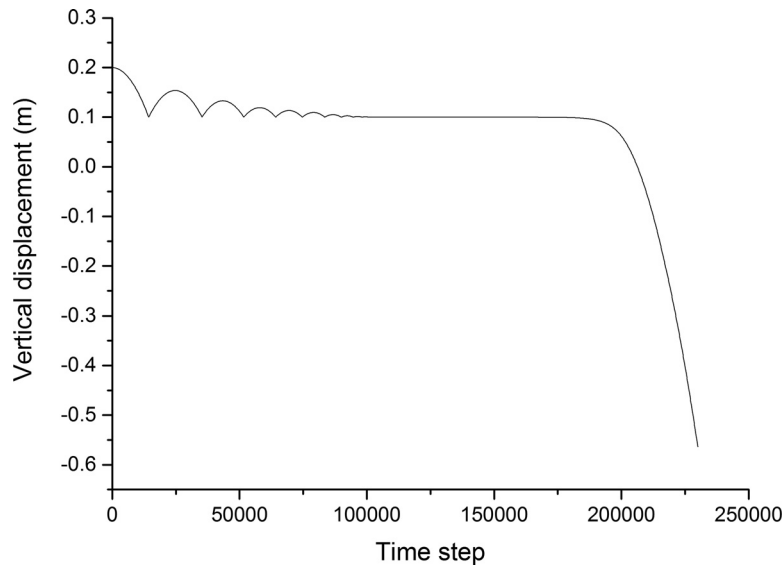**Fig. 11.** Velocity and time-step curve
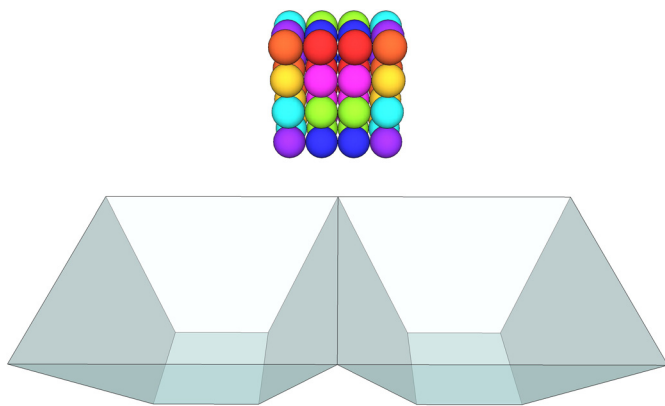
Fig. 12. Vertical displacement and time-step curve

Fig. 13. Computed model of example 3

Fig. 14. Simulation results: (a) step = 3,691; (b) step = 19,673

time-step curve are plotted in Figs. 11 and 12, respectively. As can be seen from the simulation results, the sphere system converges to a stable state when the time step $t = 107,004$. However, because of a rounding error in the computer, the sphere system begins to lose balance when $t = 151,004$, and the sphere begins to fall from the top of the tetrahedron.

Example 3: Spheres Falling into Containers

To further verify that the proposed algorithm can simulate contact between multiple spheres and edges, the falling processes of a number of rigid spheres under the action of gravity are simulated, as shown in Fig. 13. This spherical system, with no bonding strength, consists of 48 spheres.

Fig. 14 shows the simulation results of this example. As it can be seen from the results, the spheres contact the edge and fall into both the containers. This case further demonstrates that the proposed algorithm is physically viable and correct.

