Geomechanics and Engineering, *Vol. 23, No. 3 (2020) 189-206*
DOI: https://doi.org/10.12989/gae.2020.23.3.189

Efficient flexible boundary algorithms for DEM simulations of biaxial and triaxial tests

Donghai Liu* and Jiaqi Yang[a]

State Key Laboratory of Hydraulic Engineering Simulation and Safety, Tianjin University, Tianjin 3003550, China

(Received September 18, 2019, Revised September 11, 2020, Accepted October 7, 2020)

Abstract. The accurate modeling of boundary conditions is important in simulations of the discrete element method (DEM) and can affect the numerical results significantly. In conventional triaxial compression (CTC) tests, the specimens are wrapped by flexible membranes allowing to deform freely. To accurately model the boundary conditions of CTC, new flexible boundary algorithms for 2D and 3D DEM simulations are proposed. The new algorithms are computationally efficient and easy to implement. Moreover, both horizontal and vertical component of confining pressure are considered in the 2D and 3D algorithms, which can ensure that the directions of confining pressure are always perpendicular to the specimen surfaces. Furthermore, the boundaries are continuous and closed in the new algorithms, which can prevent the escape of particles from the specimens. The effectiveness of the proposed algorithms is validated by biaxial and triaxial simulations of granular materials. The results show that the algorithms allow the boundaries to deform non-uniformly on the premise of maintaining high control accuracy of confining pressure. Meanwhile, the influences of different lateral boundary conditions on the numerical results are discussed. It is indicated that the flexible boundary is more appropriate for the models with large strain or significant localization than rigid boundary.

Keywords: discrete element method; conventional triaxial compression; flexible boundary; granular materials; localization

1. Introduction

The discrete element method (DEM) is a powerful tool for studying the mechanical properties of granular materials, as it characterizes the mechanical response of materials to particles that interact via mechanical contacts (Cundall and Strack 1979). Compared with conventional continuum modeling approaches such as the finite element method and the finite difference method, DEM can provide particle-scale information such as particle displacements, velocities, rotations and the particle interactions, enabling researchers to further interpret the mechanical behavior of granular materials (Cheung and O'Sullivan 2008, Kumara and Hayano 2016, Zhou *et al.* 2016, Dai *et al.* 2019).

Generally, to make the particle-scale and macroscale information achieved from DEM simulations more persuasive, it is necessary to relate the results of DEM simulations with the granular material responses of laboratory testing. Conventional triaxial compression (CTC), which can reveal the deformation and failure characteristics of materials under three-dimensional stress states, is one of the most important and common laboratory tests used to study the mechanical behaviors of granular materials, such soils and rockfills. The simulations of CTC are usually placed in a special position by the users of DEM, which is often used to verify the effectiveness of the models and the parameters (Cheung and O'Sullivan 2008, Lee *et al.* 2012, Ma *et al.* 2014, Rakhimzhanova *et al.* 2019).

In DEM simulations, it is important to accurately describe the boundary conditions, which can significantly affect the numerical results (Cheung and O'Sullivan 2008). In CTC, the lateral confining pressure is supplied by water or hydraulic oil allowing the lateral boundaries of specimens to deform freely, which can be seen as the stress-controlled boundary. In DEM simulations, one of the simplest ways to apply lateral confining pressure is to control the velocities of rigid walls by servo algorithm (Zhou *et al.* 2016, Huang *et al.* 2019, Rakhimzhanova *et al.* 2019). This algorithm is not only easy to implement but also computationally efficient. However, the rigid boundary will inhibit the natural development of localization, which can even lead to the distortion of numerical results. Therefore, the rigid boundary is not suitable for the DEM models with large strains and localization, such as the soil and rockfill models. Compared with the rigid boundary, the flexible boundary is more appropriate for this type of modeling, which allows the specimen to deform freely.

Many scholars have applied the flexible boundary in their research work. For example, based on the DEM model with flexible boundary and periodic boundary, O'Sullivan and Cui (2009) studied the response of granular materials under load-unload cycles. Based on a polyhedral particle model with flexible boundary, Lee *et al.* (2012) successfully reproduced the particle-scale and macroscale responses of

*Corresponding author, Professor
E-mail: liudh@tju.edu.cn
[a]Ph.D. Student
E-mail: demyang@163.com

ISSN: 2005-307X (Print), 2092-6219 (Online)

sand in CTC. More recently, Shi *et al.* (2018) used the flexible boundary to study the behavior of granular materials under torsional shear with fixed principal stress.

In this paper, new flexible boundary algorithms with high computational efficiency are proposed, which can be used in DEM simulations of CTC. The paper is organized as follows. Considering that there are many types of flexible boundary algorithms proposed by different scholars according to their own needs, the characteristics of these algorithms are discussed in Section 2. In view of the limitations of existing algorithms, new flexible boundary algorithms for 2D and 3D DEM models, as well as their detailed implementation procedures, are described in Section 3. Then, the validity and computational efficiency of the present algorithms are demonstrated in Section 4. Furthermore, the influences of different boundary conditions on the numerical results are investigated in Section 5 based on a series of DEM simulations of biaxial tests in 2D and triaxial tests in 3D.

2. Overview of flexible boundary algorithms used in DEM

Various flexible boundary algorithms have been proposed based on different considerations. Existing algorithms can be divided into 2 main categories according to the components of the boundary. One can be described as the particle boundary algorithm, while the other can be addressed as the wall boundary algorithm.

The particle boundary algorithms whose boundaries are composed of particles are widely accepted and used. The general processes of this type of algorithm, which was first proposed by Bardet and Proubet (1991) in 2D models, are as follows: (1) recognize or create the boundary particles which are used to apply the equivalent force; (2) calculate the magnitude and direction of the equivalent force of each boundary particle according to the desired confining pressure; and (3) update the boundary particles and their equivalent forces to achieve desired confining pressure during loading. A similar 3D flexible boundary algorithm was proposed by Kuhn (1995). Based on Kuhn's work, the identification of boundary particles and the calculation of equivalent force were improved by Cui *et al.* (2007), Cheung and O'Sullivan (2008) and O'Sullivan and Cui (2009). The boundary particles are identified by "radial contact check" and updated constantly. They project coordinates of boundary particles onto a rectangular plane which will be divided into Voronoi polygons. Then, the equivalent force of each boundary particle is calculated by taking the area of the corresponding Voronoi polygon as weight. These improvements not only make the algorithms suitable for cylindrical specimens but also improve the control accuracy of confining pressure. These algorithms are effective but a little complex. Similar algorithms have been proposed by Wang and Tonon (2009) and Binesh *et al.* (2018). They simplify the calculation of equivalent force which can improve the efficiency of the algorithms. Wang and Tonon (2009) and Binesh *et al.* (2018) calculate the equivalent force of each boundary particle by using the cross-sectional area of the particle as the weight. Moreover, instead of recognizing outermost particles as boundary, some algorithms create the boundaries composed of regularly packed bonded particles. The supporters of such algorithms believe that the boundary composed of bonded particles is more similar to the latex membrane used in laboratory testing. However, in order to prevent the escape of particles and accurately reflect the local deformation of the boundary, such algorithms often need to add many extra particles as boundary, which can significantly influence the computational efficiency. And it is inconvenient to determine the parameters of the bonded particles which make up the boundaries. The corresponding algorithms for the 2D model are given by Wang and Leung (2008), Evans and Frost (2010), Jiang *et al.* (2011) and Meng *et al.* (2017), while those for the 3D model are given by Bono *et al.* (2012), Cil and Alshibli (2013), Lu *et al.* (2018) and Qu *et al.* (2019).

The wall boundary algorithms are generally easier to implement than the particle boundary algorithms. The general processes of this type of algorithm are as follows: (1) replace the entire wall with stacks of wall segments; (2) calculate the servo velocity of each wall segment; and (3) update the velocity of each wall segment to achieve the desired confining pressure during loading. Using stacks of cylindrical wall segments that could deform independently of one another, Zhao and Evans (2009, 2011) simulated the CTC in 3D. Based on the work of Zhao and Evans (2009), Shi *et al.* (2018) developed a wall boundary algorithm that can apply internal and external confining pressures to the hollow cylindrical specimens. Unlike the wall boundary algorithms mentioned above, Lee *et al.* (2012) and Khoubani and Evans (2018) used the planar wall segments to build the cylindrical boundary in CTC. Each planar wall segment is independent but only allowed to move in the radial direction. The difficulty of these algorithms lies in calculating the effective area of each wall segment, which determines the servo force.

Intuitively, the boundary condition applied by particle boundary algorithm is more realistic than the wall boundary algorithm. However, the particle boundary algorithm has two limitations: (1) the implementations of these algorithms are complicated; and (2) the calculation loads are commonly large, even the calculation of equivalent force has been simplified. By contrast, the wall boundary algorithms are more efficient and easier to implement. Although the wall segments can cause the local constraint on the boundary particles, this local constraint will be reduced with increasing the number of wall segments.

Most existing algorithms have been shown to be effective, but they also have some limitations. First, most flexible boundary algorithms for 3D models do not consider the vertical component of confining pressure, which is inconsistent with reality when there is a large deformation in the model boundary (Kuhn 1995, Cui *et al.* 2007, Cheung and O'Sullivan 2008, O'Sullivan and Cui 2009, Wang and Tonon 2009, Zhao and Evans 2009, Lee *et al.* 2012, Zhao and Evans 2011, Cil and Alshibli 2014, Binesh *et al.* 2018, Khoubani and Evans 2018, Lu *et al.* 2018, Shi *et al.* 2018). Second, the boundaries of many algorithms are not continuous and closed (Bardet and Proubet 1991, Kuhn 1995, Cui *et al.* 2007, Cheung and O'Sullivan 2008, Wang and Leung 2008, O'Sullivan and Cui 2009, Wang and

Tonon 2009, Zhao and Evans 2009, Evans and Frost 2010, Jiang *et al.* 2011, Zhao and Evans 2011, de Bono *et al.* 2012, Lee *et al.* 2012, Cil and Alshibli 2014, Meng *et al.* 2017, Binesh *et al.* 2018, Khoubani and Evans 2018, Lu *et al.* 2018, Shi *et al.* 2018, Qu *et al.* 2019). Some measures are needed to prevent the escape of particles, such as updating the boundary frequently or creating compactly arranged boundary particles, which will reduce the computational efficiency significantly. Finally, some algorithms are effective and accurate yet complex and difficult to implement (Kuhn 1995, Cui *et al.* 2007, Cheung and O'Sullivan 2008, O'Sullivan and Cui 2009, Lee *et al.* 2012, Khoubani and Evans 2018, Lu *et al.* 2018, Qu *et al.* 2019).

3. Flexible boundary algorithm

Based on the problems of existing flexible boundary algorithms, new flexible boundary algorithms for 2D and 3D DEM simulations are proposed in this paper. Compared with the previous algorithms, the new algorithms in this paper have the following characteristics. First, both the horizontal and the vertical components of confining pressure are considered. The confining pressure is always perpendicular to the specimen surface, which is consistent with the boundary conditions of laboratory tests. Second, the desired confining pressure is achieved by controlling the velocity of wall nodes distributed along the boundary. In other words, the boundary composed of wall segments can deform with the movement of wall nodes. Therefore, the boundary is always continuous and closed which can prevent the escape of particles. Finally, the algorithms are easy to implement and computationally efficient.

In this section, the principle and implementation of the 2D and 3D flexible boundary algorithms are provided in detail. To ensure the integrity of the content, we first offer a brief description of the servo algorithm used in DEM simulations, although the Itasca Consulting Group, Inc. (2004) has already elaborated this in detail. Subsequently, the 2D and 3D algorithms are respectively illustrated in detail to facilitate their implementation by the readers.

3.1 The servo algorithm

In DEM, the walls are defined as objects without mass. Thus, the force cannot be directly exerted to the walls whose motions are not governed by Newton's second law. Fortunately, the servo algorithm can be used to achieve the stress-controlled boundary by adjusting wall velocity continuously. The basic principle of servo algorithms can be explained by a concise sentence: when the current wall stress (σ_{cur}) is less than the target stress (σ_{tar}), the wall will be moved to increase the overlap between wall and model; otherwise, the wall will be moved in the opposite direction.

