# Computer Simulations of Multiparticle-Contacts Dynamics

Jacek S. Leszczynski

Technical University of Czestochowa, Institute of Mathematics & Computer Science,
ul. Dabrowskiego 73, 42-200 Czestochowa, Poland
`jale@k2.pcz.czest.pl`

**Abstract.** We considered the complex problem of how to simulate dynamics of multiparticle contacts under the molecular dynamics method. The understanding of interaction process is therefore crucial in order to develop theoretical studies and also to perform simulations of motion of a granular material. In opposite to binary collisions, where several contacts between particles are independent, multiparticle contacts depend on some history including several two-particle contacts. To solve this problem we applied fractional interaction law, where fractional derivatives accumulate the whole history of the function in weighted form. We proposed a novel algorithm which allows to perform calculations for an arbitrary form of multiparticle contacts.

**Keywords:** Molecular dynamics method, multiparticle contacts, fractional derivative, Caputo derivative

## 1 Introduction

Flows of granular materials have generated much interest in a lot of industrial processes involving transport of particles and in natural phenomena. The key aspect in such media is how to simulate particle contacts that may eventually take place between particles. The contact process is responsible for the dissipation and transfer of energy in a granular material. An understanding the collision process is crucial in order to develop simulations and theoretical studies. Computer simulations have turned out to be a useful tool that to investigate the physics of a flow of granular materials because there is no generally accepted theory of the flow so far, and experimental difficulties are noted. In all the considered flows the contact processes are characterised through their collisional time $t_c$. The infinitesimally short time is responsible for binary contacts. When $t_c$ increases and tends toward to infinity we observe multiparticle contacts, non-permanent cohesion in multiparticle contacts and permanent cohesion. Consequently simulations such a flow are limited by the assumptions concerning contact processes. Continuum models [3] based on binary collisions between particles. Simulations including these models do not consider the discrete nature and anisotropic properties of contacting particles. Nevertheless, two discrete approaches exist on that matter. The first one leads to molecular dynamics [4] which assume an expression

for the repulsive force acting between a pair of contacting particles. The second leads to event driven models [1] in which introduces instantaneously change the direction and values of the velocities according to conservation equations each time a contact occurs. The basic difference between event driven and molecular dynamics methods is the contact time $t_c$ between colliding bodies. In the event driven method the time $t_c$ is ideally zero. This is quite different from the molecular dynamics, for which $t_c$ is greater than zero and it depends on parameters describing structure of a granular material. However, molecular dynamics technique underestimate energy dissipation in multiparticle contacts [8,12] but in the event driven method the inelastic collapse [9] occurs. In this paper we will focus on the molecular dynamics technique. Particularly, we will simulate multiparticle contacts. Multiparticle collisions occur when an individual particle contacts with neighbouring particles so that those contacts have a direct influence on each other. Basic interaction laws of the repulsive force [2,5,14] are formulated under assumption that two particles collide. Therefore there are valid for particle collisions which are independent from one another and there are not suitable in dynamics of arbitrary multiparticle contacts. Fractional interaction law [7] used in the molecular dynamics technique extend our considerations for multiparticle contacts. Therefore we concentrate only on the fractional law, where within the multiparticle contacts we take into account direct influence of a two-particle contact to other ones.

## 2    Problem Statement

We examine a set of spherical particles moving under optional extortion. Following on results presented in [7] we consider only central collisions, therefore we neglect particle rotation and tangential forces acting on a particle. Such limits allow us to construct an algorithm without loss its generality. However, our simulation does not reflect motion of a real particulate system and requires additional modelling in physical sense. Taking into account above assumptions, we can describe a particle through its characteristic radius $r_i$, mass $m_i$, position $\mathbf{x}_i$ and linear velocity $\dot{\mathbf{x}}_i$, for $i = 1, \ldots, np$, where $np$ is the total number of considered particles. Let us turn our attention to motion of particles without collisions shown in Fig. 1a or the motion of particles with multiparticle contacts as presented on Fig. 1b. We focus on a particle $i$ which collides with neighbouring particles $j(i)$. We cannot exclude a situation that the surrounding particles contact with other ones. Therefore Fig. 1b presents a group of colliding particles moving as one body. We can simulate the body's dynamics through the exchange of particles within the body. A set of simultaneous equations describing motion of a particle $i$ is written as