Example 4: Simulation of Rockfall

Rockfall is a serious natural hazard in mountainous areas and represents a major threat to infrastructure, transportation lines, and
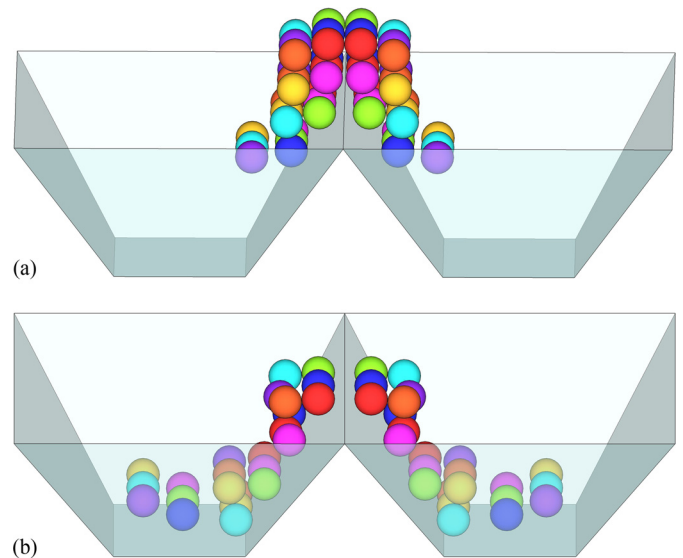
people. To verify whether the proposed algorithm has the capability to model this kind of problem, rockfall trajectories are simulated with account taken for trees.

As shown in Fig. 15, the computational model consists of ten spheres and is loaded by self-weight only. The slope is 10 m in height and 45° in slope, and its surface consists of convexes and concaves. Sphere elements simulated as clastic rocks are released from the top of the slope. It is assumed that the trees on the slope are also only acting as the boundaries of fixed planes, as shown in Fig. 16.

The simulated results of the rockfall trajectories are plotted in Fig. 17. It can be seen that the spheres fall under the gravity from the top of the slope, and some of them impact trees and change their trajectories. The falling path of the spheres is reasonable, and it confirms that the modified SDDA has the capability to simulate practical problem.
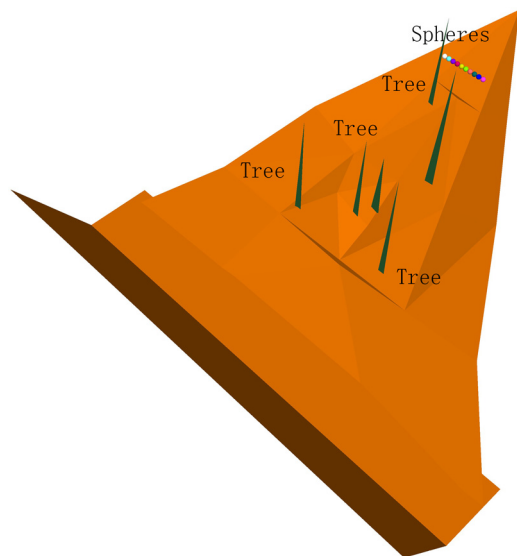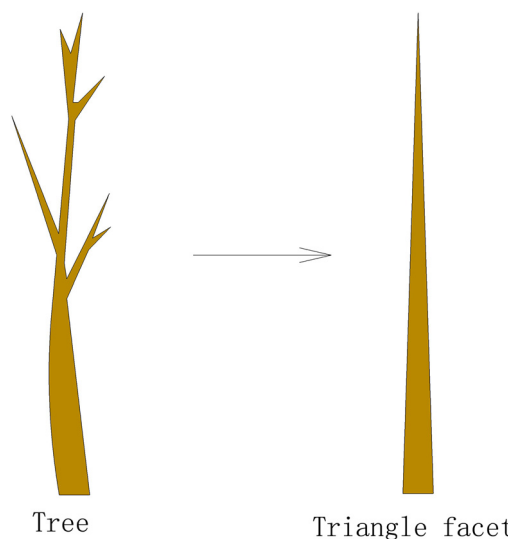
Fig. 15. Computed model of rockfall

