



**HAL**  
open science

# Long time extrapolation of DEM with heat conduction in a moving granular medium

Clara Haydar, Sylvain Martin, Olivier Bonnefoy

► **To cite this version:**

Clara Haydar, Sylvain Martin, Olivier Bonnefoy. Long time extrapolation of DEM with heat conduction in a moving granular medium. *Chemical Engineering Science*, 2023, 277, pp.118815. 10.1016/j.ces.2023.118815 . emse-04093980

**HAL Id: emse-04093980**

**<https://hal-emse.ccsd.cnrs.fr/emse-04093980>**

Submitted on 10 May 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Long time extrapolation of DEM with heat conduction in a moving granular medium

Clara HAYDAR<sup>a</sup>, Sylvain MARTIN<sup>a,\*</sup>, Olivier BONNEFOY<sup>a</sup>

<sup>a</sup>*Mines Saint-Etienne, Univ Lyon, CNRS, UMR 5307 LGF, Centre SPIN, F - 42023  
Saint-Etienne France*

---

## Abstract

This paper presents a novel approach for extrapolating DEM simulations of heat transfer over a long period of time. This method is an extension of a previously published algorithm for granular motion extrapolation, introducing heat transfer. The main idea is to perform a short-term DEM simulation for one period and then apply a conductive heat transfer extrapolation algorithm. This strategy is tested over a pilot-scale rotating drum. The outcomes of standard and extrapolated DEM simulations are compared. The results are very similar while the computational time is reduced by a factor greater than 100.

*Keywords:* Granular system, DEM, heat transfer, long time extrapolation

---

## 1. Introduction

DEM<sup>1</sup> is an effective simulation tool for granular systems [Cundall & Strack \(1979\)](#). Powder rheology is guided through the contacts occurring between the particles and it is the method that takes into account the physics of collisions  
5 between particles. DEM can be applied to many industrial processes including those in the chemical, building and pharmaceutical industries [Muzzio et al. \(2004\)](#). Even with the abundance of particles-containing systems in nature and industrial processes, a fundamental understanding of heat transfer between the particles during contact is missing [Hartmanshenn et al. \(2019\)](#). However, the

---

\*sylvain.martin@emse.fr

<sup>1</sup>Discrete Element Method

comprehension of such phenomena is essential in order to predict the intrinsic properties of the particles during collision [Fry et al. \(2019\)](#). Yet, due to the complexity of the interactions between the solid particles, these models are difficult to be applied, particularly because simulating the mechanical behavior of granular materials including heat transfer can be computationally expensive [Liu et al. \(2021\)](#). Many modeling methods are proposed in the literature for simulating heat transfer in granular media. One of them is to use coarse-graining as in [Saruwatari & Nakamura \(2022\)](#) and [De et al. \(2022\)](#). This approach enables the representation of a large number of small particles by a small number of macroscopic particles, explained in details in [Peters et al. \(2023\)](#). In [Saruwatari & Nakamura \(2022\)](#), the potency of the coarse-graining technique is demonstrated for simulating heat transfer in rotary kiln reactor. Another option is to use a multi-scale modeling approach that consists of coupling a continuum method with [DEM](#) for the simulation of heat conduction in granular materials [Zhang et al. \(2011\)](#). While [Zhao et al. \(2020\)](#) coupled [FEM](#)<sup>2</sup> with [DEM](#) to simulate the thermo-mechanical response of a granular medium. A different strategy is to employ a thermal [DEM](#) approach known in literature as [TDEM](#)<sup>3</sup> as in [Feng et al. \(2008, 2009\)](#). The [TDEM](#) is considered as a variation of the [DEM](#) applied only for heat transfer problems. It enables the representation of a large granular system as a combination of different individual particles affected by different temperature boundary conditions. This enables the determination of the temperature distribution of the particles in the system over time [Kiani-Oshtorjani & Jalali \(2019\)](#). However, all of the previously stated method share a major limitation which is the high computational cost. Therefore, the aim of our work is to challenge the limits especially those linked to the reduction of time scales [Bertrand et al. \(2005\)](#). For this paper, we will shed light on the reduction of time of [DEM](#) for simulating heat transfer in granular media. Physical time is always

---

<sup>2</sup>Finite Element Method

<sup>3</sup>Thermal Discrete Element Method

considered as the biggest challenge for numericians. Knowing that the parallelizing can give a partial gain of computation for the high number of particles, yet the time variable cannot be parallelized easily. Simulating one million of particles for one hour is still complicated. In order to face this challenge, many research teams found that the idea of pseudo-periodicity for granular process can decrease drastically the duration of numerical simulation. The first idea concerning **time extrapolation** was proposed by Doucet et al. (2008) by using a statistical approach based on Markov process. Then, the team of T. Lichtenegger has introduced the notion of recurrence CFD<sup>4</sup> Pirker & Lichtenegger (2018); Lichtenegger & Pirker (2016); Lichtenegger et al. (2019). It is a method based on the Poincaré's recurrence, which allows the long time extrapolation of fluid/granular flows. Another solution based on an extrapolation algorithm has been proposed previously by a research in our lab Bednarek et al. (2019), the major advantage of which is to be very simple. The latter method was reproduced by Siegmann et al. (2021) and Bauer et al. (2022). Siegmann et al. (2021) applied this method for continuous processes to extrapolate DEM results in the pharmaceutical industry and the extrapolated results showed a good agreement with full DEM results. Whereas Bauer et al. (2022) used time extrapolation method for SPH<sup>5</sup> flow simulations over twin screw extruders. This method led to a CPU<sup>6</sup> time gain of a factor of 10<sup>5</sup> with an accurate representation of mixing kinetics. However, the main limit of the method is that it uses the pseudo periodic behavior of processes, but since most of powder processes are pseudo periodic, these models are still very useful and promising. Hence, it is possible today to extrapolate DEM simulation for long time but it concerns only processes for which there is no need to model phenomena on physical particle scale. Nevertheless, **none of the approaches are able, up to now, to take into account local phenomena on the real particle scale like heat conduc-**

---

<sup>4</sup>Computational Fluid Dynamics

<sup>5</sup>Smoothed Particle Hydrodynamics

<sup>6</sup>Central Processing Unit

tion. In his approach, [Lichtenegger et al. \(2016\)](#) proposes a newly first method that enables obtaining the transfers inside fluidized beds. However, his method does not take into account collisions, indeed the particles are represented with passive tracers [Lichtenegger \(2020\)](#). Therefore, in this paper we present a new method which consists in an extension for the existing method developed by [Bednarek et al. \(2019\)](#) to heat transfer by conduction in granular media, taking into account particle-particle collisions.

## 2. Materials and Method

In this part, the numerical method will be presented along with the heat extrapolation algorithm. We need to mention that the novel method is simple and straightforward; it consists as a foundation for further research about time extrapolation simulations.

### 2.1. DEM model

DEM is a technique that enables the simulation of the behavior of separate particles in interaction under certain circumstances. Contrary to the continuum approach, the DEM approach considers the system's particle flow as discrete particles. It is used to calculate the behavior of the particles, their positions and the forces of collisions between them [Huang & Kuo \(2018\)](#). Particle's trajectory is then obtained by Newton's second law. The DEM model in this study is based on the soft-sphere approach [Yazdani & Hashemabadi \(2020\)](#), which allows for particles overlap as well as particles-boundary overlap. The contact forces due to collisions are calculated by the non-linear spring-dashpot Hertz-Mindlin model, which is detailed in [Zhu et al. \(2007\)](#). The DEM with heat transport will be described using the following steps in the [algorithm 1](#).