$$\begin{cases} m_i \cdot \ddot{\mathbf{x}}_i = \sum_l \mathbf{F}_l - \text{without contacts} \\ m_i \cdot \ddot{\mathbf{x}}_i + \sum_{j(i),\, j(i) \neq i} \mathbf{P}_{j(i)} = \sum_l \mathbf{F}_l - \text{with multiparticle contacts} \end{cases}, \quad (1)$$
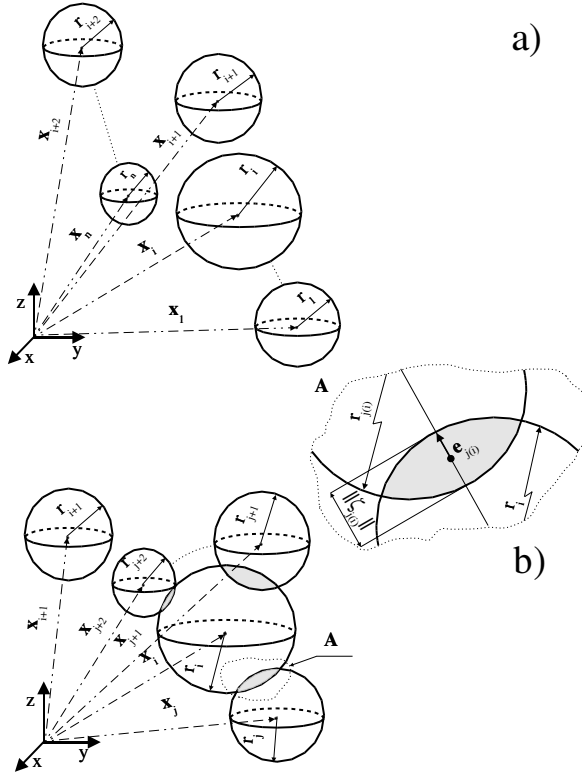
**Fig. 1.** Illustration of particles behaviour: a) without contacts, b) with multiparticle contacts.

where $\mathbf{F}_l$ is as optional force and $\mathbf{P}_{j(i)}$ is a repulsive force acting between a pair of contacting particles in a normal direction. The repulsive force is defined according to molecular dynamics models [1,4]. In such models particles virtually overlap when a contact occurs. We define the overlap of two particles experiencing a contact in the following form (see also detail A in Fig. 1b)

$$\left\| \boldsymbol{\zeta}_{j(i)} \right\| = r_{j(i)} + r_i - \left\| \mathbf{x}_{j(i)} - \mathbf{x}_i \right\|, \tag{2}$$

and a unit vector $\mathbf{e}_{j(i)}$ operating in a normal direction which connects the particles' centres of mass pointing from $i$ to $j(i)$ is defined as follows

$$\mathbf{e}_{j(i)} = \frac{\mathbf{x}_{j(i)} - \mathbf{x}_i}{\left\| \mathbf{x}_{j(i)} - \mathbf{x}_i \right\|}. \tag{3}$$

From Eqs (2) and (3) we have a vector of the normal overlap as

$$\boldsymbol{\zeta}_{j(i)} = \left( r_{j(i)} + r_i \right) \cdot \mathbf{e}_{j(i)} - \left( \mathbf{x}_{j(i)} - \mathbf{x}_i \cdot \right). \tag{4}$$

According to [7] we formulate the repulsive force in the following form

$$\mathbf{P}_{j(i)} = c_{j(i)}^{\alpha_{j(i)}} \cdot k_{j(i)}^{1-\alpha_{j(i)}} \cdot {}_{t_{j(i)}^*} D_{t_{j(i)}}^{\alpha_{j(i)}} \left( \boldsymbol{\zeta}_{j(i)} \right), \tag{5}$$

where $c_{j(i)}$ and $k_{j(i)}$ are physical properties of contacting surfaces, $\alpha_{j(i)}$ is a real order of differentiation which belongs to the range $\alpha_{j(i)} \in \langle 0 \dots 1 \rangle$, ${}_{t_{j(i)}^*} D_{t_{j(i)}}^{\alpha_{j(i)}} (\cdot)$ is a differential operator of the fractional order $\alpha_{j(i)}$, $t_{j(i)} - t_{j(i)}^*$ indicates the contact time $t_c$ between a pair of colliding particles. We introduce a definition of the fractional operator [10] as