According to the principle of the servo algorithm, v_w can be assumed to be proportional to the difference between σ_{cur} and σ_{tar}:

$$v_w = G(\sigma_{cur} - \sigma_{tar}) \tag{1}$$

where G is called the gain parameter. The G can be calculated as follows.

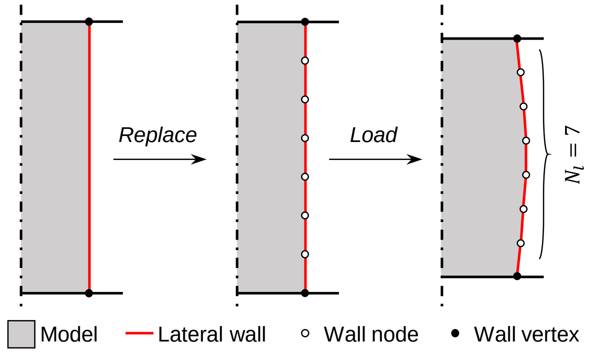


Fig. 1 Lateral walls of 2D models are replaced with articulated wall segments

The displacement increment of the wall ($\Delta\delta$) in one timestep (Δt) is calculated by:

$$\Delta\delta = v_w \Delta t \tag{2}$$

Then the increment of wall stress ($\Delta\sigma$) in Δt can be calculated by:

$$\Delta\sigma = \frac{\sum_{i=1}^{N} k_n^i \Delta\delta}{A} \tag{3}$$

where k_n^i is the normal stiffness between particle i and the wall, N is the number of contacts on the wall and A is the length or area of the wall. For stability, the absolute value of $\Delta\sigma$ should be less than the absolute value of the difference between σ_{cur} and σ_{tar}, which can be expressed as:

$$|\Delta\sigma| = \alpha |\sigma_{cur} - \sigma_{tar}| \tag{4}$$

where $\alpha \in (0,1)$ is the relaxation factor. Substituting Eqs. (1)-(3) into Eq. (4), the expression of G is obtained:

$$G = \frac{\alpha A}{\sum_{i=1}^{N} k_n^i \Delta t} \tag{5}$$

Finally, the v_w can be obtained by substituting G into Eq. (1). The desired confining pressure can be obtained by continuously updating v_w.

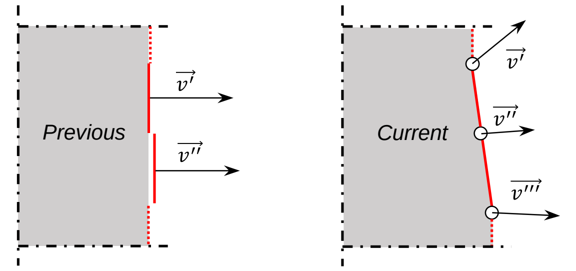
3.2 Implementation of the 2D algorithm

3.2.1 Step1 Creation of wall segments

The first step of all of the flexible boundary algorithms is to create a boundary that is able to deform. In the 2D model, the lateral wall with 2 vertices is replaced by articulated wall segments with a series of nodes which enable the boundary to deform locally (Fig. 1). As a consequence, there is a new input parameter for 2D models: the layer number of wall segments (N_l) (Fig. 1). In addition, the wall nodes located at the intersections of axial walls and lateral walls are called wall vertices to distinguish them from ordinary wall nodes. This is because there will be special treatments for wall vertices during the calculation.

3.2.2 Step2 Calculation of servo velocity

The calculation principles of servo velocity of wall



(a) Previous algorithms (b) Current algorithms

Fig. 2 The objects applied servo velocities

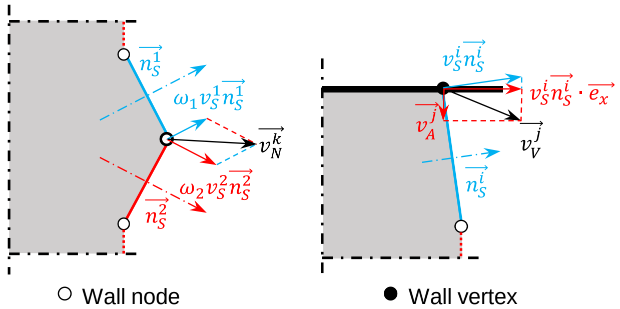


Fig. 3 Servo velocity calculations of wall nodes and wall vertices

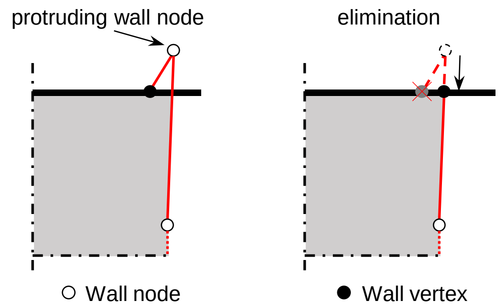


Fig. 4 The elimination of the protruding wall node

nodes and vertices are explained in this step. To ensure that the boundary consisting of wall segments is continuous and closed, the desired confining pressure is maintained by adjusting the velocities of the wall nodes in this paper instead of that of the wall segments (Fig. 2).

The servo velocity of wall nodes can be calculated by the following two steps: (1) calculate the servo velocity of each wall segment according to the servo algorithm in Section 3.1; and (2) calculate the servo velocity of each wall node based on the servo velocities and the unit normal vectors of wall segments.

According to Section 3.1, the servo velocity of the wall segment i (v_s^i) can be calculated by:

$$v_S^i = \frac{\alpha l^i}{\sum_{j=1}^{N} k_n^j \Delta t}\left(\sigma_{cur} - \sigma_{tar}\right) \tag{6}$$

where l^i is the length of wall segment i, and the other symbols are as defined in Section 3.1.

During the loading, there will be some wall segments that are not in contact with any particle. The servo velocities of these wall segments cannot be calculated according to Eq. (6) because the denominator is 0. In order to ensure that the impact on the specimens will not occur when the contactless wall segment touch the specimens again, the authors assume that the wall segment have a large virtual stiffness to reduce the servo velocity of the wall segment. The virtual stiffness (K_{vir}) is set to be:

$$K_{vir} = \left(l^i / d_{50}\right) k_n \tag{7}$$

where k_n is the average normal stiffness between wall segments and particles, and d_{50} is the average diameter of the particles. Then, the approximate servo velocities of these wall segments can be calculated by:

$$v_S^i = \frac{\alpha l^i}{\left(l^i / d_{50}\right) k_n \Delta t}\left(0 - \sigma_{tar}\right) = -\frac{\alpha d_{50}}{k_n \Delta t} \cdot \sigma_{tar} \tag{8}$$

As shown in Fig. 3, each wall node is associated with 2 wall segments in the 2D model. The servo velocity of wall node k ($\overrightarrow{v_N^k}$) can be calculated by:

$$\overrightarrow{v_N^k} = \sum_{i=1}^{2} \omega_i v_S^i \overrightarrow{n_S^i} \tag{9}$$

where ω_i is the weight which is taken as 1/2 for the 2D model in this paper, v_s^i is the servo velocity of the wall segment i related to the wall node k which can be calculated by Eqs. (6)-(8), and $\overrightarrow{n_S^i}$ is the unit normal vector of wall segment i which can be calculated by:

$$\overrightarrow{n_S^i} = \frac{\overrightarrow{N_S^i} \cdot \overrightarrow{r^i}}{\left|\overrightarrow{N_S^i} \cdot \overrightarrow{r^i}\right|} \cdot \frac{\overrightarrow{N_S^i}}{\left|\overrightarrow{N_S^i}\right|} \tag{10}$$

where $\overrightarrow{N_S^i}$ is the normal vector of wall segment i and $\overrightarrow{r^i}$ is the horizontal vector from the axial center line of the specimen to the center of the wall segment i. Suppose that $\overrightarrow{P_1^i} = (x_1^i, y_1^i)$ and $\overrightarrow{P_2^i} = (x_2^i, y_2^i)$ are the coordinates of the end nodes of wall segment i respectively. Then the $\overrightarrow{N_S^i}$ and $\overrightarrow{r^i}$ can be calculated by:

$$\overrightarrow{N_S^i} = \begin{vmatrix} \vec{i} & \vec{j} \\ x_2^i\text{-}x_1^i & y_2^i\text{-}y_1^i \end{vmatrix} \tag{11}$$

$$\overrightarrow{r^i} = \frac{\overrightarrow{P_1^i} + \overrightarrow{P_2^i}}{2} \cdot \overrightarrow{e_x} \tag{12}$$

where $\overrightarrow{e_x}$ is the unit vector in the x-direction.

Each wall vertex is only associated with 1 wall segment in the 2D model (Fig. 3). The servo velocity of the jth wall vertex ($\overrightarrow{v_V^j}$) can be calculated according to the Eq. (13):

$$\overrightarrow{v_V^j} = \overrightarrow{v_A^j} + v_S^i \overrightarrow{n_S^i} \cdot \overrightarrow{e_x} \tag{13}$$

where $\overrightarrow{v_A^j}$ is the vertical velocity of wall vertex j, which is equal to the axial velocity of the intersecting axial wall, v_s^i and $\overrightarrow{n_S^i}$ are the servo velocity and unit normal vector of

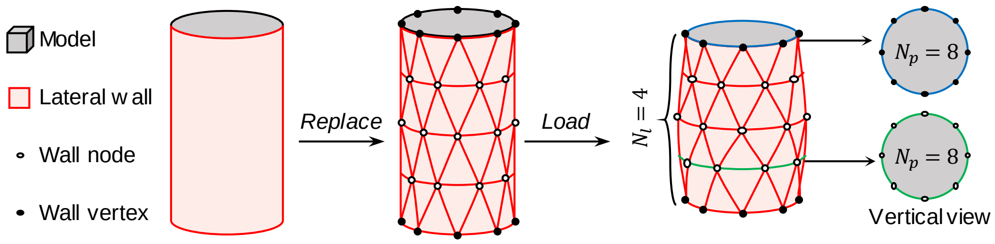
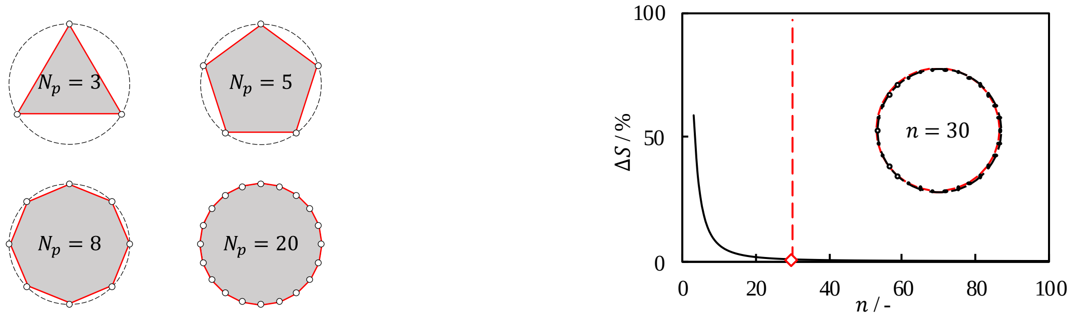


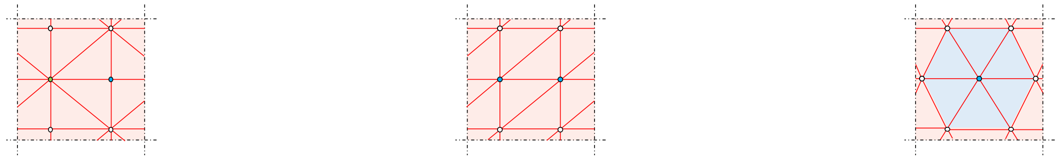
Fig. 5 Lateral walls of 3D models are replaced with articulated wall segment



(a) The area change for the wall replacement

(b) The relationship between ΔS and N_p

Fig. 6 The change of cross-section before and after the wall replacement



(a) Mode I

(b) Mode II

(c) Mode III

Fig. 7 The typical arrangements of wall segments

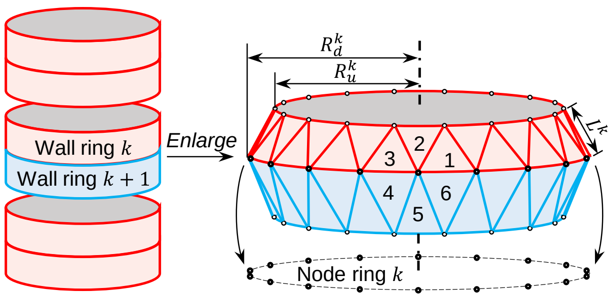


Fig. 8 Wall rings and node rings

the wall segment i related to the wall vertex j.

