Computers and Geotechnics 36 (2009) 953–959

Contents lists available at ScienceDirect

Computers and Geotechnics

journal homepage: www.elsevier.com/locate/compgeo

Potential particles: a method for modelling non-circular particles in DEM

G.T. Houlsby *

Department of Engineering Science, Oxford University, Parks Road, Oxford OX1 3PJ, United Kingdom

ARTICLE INFO

Article history:
Received 3 September 2008
Received in revised form 3 March 2009
Accepted 4 March 2009
Available online 14 April 2009

Keywords:
DEM
Non-circular particles

ABSTRACT

Options for defining non-circular (or non-spherical) particles in DEM codes are briefly reviewed before introducing a new concept: potential particles. This method can be used to define convex particles with a wide variety of shapes, from almost polygonal to circular. It is readily applicable in both two and three dimensions. Although contact detection and overlap calculations are not as fast as for circular particles, they can nevertheless be made sufficiently simple and fast for implementation in DEM codes.

1. Introduction

It is well known that an important fraction of the computation time in a DEM calculation may devoted to contact detection and calculation of the overlap distance between particles. For circular particles (in 2D) or spherical particles (in 3D) the calculation is trivial: the overlap is simply the distance between the particle centres, minus the sum of the radii of the two contacting particles. For almost any other particle shape the calculation is much more complex, and the resulting computation can become sufficiently lengthy that it impacts significantly on overall run time. For most applications real particles are, of course, not spherical. However, the simplicity of the contact detection and overlap calculations is a powerful driver that encourages the use of the spherical idealisation. Some leading commercial DEM codes, for instance, only offer circular or spherical particles.

Particle shape can in some applications have a very important influence on the behaviour of a granular medium, and so a means of modelling non-circular (or non-spherical) particles is required. Several methods are available, and we briefly review these options before introducing a new technique called "potential particles". The new method has the advantage that particles of widely differing shapes can be defined, whilst the contact detection and overlap calculations are relatively straightforward. Unlike some other techniques, the new method is readily extended from 2D to 3D.

With the exception of Option 2, discussed below, all the techniques addressed here can generate only convex particles. Real particles often involve concavities, and in some cases it will be important to be able to model them. The detection of contacts between con-convex particles is, however, significantly more complex than the detection of contacts between convex particles, principally because of the possibility for multiple contacts between the same pair of particles. (Any pair of convex particles can of course only touch at one point.) Contacts between concave particles have been considered (*e.g.* [5,6]), but we avoid that necessity here.

2. Some options for non-circular particles

2.1. Option 1

Perhaps the most obvious extension of circular or spherical particles is to modify them to ellipses or ellipsoids. Although this approach seems superficially attractive, there are two main disadvantages. Firstly the range of particle shapes available is still strictly limited. Secondly, however, the detection of contacts between ellipses is mathematically not particularly tractable (it involves solving a quartic), and the problem for ellipsoids in 3D is even more difficult. For these reasons this does not appear to be a popular option.

2.2. Option 2

A simpler way to create non-circular particles in a DEM code is in fact to aggregate circular particles into "molecules" (or clusters or agglomerates), each consisting of circular "atoms" that are bound rigidly together, Fig. 1 (see for example [7,2]). Contacts (and overlaps) between atoms in the same molecule are ignored. Contacts between atoms in neighbouring molecules are detected in the usual way, the forces on an entire molecule are aggregated, and the motion of the whole molecule moving as a rigid body is determined. The advantage is that little new mechanics has to be introduced to the program, with most of the new computation

* Tel.: +44 1865 273138.
E-mail address: guy.houlsby@eng.ox.ac.uk

0266-352X/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compgeo.2009.03.001

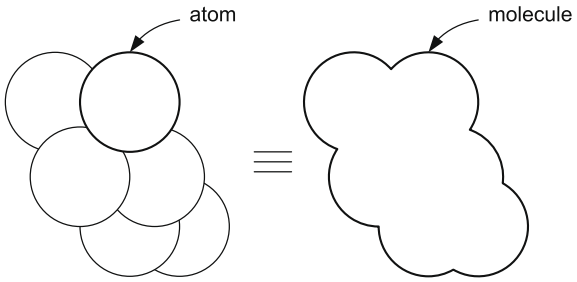

Fig. 1. An "atom and molecule" particle or agglomerate.

simply involving a "housekeeping" exercise in attributing atoms to molecules. The disadvantages are that very large numbers of constituent atoms may be needed to make a realistically shaped particle, and the resulting molecules are often too "knobbly" by comparison with real particles. The method can be extended to include non-rigid connections within each molecule, and even to include criteria for molecule breakage. The extension from 2D to 3D is trivial. Unlike the other techniques addressed here, this method is as suitable for modelling non-convex particles as it is for convex.