$$_{a}D_t^\alpha f(t) = \sum_{l=0}^{n-1} \frac{(t-a)^{l-\alpha}}{\Gamma(l-\alpha+1)} \cdot f^{(l)}(a) + {}_{a}^{C}D_t^\alpha f(t), \tag{6}$$

where $_{a}D_t^\alpha f(t)$ denotes the Caputo derivative [10]

$$^{C}D_t^\alpha f(t) = \begin{cases} \frac{1}{\Gamma(n-\alpha)} \cdot \int_a^t \frac{\frac{d^n f(\tau)}{d\tau^n}}{(t-\tau)^{\alpha+1-n}} d\tau & \text{for } n-1 < \alpha < n \\ \frac{d^n}{d(t-a)^n} f(t) & \text{for } \alpha = n \end{cases}, \tag{7}$$

and $n = [\alpha] + 1$, and $[\cdot]$ is integer part of a real number. Unlike derivatives of integer order, which depend only on the local behaviour of the function, derivatives of fractional order accumulate the whole history of the function (in our case the function is represented by the overlap) in weighted form. In multiparticle contacts we need to take into consideration the sum of forces (5) over all the contacting particles. It should be noted that eqn. (5) reflects the fractional interaction law for a two particle contacts but we can extend this formula when the parameter $\alpha_{j(i)}$ varies over a number of contacting particles and becomes a common parameter for the total number of particles experiencing a contact. Some computational examples shown in [7]. Nevertheless, we need to solve a problem of how to simulate dynamics of multiparticle contacts, especially when we introduce the fractional operator (6) into formula (5) which occurs in expression (1).

## 3   Algorithm for Multi-collision Detection and Administration for the Repulsive Force-Overlap Path

Let us turn our attention to an initial value problem described by a set of the upper and lower equations in eqn. (1) taken simultaneously into account. The upper expression in eqn. (1) is connected with motion of a particle $i$ without any collisions but the lower expression in eqn. (1) is dedicated to motion of a particle $i$ which collides with other particles. Therefore we indicate a particle $i$ which can collide neighbouring particles or does not collide with any particle, for $i = 1, \dots, np$, where $np$ is the total number of particles. We need to introduce initial conditions $\mathbf{x}_i^0$, $\dot{\mathbf{x}}_i^0$, where the superscript 0 refers time $t = 0$. We also divide a time of calculations into $0 < t_0 < \dots < t_k < \dots < T$, where $T$ denotes the total time in which the calculations are considered and $h = t_k - t_{k-1}$ is the time step. For both upper and lower expressions in eqn. (1) the Runge-Kutta scheme [11] is used. However, for the lower expression in eqn. (1),

where the fractional derivative is applied, some modification of the Runge-Kutta method needs to be done. The general idea of how to solve numerically ordinary differential equations of fractional order is presented in [6]. Nevertheless, we cannot apply this idea because the lower expression in eqn. (1) is represented by a mixture of integer and fractional operators. Taking into account a fact that the parameter $\alpha$ varies between 0 and 1 we obtained integer order equals to 2 as a maximal degree of the ordinary differential equation. Therefore the fractional derivative (7) included in this equation needs to be represented by some numerical form. We introduced division of a local time $t_{j(i)}^* = t_0 < t_1 < \ldots < t_l < \ldots < t_m = t_{j(i)}$ where the fractional derivative is defined. Note that $t_{j(i)}^*$ indicates time when a collision between a pair of particles begins, $t_{j(i)}$ denotes time when the collision ends and $t_c = t_{j(i)} - t_{j(i)}^*$ is duration of a collision. The local index $m$ is the total number of divisions of the collisional time $t_c$ into time steps. Note that $m$ varies over the overlap path (2) and therefore the index $m$ is a result of direct calculations. According to [10] we use the numerical scheme for the fractional derivative (7) in the following form

$$
{}_{t_0}^{C}D_{t_m}^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha+1)} \sum_{l=0}^{m-1} B_l \cdot \left[ (t_m - t_l)^{n-\alpha} - (t_m - t_{l+1})^{n-\alpha} \right], \qquad (8)
$$