During the loading, some wall nodes may protrude slightly (Fig. 4). To eliminate this phenomenon, the original wall vertex will be deleted, while the protruding wall node will be set as a new wall vertex and its vertical position will be equal to that of the original wall vertex (Fig. 4).

3.2.3 Step3 Iteration

The desired confining pressure will be obtained by updating the servo velocity of each wall node and vertex in each timestep.

3.3 Implementation of the 3D algorithm

3.3.1 Step1 Creation of wall segments

The first step of the 3D algorithm, which is similar to the 2D algorithm, is to replace the rigid wall as articulated wall segments. As shown in Fig. 5, the cylindrical lateral boundary is replaced by a polyhedron consisting of wall segments. As a result, there are two input parameters about boundaries for the 3D models: the layer number of wall segments (N_l) and the node number in each layer (N_p). Taking Fig. 5 as an example, when N_l and N_p are set as 4 and 8, respectively, the lateral boundary will be divided into 4 layers along the axial direction, and the number of wall nodes (or wall vertices) in each layer is equal to 8.

As shown in Fig. 6(a), the cross-sectional area of the lateral boundary will decrease when the whole wall is replaced by the articulated wall segments. If the reduction of cross-sectional area is too much, the model will be unstable due to the sudden surge of boundary forces. The percentage change in cross-sectional area (ΔS) can be calculated as:

$$\Delta S = (1 - \frac{S_P}{S_C}) \times 100\% = [1 - \frac{N_p}{2\pi} \cdot \sin\left(\frac{2\pi}{N_p}\right)] \times 100\% \qquad (14)$$

where S_P and S_C are the area of the regular polygon and its circumscribed circle respectively.

According to Eq. (14), ΔS gradually approaches 0 with the increase of N_p (Fig. 6(b)). Therefore, N_p must be large enough to guarantee the stability of the model after wall replacement. In the 3D algorithm of this paper, N_p is recommend to be set as 30 to guarantee that ΔS is small enough (ΔS<0.75%). In this way, not only the stability of the model after wall replacement can be ensured, but also the increase of computation load can be accepted.

The wall nodes in this algorithm are moving continuously during the calculation. Therefore, the shape of wall segments must be triangle to ensure the coplanarity of wall nodes on the same wall segment. The rectangular arrangement used by Lee *et. al.* (2012) and Khoubani and Evans (2018) is no longer suitable for the algorithm in this paper. The typical arrangements of wall segments are shown in Fig. 7. In this paper, wall nodes are set to be hexagonally arranged (Mode III), such that each node will be associated with 6 wall segments which can simplify the complexity of the algorithm during the traversal process.

3.3.2 Step 2 Calculation of applied velocity

Similar to the 2D algorithm, the servo velocities of wall segments must be first calculated in the 3D algorithm before calculating the servo velocities of wall nodes or vertices.

When the number of particles is too small (e.g., less than 3000) in 3D model, there will be a lot of non-contact wall segments, which will cause the instability of the 3D algorithm. To conquer this problem and increase the applicability of the 3D algorithm, the wall segments on the same layer are treated as an integrity called "wall ring" which is the minimum unit used to calculate the servo velocity (Fig. 8). Although this treatment may restrict the motion of the wall nodes to some extent, it significantly improves the applicability and computational efficiency of the 3D algorithm. To verify the effectiveness and feasibility of this treatment, the algorithm taking wall ring as the minimum unit to calculate servo velocity is compared with that taking wall segment as the minimum unit, as shown in Appendix. For the convenience of expression, the former is called as wall ring algorithm (WRA), while latter is called as wall segment algorithm (WSA). The results show that the numerical results obtained from WRA and WSA have no significant difference. Considering the better performance of WRA in efficiency, stability and applicability, it is presented and studied explicitly in the following section. The WSA is introduced and discussed in the Appendix.

For the wall ring algorithm which take the wall ring as the minimum unit for calculating the servo velocity, the servo velocity of wall segment i (v_S^i) is equal to that of wall ring k (v_R^k) when wall segment i belong to the wall ring k:

$$v_S^i = v_R^k \tag{15}$$

Then, v_R^k can be calculated as:

$$v_R^k = \frac{\alpha A^k}{\sum_{j=1}^{N} k_n^j \Delta t}\left(\sigma_{cur} - \sigma_{tar}\right) \tag{16}$$

where A^k is the surface area of wall ring k which can be calculated by:

$$A^k = \pi L^k \left(R_u^k + R_d^k\right) \tag{17}$$

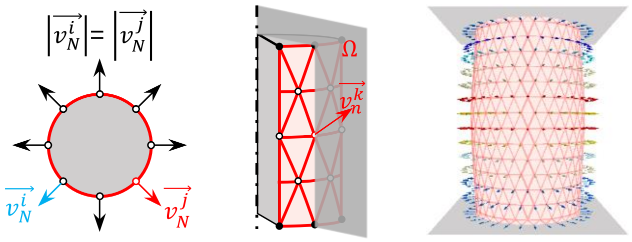


Fig. 9 The characteristics of the wall nodes' servo velocities in 3D models

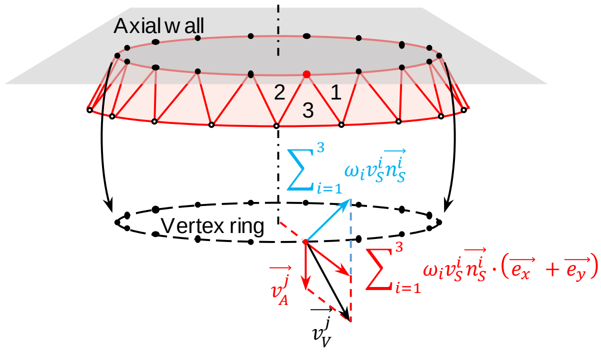


Fig. 10 Servo velocity calculations of wall vertices

where L^k is the generatrix length of the conical frustum, and R_u^k and R_d^k are the radii of the top and bottom surfaces (Fig. 8).

In addition, for the same reason as 2D algorithm, the virtual stiffness (K_{vir}) of the wall ring with no contact is set to be:

$$K_{vir} = \left(A^k / d_{50}^2\right) k_n \tag{18}$$

Then, the servo velocities of such wall ring can be calculated according to Eq. (19):

$$v_R^k = \frac{\alpha A^k}{\left(A^k / d_{50}^2\right) k_n \Delta t}\left(0 - \sigma_{tar}\right) = -\frac{\alpha d_{50}^2}{k_n \Delta t}\sigma_{tar} \tag{19}$$

As shown in Fig. 8, each wall node is associated with 6 wall segments. The servo velocity of wall node k can be calculated by:

$$\overrightarrow{v_N^k} = \sum_{i=1}^{6} \omega_i v_S^i \overrightarrow{n_S^i} \tag{20}$$

where ω_i is the weight which is taken as 1/6 for the 3D model in this paper, v_S^i is the servo velocity of the wall segment i related to the wall node k which can be calculated by Eq. (16) and Eq. (19), and $\overrightarrow{n_S^i}$ is the unit normal vector of wall segment i which can be calculated by Eq. (10). In the 3D algorithm, it is assumed that $\overrightarrow{P_1^i} = (x_1^i, y_1^i)$, $\overrightarrow{P_2^i} = (x_2^i, y_2^i)$ and $\overrightarrow{P_3^i} = (x_3^i, y_3^i)$ are the coordinates of the end nodes respectively which are arranged anticlockwise. Then the $\overrightarrow{N_S^i}$ and $\overrightarrow{r^i}$ in 3D models can be calculated by:

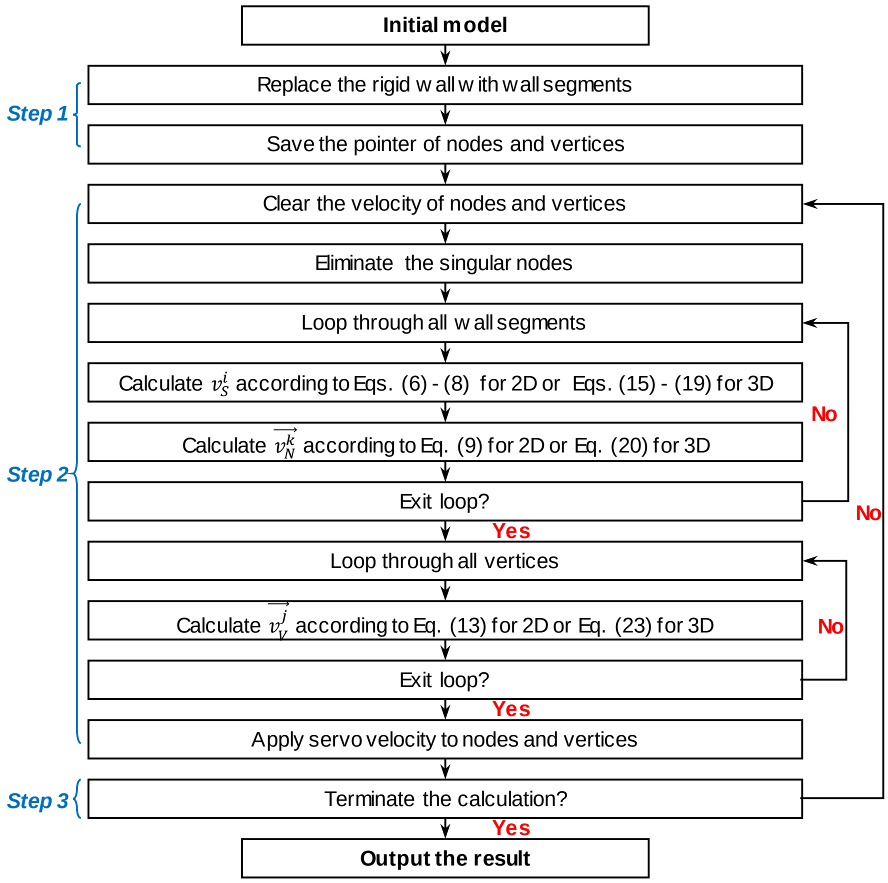


Fig. 11 Flow chart of 2D and 3D flexible boundary algorithms

$$\overrightarrow{N_S^i} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ x_2^i\text{-}x_1^i & y_2^i\text{-}y_1^i & z_2^i\text{-}z_1^i \\ x_3^i\text{-}x_2^i & y_3^i\text{-}y_2^i & z_3^i\text{-}z_2^i \end{vmatrix} \tag{21}$$

$$\overrightarrow{r^i} = \frac{\overrightarrow{P_1^i} + \overrightarrow{P_2^i} + \overrightarrow{P_3^i}}{3} \cdot \left(\overrightarrow{e_x} + \overrightarrow{e_y}\right) \tag{22}$$

where $\overrightarrow{e_x}$ and $\overrightarrow{e_y}$ are the unit vectors in x- and y-directions.

The servo velocities of the wall nodes calculated by Eqs. (15)-(22) have 3 characteristics: (1) the magnitudes of the wall nodes' servo velocities on the same node ring are equal (Fig. 9(a)); (2) the directions of the wall nodes' servo velocities are parallel to the plane formed by the center axis and the node itself (Fig. 9(b)); and (3) the servo velocities of wall nodes on each node ring are symmetric (Fig. 9(c)).