2.3. Option 3

Another obvious way to create a non-circular particle is the use of polygonal particles. If these are limited to convex particles then contact detection is relatively straightforward, with the important computation being detection of edge/vertex contacts. The computation time is proportional to the product of the number of vertices in the two neighbouring particles. If concave particles are allowed then the computation becomes much more complex, as each particle may contact another at multiple points (*e.g.* [6]). The advantage of polygonal particles is clearly that they can model angular particles more realistically. The main disadvantage is the computational complexity. The extension to 3D is not trivial, as contacts between polyhedra may involve a number of possibilities such as vertex/face and edge/edge contacts. Latham et al. [3] and Latham and Munjiza [4] have explored the use of simple polyhedra such as tetrahedra and cubes.

2.4. Option 4

More realistic particles can be made by adopting a hybrid between Options 2 and 3 above. In 2D a particle may be specified by specifying a number of corner "atoms" as in method 2. The gaps between the atoms may then be bridged by straight segments, see Fig. 2. Crammond [1] termed these "racetrack particles". In the limit as the diameters of the corner atoms become very small this method approaches Option 3 as the particles become polygonal. In fact for computation there is a distinct advantage if the particles are strictly convex, not merely convex, and it is preferable to bridge

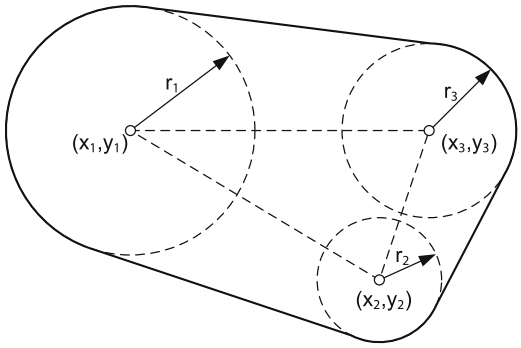

Fig. 2. A "racetrack" particle.

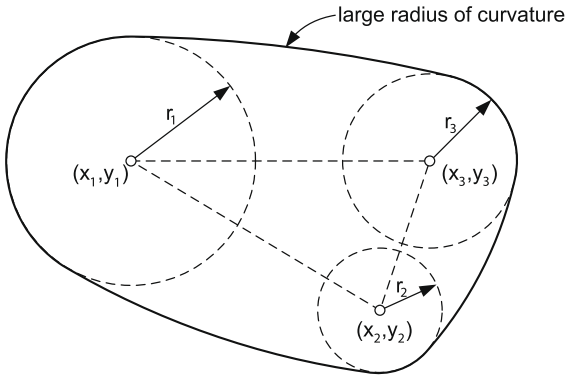

Fig. 3. A "modified racetrack" particle with curved sides.

the corner atoms with circular arcs, see Fig. 3. If the radii of the arcs are large compared to those of the corner atoms, then the sides approach straight lines. The modified racetrack particles only involve contacts between circular arcs, which is straightforward computationally. The main change is that contacts can only be made within a certain range of angles for each arc. The transition points between the different arcs can easily be determined by standard geometry. Although Crammond [1] successfully implemented such particles, he found that simple particle contact detection algorithms often falsely detected a contact when none had occurred, and that a number of checking routines were required to eliminate these false positives. In 2D, however, this technique can be used to define a very wide range of realistic looking particles of varying shapes and angularity.

Fig. 4 shows an example of an assembly of 100 "modified racetrack" particles of three different shapes, deposited under gravity into a rectangular hopper. The possibilities of this method by comparison with just circular particles are immediately apparent. By comparison with an equivalent analysis with the same number of circular particles, this analysis ran about 11.8 times slower, but for some applications this would be a penalty worth paying for the benefits of the non-circular particles.

The extension to 3D could be made in principle, with small spheres at each vertex of a polyhedron, and each face being bridged by a sphere of much larger radius. Provided that all the vertex spheres are the same size the edges would then be sections of toruses. The problem of detecting all possibilities of sphere/sphere,

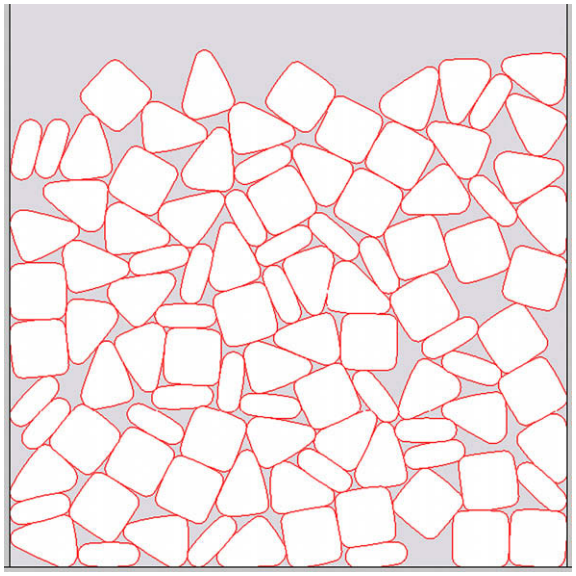

Fig. 4. An assembly of modified racetrack particles.

sphere/torus and torus/torus contacts, including elimination of all possible false positives is, however, horrendous. It is the contemplation of this problem that provided the impetus for the development of potential particles.