where

$$
B_l = \frac{f(t_{l+1}) - f(t_l)}{t_{l+1} - t_l} = f^{(n)}(t_l). \qquad (9)
$$

Above scheme is defined when a function $f(t)$ has a constant value in the time interval $t_l < t < t_{l+1}$. We improve formula (8) in order to assume a linear form of the function $f(t)$ and thus we have

$$
{}_{t_0}^{C}D_{t_m}^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \sum_{l=0}^{m-1} \left\{ \frac{A_l}{n-\alpha+1} \cdot \left[ (t_m - t_{l+1})^{n-\alpha+1} - (t_m - t_l)^{n-\alpha+1} \right] \right.
$$
$$
\left. - \frac{A_l \cdot t_m + B_l}{n-\alpha} \cdot \left[ (t_m - t_{l+1})^{n-\alpha} - (t_m - t_l)^{n-\alpha} \right] \right\}, \qquad (10)
$$

where

$$
A_l = \frac{f^{(n)}(t_{l+1}) - f^{(n)}(t_l)}{t_{l+1} - t_l}, \qquad B_l = f^{(n)}(t_l) - A_l \cdot \frac{t_{l+1} + t_l}{2}. \qquad (11)
$$

Formula (10) is more precise in numerical calculations of the fractional derivative than formula (8). Note that eqn. (10) becomes eqn. (8) when $A_l = 0$. This simple proof does not require any additional evidence. In this paper we neglect analysis of stability and convergence of such formulae. Some details concerning the analysis one can find in [10].

Before the governing calculations, where the set of ordinary differential equations is applied, we need to introduce an algorithm for collision detection. This algorithm has to be useful for multiparticle contacts and it needs to detect a begin of a collision and of course the collision end. The following two methods are successfully applied in collision detection. The first method, called the

Verlet-Neighbour List (VNL) [1,13], basis on an imaginary sphere which is drawn around each particle of the system. particles within the imaginary sphere are considered as neighbours and can be taken into account as potential particles colliding with the particle. For each particle a list is generated, where all colliding particles are stored. The second method, called the Linked Cell (LC) [1, 13], determines the neighbours of a particle, where the system is divided into a lattice. We chosen the Liked Cell method that to find neighbours of a particle. More details concerning for both methods one can find in [1,13]. For additional considerations of the collision detection algorithm we need to introduce some definitions.

**Definition 1.** *If within a time interval $t_k \ldots t_{k+1}$ detects a collision, the virtual overlap (2) fulfils the following conditions*

$$\left\| \boldsymbol{\zeta}_{j(i)} \left( t_k \right) \right\| \leq 0 \qquad and \qquad \left\| \boldsymbol{\zeta}_{j(i)} \left( t_{k+1} \right) \right\| \geq 0 \qquad (12)$$

*or*

$$\left\| \boldsymbol{\zeta}_{j(i)} \left( t_k \right) \right\| \geq 0 \qquad and \qquad \left\| \boldsymbol{\zeta}_{j(i)} \left( t_{k+1} \right) \right\| \leq 0. \qquad (13)$$

**Definition 2.** *In a time interval $t_k \ldots t_{k+1}$ searches for time $t^*_{j(i)}$, which issues from the begin of a collision between two particles, under a condition that the virtual overlap (2) obeys the following condition*

$$\left\| \boldsymbol{\zeta}_{j(i)} \left( t_k \right) \right\| \leq 0 \qquad and \qquad \left\| \boldsymbol{\zeta}_{j(i)} \left( t^*_{j(i)} \right) \right\| \leq \varepsilon, \qquad (14)$$

*where $\varepsilon$ denotes accuracy of calculations.*

**Definition 3.** *In a time interval $t_k \ldots t_{k+1}$ searches for time $t_{j(i)}$, which issues from the end of a collision between two particles, under a condition that the virtual overlap (2) obeys the following condition*

$$\left\| \boldsymbol{\zeta}_{j(i)} \left( t_k \right) \right\| \geq 0 \qquad and \qquad \left\| \boldsymbol{\zeta}_{j(i)} \left( t_{j(i)} \right) \right\| \leq \varepsilon, \qquad (15)$$