A linear heat transfer model between two particles was used based on applying the one dimensional heat transfer equation by conduction. To obtain the heat transfer due to particle-particle conduction, the transport equation is influenced by many parameters such as temperature of the material, the heat

---

**Algorithm 1:** General algorithm for DEM along with heat transfer.

---

```
for  $t = 0 \rightarrow t_{max}$  do
    Establish the contact list between the particles and detect changes
    with previous one;
    For each particle and each contact, compute the force using contact
    law;
    For each particle and each contact, compute the conductive heat
    flux using Fourier's law;
    For each particle, sum all forces and heat fluxes;
    Compute the new particles temperature through energy balance;
    Compute the new particles position through Newton's 2nd law;
```

---

capacity and the heat transfer coefficient. Heat flux  $Q_{ij}$  due to particle-particle collision is calculated using Equation 1 where  $T_i$  and  $T_j$  are respectively the  
95 temperatures of the particles and  $H_{ij}$  is the conductance between the particles.

$$Q_{ij} = H_{ij}(T_i - T_j) \quad (1)$$

Modeling granular material will be achieved with the use of LIGGGHTS,<sup>7</sup> an open-source software package Kloss et al. (2012). Based on the heat equations developed in the software, the heat transfer coefficient known as the conductance between particle  $i$  and particle  $j$  will be obtained by the Equation 2 where  $K_{ij}$   
100 is the thermal conductivity and  $a_{ij}$  is the area of contact between  $i$  and  $j$ :

$$H_{ij} = \frac{4K_i K_j}{K_i + K_j} \sqrt{a_{ij}} \quad (2)$$

And since the particles taken are identical  $K_i$  is equal to  $K_j$ , hence Equation 1 is simplified to Equation 3:

$$Q_{ij} = 2K_{ij} \sqrt{a_{ij}} (T_i - T_j) \quad (3)$$

---

<sup>7</sup>LAMMPS Improved for General Granular and Granular Heat Transfer Simulations

On the other hand, the temperature change for a particle  $i$  can be calculated using Equation 4 where  $\sum(Q_i)$  is the sum of all heat fluxes involving particle  $i$  and  $\rho_i C_i V_i$  is the thermal capacity of particle  $i$  as in Peng et al. (2020).

$$\frac{dT_i}{dt} = \frac{\sum(Q_i)}{\rho_i C_i V_i} \quad (4)$$

The following simple ordinary differential equation represents the final equation that describes the evolution of the temperature for a given particle  $i$  in contact with different particles  $j$ :

$$\frac{dT_i}{dt} = \frac{\sum_j 2K_{ij} \sqrt{a_{ij}} (T_i - T_j)}{\rho_i C_i V_i} \quad (5)$$

For our case of study,  $Biot^* = \frac{2K_{ij}r_c}{K_{ij}A_p/r_p} = \frac{2r_c}{\pi r_p}$ , with  $r_c$  is the contact radius between the two particles,  $A_p$  is the cross-sectional area of the particle and  $r_p$  is the particle's radius. The dimensionless group  $Biot^*$  refers to the ratio of heat conduction transferred from one particle to the other over the heat conduction inside the particle as in Gui et al. (2013) and Vargas & McCarthy (2001).  $r_c = r_p - 0.5 * \Delta_n$  is obtained using LIGGGHTS. Since  $Biot^* \ll 1$ , thus the heat conduction within the particle can be neglected in comparison with the conduction between the particles Beaulieu et al. (2021); Kiani-Oshtorjani et al. (2022).

The newly presented approach is for heat transport extrapolation in granular systems. The approach is an extension of the pairing algorithm proposed by Bednarek et al. (2019). More precisely, it consists of a combination between the pairing algorithm along with an implementation of scalar transport for conductive heat transfer. The pairing algorithm enables the extrapolation of granular motion for DEM results, at a very low computational cost, from one period of time over a longer period for any pseudo-periodic process. The previous method is explained in details in Bednarek et al. (2019). The new method is based on the idea that whatever happening during a period will be recurring. This mean that using the data obtained from the normal DEM simulation for one period on the microscopic level will enable the acquisition of what is happening over a long period by the means of paired particles. The details to get the paired

particles are found in [Bednarek et al. \(2019\)](#), but physically speaking the paired particles refers to the particles connected by a relation which is, here, a bijective function  $f_{ext}$ .

$$f_{ext} = \left| \begin{array}{l} [1 \rightarrow N] \rightarrow [1 \rightarrow N] \\ j \rightarrow i = f_{ext}(j) \end{array} \right. \quad (6)$$

The function  $f_{ext}$  minimizes the sum of the euclidean distance between the paired particles. If particle  $i$  is paired with particle  $j$  then the position of  $i$  at  $t_2$  is obtained by the following equation:  $\vec{x}_i(t_2) = \vec{x}_j(t_1)$ .

115 Some assumptions are required for the implementation of this method:

- The system studied should reach a state of pseudo-periodicity which is described in [Siegmann et al. \(2021\)](#). However, since many processes are pseudo-periodic, the application of the method is allowed for a wide range of systems.
- 120 • The particle's size is identical [Bednarek et al. \(2019\)](#)
- The void between the particles which, in real systems, is filled with an interstitial fluid mostly air is ignored [Chaudhuri et al. \(2006\)](#)
- The collisions are treated by apparition order which means that the simultaneous contacts of a particle are treated one after the other and the collisions detected at the same time are also processed one after the other
- 125 as in the output from the [DEM](#) simulation.

The first two previously stated conditions are common with the pairing algorithm and this is due to the use of the same algorithm as a basis of our work. However all of these conditions are soft conditions since they do not affect the execution of the extrapolation algorithm. It should be noted that in this preliminary study, the aim is to show the ability of the new algorithm to speed-up granular flow simulations. Hence for simplicity sake, collision area is taken as a constant parameter  $a_{ij}$  but the code already includes the calculation of  $a_{ij}$  from standard [DEM](#) results.

130



135 This section will be divided into 3 parts as in the flowchart [Figure 1](#) that refers to the workflow.

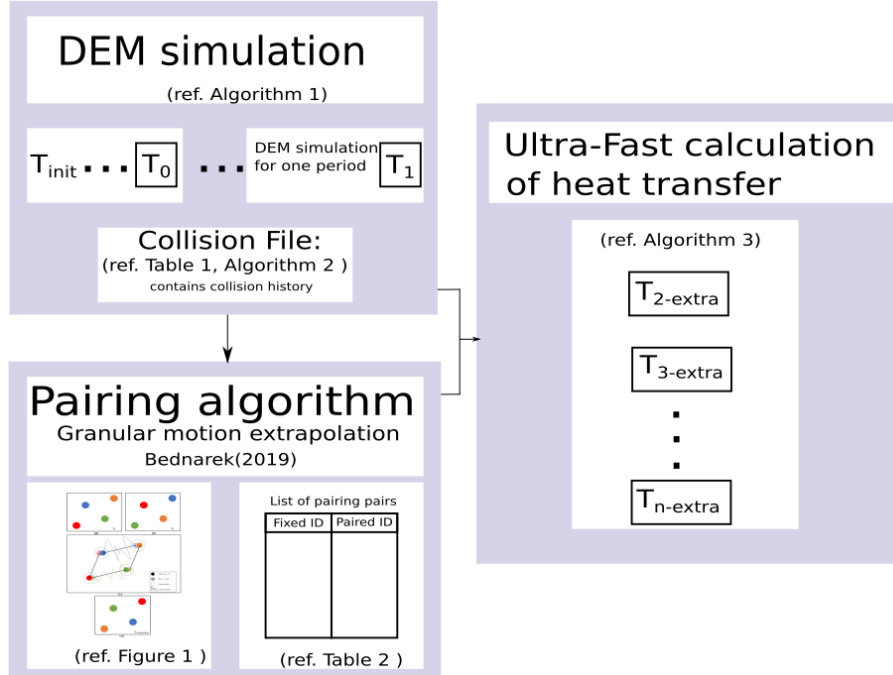


Figure 1: Flowchart of heat extrapolation algorithm.

### 2.1.1. DEM data

DEM simulations are performed until  $t_1$ . It is significant to point out that the DEM simulation should reach a steady state at  $t_0$ . In the framework of the extrapolation, all the collisions between  $t_0$  and  $t_1$  are saved and sorted in  
 140 apparition order. The list of contacts along with its duration and the area of collisions are saved in a table. The final structure of the contact list table is stocked as shown in [Table 1](#).