3. Potential particles

An alternative to the definition of a particle by geometric construction is to define the particle simply in terms of some function of a local coordinate system. The following analysis is presented in terms of two dimensions, but the approach can very readily be extended to three dimensions.

We define a "potential particle" by means of a function $f(x,y) = 0$, where (x,y) are local coordinates. The function is chosen so that:

- $f = 0$ defines the particle surface,
- $f < 0$ "inside" the particle,
- $f > 0$ "outside" the particle,
- the particle is strictly convex, and furthermore any surface $f = const$ is strictly convex. This restriction is introduced to eliminate the possibility of multiple contact points between particles.

An example is illustrated in Fig. 5. The shaded area represents the particle, and the lines are contours $f = const$. The illustration shows an elliptical particle, but the method can readily be extended to particles with other shapes (see below for one possible approach).

For any point (x,y) the function must be defined such that it is possible to evaluate not just the value of the function f, but also the first and second partial derivatives $\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$, $\frac{\partial^2 f}{\partial x^2}$, $\frac{\partial^2 f}{\partial x \partial y}$ and $\frac{\partial^2 f}{\partial y^2}$.

3.1. Contacts between potential particles

Now consider two "potential particles" defined by $f_1(x,y) = 0$ and $f_2(x,y) = 0$. We seek first point P_1 which is on the surface of particle 1 and "closest" to particle 2, see Fig. 6. This point can be found by minimising f_2 subject to the constraint $f_1 = 0$. This defines a point that is "closest" in a certain sense. For certain functions it may not correspond exactly to the closest point in terms of Cartesian distance, but for the purposes of establishing contacts between particles it is quite sufficient.

We use the method of Lagrangian multipliers, so that we minimise $f_2 + \Lambda f_1$. This gives the conditions:

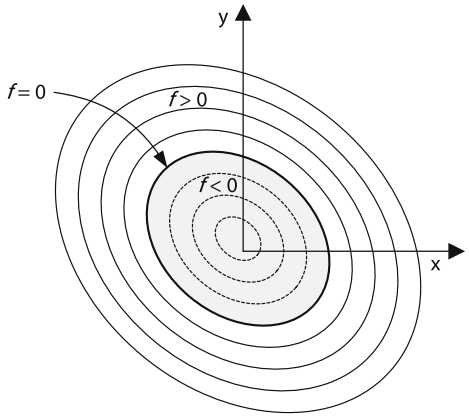

Fig. 5. A simple "potential particle".

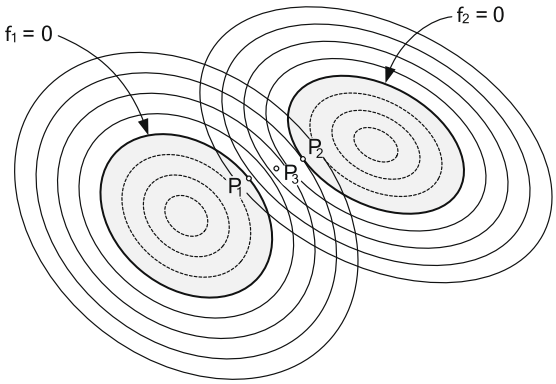

Fig. 6. Two potential particles not in contact.

$$\frac{\partial}{\partial x}(f_2 + \Lambda f_1) = \frac{\partial f_2}{\partial x} + \Lambda \frac{\partial f_1}{\partial x} = 0 \tag{1}$$

$$\frac{\partial}{\partial y}(f_2 + \Lambda f_1) = \frac{\partial f_2}{\partial y} + \Lambda \frac{\partial f_1}{\partial y} = 0 \tag{2}$$

Eliminating the Lagrangian multiplier Λ between these two equations leads to:

$$\frac{\partial f_1}{\partial x}\frac{\partial f_2}{\partial y} - \frac{\partial f_1}{\partial y}\frac{\partial f_2}{\partial x} = 0 \tag{3}$$

We then seek a solution to the above equation together with $f_1 = 0$.

There are a variety of ways of solving these equations. It is convenient to use the Newton–Raphson method. Starting from some trial point (x_0, y_0), and linearising the variation of f_1 about that point we have $\delta f_1 \approx \frac{\partial f_1}{\partial x}\delta x + \frac{\partial f_1}{\partial y}\delta y$. We therefore seek increments that satisfy:

$$\delta f_1 = -f_1(x_0, y_0) \approx \frac{\partial f_1}{\partial x}\delta x + \frac{\partial f_1}{\partial y}\delta y \tag{4}$$

If we write $g_1 = \frac{\partial f_1}{\partial x}\frac{\partial f_2}{\partial y} - \frac{\partial f_1}{\partial y}\frac{\partial f_2}{\partial x}$, then the increments must also satisfy:

$$\delta g_1 = -g_1(x_0, y_0) \approx \frac{\partial g_1}{\partial x}\delta x + \frac{\partial g_1}{\partial y}\delta y \tag{5}$$

Thus we solve:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial g_1}{\partial x} & \frac{\partial g_1}{\partial y} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} -f_1(x_0, y_0) \\ -g_1(x_0, y_0) \end{bmatrix} \tag{6}$$

for $(\delta x, \delta y)$ and then compute $x_1 = x_0 + \delta x$, $y_1 = y_0 + \delta y$.