*where $\varepsilon$ denotes accuracy of calculations.*

Note that the condition (12) represents the begin of a collision and the condition (13) is the end of a collision between two particles. Above definitions are necessary that to construct an algorithm which detects a collision. Moreover, the definitions are useful for multiparticle contacts because a particle can contact with another one which is eventually in contact with other ones. In solving this problem, the following iterative algorithm can be used:

**Algorithm 1**

**Step 1.** *A collision is detected during a transition from $t_k$ to $t_{k+1}$ when formula (14) or (15) is filled. In this way it searches for time $t^*_{j(i)}$ which defines the collision begins or time $t_{j(i)}$ in which the collision ends.*

**Step 2.** *Perform calculations over the next time step $t_{k+1}$ for two group of particles:*
- *which do not collide - the upper formula in eqn. (1),*
- *which are under collision - the lower formula in eqn. (1).*

**Step 3.** *Calibration of previous calculations in order to find several times $t^*_{j(i)}$ when collisions begin and the times $t_{j(i)}$ when collisions end.*

Above algorithm is suitable in calculations of dynamics of multiparticle contacts. In this way the multiparticle contacts composes with a sum running after several binary contacts.
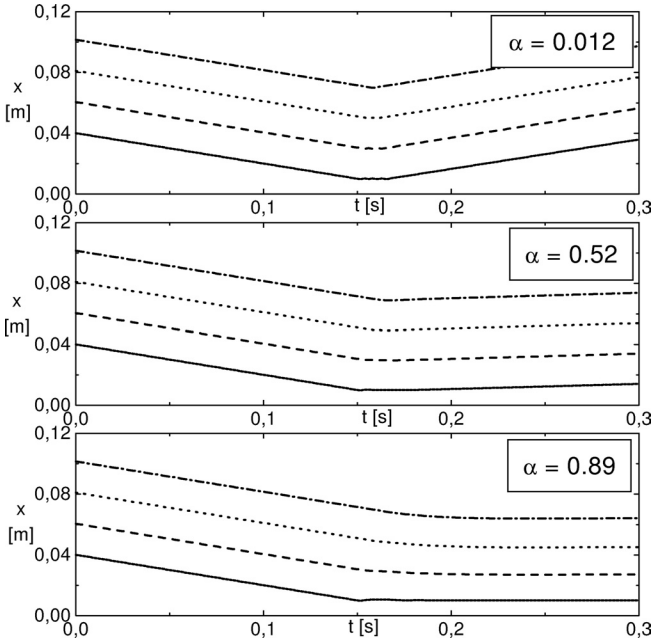


**Fig. 2.** Simulations of particle trajectories over a bottom plate depending on the parameter $\alpha$.

## 4   Examples of Simulation

In order to show how our algorithm operates during particle contacts, simulations for different physical situations have been performed. First example presents motion of a set of four identical particles vertically stacked over a bottom plate. We assume spherical particles with their characteristic parameters $r_i = 0.0014 \ m$, $m_i = 1.35 \cdot 10^{-5} \ kg$, $\dot{x}_i^0 = -0.21 \ m/s$, for $i = 1, \ldots, 4$, where the superscript 0 denotes the positions examined at time $t = 0 \ s$.