The servo velocity of wall vertex j ($\overrightarrow{v_V^j}$) on the top or bottom node ring can be calculated by Eq. (23) in the 3D model (Fig. 10):

$$\overrightarrow{v_V^j} = \overrightarrow{v_A^j} + \sum_{i=1}^{3} \omega_i v_S^i \overrightarrow{n_S^i} \cdot \left(\overrightarrow{e_x} + \overrightarrow{e_y}\right) \tag{23}$$

where ω_i is the weight which is taken as 1/3 for the 3D model in this paper, $\overrightarrow{v_A^j}$ is the vertical velocity of wall vertex j which is equal to the axial velocity of the intersecting axial wall, v_S^i and $\overrightarrow{n_S^i}$ are the servo velocity and the unit vector of wall segment i related to wall vertex j. It should be noted that the weights in Eq. (9), Eq. (20) and Eq. (23) are taken as constant according to the initial area proportion of the wall segments, e.g., the weight in Eq. (20) is set as 1/6 because the area ratio of single wall segment to all the wall segments around the wall node k is 1/6 at initial states. In fact, the weights in Eq. (9), Eq. (20) and Eq. (23) can be calculated iteratively by:

$$\omega = \frac{A^i}{\sum A^i} \tag{24}$$

where A^i is the area of wall segment i and $\sum A^i$ is the total area of the wall segments around the wall nodes k. It is more reasonable to calculate the weights according to Eq. (24). However, this treatment will significantly affect the computational efficiency of the algorithms. Considering the efficiency of the algorithms, the weights in Eq. (9), Eq. (20) and Eq. (23) are set as constant. The feasibility and effectiveness of the 2D and 3D algorithms with constant weights are verified in Section 4

When protruding wall nodes appear in the 3D model, they can be eliminated by referring to the method for the 2D model mentioned above.

3.3.3 Step3 Iteration

The desired confining pressure will be obtained by updating the servo velocity of each wall node and vertex in each timestep.

3.4 Algorithm flow

The flow chart of the 2D and 3D algorithms mentioned above is shown in Fig. 11.

4. Validation

In this section, several 2D and 3D DEM simulations are carried out to validate the 2D and 3D flexible boundary algorithms proposed in this paper. The purpose of this paper is to propose the flexible boundary algorithms for the biaxial tests and triaxial tests which are efficient and easy to implement. The algorithms are not limited to the study of a specific material. Therefore, the unbonded-particle model of Khoubani and Evans (2018) is taken as an example to validate the flexible boundary algorithms proposed in this paper. Based on the numerical results of biaxial and triaxial tests, three aspects of the algorithms are discussed (i.e., the control accuracy of confining pressure, the deformability of boundary and the computational efficiency).

It should be noted that the 2D and 3D algorithms are implemented in Particle flow code 2D and 3D (Version 5.0), respectively. And all numerical simulations in this paper were performed on a personal computer with an Intel (R) Core (TM) i5-4590 CPU @ 3.30GHz and 8.00GB of RAM.

4.1 The DEM model

Instead of the bonded-particle model, the unbonded-particle model, which has more significant local deformation during loading, was selected to validate the effectiveness and efficiency of the 2D and 3D algorithms in this paper. The model establishment and parameter selection of DEM models refer to the paper of Khoubani and Evans (2018). Only the main settings and parameters of the DEM models are described in the following.

The DEM model, which is a rectangle in 2D and a cylinder in 3D, consists of unbonded particles assumed to be disks of unit thickness in 2D and spheres in 3D. These particles are generated with radii uniformly distributed between the minimum radius ($r_{\min}$) and maximum radius ($r_{\max}$). The rolling resistance linear model is applied to the DEM models to reflect the effect of rolling resistance at contacts between particles (Ai *et al.* 2011). The parameters of the DEM models are presented in Table 1.

It should be noted that two measures are taken to reduce the computational time of the 3D model. First, the sizes of particles in the 3D models are enlarged 1000 times, which can significantly enlarge the timestep of the model. This is a common method to decrease the simulation time, which is called the mass scaling method (Jacobson *et al.* 2007, Belheine *et al.* 2009, Evans and Valdes 2011, Khoubani and Evans 2018). The readers interested in scale effect can refer to the work of Feng *et al.* (2009) and Feng and Owen (2011). Second, H/d_{50} in the 3D model is set to be 25, which is considered appropriate by Khoubani and Evans (2018).

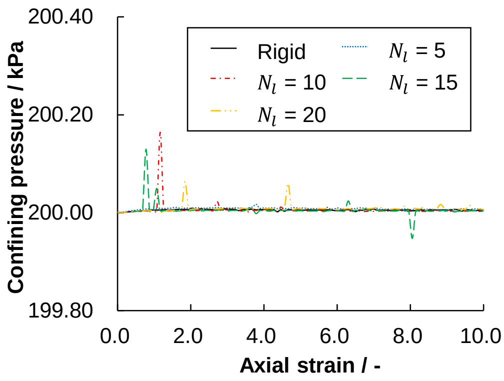
Table 1 Parameters of the 2D and 3D DEM models

Parameters	Value in 2D	Value in 3D
Geometry parameters		
Specimen height (H)	100 mm	100 mm × 1000
Specimen diameter (D)	50 mm	50 mm × 1000
Porosity under 200 kPa confining pressure (n)	0.159	0.362
Specimen height / mean particle radius ($H/r_{mean}\overrightarrow{n_S^i}$)	80	25
Max particle radius / min particle radius ($r_{\max}/r_{\min}$)	2.0	2.0
Number of particles (N_p)	13180	27588
Mechanical parameters		
Particle density (ρ)	2650 kg/m^3	2650 kg/m^3
Particle-particle normal stiffness (k_n)	1.2×10^8 N/m	1.2×10^8 N/m
Particle-particle shear stiffness (k_s)	1.0×10^8 N/m	1.0×10^8 N/m
Particle-wall normal stiffness (k_n^w)	1.2×10^7 N/m	1.2×10^7 N/m
Particle-wall normal stiffness (k_n^s)	1.0×10^7 N/m	1.0×10^7 N/m
Friction coefficient of particle-particle ($\mu^{p\text{-}p}$)	0.2	0.2
Friction coefficient of particle-wall ($\mu^{p\text{-}w}$)	0.2	0.2
Rolling resistance coefficient of particle-particle ($\mu^{p\text{-}p}$))	0.31	0.31
Local damping coefficient (d_p)	0.70	0.70
Load parameters		
Strain rate of axial loading (ε_a)	10^{-3} s^{-1}	10^{-3} s^{-1}
Lateral confining pressure (σ_3)	200 kPa	200 kPa
Acceleration of gravity (g)	0.0	0.0

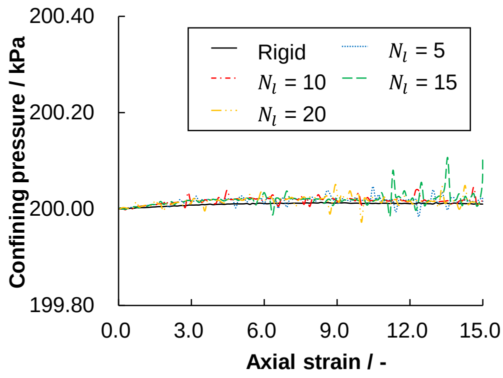
4.2 Effectiveness of 2D and 3D algorithms

The control accuracy of confining pressure and the deformability of boundary are the two most important criteria for evaluating the effectiveness of the flexible boundary algorithms. The CTC simulation is one of the most common application scenarios of flexible boundary algorithms. Therefore, the effectiveness of flexible boundary algorithms in this paper is verified and discussed by biaxial tests in 2D and triaxial tests in 3D. Because of the lack of available results about the other flexible boundary algorithms, the algorithms in this paper have not been demonstrated with other studies. To verify the effectiveness of the algorithms, the proposed algorithms are compared with the rigid boundary algorithm which is a common treatment of previous studies (Cheung and O'Sullivan 2008, Wang and Tonon 2009, Cil and Alshibli 2014, Khoubani and Evans 2018, Lu *et al.* 2018, Binesh *et al.* 2018, Qu *et al.* 2019). Consequently, the models with flexible boundaries are selected as the experimental group in which N_l are taken as 5, 10, 15 and 20 respectively, while those with rigid boundaries are selected as the control group.

In the numerical experiments, the load parameters of models are shown in Table 1. As shown in Fig. 12, both the rigid and flexible boundaries achieve high control accuracy



(a) 2D models



(b) 3D models

Fig. 12 Variation of average confining pressure under different boundary conditions

Table 2 The means and standard deviations of the confining pressures

Boundary condition	Means and standard deviation			
	2D		3D	
	0%	10%	0%	15%
Rigid boundary	200.14+17.742	200.01+34.355	200.00+15.153	200.63+51.614
N_l=5	199.97 ± 0.016	200.01 ± 0.005	200.00 ± 0.000	200.08 ± 0.295
N_l=10	199.99 ± 0.019	200.02 ± 0.017	200.00 ± 0.001	200.10 ± 0.170
N_l=15	199.98 ± 0.034	200.01 ± 0.037	200.00 ± 0.000	200.09 ± 0.237*
N_l=20	200.01 ± 0.014	200.01 ± 0.010	200.00 ± 0.039	200.08 ± 0.466

*There is one wall ring with no contact in the 3D model (N_l = 15) at axial strain of 15%. The data of this wall ring is excluded when calculating the mean and standard deviation

of the average confining pressure. In both the 2D and 3D models, the fluctuation of average confining pressure under different boundary conditions are less than 0.2 kPa (0.1% of the desired confining pressure (200 kPa)).

The numerical results show that the rigid boundary performs better in control accuracy of the average confining pressure than the flexible boundary. However, its performance in the uniformity of confining pressure is the worst. The means and standard deviations of the confining pressures of individual wall segments (right side boundary) in 2D models and that of individual wall rings in 3D models are shown in Table 2. It should be noted that to investigate the distribution of local stress applied by rigid boundary, the boundary is divided into 15 regions along the axial direction, and the local stress is calculated and counted according to these 15 regions. As shown in Table 2, the standard deviations of local stress under the rigid boundary are several hundred times of that under the flexible boundary.

To be more specific, the local stress distribution along the axial direction of different models are shown in Fig. 13. As shown in Fig. 13, the fluctuation of local stress applied by rigid boundary is substantially greater than that applied by flexible boundary.

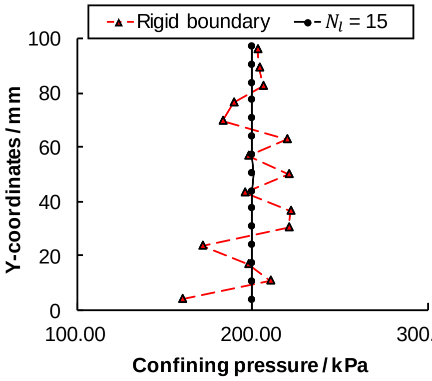
Taking the Fig. 13(d) as an example, under rigid boundary condition, the local confining pressure in the bottom of model less than 100 kPa, while that in the middle higher than 250 kPa. Moreover, the confining pressure on the middle of the rigid boundary is significantly greater than that on both ends, which is particularly evident in 3D models (Fig. 13). This reflects that the rigid boundary overly restrains the movement of particles in the middle of models. The 2D and 3D flexible boundary algorithms in this paper can overcome this problem. The way they apply confining pressure is more accordance with the actual conditions.

Furthermore, Figs. 14(a) and 14(b) presents the confining pressure distributions along the Y-direction of wall segments of 2D models in which N_l are set to be 5, 10, 15 and 20. At the beginning (axial strain of 0%) and ending (axial strain of 10%) of simulations, the confining pressures of individual wall segments vary in a similar and small interval, which shows the control accuracy and stability of 2D algorithm in local confining pressures.