We note that:

$$\frac{\partial g_1}{\partial x} = \frac{\partial^2 f_1}{\partial x^2}\frac{\partial f_2}{\partial y} + \frac{\partial f_1}{\partial x}\frac{\partial^2 f_2}{\partial x \partial y} - \frac{\partial^2 f_1}{\partial x \partial y}\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y}\frac{\partial^2 f_2}{\partial x^2} \tag{7}$$

$$\frac{\partial g_1}{\partial y} = \frac{\partial^2 f_1}{\partial x \partial y}\frac{\partial f_2}{\partial y} + \frac{\partial f_1}{\partial x}\frac{\partial^2 f_2}{\partial y^2} - \frac{\partial^2 f_1}{\partial y^2}\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y}\frac{\partial^2 f_2}{\partial x \partial y} \tag{8}$$

So that, expanding the various terms:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \left(\frac{\partial^2 f_1}{\partial x^2}\frac{\partial f_2}{\partial y} + \frac{\partial f_1}{\partial x}\frac{\partial^2 f_2}{\partial x \partial y} - \frac{\partial^2 f_1}{\partial x \partial y}\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y}\frac{\partial^2 f_2}{\partial x^2}\right) & \left(\frac{\partial^2 f_1}{\partial x \partial y}\frac{\partial f_2}{\partial y} + \frac{\partial f_1}{\partial x}\frac{\partial^2 f_2}{\partial y^2} - \frac{\partial^2 f_1}{\partial y^2}\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y}\frac{\partial^2 f_2}{\partial x \partial y}\right) \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} -f_1 \\ \left(-\frac{\partial f_1}{\partial x}\frac{\partial f_2}{\partial y} + \frac{\partial f_1}{\partial y}\frac{\partial f_2}{\partial x}\right) \end{bmatrix} \tag{9}$$

The iteration can be continued until either (a) the absolute values of f_1 and g_1 are below some tolerance, or (b) the magnitude of the increment $(\delta x, \delta y)$ is below some tolerance. The efficiency of the iteration can be improved in some cases by adopting acceleration (or deceleration) techniques which involve applying a factor to the increment $(\delta x, \delta y)$. The iteration is also sensitive to the starting point, which must be chosen sensibly.

Note that it is important that:

- The forms of f_1 and f_2 are such that the Newton–Raphson procedure converges.
- The initial choice of (x_0, y_0) is such that the solution obtained is the minimum value of f_2 and not the maximum value (which would correspond to the point on particle 1 that is "furthest" from particle 2).

A typical solution for point P_1 is shown in Fig. 6.

Having found a solution (x_1, y_1), if $f_2(x_1, y_1) > 0$, then the particles are not in contact, as illustrated in Fig. 6. If $f_2(x_1, y_1) = 0$, they are just touching but not overlapping. If $f_2(x_1, y_1) < 0$, they are overlapping, as illustrated in Fig. 7. In this last case we need to carry out a second computation, interchanging the roles of particles 1 and 2 above to find a point P_2 at coordinates (x_2, y_2) on particle 2 which is "closest" to particle 1 (this is the point that minimises f_1 subject to $f_2 = 0$). The value $f_1(x_2, y_2)$ must in this case always be negative.

The "overlap distance" between the particles may be defined as $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. The actual apparent contact point can be determined as the weighted mean of points P_1 and P_2, where the weighting factors may depend on the relative stiffnesses of the particles. Note that other possible definitions of the interpenetration of the particles could be made, but this simple definition proves to be sufficient for most purposes.

The direction of the normal (from particle 1 to particle 2) at the contact point is also required in most DEM codes. It can be evaluated by a number of means:

1. The direction of the vector $(x_1 - x_2, y_1 - y_2)$.
2. The local normal to particle 1 at (x_1, y_1), which has direction $\left(\frac{\partial f_1}{\partial x}, \frac{\partial f_1}{\partial y}\right)$ at this point.
3. (Minus) the local normal to the function $f_2 = const$ at (x_1, y_1), *i.e.* the direction $\left(-\frac{\partial f_2}{\partial x}, -\frac{\partial f_2}{\partial y}\right)$ at this point.
4. (Minus) the local normal to particle 2 at (x_2, y_2), which has direction $\left(-\frac{\partial f_2}{\partial x}, -\frac{\partial f_2}{\partial y}\right)$ at this point.
5. The local normal to the function $f_1 = const$ at (x_2, y_2), *i.e.* the direction $\left(\frac{\partial f_1}{\partial x}, \frac{\partial f_1}{\partial y}\right)$ at this point.