Fig. 2 shows particle displacements over time depending on the parameter $\alpha$ where initial distance between neighbouring particles assumed as $l_i = 4 \cdot 10^{-4}$ $m$, for $i = 2, \ldots, 4$. The initial distance between the first particle and the bottom plate is set to $l_1 = 0.029$ $m$. Gravity is set to 0. Coefficients $k_{j(i)}$ and $c_{j(i)}$ are calculated under formulae found in [12], where we assumed the contact time between two colliding bodies $t_c = 10^{-4}$ $s$ and the restitution coefficient $e = 0.9$. Small separation time between particles $\frac{l_i}{|\dot{x}_i^0|}$ leads to multiparticle contacts. For low values of the parameter $\alpha$ (upper chart in Fig. 2) notices that first particle located close to the bottom plate contacts twice with the bottom plate. When $\alpha$ increases up to 0.89 (lower chart in Fig. 2) the cohesion effect takes place and after collision particles stay clustered on the bottom plate.
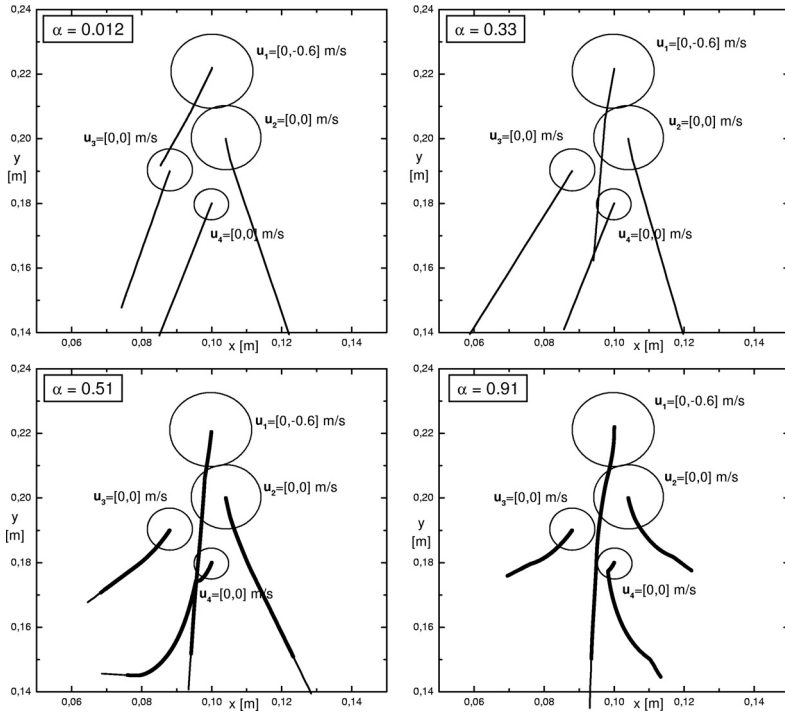


**Fig. 3.** Simulations of multiparticle contacts in two dimensional space.

The next example presents some behaviour of four particles in two dimensional space when the parameter $\alpha$ changed from 0.012 to 0.91. Note that values of coefficients and additional parameters one can find in [7]. Fig. 3 shows a situation where a particle with the initial velocity $\dot{\mathbf{x}}_1^0 = [0, \ 0.6]$ $m/s$ contacts at different moments in time with particles which initially do not move. The thin lines represents particle trajectories when particles move separately and the thick lines are common trajectories when particles move as one object. At low values

we do not notice these common trajectories and therefore binary contacts occur. When the parameter $\alpha$ increases to $\alpha = 0.91$ we can see common trajectories of two or more particles being dependent on mutual positions. We also noted different lengths of the common trajectories. This fact issues from different begins of contacting times $t^*_{j(i)}$ between two interacting particles. Therefore we can simulate multiparticle contacts where we do not assume the same collisional time. Note that the collisional time $t_c$ varies over a number of contacting particles and also varies over the parameter $\alpha$.

$\alpha = 0.1$                                               $\alpha = 0.7$



$t = 6.0\ s$                                               $t = 6.0\ s$
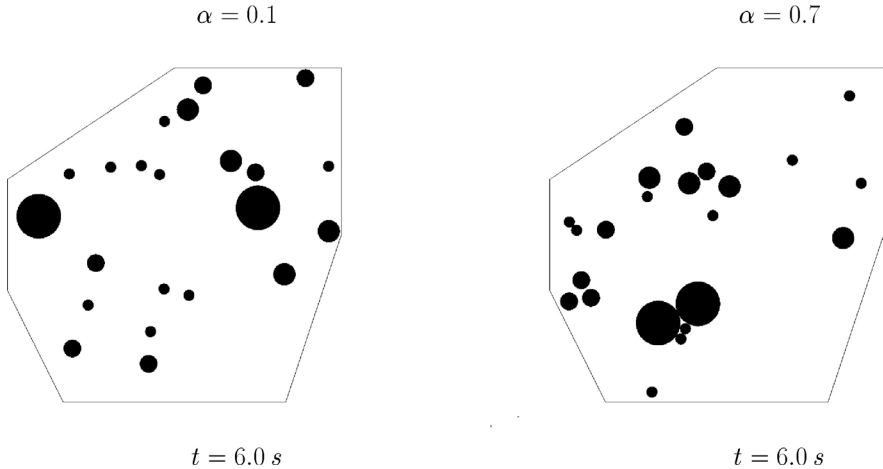
**Fig. 4.** Simulations of granular cohesion dynamics.