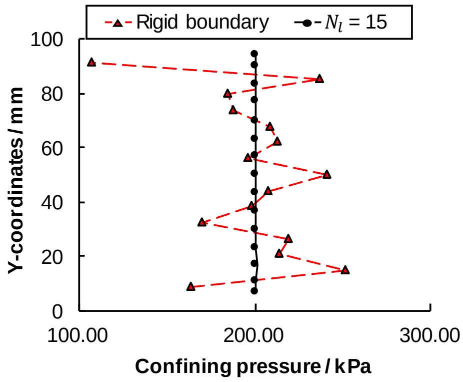
Under different axial strains and different boundary conditions, the confining pressure distributions of wall rings in 3D models are shown in Figs. 14(c) and 14(d). The confining pressure fluctuations of wall rings in 3D models are less than 0.3% of the desired confining pressure, which is acceptable in the CTC test. Comparing Fig. 14(c) with Fig. 14(d), the distributions of confining pressures at axial strain of 15% is obviously more non-uniform than that at axial strain of 0%. This is mainly due to that the mass scaling method was used in the 3D model, which can enlarge the timestep significantly. In fact, if the timestep of 3D model is artificially set to a small value, such as 10^{-5}, the confining pressure distribution of wall rings at the process of loading will be quite uniform. However, the simulation time will also increase by nearly 100 times. To reflect the accuracy of the algorithm in 3D models with scaled density, the authors have not artificially shortened the timestep to obtain higher control accuracy of confining pressure.

The boundary deformations of 2D and 3D models with different boundary conditions are shown in Figs. 15 and 16. As illustrated in Figs. 15 and 16, the models with applied flexible boundaries can deform non-uniformly, while those with rigid boundaries can only deform regularly. Furthermore, the local bulge in the middle of the models, which is a typical localization phenomenon in the CTC of granular material, can only be reflected in the models with flexible boundaries (Figs. 15(b) and 16(b)).

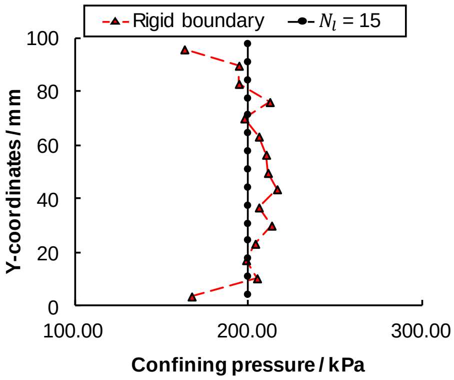
In summary, both 2D and 3D flexible boundary algorithms in this paper allow the boundaries to deform



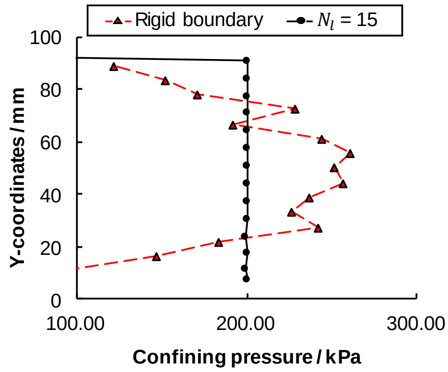
(a) Axial strain of 0% in 2D models



(b) axial strain of 10% in 2D models

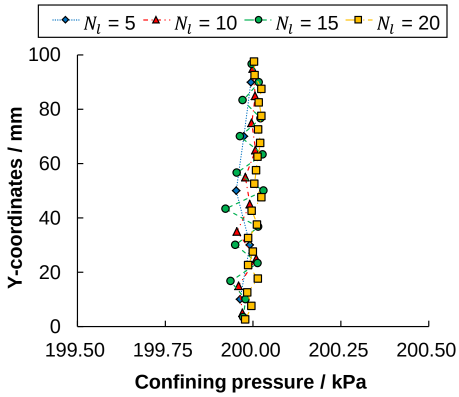


(c) Axial strain of 0% in 3D models

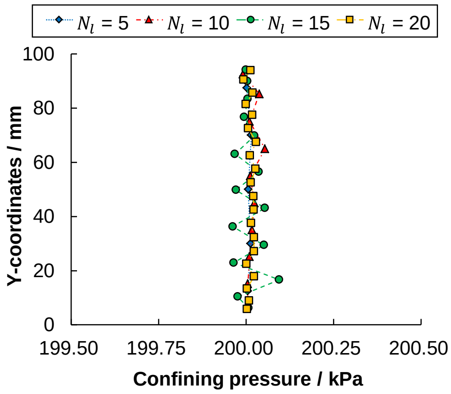


(d) Axial strain of 15% in 3D models

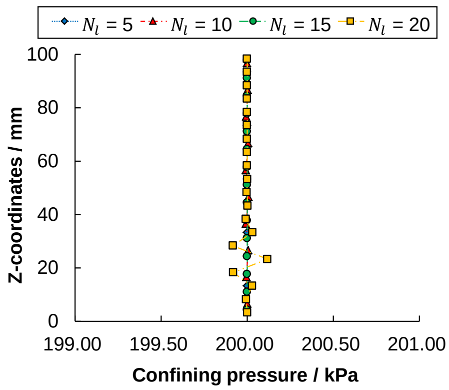
Fig. 13 The local stress distribution in boundary along the axial direction



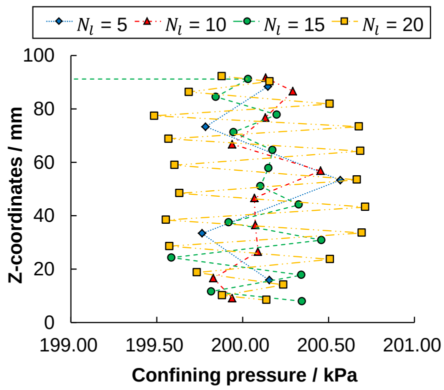
(a) Axial strain of 0% in 2D models



(b) axial strain of 10% in 2D models

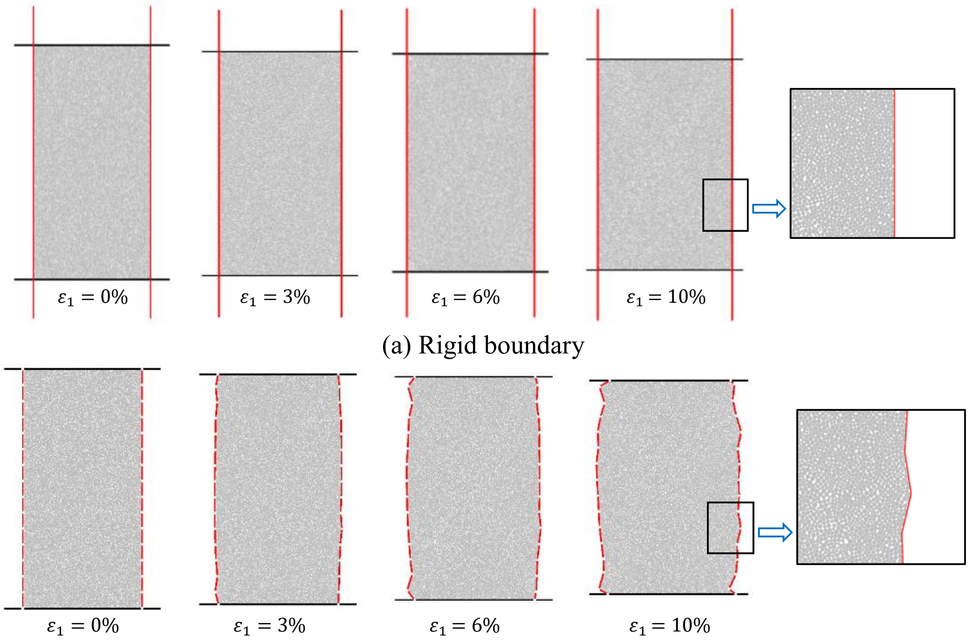


(c) Axial strain of 0% in 3D models



(d) Axial strain of 15% in 3D models

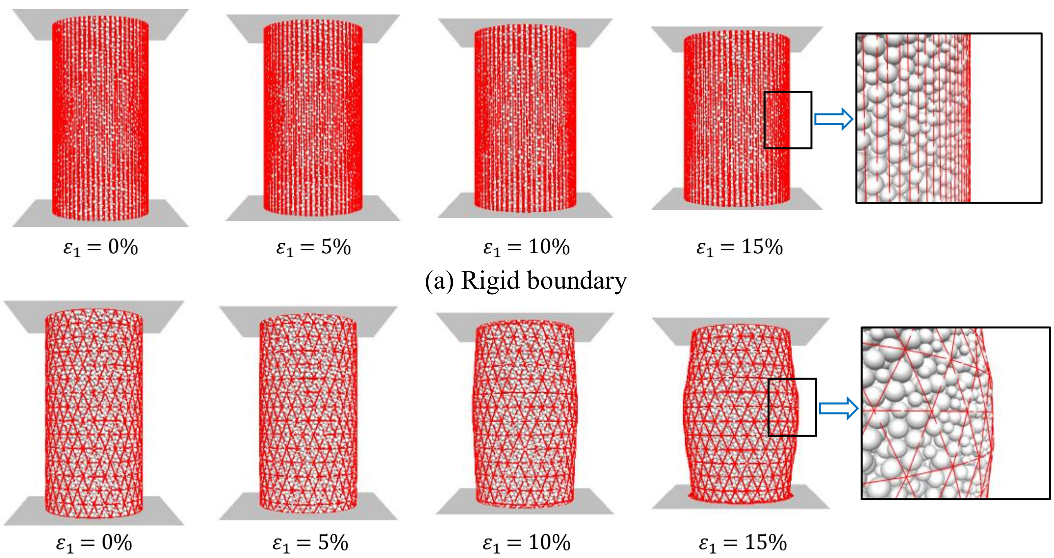
Fig. 14 The confining pressure distributions of wall segments along the vertical direction



(a) Rigid boundary

(b) Flexible boundary (N_l=15)

Fig. 15 Boundary deformation of 2D models



(a) Rigid boundary

(b) Flexible boundary (N_l=15)

Fig. 16 Boundary deformation of 3D models

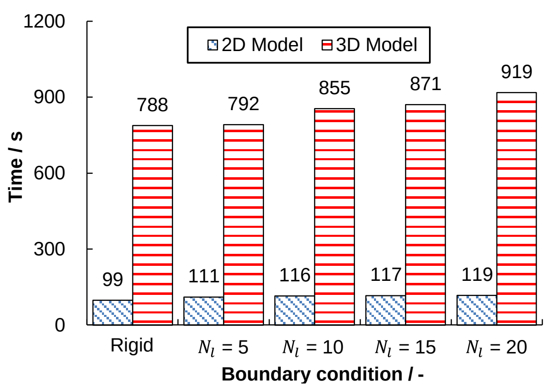
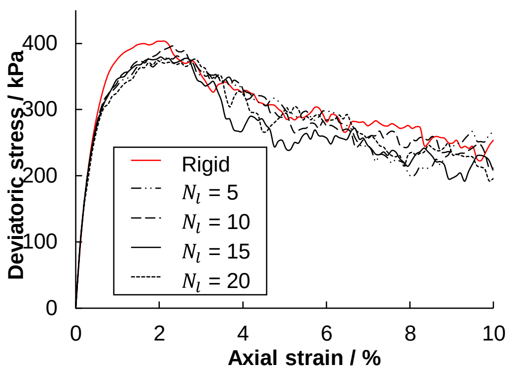
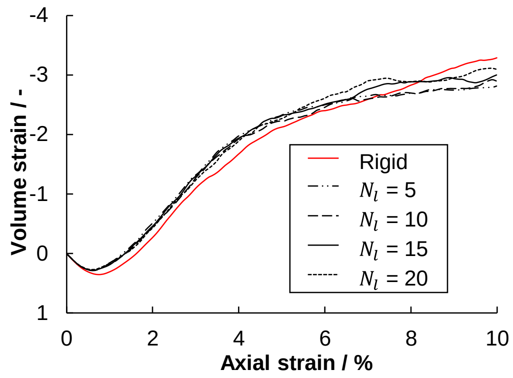


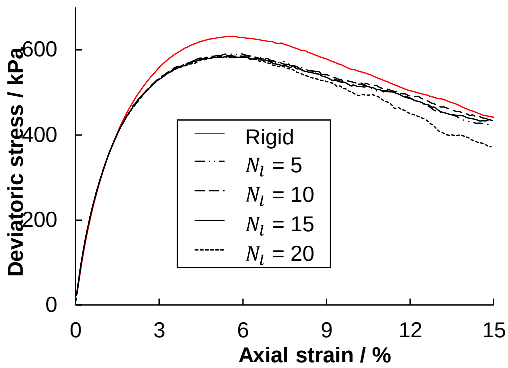
Fig. 17 Computational times (per 10,000 timesteps) of the 2D and 3D models with different boundary conditions



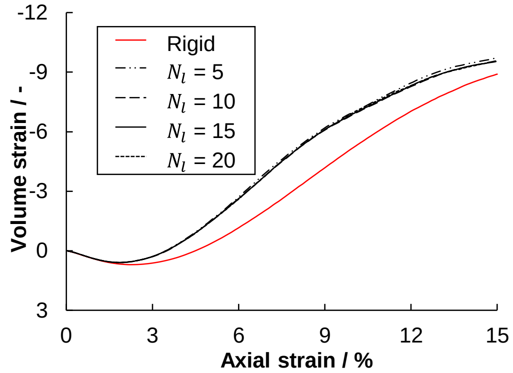
(a) Deviatoric stress and axial strain curve in 2D models



(b) Volume strain and axial strain curve in 2D models



(c) Deviatoric stress and axial strain curve in 3D models



(d) Volume strain and axial strain curve in 3D models

Fig. 18 he response of stress and strain under different boundary conditions

non-uniformly on the premise of retaining high control accuracy of confining pressure.