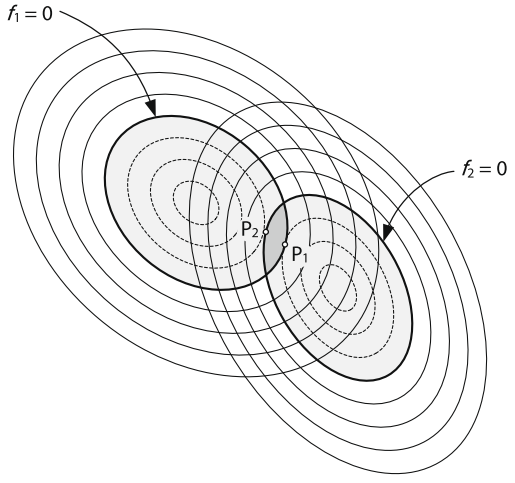

Fig. 7. Two potential particles in contact.

As the two points approach each other for very small overlaps, the first method becomes ill conditioned, so we explore further the remaining four methods. Examining the second and third methods, we observe that a vector at right angles to the third criterion would have direction $\left(\frac{\partial f_2}{\partial y}, -\frac{\partial f_2}{\partial x}\right)$. Taking the dot product of this vector with that derived from method 2 we obtain $\frac{\partial f_1}{\partial x}\frac{\partial f_2}{\partial y} - \frac{\partial f_1}{\partial y}\frac{\partial f_2}{\partial x}$, which is of course zero by the condition that $g_1 = 0$. Thus the second and third criteria give the same vector (to within the tolerance on the solution). Similarly the fourth and fifth criteria give the same vector. The 2, 3 or 4, 5 methods differ in that the normals are evaluated at different points.

It is suggested that the direction of the normal is therefore best evaluated by averaging in some way the 2, 3 and 4, 5 methods. A (weighted) average of methods 2 and 4 may be the most appropriate.

3.2. Alternative approach for contact detection

Provided that f_1 and f_2 are strictly convex functions, an alternative approach is possible. Instead of seeking P_1 one could seek first P_3, at coordinates (x_3, y_3) a point "midway" between the particles and "closest" to both (also shown in Fig. 6). This would be at $f_1 = f_2$, and minimising $f_1 + f_2$. This leads to minimisation of $f_1 + f_2 + \Lambda(f_1 - f_2)$ subject to $f_1 - f_2 = 0$. We obtain:

$$\frac{\partial f_1}{\partial x} + \frac{\partial f_2}{\partial x} + \Lambda\frac{\partial f_1}{\partial x} - \Lambda\frac{\partial f_2}{\partial x} = 0 \tag{10}$$

$$\frac{\partial f_1}{\partial y} + \frac{\partial f_2}{\partial y} + \Lambda\frac{\partial f_1}{\partial y} - \Lambda\frac{\partial f_2}{\partial y} = 0 \tag{11}$$

And eliminating the Lagrangian multiplier gives:

$$\left(\frac{\partial f_1}{\partial x} + \frac{\partial f_2}{\partial x}\right)\left(\frac{\partial f_1}{\partial y} - \frac{\partial f_2}{\partial y}\right) - \left(\frac{\partial f_1}{\partial y} + \frac{\partial f_2}{\partial y}\right)\left(\frac{\partial f_1}{\partial x} - \frac{\partial f_2}{\partial x}\right) = 2\left(\frac{\partial f_2}{\partial x}\frac{\partial f_1}{\partial y} - \frac{\partial f_1}{\partial x}\frac{\partial f_2}{\partial y}\right) = 0 \tag{12}$$

i.e. the same condition as Eq. (3). This time though we solve this with $f_1 - f_2 = 0$. If $f_1(x_3, y_3)$ is negative then the two particles are in contact, and points P_1 and P_2 must be determined as before to calculate the overlap distance. The notional contact point may rationally be taken as P_3, and the contact normal determined from $\left(\frac{\partial f_1}{\partial x}, \frac{\partial f_1}{\partial y}\right)$ at this point.

4. Definition of potential particles

A very wide range of particle shapes can be defined by the above method, subject only to the constraint that they must be convex. Some simple forms of f to define potential particles would be:

- $f = x^2 + y^2 - a^2$ a circle of radius a
- $f = \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1$ an ellipse with axes a, b
- $f = \frac{|x|^n}{a^n} + \frac{|y|^n}{b^n} - 1, n > 1$ a superellipse with axes a, b.

More complex shapes of particles can be built up by the following procedure. A line in two dimensions may be defined by $f = ax + by - d = 0$, where d is the distance of the line from the origin and (a, b) the direction cosines of the normal to the line. The coefficients should be normalised so that $a^2 + b^2 = 1$. The Cartesian distance from the line is simply given by the value of f, with the sign indicating the side of the line (positive values being in the direction of the normal).

A function which is positive one side of a line (and indicates the distance from it), but zero the other side would be $f = \langle ax + by - d\rangle$, where $\langle\rangle$ indicate Macaulay brackets ($\langle x\rangle = x, x > 0; \langle x\rangle = 0, x \leqslant 0$).

Thus a convex particle can first be defined in terms of a suitable number of lines (at least 3). Such a particle is shown in Fig. 8, with the particle itself shaded and contours of the function defining each individual surface shown.