Since the collision properties between  $t_0$  and  $t_1$  are stored, high memory is  
 145 required. To solve this limitation, collision sampling is employed. Two parameters are introduced; the first is  $\Delta t_c^{max}$ , the maximum collision time in seconds, which is a maximum imposed collision duration for two particles in contact.

Table 1: Structure of the saved information of the extrapolation for the collisions.

| Index               | Fixed ID | Fixed ID | $t_{init}$   | $m$ | $\sqrt{(a_{ij})}$ |
|---------------------|----------|----------|--|-----|-------------------|
| 1                   | 5        | 8        | $t_0$  | 10. | $8.8e^{-3}$       |
| 2                   | 10       | 6        | ...  | ... | ...               |
| ...                 | ...      | ...      | ...  | ... | ...               |
| $x$                 | $i$      | $j$      | starts at $t_{init(ij)}$ for $\Delta t_c = 25$ s for $\Delta t_c^{max} = 10$ s |     |                   |
| $x$                 | $i$      | $j$      | $t_{init(ij)}$   | 10. | $7.8e^{-5}$       |
| $x+1$               | $i$      | $j$      | $t_{init(ij)} + 10$  | 10. | ...               |
| $x+2$               | $i$      | $j$      | $t_{init(ij)} + 20$  | 5.  | ...               |
| $x$                 | $i$      | $j$      | starts at $t_{init(ij)}$ for $\Delta t_c = 25$ s for $\Delta t_c^{max} = 20$ s |     |                   |
| $x$                 | $i$      | $j$      | $t_{init(ij)}$   | 20. | $7.8e^{-5}$       |
| $x+1$               | $i$      | $j$      | $t_{init(ij)} + 20$  | 5.  | ...               |
| ...                 | ...      | ...      | ...  | ... | ...               |
| $nb_{collisions}-1$ | ...      | ...      | ...  | ... | ...               |
| $nb_{collisions}$   | ...      | ...      | $t_1$  | ... | ...               |

And, the second is a sampling factor  $n$ , identified using statistical analysis by showing how much time the collision pairs remain in contact. Both of the previously stated parameters aim to reduce the size of the collision table.  $\Delta t_c^{max}$  is an optimization variable that enables us to adjust the size of the output file and the times of operations executed.  $n$  enables us to conclude the timesteps' frequency, thus not saving every single timestep, instead saving every  $n$  timesteps if the minimum collision duration is  $n$  timesteps. The two lines for  $x^{th}$  collision in the Table 1 in gray color represent the same phenomenon: a collision between particles  $i$  and  $j$  that lasts for  $\Delta t_c = 25$  seconds. As it can be seen, this collision can be interpreted in two ways: if  $\Delta t_c^{max} = 10$  seconds then the output will be divided in three rows (the collision will be seen as three different collisions that last for  $10 + 10 + 5$ ), while if  $\Delta t_c^{max} = 20$  seconds, the output will be two rows (two collisions  $20 + 5$ ). This will reduce the size of the output file and the time of operations for the numerical calculation. Note that the gray line in Table 1 will not be seen in the final output file, it only serves for detailed explanation. The general algorithm for collision splitting is presented in algorithm 2. It is considered as a refining method to improve the speed of the simulations. It

165 takes an input, the total collision duration ( $\Delta t_c$ ), and compare it to the imposed collision duration ( $\Delta t_c^{max}$ ). If  $\Delta t_c$  is less than  $\Delta t_c^{max}$  then  $m$ , the single collision duration in [Table 1](#) is equal to  $\Delta t_c$ . Else if it is higher, the collision is splitted into several lines. The number of lines is equal to the floor of  $\frac{\Delta t_c}{\Delta t_c^{max}}$  (floor is the function that converts the floating number to the smaller integer  
170 (2.7 to 2)) and the last line will be the remainder of the same fraction. Eventually, the [algorithm 2](#) is an external tool that shows how the input table for the extrapolation was handled. It serves only for optimization purpose and for the seek of bigger gain and reduction of the computational time. The extrapolation algorithm can be performed without this algorithm but will need more memory  
175 and computational time.

---

**Algorithm 2:** Collision splitting algorithm .

---

```

for  $k = 1 \rightarrow nb_{collisions}$  do
  if  $\Delta t_c \leq \Delta t_c^{max}$  then
     $m = \Delta t_c$ ;
  else
    for  $i = 1 \rightarrow floor(\frac{\Delta t_c}{\Delta t_c^{max}})$  do
       $m_i = \Delta t_c^{max}$ ;
       $m_{i+1} = \Delta t_c \text{ modulo}(\Delta t_c^{max})$ ;

```

---

### 2.1.2. Granular motion extrapolation

In [Table 1](#), the fixed ID, represents the identifier of the particles for which the collisions were recorded between  $t_0$  and  $t_1$  during the [DEM](#) simulation. These same collisions will be reproduced by the paired particles during the  
180 extrapolation steps obtained using the extrapolation function  $f_{ext}$ . [Figure 2a](#) and [Figure 2b](#) illustrate the two states at  $t_0$  and  $t_1$ , which are subsequently superposed to indicate the paired particles. The latter are shown inside the dashed black circles in [Figure 2c](#) i.e. for a deeper explanation about  $f_{ext}$  refer to [Bednarek et al. \(2019\)](#) article. For the example presented in [Figure 2](#), the pairs

Table 2: List of paired particles.

| Fixed ID | Paired particle ID |
|----------|--------------------|
| Red      | Blue               |
| Blue     | Orange             |
| Orange   | Green              |
| Green    | Red                |

185 of paired particles are shown in Table 2, and the new position (extrapolated) of the particles is shown in Figure 2d.

### 2.1.3. Heat transfer extrapolation

Based on the presented Equation 5, the newly developed algorithm will calculate the new temperature for the particle  $i$  by calculating the sum of the  
 190 temperature change due to the particles in collisions with  $i$ .

In other terms, the temperature of a given particle  $i$  at  $t_2$  will be obtained through the heat conduction equation by using the history of the collisions that particle  $i$  has experienced. The table of contacts is then traversed for each extrapolation. The temperature of the particles paired with  $i$  and  $j$  are  
 195 updated as the collisions with  $i$  and  $j$  occur. It is simply a matter of numerically integrating the Equation 5.

The objective here is not to aim for extreme precision but to limit the number of operations as much as possible. On the other hand, the stability of the scheme must be guaranteed. The idea is therefore to use a single time step equal to the  
 200 duration of the collision. As used in the DEM software, an explicit numerical scheme is also employed. Hence, the condition of Courant–Friedrichs–Lewy CFL<sup>8</sup> less than 1 must be checked to assure the convergence of the explicit scheme. This condition is respected if the  $m < \frac{\rho_i C_i V_i}{2K_{ij} \sqrt{a_{ij}}}$  indicating that  $m <$