4.3 Computational efficiency

Computational efficiency is also an important criterion for flexible boundary algorithms. The computational times per 10,000 timesteps of 2D and 3D models with different boundary conditions are compared in Fig. 17. The calculation time of the rigid boundary is shorter than that of the flexible boundary when N_l is set as 5, 10, 15, and 20. In fact, the rigid boundary can be considered as the flexible boundary with N_l=1. From this perspective, the computational time increases gradually with the increase of N_l. This is because the increase of N_l will leads to the increase of the number of wall nodes which is directly related to the computational time of a single timestep. When the N_l increases from 1 to 20, the computational time of the 2D model increases by 19.7%, and that of the 3D model increases by 16.6%. Generally, the computational efficiency will not be reduced significantly when the flexible boundary algorithms of this paper are implemented to the models.

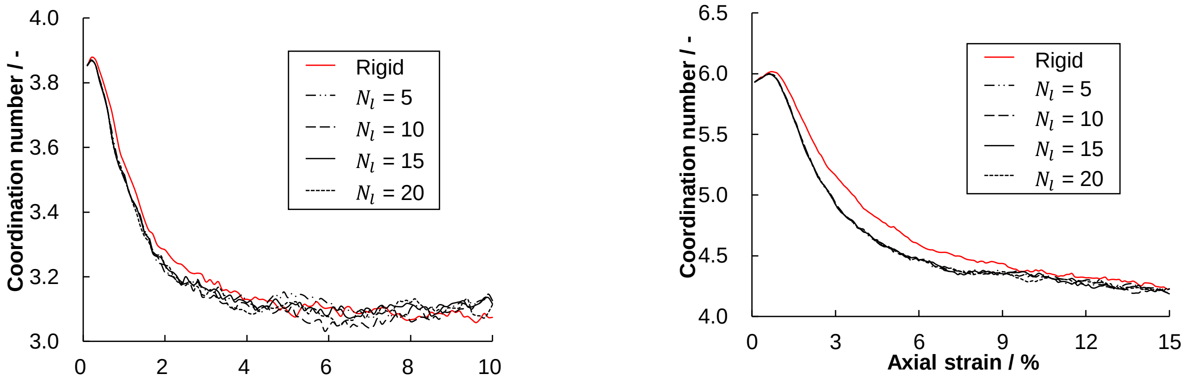
The control accuracy of confining pressure, the deformability of boundary and the computational efficiency of the flexible boundary algorithms of this paper are discussed in this section. The numerical results show that (1) the control accuracy of confining pressure of can meet the requirements of biaxial and triaxial tests; (2) the boundary deformation is consistent with the practical experience of laboratory; and (3) the computational efficiency is close to that of the rigid boundary algorithm. In conclusion, the proposed flexible boundary algorithms are considered to be efficient, reasonable and easy to implement. In the future research, more comparative studies will be carried out to further verify the effectiveness of the proposed algorithm.

5. Sensitivity analysis

The different boundary conditions will lead to different numerical results. A series of biaxial and triaxial simulations with different boundary conditions are carried out in this section. The results of these numerical experiments are discussed in order to illustrate the influence of boundary conditions on macroscale and particle-scale responses of the models. In this section, the influence of N_l on the models are discussed first to provide suggestions on the selection of N_l. Then, based on several biaxial and triaxial simulations with different confining pressures, the differences of numerical results caused by different boundary conditions are discussed and analyzed.

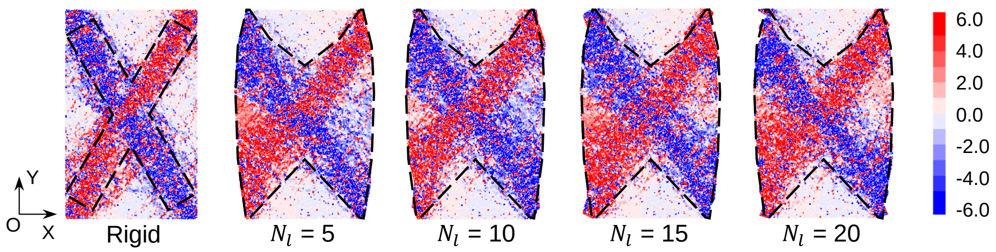
5.1 Sensitivity analysis of N_l

N_l can not only affect the computational efficiency of the models but also determines the deformation capacity of the flexible boundary. The biaxial and triaxial simulations with 200kPa confining pressure are carried out to analyze the influence of N_l on the responses of 2D and 3D models in

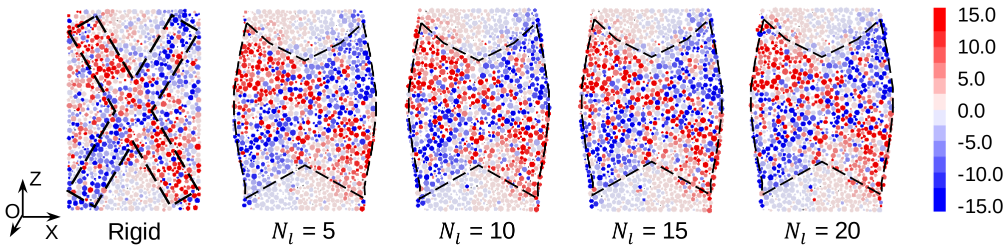


(a) 2D models (b) 3D models

Fig. 19 Coordination number and axial strain curve under different boundary conditions



(a) Axial strain of 10% in 2D models (clockwise angle)



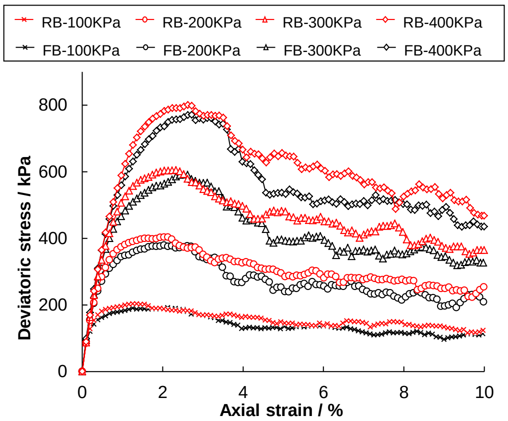
(b) Axial strain of 15% in 3D models (Euler angle in Y direction according to right hand rule)

Fig. 20 The distribution of particle rotations in degrees

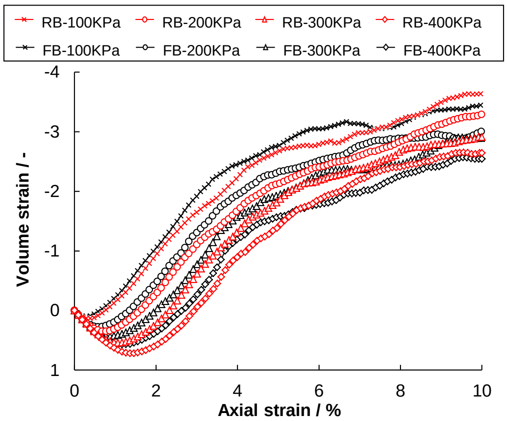
which N_l is set as 5, 10, 15, and 20. For comparison, the simulations with rigid boundary (N_l =1) are also carried out. According to Fig. 18, the stress-strain response under rigid boundary (N_l =1) is obviously different from that under flexible boundary ($N_l \in$[5,20]). However, when $N_l \in$[5,20], there are no significant differences among the stress-strain responses with different N_l. As mentioned above, deformation capacity of the flexible boundary is influenced by N_l. With the increase of N_l, the deformability of the boundary will continuously increase which can reduce the restriction of the boundary on the local deformation. Based on the numerical results, it can be concluded that the number of wall nodes will be sufficient to describe the deformation of the lateral boundary if N_l is set to be greater than 5. In the other words, the macroscale responses of the models are not affected by N_l when $N_l \in$[5,20].

The coordination number C_n, which is defined as the average number of active contacts per particle in the model, is positively correlated with the structure stability of granular materials. As shown in Fig. 19, the trends of the under different boundary conditions are consistent with each other. The change of C_n, can be divided into three stages: rising, falling and then tending to be gentle, corresponding to the volume contraction, volume dilatancy and shear failure of granular materials. The differences among C_n, curves of flexible boundaries (N_l) can be ignored while the differences between that of rigid boundaries and flexible boundaries cannot (Fig. 19). This indicates that when $N_l \in$[5,20], N_l has little influence on the particle-scale responses of the models. This conclusion is also confirmed by Fig. 20.

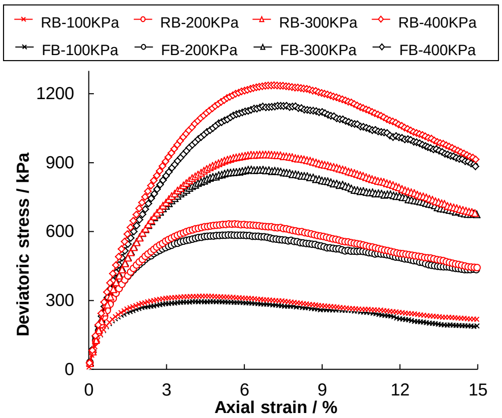
Particle rotation is a good indicator of localizations in granular materials (Iwashita and Oda 1998). According to the distributions of particle rotations shown in Fig. 20, localizations can be observed in the models under both the rigid and flexible boundary conditions. The localization patterns under rigid boundary condition (X-shaped pattern) are different from that under flexible boundary condition (butterfly-shaped pattern), which is coincide with the result of Khoubani and Evans (2018). To be more specific, the rotations of particles near the middle of the lateral boundary under rigid boundary are significantly constrained, which



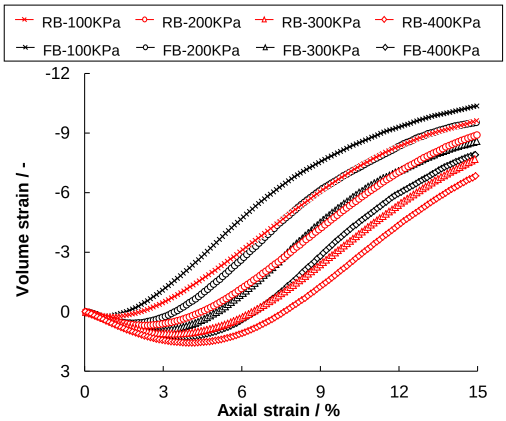
(a) Deviatoric stress and axial strain curve in 2D models



(b) Volume strain and axial strain curve in 2D models

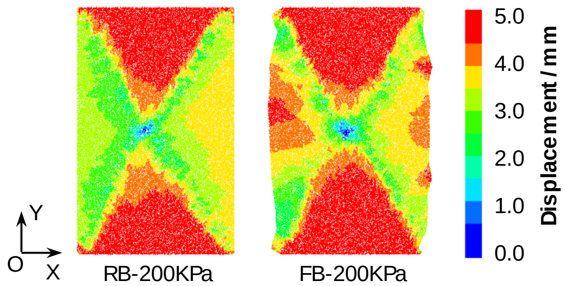


(c) Deviatoric stress and axial strain curve in 3D models

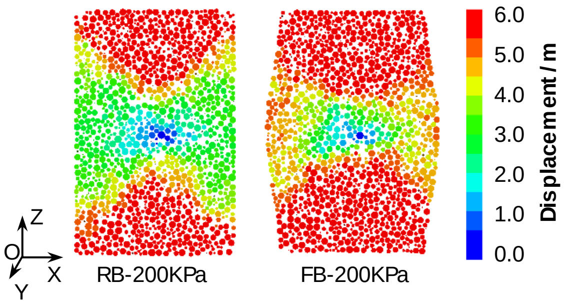


(d) Volume strain and axial strain curve in 3D models

Fig. 21 The response of stress and strain under different boundary conditions (RB: rigid boundary; FB: flexible boundary) and different confining pressures



(a) 2D models (ε_1=10%)



(b) 3D models (ε_1=15%)

Fig. 22 Comparison of particle displacements under rigid and flexible boundary (N_l=15) conditions

strongly contrast with that under flexible boundary. As a result, the region with high rotations under rigid boundary is X-shaped, while the counterpart under flexible boundary is butterfly shaped. Furthermore, it should be noted that the distributions of particle rotations under flexible boundary condition are almost identical when $N_l \in [5,20]$ (Fig. 20).