If the function for the particle is defined as $f = \sum_{i=1}^{n} \langle a_i x + b_i y - d_i\rangle$, the value of this function is zero "within" the particle, and increases linearly with distance from any surface. The effect of the summation in the definition of the function is such that at the corners of the particle two terms in the summation (corresponding to the adjacent planes) contribute to f, with the effect of "chamfering" each corner by an additional line (so that a 4-sided particle becomes 8-sided *etc.*). The contours of f outside the particle are as illustrated in Fig. 9, in which the "chamfer" sections of the contours for which two of the planes contribute to the value of f are shown bold.

Rounded instead of chamfered corners may instead be achieved by adopting $f = \sum_{i=1}^{n} \langle a_i x + b_i y - d_i\rangle^2$, in which case the "corners" outside the original particle become parts of ellipses, as shown in Fig. 10a.

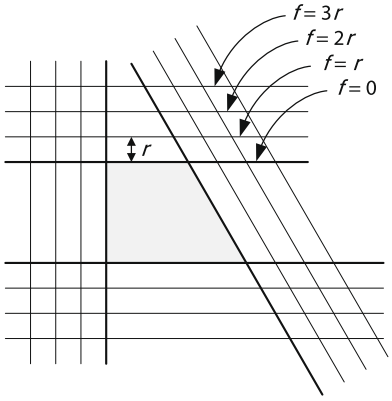

Fig. 8. A polygonal particle defined by a series of planes.

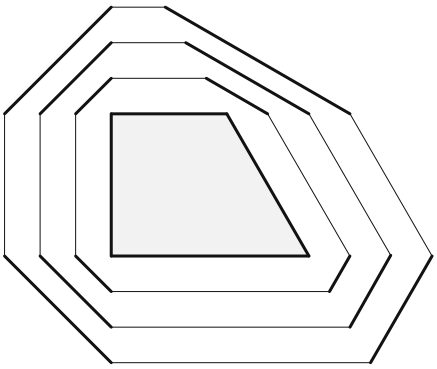

Fig. 9. Contours of function using linear summation.

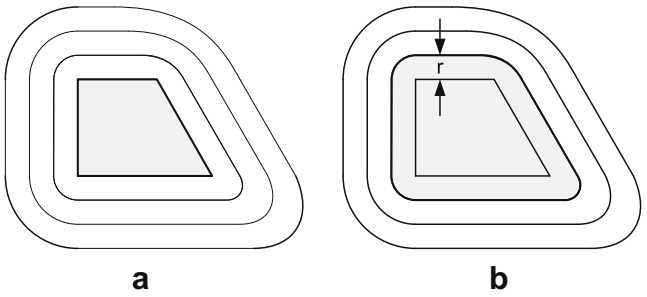

Fig. 10. (a) Contours of function using quadratic summation and (b) expanded particle with rounded corners.

The particle itself may be rounded by defining the function as $f = \sum_{i=1}^{n} \langle a_i x + b_i y - d_i\rangle^2 - r^2$, in which r is the distance by which the particle is "expanded", and related to the radius of curvature at the corners (and equal to the radius of curvature where two faces meet at right angles). The value of the function now becomes negative within the particle, taking a constant value $-r^2$ within the core defined by the original planes. The new particle is shown shaded in Fig. 10b.

The remaining problem with the above definition is that the particle is convex, but not strictly convex (*i.e.* it includes flat faces). These may be removed by defining a "shadow" circular particle $f = x^2 + y^2 - R^2$, and then combining this with the original particle:

$$f = (1-k)\left(\sum_{i=1}^{n} \langle a_i x + b_i y - d_i\rangle^2 - r^2\right) + k\left(x^2 + y^2 - R^2\right) \tag{13}$$

in which $0 < k \ll 1$ is a coefficient which determines the curvature of the faces of the final particle. For computational purposes it is convenient to normalise the function in the form:

$$f = (1-k)\left(\frac{\sum_{i=1}^{n} \langle a_i x + b_i y - d_i\rangle^2}{r^2} - 1\right) + k\left(\frac{x^2}{R^2} + \frac{y^2}{R^2} - 1\right) \tag{14}$$

and derive also

$$\frac{\partial f}{\partial x} = \frac{2(1-k)}{r^2}\left(\sum_{i=1}^{n} a_i \langle a_i x + b_i y - d_i\rangle\right) + \frac{2k}{R^2}x \tag{15}$$

$$\frac{\partial f}{\partial y} = \frac{2(1-k)}{r^2}\left(\sum_{i=1}^{n} b_i \langle a_i x + b_i y - d_i\rangle\right) + \frac{2k}{R^2}y \tag{16}$$

$$\frac{\partial^2 f}{\partial x^2} = \frac{2(1-k)}{r^2}\left(\sum_{i=1}^{n} a_i^2 H(a_i x + b_i y - d_i)\right) + \frac{2k}{R^2} \tag{17}$$

$$\frac{\partial^2 f}{\partial x \partial y} = \frac{2(1-k)}{r^2}\left(\sum_{i=1}^{n} a_i b_i H(a_i x + b_i y - d_i)\right) \tag{18}$$

$$\frac{\partial^2 f}{\partial y^2} = \frac{2(1-k)}{r^2}\left(\sum_{i=1}^{n} b_i^2 H(a_i x + b_i y - d_i)\right) + \frac{2k}{R^2} \tag{19}$$

where we make use of the Heaviside step function $H(x) = 1,\ x > 0$; $H(x) = 0,\ x \leqslant 0$.