---

<sup>8</sup>Courant Friedrichs Lewy

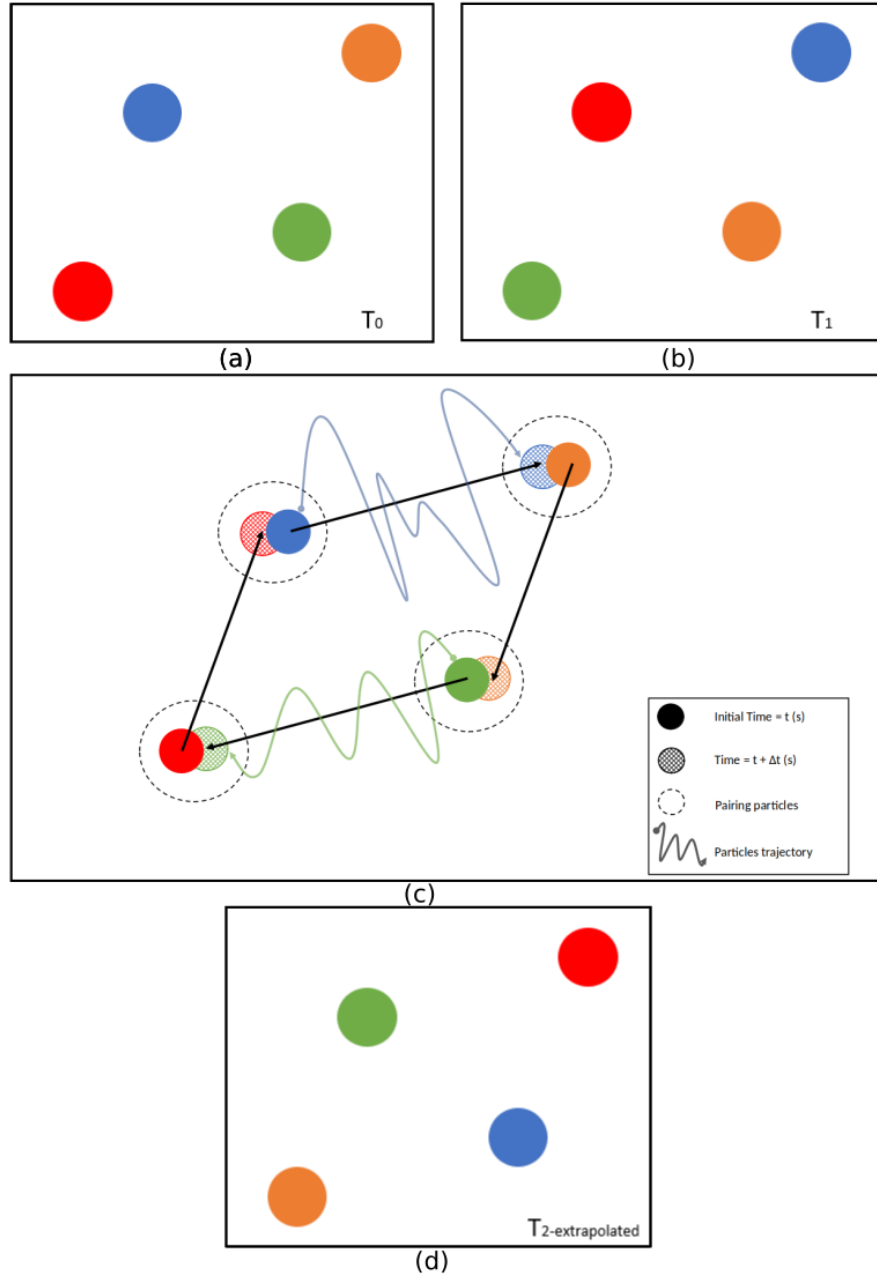


Figure 2: (a) Particles positions at  $t_0$  (b) Particles positions at  $t_1$  (c) Pairing algorithm concept (d) Particles new extrapolated positions at  $t_2$ (*extrapolated*) .

$\Delta t_{critical}$ .

205 Then, for a particle  $i$  in contact with a particle  $j$ , the temperature at the end of the collision  $\{T_i^f, T_j^f\}$  is obtained by solving the following system :

$$T_i^f = T_i^{t_{ini}} - \frac{Q_{ij}m}{\rho_i C_i V_i} \quad (7)$$

$$T_j^f = T_j^{t_{ini}} + \frac{Q_{ij}m}{\rho_i C_i V_i} \quad (8)$$

It should be noted that the couple  $\{T_i^f, T_j^f\}$  is not physically equivalent to the temperature of the particles  $i$  and  $j$  at the final instant of the collision  $\{T_i^{t_{ini}+m}, T_j^{t_{ini}+m}\}$ .

210 Indeed, the collisions are treated by order of occurrence. The simultaneous contacts of a particle will be processed one after the other. The contacts detected at the same time are also processed one after the other.

Finally, the conduction transfer extrapolation algorithm is given in the [algorithm 3](#).

---

**Algorithm 3:** General algorithm for the extrapolation of heat transfer by conduction.

---

```

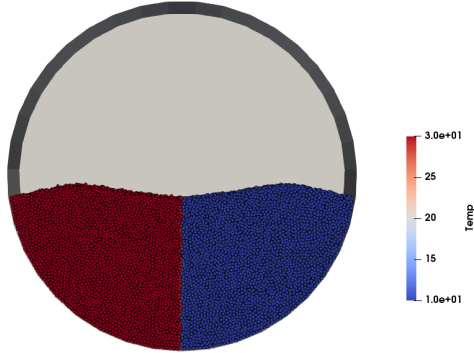
for  $p = 1 \rightarrow nb_{collisions}$  do
  Read  $i(p)$  in Table 1;
  Read  $j(p)$  in Table 1;
  Define  $k = f_{ext}(i)$  with respect to pairing algorithm;
  Define  $l = f_{ext}(j)$  with respect to pairing algorithm;
  Update by solving  $T_k^f, T_l^f$  by solving Equation 7 and Equation 8;

```

---

## 215 2.2. Rotative drum/Simulation Setup

In granular material, heat transfer can be generated through three different forms: conduction, convection and radiation [Nguyen et al. \(2014\)](#). In this paper, only the heat transfer during particle-particle collisions by conduction will be



$t_{init}$

Figure 3: Two halves of particles in rotating drum.

taken into account [Gui et al. \(2013\)](#). Radiation and convection will be neglected  
 220 as in [Chaudhuri et al. \(2006\)](#). In a rotating kiln, there are two sorts of heating  
 techniques: direct and indirect. Direct heating entails infusing a hot fluid that  
 has undergone a chemical reaction to raise its temperature [Herz et al. \(2012\)](#).  
 The indirect one is caused by increasing the temperature of the drum's casing.  
 In our research, the example that will be employed is the one of a rotating drum  
 225 containing two halves of particles, with the first half being hot and the other  
 half being cold as shown in [Figure 3](#). Note that the new method works for any  
 heating type or application.

[Table 3](#) and [Table 4](#) show the characteristic parameters used for the simula-  
 tion. The code can be requested for free on the git repository: [code source](#).

### 230 **3. Results and discussion**

The previously proposed method has been applied over the mixing process  
 of particles in a rotating drum. Many scenarios are tested and the impact of  
 numerous parameters is investigated. Results from standard and extrapolated

Table 3: DEM Parameters.

| Parameter                     | Type              | Value                  |
|-------------------------------|-------------------|------------------------|
| Number                        | Particles         | 80000                  |
| Radius( $r^*$ )               | Particles         | 0.005 mm               |
| Friction coefficient          | Particle-Particle | 0.3                    |
|                               | Particle-Wall     | 0.5                    |
| Rolling Friction coefficient  | Particle-Particle | 0.002                  |
|                               | Particle-Wall     | 0.002                  |
| Restitution coefficient       | Particle-Particle | 0.2                    |
|                               | Particle-Wall     | 0.2                    |
| Poisson ratio                 | Particle          | 0.3                    |
|                               | Wall              | 0.3                    |
| Young Modulus                 | Particle          | $1e^7$ Pa              |
|                               | Wall              | $1e^7$ Pa              |
| Density ( $\rho_i$ )          | Particles         | 2500 Kg/m <sup>3</sup> |
| Rotational speed ( $\omega$ ) | Drum              | 5 s per revolution     |

Table 4: Heat Transfer Parameters for half cold-half hot.

| Parameter  | Value | Unit      |
|--|-------|-----------|
| Initial Temperature of the right subdomain ( $T$ ) | 10    | K         |
| Initial Temperature of the left subdomain ( $T$ )  | 30    | K         |
| Thermal conductivity ( $K_{ij}$ )                  | 100   | W/(m. K ) |
| Thermal capacity ( $C_i$ )                         | 840   | J/K       |



DEM are compared and assessed. All the results shown as temperature profiles  
 235 are obtained using Paraview.