Fig. 16. Boundary condition of tree

Concluding Remarks

To make up for the incompleteness of contact type in the original SDDA, a sphere–edge contact model is presented in this paper, and a corresponding contact detection algorithm has been incorporated into the original SDDA source code. The proposed sphere–boundary contact detection algorithm includes two main loops, i.e., a sphere–plane contact search and sphere–edge contact detection. Contact detection begins with determining spheres that have potential to be in contact with a fixed triangle or other polyhedral planes. Then, contact pairs are further examined to judge whether the contact type is sphere–plane contact or sphere–edge contact. According to the geometrical information obtained from the sphere–plane or sphere–edge contact pair, corresponding contact stiffness submatrices and contact force vectors are derived for calculation. To verify the proposed algorithm, four examples were calculated. The numerical results demonstrate that the proposed sphere–boundary contact model is effective in dealing with complex boundaries,
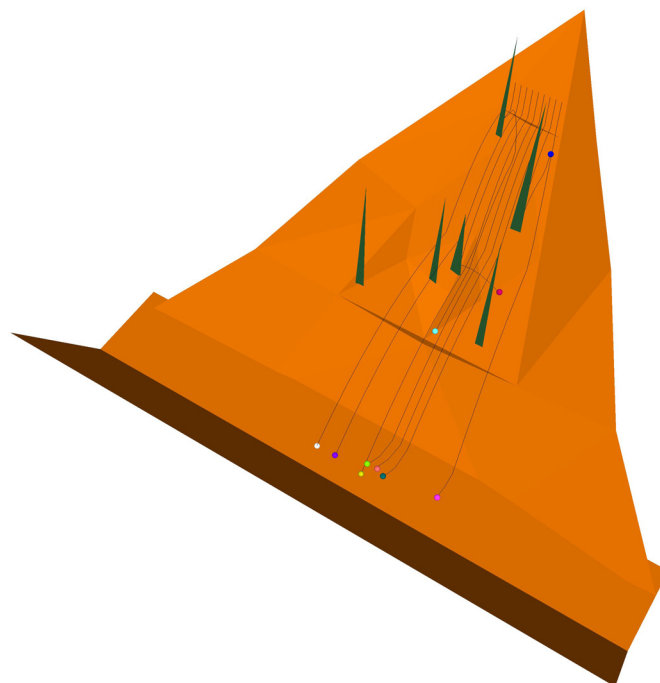

Fig. 17. Results of rockfall trajectories

including planes, edges and corners, and the modified SDDA has the capability to simulate practical contact problems.

Acknowledgments

References

Bao, H. R., Hatzor, Y. H., and Huang, X. (2012). "A new viscous boundary condition in the two-dimensional discontinuous deformation analysis method for wave propagation problems." *Rock Mech. Rock Eng.*, 45(5), 919–928.

Beyabanaki, S. A. R., and Bagtzoglou, A. C. (2012). "Three-dimensional discontinuous deformation analysis (3-D DDA) method for particulate media application." *Geomech. Geoeng.*, 7(4), 239–253.

Chen, G. Q., Zheng, L., Zhang, Y. B., and Wu, J. (2013). "Numerical simulation in rock analysis: A close comparison of 2-D and 3-D DDA." *Rock Mech. Rock Eng.*, 46(3), 527–541.

Chen, H. M., Zhao, Z. Y., Choo, L. Q., and Sun, J. P. (2016). "Rock cavern stability analysis under different hydro-geological conditions using the coupled hydro-mechanical model." *Rock Mech. Rock Eng.*, 49(2), 555–572.

Cheng, Y. M. (1998). "Advancements and improvement in discontinuous deformation analysis." *Comput. Geotech.*, 22(2), 153–163.

Fan, H., and He, S. M. (2015). "An angle-based method dealing with vertex-vertex contact in the two-dimensional discontinuous deformation analysis (DDA)." *Rock Mech. Rock Eng.*, 48(5), 2031–2043.

Jiang, Q. H., Chen, Y. F., Zhou, C. B., and Yeung, M. C. R. (2013). "Kinetic energy dissipation and convergence criterion of discontinuous deformation analysis (DDA) for geotechnical engineering." *Rock Mech. Rock Eng.*, 46(6), 1443–1460.