Note that of course that each particle is originally defined in terms of the local (x, y) coordinates as shown in Fig. 11a. When each particle is translated and rotated the global coordinates must first be transformed back to the local coordinates before the functions are evaluated. If the particle reference point is at (x_0, y_0) and it is rotated by θ anticlockwise, as shown in Fig. 11b, then first define:

$$x' = x - x_0 \tag{20}$$

$$y' = y - y_0 \tag{21}$$

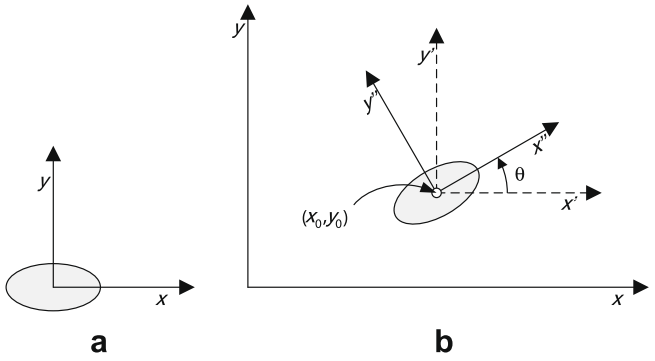

Fig. 11. Potential particle in (a) reference position and (b) current position.

then define:

$$x'' = x'\cos\theta + y'\sin\theta \tag{22}$$

$$y'' = -x'\sin\theta + y'\cos\theta \tag{23}$$

and evaluate f and its derivatives as functions of the local coordinates (x'', y''). Then note that:

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial x''}\frac{\partial x''}{\partial x} + \frac{\partial f}{\partial y''}\frac{\partial y''}{\partial x} = \frac{\partial f}{\partial x''}\cos\theta + \frac{\partial f}{\partial y''}\sin\theta \tag{24}$$

$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial x''}\frac{\partial x''}{\partial y} + \frac{\partial f}{\partial y''}\frac{\partial y''}{\partial y} = -\frac{\partial f}{\partial x''}\sin\theta + \frac{\partial f}{\partial y''}\cos\theta \tag{25}$$

$$\begin{aligned}\frac{\partial^2 f}{\partial x^2} &= \frac{\partial^2 f}{\partial x''^2}\frac{\partial x''}{\partial x}\frac{\partial x''}{\partial x} + \frac{\partial^2 f}{\partial x''\partial y''}\frac{\partial x''}{\partial x}\frac{\partial y''}{\partial x} + \frac{\partial^2 f}{\partial x''\partial y''}\frac{\partial x''}{\partial x}\frac{\partial y''}{\partial x}\\ &\quad + \frac{\partial^2 f}{\partial y''^2}\frac{\partial y''}{\partial x}\frac{\partial y''}{\partial x} = \frac{\partial^2 f}{\partial x''^2}\cos^2\theta - 2\frac{\partial^2 f}{\partial x''\partial y''}\sin\theta\cos\theta\\ &\quad + \frac{\partial^2 f}{\partial y''^2}\sin^2\theta\end{aligned} \tag{26}$$

$$\begin{aligned}\frac{\partial^2 f}{\partial x\partial y} &= \frac{\partial^2 f}{\partial x''^2}\frac{\partial x''}{\partial x}\frac{\partial x''}{\partial y} + \frac{\partial^2 f}{\partial x''\partial y''}\frac{\partial x''}{\partial x}\frac{\partial y''}{\partial y} + \frac{\partial^2 f}{\partial x''\partial y''}\frac{\partial x''}{\partial y}\frac{\partial y''}{\partial x}\\ &\quad + \frac{\partial^2 f}{\partial y''^2}\frac{\partial y''}{\partial y}\frac{\partial y''}{\partial x} = \frac{\partial^2 f}{\partial x''^2}\sin\theta\cos\theta + \frac{\partial^2 f}{\partial x''\partial y''}\left(\cos^2\theta - \sin^2\theta\right)\\ &\quad - \frac{\partial^2 f}{\partial y''^2}\sin\theta\cos\theta\end{aligned} \tag{27}$$

$$\begin{aligned}\frac{\partial^2 f}{\partial y^2} &= \frac{\partial^2 f}{\partial x''^2}\frac{\partial x''}{\partial y}\frac{\partial x''}{\partial y} + \frac{\partial^2 f}{\partial x''\partial y''}\frac{\partial x''}{\partial y}\frac{\partial y''}{\partial y} + \frac{\partial^2 f}{\partial x''\partial y''}\frac{\partial x''}{\partial y}\frac{\partial y''}{\partial y}\\ &\quad + \frac{\partial^2 f}{\partial y''^2}\frac{\partial y''}{\partial y}\frac{\partial y''}{\partial y} = \frac{\partial^2 f}{\partial x''^2}\sin^2\theta + 2\frac{\partial^2 f}{\partial x''\partial y''}\sin\theta\cos\theta\\ &\quad + \frac{\partial^2 f}{\partial y''^2}\cos^2\theta\end{aligned} \tag{28}$$

Although the function itself is smooth, the second derivatives are discontinuous, because of the piecewise definition of the potential. This can cause some problems with the Newton–Raphson iteration if not treated carefully in a numerical analysis.