### 3.1. Reference case

The reference case that will be presented here is described in details in the preceding section while taking into account all what happens between the two extrapolation instants ( $n = 1$ ) in the Equation 7 and Equation 8. Note  
 240 that the maximum collision duration is fixed to  $\Delta t_c^{max} = 0.001$  seconds which corresponds to 10 DEM time steps. Considering the pairing, after packing the particles, the drum rotates for one rotation before selecting the initial time of the pairing  $t_0$ . The initial time of extrapolation,  $t_1$ , is then selected when half a rotation has been completed as in Figure 4. Here  $\tau$ , the extrapolation  
 245 period, is equal to a rotation. It should be mentioned that a pseudo-periodic state has already been seen by looking at the interface of the particles that is not changing. The outcomes of the standard and extrapolated DEM are then visualized as shown in Figure 5 and Figure 6. In Figure 7, in red, the average temperature of the hot particles DEM  $\overline{T_{hot}}$  is compared with the extrapolated  
 250 average temperature of the same hot particles *extra*  $\overline{T_{hot}}$  as in Wang et al. (2019). Analogically, in blue, the standard average temperature DEM  $\overline{T_{cold}}$  and the extrapolated average temperature *extra*  $\overline{T_{cold}}$  for the cold particles is also evaluated. More quantitative comparisons are presented in order to compare the standard to the extrapolated DEM. To do so, Voronoi diagram is used. It  
 255 is a diagram that divides the region into smaller regions, the voronoi cells. The cells are generated based on a number of random points, the seeds. The data, temperature for this case, is affected to the closest seed forming the voronoi cell. As it can be seen in Figure 8 the system was divided into  $v = 200$  voronoi cells and the error is calculated using the Equation 9 below:

$$\varepsilon_{rel} = \frac{1}{v} \sum_{i=1}^v \frac{|\overline{T_{DEM}}(i) - \overline{T_{extra}}(i)|}{|\overline{T_{hot}} - \overline{T_{cold}}|} = \frac{\varepsilon_{abs}}{|\overline{T_{hot}} - \overline{T_{cold}}|} \quad (9)$$

260 The physical meaning of the absolute error is that for 200 random cells, the average temperature of the particles  $\overline{T_{DEM}}$  in a random cell for the DEM is

Table 5: Absolute and relative errors per extrapolation step.

| Extrapolation step   | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ | $t_7$ |
|----------------------|-------|-------|-------|-------|-------|-------|
| $\epsilon_{abs}$ (K) | 0.241 | 0.126 | 0.083 | 0.062 | 0.042 | 0.03  |
| $\epsilon_{rel}$ (%) | 1.84  | 1.275 | 1.27  | 1.69  | 2.25  | 3.09  |

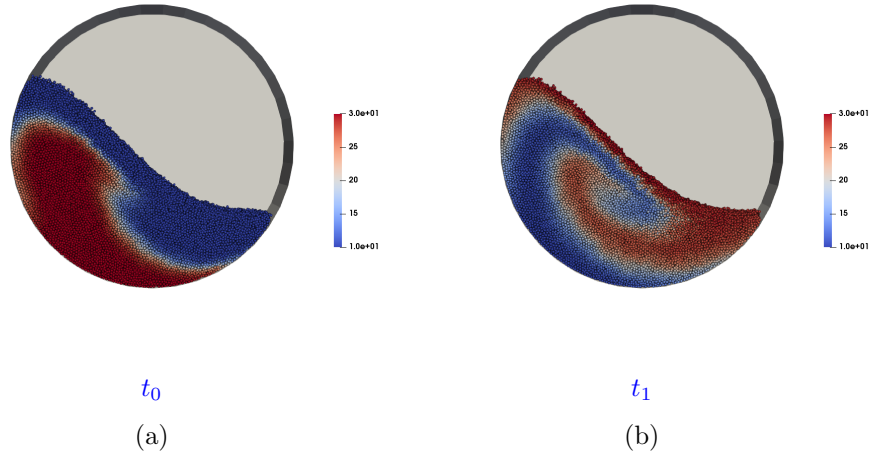


Figure 4: Initial configuration for pairing algorithm.

compared with the extrapolated average temperature of the particles  $\overline{T_{extra}}$  in the same cell and the sum of the difference of those temperature over the number of cells is calculated. The maximum absolute error is of order of 0.2 Kelvin which is small. Furthermore, the relative error is obtained by dividing the absolute error by the difference of the DEM  $\overline{T_{hot}}$  and DEM  $\overline{T_{cold}}$ . The maximum relative error is also smaller than 3% which is considered as an acceptable error as shown in Table 5. This implies that the method is efficient and reliable. It is to be noted that the Figure 7 represents a temporal comparison for the results whereas the Figure 8 stands for the spacial comparison.

As we can visualize, the extrapolated findings of DEM represents a realistic reproduction of the results of standard DEM with a significant reduction of the

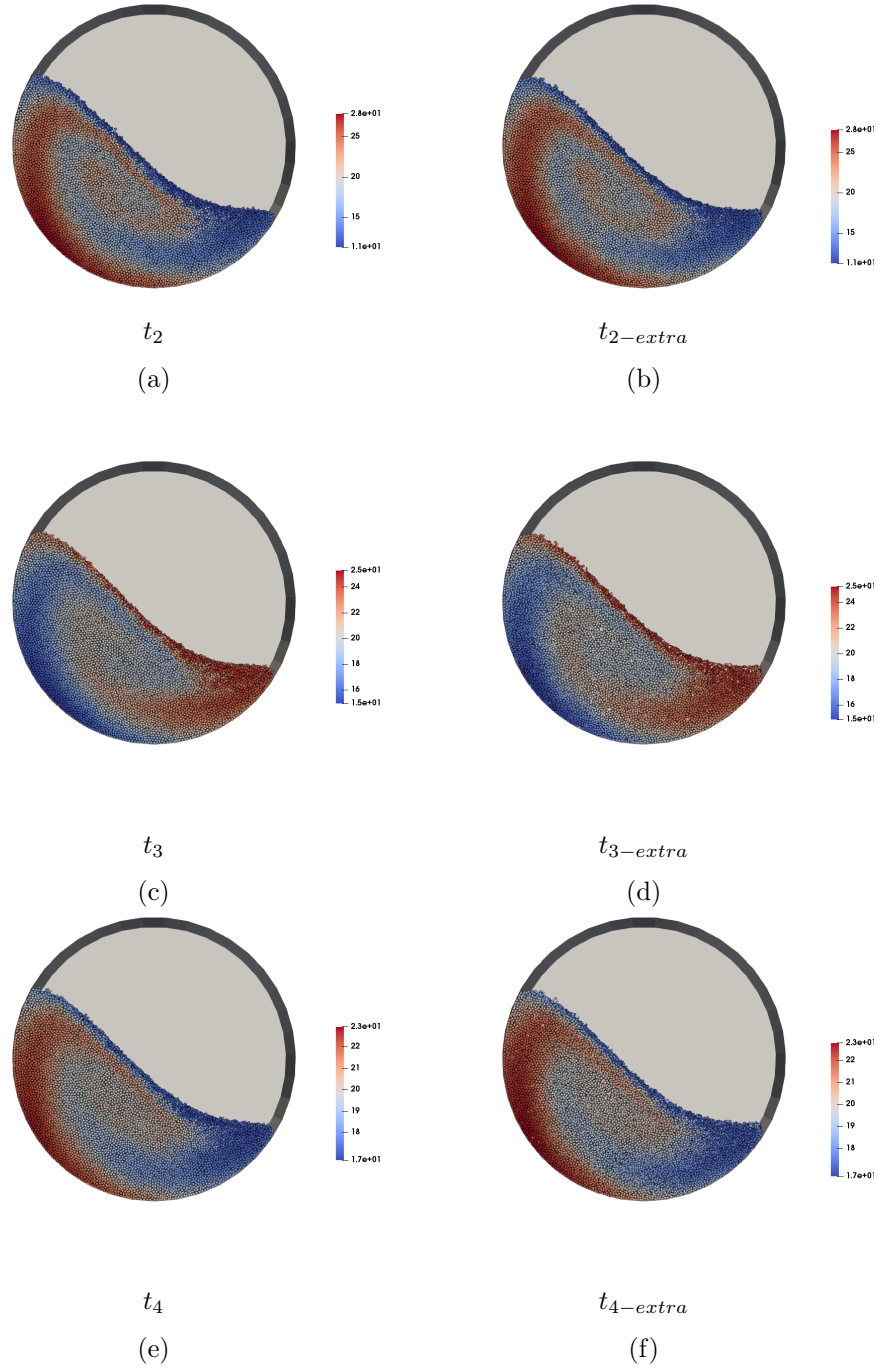


Figure 5: Comparison between standard and extrapolated DEM.