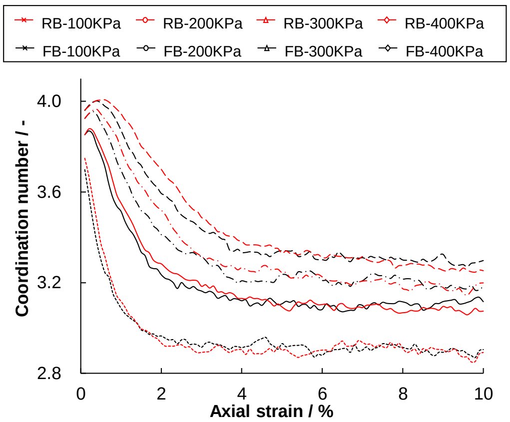
In summary, when $N_l \in [5,20]$, the change of N_l has no significant influence on the macro-scale and particle-scale responses of the models.

5.2 Sensitivity analysis of boundary conditions

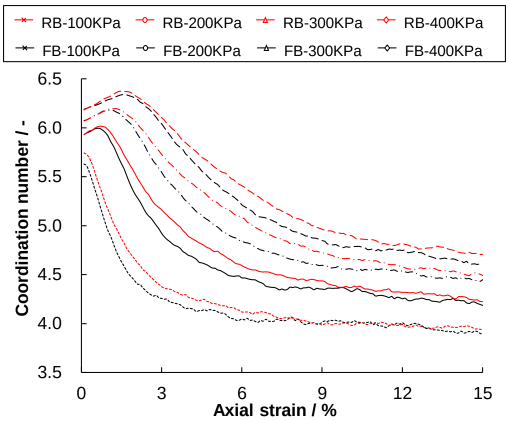
To further study the influence of different boundary conditions on the macro-scale and particle-scale responses of models, the biaxial and triaxial simulations with different boundary conditions (rigid boundary and flexible boundary) and different confining pressures (100 kPa, 200 k Pa, 300 kPa and 400 kPa.) are carried out. According to the conclusions of Section 5.1, N_l is taken as 15 in this section to fully ensure the deformability of the flexible boundary.

The responses of stress and strain in the simulations are shown in Fig. 21. Two tentative conclusions for both 2D and 3D models can be drawn as follows.

(1) The peak deviation stress under the rigid boundary is greater than that under the flexible boundary. In fact, there are still some controversies about this conclusion. Some researchers (Cil and Alshibli 2014, Binesh *et al.* 2018, Qu *et al.* 2019) believe that the flexible boundary can lead to higher strength of specimens comparing with rigid

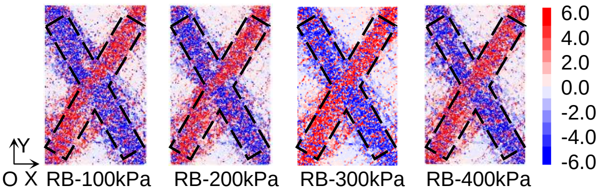


(a) 2D models

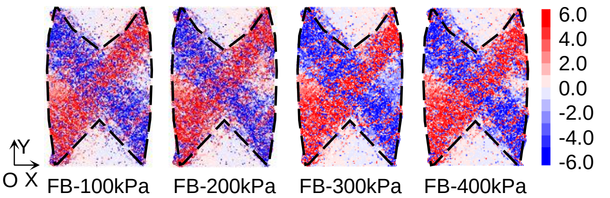


(b) 3D models

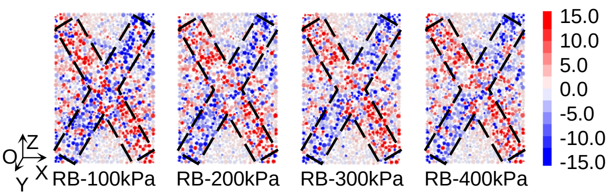
Fig. 23 Coordination number and axial strain curve under different boundary conditions and different confine pressure



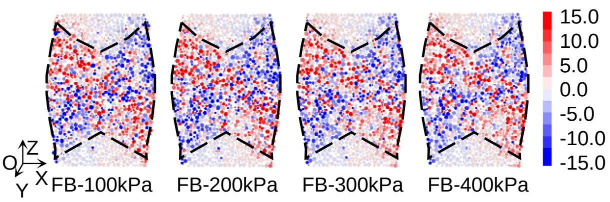
(a) Axial strain of 10% in 2D models (RB)



(b) Axial strain of 10% in 2D models (FB)



(c) Axial strain of 10% in 3D models (RB)



(d) Axial strain of 10% in 3D models (FB)

Fig. 24 Particle rotations in degrees under different boundary conditions and different confine pressures

boundary, while some researchers (Cheung and O'Sullivan 2008) demonstrate that the stress-strain responses under flexible and rigid boundary have no significant divergence. However, the numerical results of this paper show that the specimens under rigid boundary will exhibit higher strength and stiffness than that under flexible boundary, which is consistent with the conclusion of Wang and Tonon (2009). It is inferred that the higher strength and stiffness result from the excessive constraint of rigid boundary which restrains the deformability processes and the formation of shear bands. As shown in Fig. 22, the displacement magnitudes near the lateral boundary under rigid boundary approximate the uniform distribution while that under flexible boundary are distributed in a different manner, i.e., displacement magnitudes near the middle of the lateral boundary are apparently higher than that near the end of the lateral boundary. This phenomenon is caused by the fact that the rigid boundary can only move uniformly while the flexible boundary can deform flexibly. Furthermore, the restrictions of rigid boundary will become more obvious with the increase of confining pressure.

(2) The volume contraction under the rigid boundary is larger than that under the flexible boundary. And the volume dilatancy also occurs earlier in the models with flexible boundary. This phenomenon results from the restrictions of rigid boundary which increase the compressibility of the models and delay the occurrence of dilatancy. According to the Fig. 23, the C_n under rigid boundary is significantly larger than that under flexible boundary in the stage of volume dilatancy. This reflects the fact that the models with rigid boundary are more compact than those with flexible boundary during the stage of volume dilatancy. This confirms the restriction of the rigid boundary on the models from the particle-scale perspective. Moreover, it also explains why the strength and stiffness under rigid boundary are higher.

The distributions of particle rotations in the models with different boundary conditions and confining pressures are shown in Fig. 24. The boundary conditions have an evidently effect on the distributions of particle rotations. The region with high rotations is X-shaped under rigid boundary, while it is butterfly shaped under flexible boundary. This indicates that the particle rotations near the middle of the lateral boundary are restricted by the rigid boundary, which is caused by the fact that the rigid boundary can only move uniformly. The localizations and deformations of granular materials can be better presented in the models with flexible boundaries, which are more consistent with the results of laboratory testing.

To summarize, the macroscale and particle-scale responses under rigid boundaries are different from those under flexible boundaries. It primarily results from the restriction of the rigid boundary on the deformation and localization of the models. Therefore, it can be concluded that the flexible boundary, which has less restriction on the models, is more suitable for the models with large strain and significant localization.

6. Conclusions

Based on the servo algorithm of the rigid boundary, the flexible boundary algorithms for 2D and 3D DEM models are proposed. The algorithms are not only effective and efficient but also easy to implement. The algorithms proposed in this paper apply the desired confining pressure to the specimens by controlling the velocity of wall nodes distributed along the boundary. Therefore, the boundary is naturally continuous and closed, which can prevent the escape of particles. Moreover, both the horizontal and the vertical components of confining pressure are considered in the 2D and 3D algorithms. The confining pressure will always be perpendicular to the surface of the specimen.

The results of numerical experiments indicate that the algorithms in this paper allow the boundaries to deform non-uniformly on the premise of maintaining the high control accuracy of confining pressure. Furthermore, these algorithms will not significantly increase the computational cost of the models.

In addition, in order to study the influences of different boundary conditions on the numerical results, a series of biaxial and triaxial simulations with different boundary conditions and different confining pressures are carried out in this paper. The numerical results indicate the following:

(1) The macro deformation of the unbonded-particle model in biaxial or triaxial simulations is not complicated. The number of wall nodes will be sufficient to describe the deformation of the lateral boundary when N_l>5. When $N_l \in [5,20]$, the change of N_l has no significant influence on the macro-scale and particle-scale responses of the models.

(2) The stress-strain responses of models with rigid boundaries are different from those with flexible boundaries. The differences will be more significant with the increase of confining pressure. The differences are primarily caused by the restriction of rigid boundary on the natural development of localizations within granular materials.

(3) The deformation and failure behaviors of the granular materials will be affected by different boundary conditions. This results from that the rigid boundary can only move uniformly which constrains the movements of the particles near the boundary.

Acknowledgments

This study is supported by the National Key Research and Development Program of China (No. 2017YFC0405105) and National Natural Science Foundation of China (No. 51679164).

References

Ai, J., Chen, J.F., Rotter, J.M. and Ooi, J.Y. (2011), “Assessment of rolling resistance models in discrete element simulations”, *Powder Technol*., **206**(3), 269-282. http://doi.org/10.1016/j.powtec.2010.09.030.

Bardet, J.P. and Proubet, J. (1991), “A numerical investigation of the structure of persistent shear bands in granular media”, *Géotechnique*, **41**(4), 599-613. http://doi.org/10.1680/geot.1991.41.4.599.

Belheine, N., Plassiard, J., Donze, F., Darve, F. and Seridi, A. (2009), “Numerical simulation of drained triaxial test using 3D discrete element modeling”, *Comput*. *Geotech*., **36**(1-2), 320-331. https://doi.org/10.1016/j.compgeo.2008.02.003.

Binesh, S.M., Eslami-Feizabad, E. and Rahmani, R. (2018), “Discrete element modeling of drained triaxial test: Flexible and rigid lateral boundaries”, *Int*. *J*. *Civ*. *Eng*., **16**(10), 1463-1474. http://doi.org/10.1007/s40999-018-0293-0.

Cheung, G. and O’Sullivan, C. (2008), “Effective simulation of flexible lateral boundaries in two- and three-dimensional DEM simulations”, *Particuology*, **6**(6), 483-500. http://doi.org/10.1016/j.partic.2008.07.018.

Cil, M.B. and Alshibli, K.A. (2013), “3D analysis of kinematic behavior of granular materials in triaxial testing using DEM with flexible membrane boundary”, *Acta Geotech*., **9**(2), 287-298. http://doi.org/10.1007/s11440-013-0273-0.