5. Discussion

Fig. 12 shows an example of an assembly of 100 potential particles of three different shapes sedimented into a hopper under gravity. The particles are much more angular than those shown in Fig. 4, but the angularity is not an essential feature of the potential particles. Either technique can be used to generate more or less angular particles. By comparison with the equivalent analysis with circular particles this analysis ran about 14.8 times slower (*i.e.* 1.25 times slower than the modified racetrack particles). The run time depends, however, significantly on the efficiency of the Newton–Raphson algorithm – specifically on the choice of starting point and whether any acceleration techniques are used. The current algorithm is not fully optimised, so some speed gain can be expected in future.

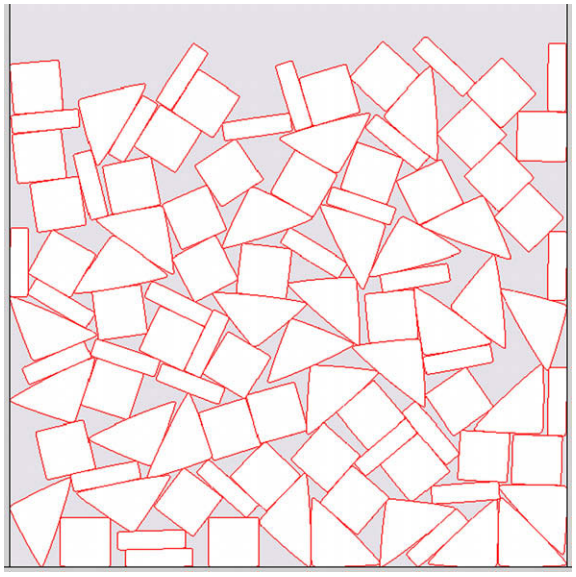

Fig. 12. An assembly of potential particles.

Potential particles may be used to define a very wide range of shapes. For instance the particular form of the potential, as described above, can be used to define particles that are roughly polygonal, with a controlled amount of curvature both at the corners and along the sides. Their real advantage is, however, realised when the method is extended to three dimensions. The contact detection and overlap calculations are only slightly more complex in 3D than in 2D, and special treatment of particular cases is unnecessary.

Note that if concave particles are required, the potential particle method can readily be combined with Option 2 (clusters, agglomerates or "molecules", discussed above). A cluster of a small number of potential particles could be used to create a realistic particle including concavities. This approach has advantages in that it avoids many of the difficulties in detecting multiple contacts between particles, and the advantage of the potential particle approach in this context would be that rather fewer particles would be needed in the cluster.

6. Conclusions

Techniques for defining non-circular particles in DEM have been addressed. Although the "modified racetrack" method is promising in 2D it is would be too complex in 3D. An alternative is the "potential particle" approach. The overall method has been described, and a specific form of the potential given which allows definition of roughly polygonal particles. The method is equally applicable in two and three dimensions.

Acknowledgements

Much of the programming for "racetrack particles" was conducted by E. Crammond. Development work on "potential particles", including 3D applications, was carried out by S. Hookham, supported by an EPSRC Vacation Bursary.

References

[1] Crammond E. Squaring the circle: arbitrarily shaped particle modelling in distinct element analysis, M. Eng. Project report, Dept. of Engineering Science, Oxford University; 2007.
[2] Jensen RP, Bosscher PJ, Plesha ME, Edil TB. DEM simulation of granular media – structure interface: effects of surface roughness and particle shape. Int J Numer Anal Methods Geomech 1999;23(6):531–47.
[3] Latham J-P, Lu Y, Munjiza A. A random method for simulating loose packs of angular particles using tetrahedra. Géotechnique 2001;51(10):871–9.
[4] Latham J-P, Munjiza A. The modelling of particle systems with real shapes. Philos Trans Roy Soc London, Math Phys Eng Sci 2004;362(1822): 1953–72.
[5] Shi G. Discontinuous deformation analysis – a new numerical model for the statics and dynamics of block systems. PhD thesis, University of California, Berkeley; 1988.
[6] Wu CY, Cocks ACF. Numerical and experimental investigations of the flow of powder into a confined space. Mech Mater 2006;38(4):304–24.
[7] Yamane K, Nakagawa M, Altobelli SA, Tanaka T, Tsuji Y. Steady particulate flows in a horizontal rotating cylinder. Phys Fluids 1998;10:1419–27.