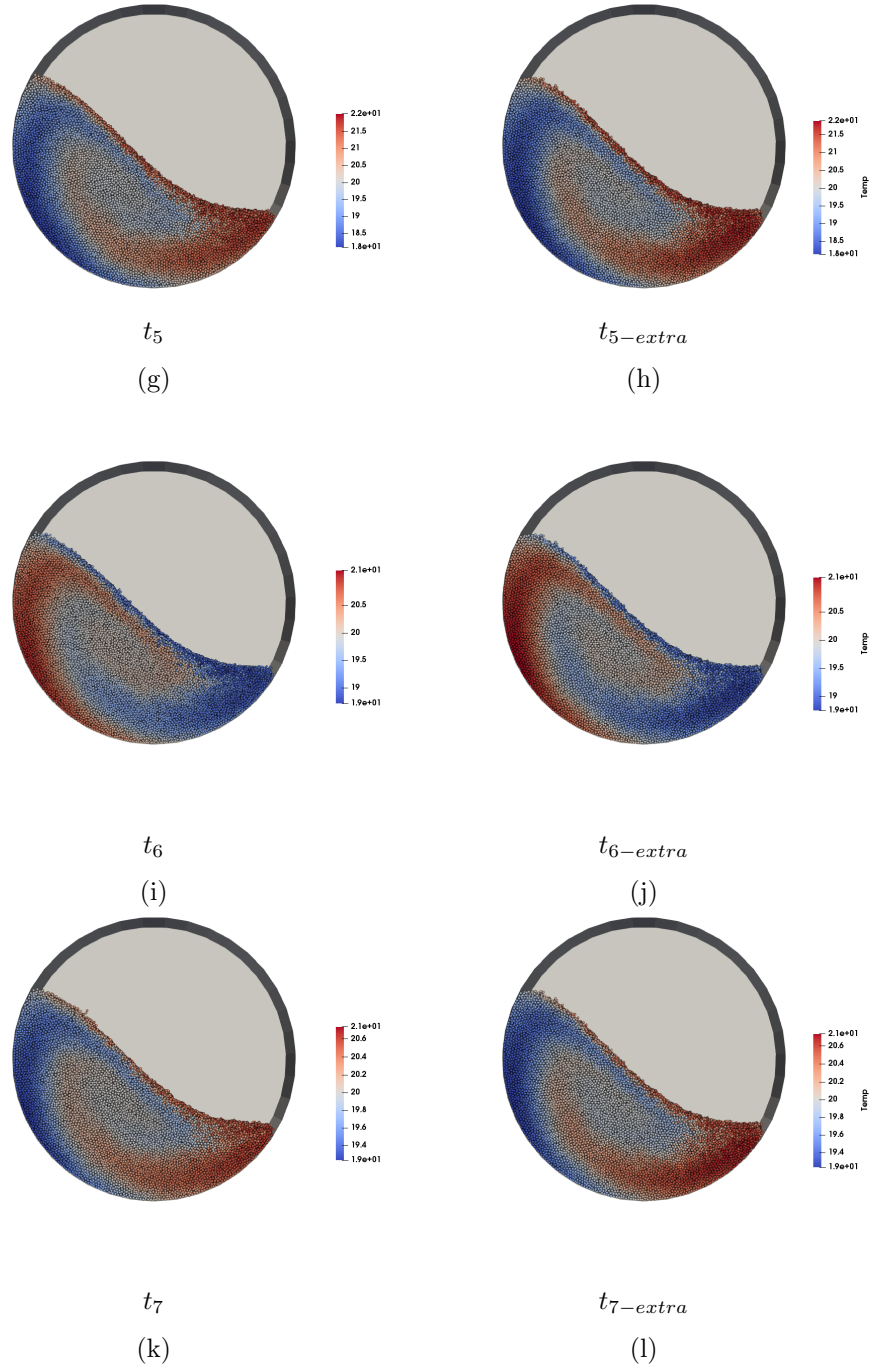


Figure 6: Comparison between standard and extrapolated DEM.

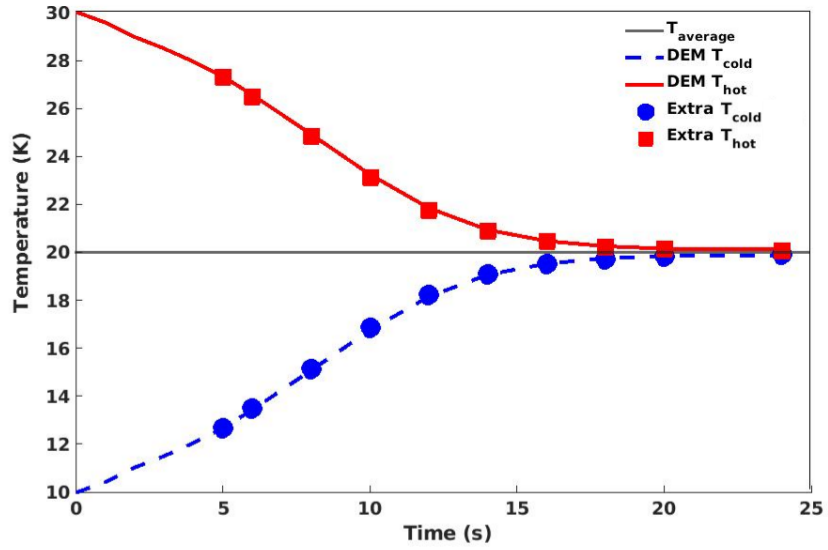


Figure 7: Comparison between DEM results with extrapolated DEM results for heat transfer for the reference case.

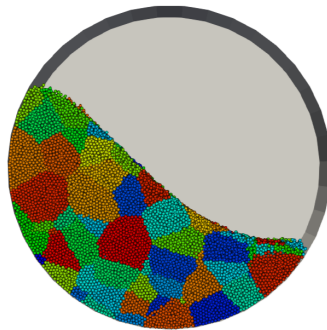


Figure 8: Repartition of 200 voronoi cells in the rotating drum for the study of efficiency.

Table 6: Simulations Performance for 100  $\tau$  of real time.

| Case                                     | Complete DEM | Extrapolated DEM |
|--|--------------|------------------|
| 80000 particles                          |              |                  |
| Until $t_1$ (s)                          | 8580         | 8580             |
| Pairing (s)                              | 0            | 551              |
| Collision file (s)                       | 0            | 7619             |
| For one $\tau$ (s)                       | 3900         | 6                |
| Total simulation time for 100 $\tau$ (s) | 398580       | 17350            |

computational time. All the simulations ran over the cluster centaure of **Mines Saint-Etienne** with one node (one thread only) at 2.66 *GHz* to evenly compare the computational time. Since they require high **RAM**<sup>9</sup>, the node that was used is equipped with big memory that consists of 500 *Gb*. The extrapolated **DEM** requires a **CPU** of just 6 seconds/  $\tau$  whereas the standard **DEM** takes a **CPU** of 3900 seconds/  $\tau$  as in [Table 6](#). This method is efficient if the results required are for longer industrial time. The reduction of the **CPU** is of an order of 650 per  $\tau$  while illustrating accurately the results. The computational time can be changed by modifying other parameters and this will be presented in the next sections.