Fig. 4 shows the last example – simulations of particle motions in a box for different values of the parameter $\alpha$.

We assumed the same initial conditions and parameters of a structure for both $\alpha = 0.1$ and $\alpha = 0.7$. This figure presents positions of particles in time of calculations $t = 6\ s$. For a low value of the parameter $\alpha$ ($\alpha = 0.1$) binary contacts can be observed. Therefore we cannot see particle contacts after $t = 6\ s$ because during binary contacts the contact time is very low and equals to $t_c = 10^{-4}\ s$. In opposite to previous situation, when the parameter $\alpha$ increases to $\alpha = 0.7$ we notice particles cohesion. In this case increases the contact time between particles and particles stay clustered through some part of the time of calculations. In this case we are able to simulate granular cohesion dynamics where the collisional time varies over the parameter $\alpha$. We also found that $\alpha \sim (np)^{0.47}$ that to keep qualitative and quantitative agreement with the event driven technique [1,9].

## 5   Conclusions

We have examined computer simulations of arbitrary multiparticle contacts under the molecular dynamics technique. We based on the fractional interaction

law where the repulsive force is defined under fractional calculus. Note that fractional derivatives accumulate the whole history of the virtual overlap over time in weighted form. Therefore it requires novel algorithms of how to detect collisions and how to a system of ordinary differential equations involving fractional derivatives. We proposed an algorithm that to solve the system of ordinary differential equations composed from integer and fractional operators. We also applied modified collision-detection algorithm which detects a time when a collision begins and the time when the collision ends. We shown how our algorithm operates in different behaviours of particle motions. We noted that using this algorithm one can simulate dynamics of multiparticle contacts in arbitrary form as well as granular cohesion dynamics.

# References

1. Allen M.P. and Tidesley D.J.: Computer simulation of liquids, Oxford Univ. Press, New York (1989)
2. Cundall P.A. and Strack O.D.L.: A discrete numerical model for granular assemblies, Geotechnique **29** (1979) pp. 47–65
3. Gidaspow D.: Multiphase flow and fluidization. Continuum and kinetic theory descriptions, Academic Press, San Diego (1994)
4. Greenspan D.: Discrete models, Addison-Wesley, London (1973)
5. Kuwabara G., Kono K.: Restitution coefficient in a collision between two spheres, Jap. J. Appl. Phys. **26** Part 1 (1987) pp. 1230–1233
6. Leszczynski J.S.: A numerical method for solution of ordinary differential equations of fractional order, LNCS Vol. 2328, Springer-Verlag (2001) pp. 695–702
7. Leszczynski J.S.: The calculation of a normal force between multiparticle contacts using fractional operators, arXiv:physics/0209085, to appear in The Second MIT Conference on Computational Fluid and Solid Mechanics, MIT (2003); Leszczynski J.S.: A discrete model of a two-particle contact applied to cohesive granular materials, (to appear in Granular Matter, 2003)
8. Luding S. et al: Anomalous energy dissipation in molecular dynamics simulations of grains, Phys. Rev. E **50** (1994) pp. 4113–4122
9. McNamara S. and Young W.R.: Inelastic collapse and clumping in a one dimensional granular medium, Phys. Fluids A **4** (1992) pp. 496–504
10. Oldham K.B., Spanier J.: The fractional calculus. Theory and applications of differentiation and integration to arbitrary order, Academic Press, New York (1974)
11. Palczewski A.: Ordinary differential equations, WNT, Warsaw (1999) (in Polish)
12. Pournin L., Liebling Th.M.: Molecular dynamics force models for better control of energy dissipation in numerical simulations of dense granular media, Phys. Rev. E **65** (2001) pp. 011302-1-011302-7
13. Rappaport D.C.: The art of molecular dynamics simulation, Cambridge Univ. Press, Cambridge (1995)
14. Walton O.R., Braun R.L.: Viscosity, granular-temperature and stress calculations for shearing assemblies of inelastic frictional disks, J. Rheol. **30** (1986) pp. 949–980