Jiang, Q. H., and Yeung, M. R. (2004). "A model of point-to-face contact for three-dimensional deformation analysis." *Rock Mech. Rock Eng*, 37(2), 95–116.

Jiao, Y. Y., Zhang, H. Q., Tang, H. M., Zhang, X. L., Adoko, A. C., and Tian, H. N. (2014). "Simulating the process of reservoir-impoundment-induced landslide using the extended DDA method." *Eng. Geol.*, 182, 37–48.

Jiao, Y. Y., Huang, G. H., Zhao, Z. Y., Zheng, F., and Wang, L. (2015). "An improved three-dimensional spherical DDA model for simulating rock failure." *Sci. China Tech. Sci.*, 58(9), 1533–1541.

Nie, W., Zhao, Z. Y., Ning, Y. J., and Sun, J. P. (2014). "Development of rock bolt elements in two-dimensional discontinuous deformation analysis." *Rock Mech. Rock Eng.*, 47(6), 2157–2170.

Shi, G. H. (1998). "Discontinuous deformation analysis: A new numerical model for the statics and dynamics of block systems." Ph.D. thesis, Univ. of California, Berkeley, CA.

Shi, G. H. (2015). "Contact theory." *Sci. China Tech. Sci.*, 58(9), 1450–1496.

Sitar, N., Maclaughlin, M. M., and Doolin, D. M. (2005). "Influence of kinematics on landslide mobility and failure mode." *J. Geotechnol. Geoenviron.*, 10.1061/(ASCE)1090-0241(2005)131:6(716), 716–728.

Wu, J. H., Ohnishi, Y., and Nishiyama, S. (2004). "Simulation of the mechanical behavior of inclined jointed rock masses during tunnel construction using discontinuous deformation analysis (DDA)." *Int. J. Rock Mech. Min. Sci.*, 41(5), 731–743.

Wu, J. H., Ohnishi, Y., and Nishiyama, S. (2005). "A development of the discontinuous deformation analysis for rock fall analysis." *Int. J. Numer. Anal. Meth. Geomech.*, 29(10), 971–988.

Wu, W., Zhu, H. H., Zhuang, X. Y., Ma, G. W., and Cai, Y. C. (2014). "A multi-shell cover algorithm for contact detection in the three

dimensional discontinuous deformation analysis." *Theor. Appl. Fract. Mech.*, 72, 136–149.

Yeung, M. R. and Leong, L. L. (1997). "Effects of joint attributes on tunnel stability." *Int. J. Rock Mech. Min. Sci.*, 34(3-4).

Yeung, M. R., Jiang, Q. H., and Sun, N. (2007). "A model of edge-to-edge contact for three-dimensional discontinuous deformation analysis." *Comput. Geotech.*, 34(3), 175–186.

Zhang, Y., Chen, G., Zheng, L., Li, Y., and Wu, J. (2013). "Effects of near-fault seismic loadings on run-out of large-scale landslide: A case study." *Eng. Geol.*, 166, 216–236.

Zhang, Y., Xu, Q., Chen, G., Zhao, J. X., and Zheng, L. (2014). "Extension of discontinuous deformation analysis and application in cohesive-frictional slope analysis." *Int. J. Rock Mech. Min. Sci.*, 70, 533–545.

Zhang, Y., Wang, J., Xu, Q., Chen, G., Zhao, J., Zhen, L., Han, Z., and Yu, P. (2015). "DDA validation of the mobility of earthquake-induced landslides." *Eng. Geol.*, 2015, 194, 38–51.

Zhao, S. L. (2000). "Development of three dimensional spherical discontinuous deformation analysis for granular materials." Ph.D. thesis, North Carolina State Univ., Raleigh, NC.

Zheng, F., Jiao, Y. Y., Zhang, X. L., Tan, F., Wang, L., and Zhao, Q. (2016). "Object-oriented contact detection approach for three-dimensional discontinuous deformation analysis based on entrance block theory." *Int. J. Geomech.*, 10.1061/(ASCE)GM.1943-5622.0000718, E4016009.