### 3.2. Influence of the sampling parameters

The new method presented is really promising in terms of the gain of the computational time and rapidity of the simulation. But it requires a high use of **RAM** due the obligation of saving the microscopic characteristic of the particles as  $m$  and the  $\sqrt{(a_{ij})}$  of collisions. Therefore, to solve the latter problem and to reduce the computational time more, statistical approaches are done. These approaches study the effect of the parameter affecting the temperature evolution as shown in [Equation 5](#). The temperature term is proportional to the multipli-

---

<sup>9</sup>Random Access Memory

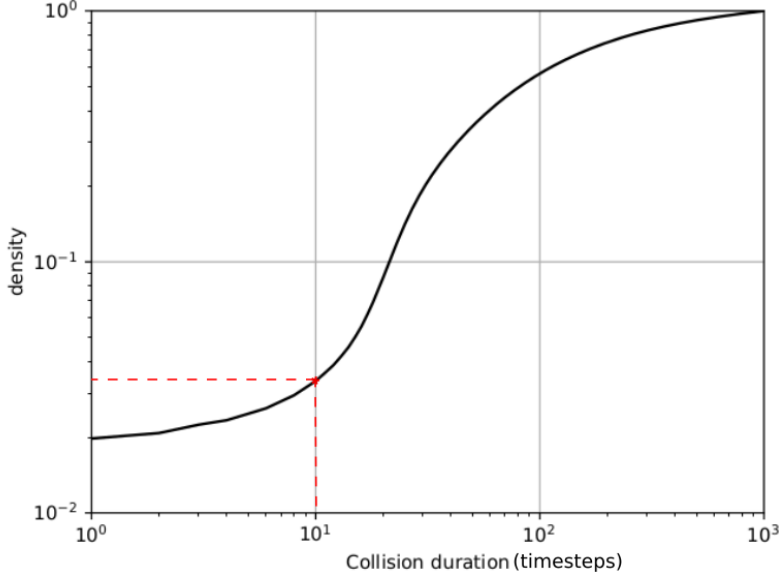


Figure 9: Cumulated density function for collision occurrence.

cation of  $m$  and the  $\sqrt{(a_{ij})}$  of the particles. Therefore, the statistics show the density of the collisions duration or how much time does the majority of the collision lasts. The graph output's can explicit if the omission of small duration collision will affect highly the results or not. According to Figure 9,  $n$  can be deduced. For this case, 97% of the collisions lasts more than 10 timesteps. This enables saving every  $n = 10$  timesteps instead of every one timestep. Indeed the results will be affected a little bit, but this can be explained by not aiming for exact results instead for acceptable, fast and low cost results. Thus, the extrapolated heat transfer results are expected to loose about 3 % in comparison with the normal DEM simulations' values since the collisions with  $\Delta t_c < 10\Delta t_{DEM}$  seconds will be neglected due to collision sampling as in Figure 9.

To study the impact of the sampling parameters on the results,  $n$  and  $\Delta t_c^{max}$  were modified. The Figure 10 shows the percentage error for the energy transferred from  $t_n$  to  $t_{n+1}$  in standard DEM in comparison with the energy transferred from  $t_n$  to  $t_{n+1}$  in extrapolated DEM as in Equation 10. This will show

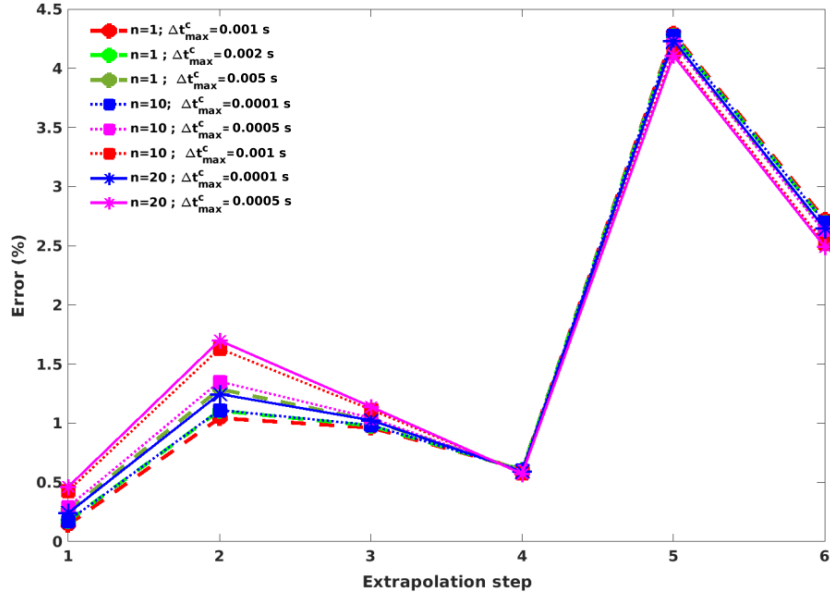


Figure 10: Percentage error for different sampling parameters with saving every  $n$  timesteps and each collision lasts a maximum of  $\Delta t_c$  seconds.

if the transfer for the extrapolated DEM is reliable.

$$\varepsilon = \frac{\sum(|T_{n+1} - T_n|) - \sum(|T_{n+1-extra} - T_{n-extra}|)}{\sum(|T_{n+1} - T_n|)} \quad (10)$$

The difference between the results is not significant. The errors varies between 0.1 % and 4.5% as in Figure 10. The scenario that will be used for further studies is that of  $n = 10$  and  $m = 10$  since it has the least computational time with reasonable results.

310

### 3.3. Influence of the setup parameters

Several setup parameters can be changed to evaluate the results. Some of them are: the number of particles, the initial condition (adiabatic wall or not)), the rotating velocity of the drum that affect the regime inside the drum and the pairing period (half rotation or full rotation) and etc... This analysis is presented below.

315



Table 7: Testing several particles number.

| Case                                     | Complete DEM | Extrapolated DEM |
|--|--------------|------------------|
| 400000 particles                         |              |                  |
| Until $t_1$ (s)                          | 25620        | 25620            |
| Pairing (s)                              | 0            | 3060             |
| Collision file (s)                       | 0            | 19680            |
| For one $\tau$ (s)                       | 4860         | 24               |
| Total simulation time for 100 $\tau$ (s) | 493620       | 50760            |

### 3.3.1. The number of particles

The DEM simulations were performed for different particles' number inside the rotating drum. Many cases were simulated. The following section will present another scenario with 400000 particles. All the physical parameters chosen are the same as of those in the reference case, it is only the number of particles packed that changes. Here,  $n = 1$  and  $\tau$  is taken to be  $0.1\omega$  instead of  $0.5\omega$  as in the reference case since we are limited with the memory. It should be noted that the higher the number of particles, the more the method is efficient and the bigger the gain in the computational time Table 7. Also, if the particle's number increases the CPU time increases as well. However, for the extrapolated simulation, the pairing method is efficient when bigger system is employed as in Bednarek et al. (2019). Then the time for the creation of the collision file can be minimized easily by using the sampling; and finally each extrapolation step takes no more than a minute.

### 3.3.2. Regimes in rotating drum

An analysis for the previously described method is performed for different regimes in the rotary drum. Six different regimes can identified in the rotary drum. These regimes depend on a dimensionless group known as Froude number which is proportional to the angular velocity of the drum and the percentage filling of the drum. This is explained in details by Mellmann (2001). By increas-

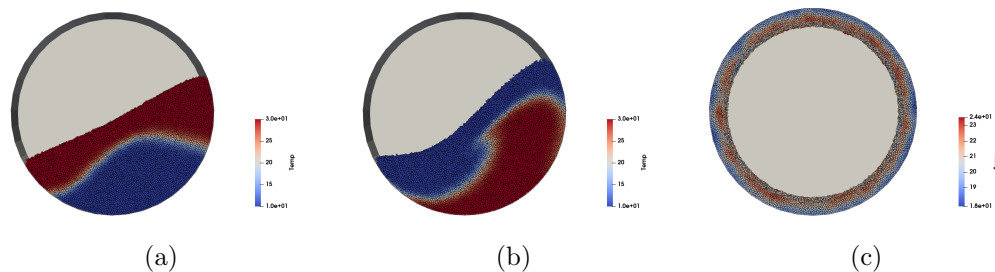


Figure 11: Different regimes used for the extrapolation case: (a) Rolling regime: 15 seconds/revolution; (b) Cascading regime: 5 seconds/revolution; (c) Centrifuging regime: 0.1 seconds/revolution.

ing or decreasing the angular velocity, the regimes can be visualized; the rolling regime [Figure 11a](#) takes 15 seconds per rotation, the cascading [Figure 11b](#) takes 5 seconds per rotation and finally the centrifuging [Figure 11c](#) takes 0.1 seconds per rotation. As shown in the sections before, the method is applied over a cascading regime, [Figure 11b](#). To demonstrate that the used approach is general, it was tested over other regimes. The following results presented in [Figure 12](#) shows the extrapolation for the rolling regime is possible. The extrapolation for [Figure 11c](#) was not performed since it does not present an interest specially because the particles do not have a relative movement. Hence, the method is an accurate description of what is happening with a drastic reduction of computational time. It should be mentioned that the proposed new method is general and may be used to a variety of pseudo-periodic granular systems.