Cui, L., O’Sullivan, C. and O’Neill, S. (2007), “An analysis of the triaxial apparatus using a mixed boundary three-dimensional discrete element model”, *Géotechnique*, **57**(10), 831-844. http://doi.org/10.1680/geot.2007.57.10.831.

Cundall, P.A. and Strack, O.D.L. (1979), “A discrete numerical model for granular assemblies”, *Géotechnique*, **29**(1), 47-65. http://doi.org/10.1680/geot.1980.30.3.331.

Dai, B., Yang, J. Gu, X.Q. and Zhang, W. (2019), “A numerical analysis of the equivalent skeleton void ratio for silty sand”, *Geomech*. *Eng*., **17**(1), 19-30. http://doi.org/10.12989/gae.2019.17.1.019.

de Bono, J., McDowell, G. and Wanatowski, D. (2012), “Discrete element modelling of a flexible membrane for triaxial testing of granular material at high pressures”, *Géotech*. *Lett*., **2**(4), 199-203. http://doi.org/10.1680/geolett.12.00040.

Evans, T.M. and Frost, J.D. (2010), “Multiscale investigation of shear bands in sand: Physical and numerical experiments”, *Int*. *J*. *Numer*. *Anal*. *Meth*. *Geomech*., **34**(15), 1634-1650. http://doi.org/10.1002/nag.877.

Evans, T.M. and Valdes, J.R. (2011), “The microstructure of particulate mixtures in one-dimensional compression: Numerical studies”, *Granul*. *Matter*, **13**(5), 657-669. http://doi.org/10.1007/s10035-011-0278-z.

Feng, Y.T. and Owen, D.R.J. (2014), “Discrete element modelling of large scale particle systems—I: Exact scaling laws”, *Comp*. *Part*. *Mech*., **1**(2), 159-168. http://doi.org/10.1007/s40571-014-0010-y.

Feng, Y.T., Han, K., Owen, D.R.J. and Loughran, J. (2009), “On upscaling of discrete element models: Similarity principles”, *Eng*. *Comput*., **26(**6), 599-609. http://doi.org/10.1108/02644400910975405.

Huang, X., Hanley, K.J., Zhang, Z. and Kwok, C.Y. (2019), “Structural degradation of sands during cyclic liquefaction: Insight from DEM simulations”, *Comput*. *Geotech*., **114**, 103139. http://doi.org/10.1016/j.compgeo.2019.103139

Itasca Consulting Group Inc. (2004), *Particle Flow Code in 3D Dimensions (PFC3D) User’s Manual*, Version 3.1, Itasca, Minneapolis, Minnesota, U.S.A.

Iwashita, K. and Oda, M. (1998), “Rolling resistance at contacts in simulation of shear band development by DEM”, *J*. *Eng*. *Mech*., **124**(3), 285-292.

http://doi.org/10.1061/(ASCE)0733-9399(1998)124:3(285).
Jacobson, D.E., Valdes, J.R. and Evans, T.M. (2007), "A numerical view into direct shear specimen size effects", *Geotech. Test. J.*, **30**(6), 512-516. http://doi.org/10.1520/GTJ100923.
Jiang, M.J., Yan, H.B., Zhu, H.H. and Utili, S. (2011), "Modeling shear behavior and strain localization in cemented sands by two-dimensional distinct element method analyses", *Comput. Geotech.*, **38**(1), 14-29. http://doi.org/10.1016/j.compgeo.2010.09.001.
Khoubani, A. and Evans, T.M. (2018), "An efficient flexible membrane boundary condition for DEM simulation of axisymmetric element tests", *Int. J. Numer. Anal. Meth. Geomech.*, **42**(4), 694-715. http://doi.org/10.1002/nag.2762.
Kuhn, M.R. (1995), "A flexible boundary for three-dimensional DEM particle assemblies", *Eng. Comput.*, **12**(2), 175-183. http://doi.org/10.1108/02644409510799541.
Kumara, J.J. and Hayano, K. (2016), "Importance of particle shape on stress-strain behaviour of crushed stone-sand mixtures", *Geomech. Eng.*, **10**(4), 455-470. http://doi.org/10.12989/gae.2016.10.4.455.
Lee, S.J., Hashash, Y.M.A. and Nezami, E.G. (2012), "Simulation of triaxial compression tests with polyhedral discrete elements", *Comput. Geotech.*, **43**, 92-100. http://doi.org/10.1016/j.compgeo.2012.02.011.
Lu, Y., Li, X. and Wang, Y. (2018), "Application of a flexible membrane to DEM modelling of axisymmetric triaxial compression tests on sands", *Eur. J. Environ. Civ. Eng.*, **22**(s1), s19-s36. http://doi.org/10.1080/19648189.2018.1425157.
Ma, G., Chang, X.L., Zhou, W. and Ng, T.T. (2014), "Mechanical response of rockfills in a simulated true triaxial test: A combined FDEM study", *Geomech. Eng.*, **7**(3), 317-333. http://doi.org/10.12989/gae.2014.7.3.317.
Meng, J., Huang, J., Sheng, D. and Sloan, S.W. (2017), "Quasi-static rheology of granular media using the static DEM", *Int. J. Geomech.*, **17**(11), 04017094. http://doi.org/10.1061/(asce)gm.1943-5622.0001001.
O'Sullivan, C. and Cui, L. (2009), "Micromechanics of granular material response during load reversals: Combined DEM and experimental study", *Powder Technol.*, **193**(3), 289-302. http://doi.org/10.1016/j.powtec.2009.03.003
Qu, T., Feng, Y.T., Wang, Y. and Wang, M. (2019), "Discrete element modelling of flexible membrane boundaries for triaxial tests", *Comput. Geotech.*, **115**, 103154. http://doi.org/10.1016/j.compgeo.2019.103154.
Rakhimzhanova, A.K., Thornton, C., Minh, N.H., Fok, S.C. and Zhao, Y. (2019), "Numerical simulations of triaxial compression tests of cemented sandstone", *Comput. Geotech.*, **113**, 103068. http://doi.org/10.1016/j.compgeo.2019.04.013.
Shi, D., Yang, C., Xue, J. and Wang, W. (2018), "Discrete element modeling of hollow cylinder shear behavior of granular material with fixed principal stress diection", *J. Hydraul. Eng.*, **49**(8), 917-925. (In Chinese). http://doi.org/10.13243/j.cnki.slxb.20180233.
Wang, Y. and Tonon, F. (2009), "Modeling triaxial test on intact rock using discrete element method with membrane boundary", *J. Eng. Mech.*, **135**(9), 1029-1037. http://doi.org/10.1061/(ASCE)EM.1943-7889.0000017.
Wang, Y.H. and Leung, S.C. (2008), "A particulate-scale investigation of cemented sand behavior", *Can. Geotech. J.*, **45**(1), 29-44. http://doi.org/10.1139/t07-070.
Zhao, X. and Evans, T.M. (2009), "Discrete simulations of laboratory loading conditions", *Int. J. Geomech.*, **9**(4), 169-178. http://doi.org/10.1007/s10035-011-0284-1.
Zhao, X. and Evans, T.M. (2011), "Numerical analysis of critical state behaviors of granular soils under different loading conditions", *Granul. Matter*, **13**(6), 751-764. http://doi.org/10.1061/(ASCE)1532-3641(2009)9:4(169).
Zhou, L.L., Chu, X.H., Zhang, X. and Xu, Y.J. (2016), "Numerical investigations on breakage behaviour of granular materials under triaxial stresses", *Geomech. Eng.*, **11**(5), 639-655. http://doi.org/10.12989/gae.2016.11.5.639.

CC

Appendix A

The difference between WRA and WSA is the calculation method of v_s^i. Based on the WRA purposed in Section 3.3, the implementation of WSA should calculate the v_s^i according to, rather than Eqs. (15)-(20), the following equations:

$$v_S^i = \frac{\alpha S^i}{\sum_{j=1}^{N} k_n^j \Delta t}\left(\sigma_{cur} - \sigma_{tar}\right) \tag{A1}$$

where S^i is the area of the wall segment i, and the other symbols are as defined in Section 3.1. S^i can be calculated by:

$$S^i = \sqrt{p(p-a)(p-b)(p-c)} \tag{A2}$$

where a, b and c are the side length of triangular wall segment i, and p can be calculated by:

$$p = \frac{a+b+c}{3} \tag{A3}$$

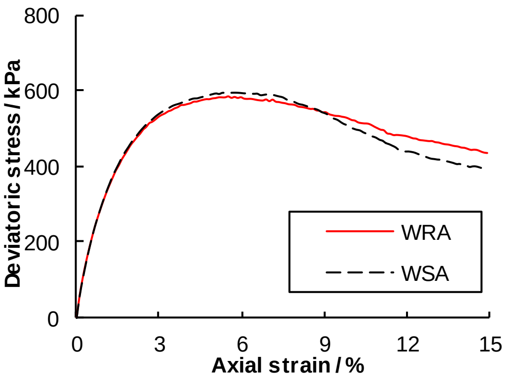
Furthermore, the servo velocities of the wall segments without contact can be calculated by:

$$v_S^i = \frac{\alpha S^i}{\left(S^i / d_{50}^2\right) k_n \Delta t}\left(0 - \sigma_{tar}\right) = -\frac{\alpha d_{50}^2}{k_n \Delta t}\sigma_{tar} \tag{A4}$$

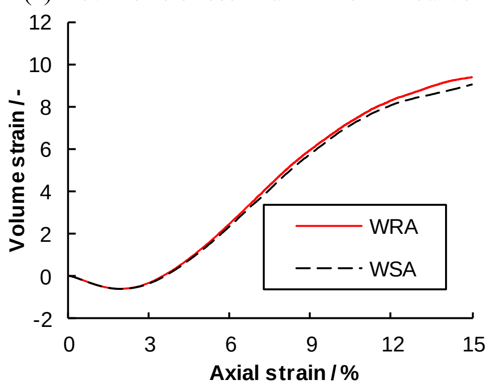
The boundary of WSA is more flexible than that of WRA. By using the WSA, the macroscopic asymmetric deformation of model can be reflected. However, there are two limitations in WSA: (1) To guarantee the stability of WSA, the particle number in the 3D model must be large enough, which limits the applicability of this algorithm; and (2) Servo velocity of each wall segment is calculated independently, which will reduce the computational efficiency of this algorithm.

WRA and WSA are compared by simulations of conventional triaxial compression. As shown in Fig. A1, for unbonded-particle models, the numerical results by conventional triaxial tests with WRA and WSA have no significant difference. This is mainly due to the specimen is nearly homogeneous which lead to the symmetric deformation under conventional triaxial compression.

In summary, for 3D models, WRA is recommend when the boundary conditions are symmetric and the specimens are nearly homogeneous.

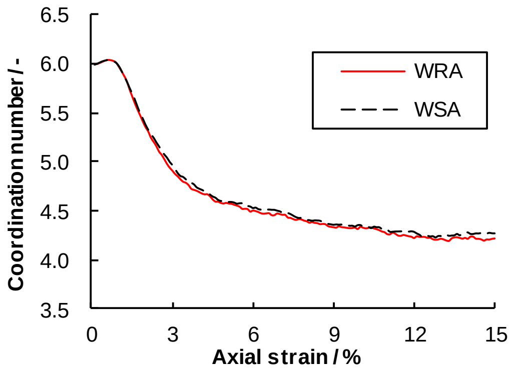


(a) Deviatoric stress and axial strain curve

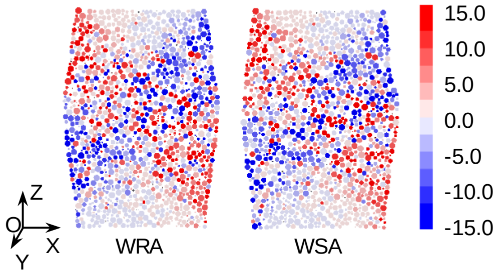


(b) Volume strain and axial strain curve

Fig. A1 Comparison of the numerical results of wall ring and segment algorithms (N_l=15 and N_p=20)



(c) Coordination number and axial strain curve



(d) Particle rotations in degrees

Fig. A1 Continued