### 3.3.3. The pairing period

The period for the pairing algorithm (initial time of period  $t_0$  and initial extrapolation time  $t_1$ ) is selected to be proportional to the time of the drum rotational speed  $\omega$ , i.e.  $\alpha * \omega$  is equal to  $0.1\omega$ ,  $0.5\omega$ ,  $1\omega$ ,  $2\omega$ , etc...

Various extrapolation periods were studied; quarter rotation, half rotation and full rotation. In the previous section, results for full rotation's extrapolation were evoked and in this section we will present different  $\tau$  values. [Figure 14](#) shows extrapolating every quarter turn and [Figure 15](#) for half rotation extrap-

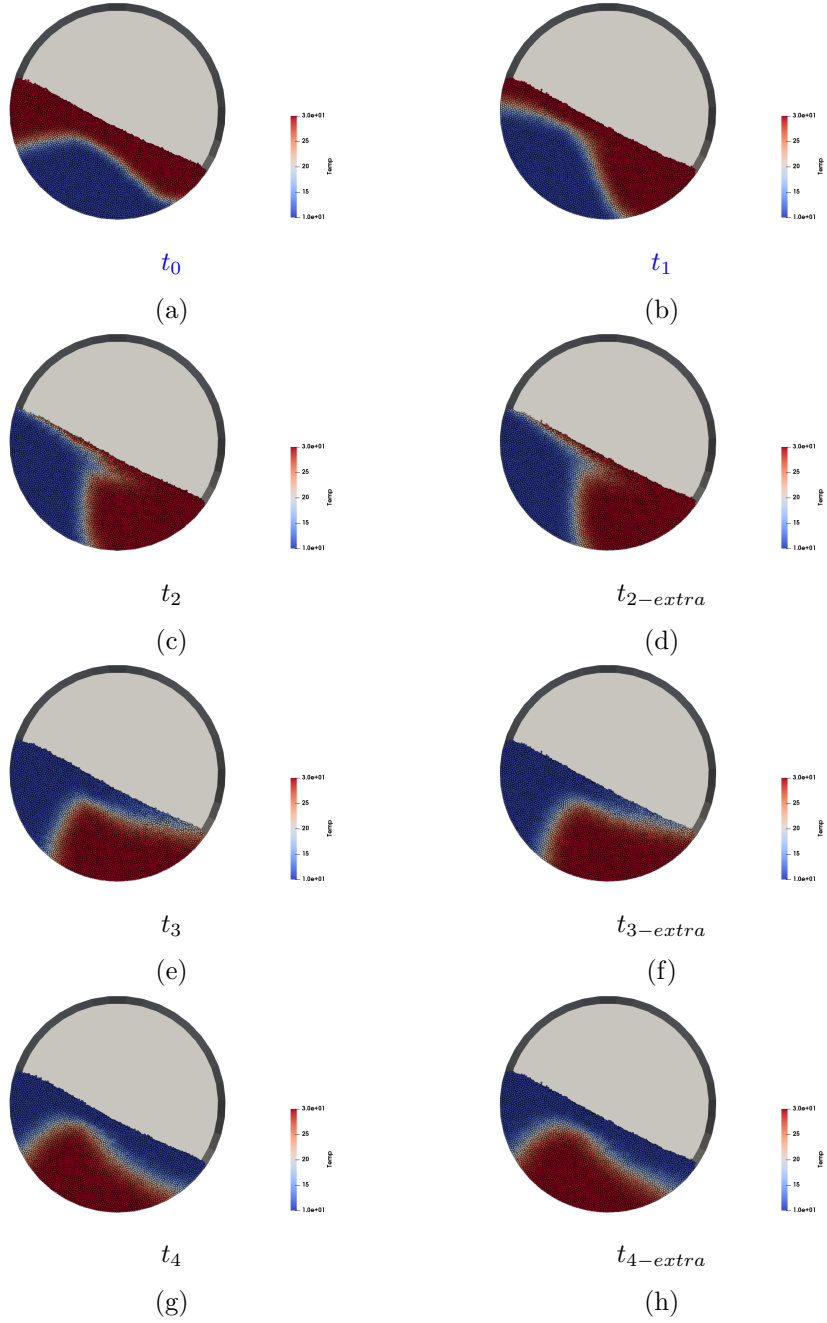


Figure 12: Extrapolation for the rolling regime.

olation. The method works effectively in all scenarios used as it can be seen in Figure 13 describing the evolution of the temperature with the change of  $\alpha$  a constant proportional to the pairing period.

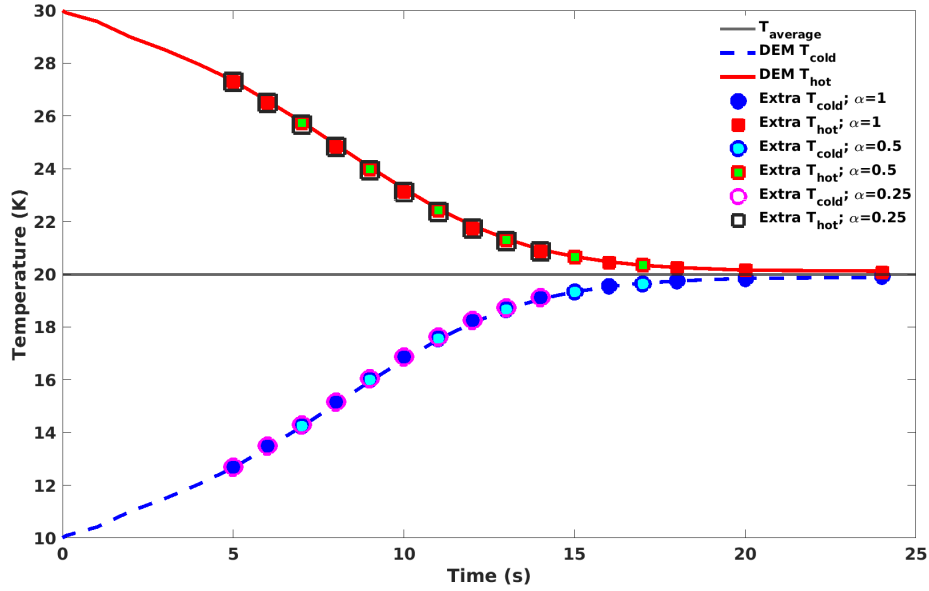


Figure 13: Comparison between DEM results and extrapolated DEM results for heat transfer changing  $\alpha$  for different pairing period.

360 It should be also noted that the bigger the extrapolation time, the bigger the output file, the higher the RAM used and the lower the computational time. It is always a compromise between the computational time and the RAM. In order to use higher extrapolation period with an acceptable memory usage, the sampling should be higher, i.e. saving every 10 or 20 timesteps. Also, the speed  
 365 of the extrapolation is linked to the parameter alpha (the difference between  $t_0$  and  $t_1$ ). Hence if the objective is to find what is happening for very long time in the simulation, higher value of alpha will give the required results faster. This can be seen in following comparative Table 8.

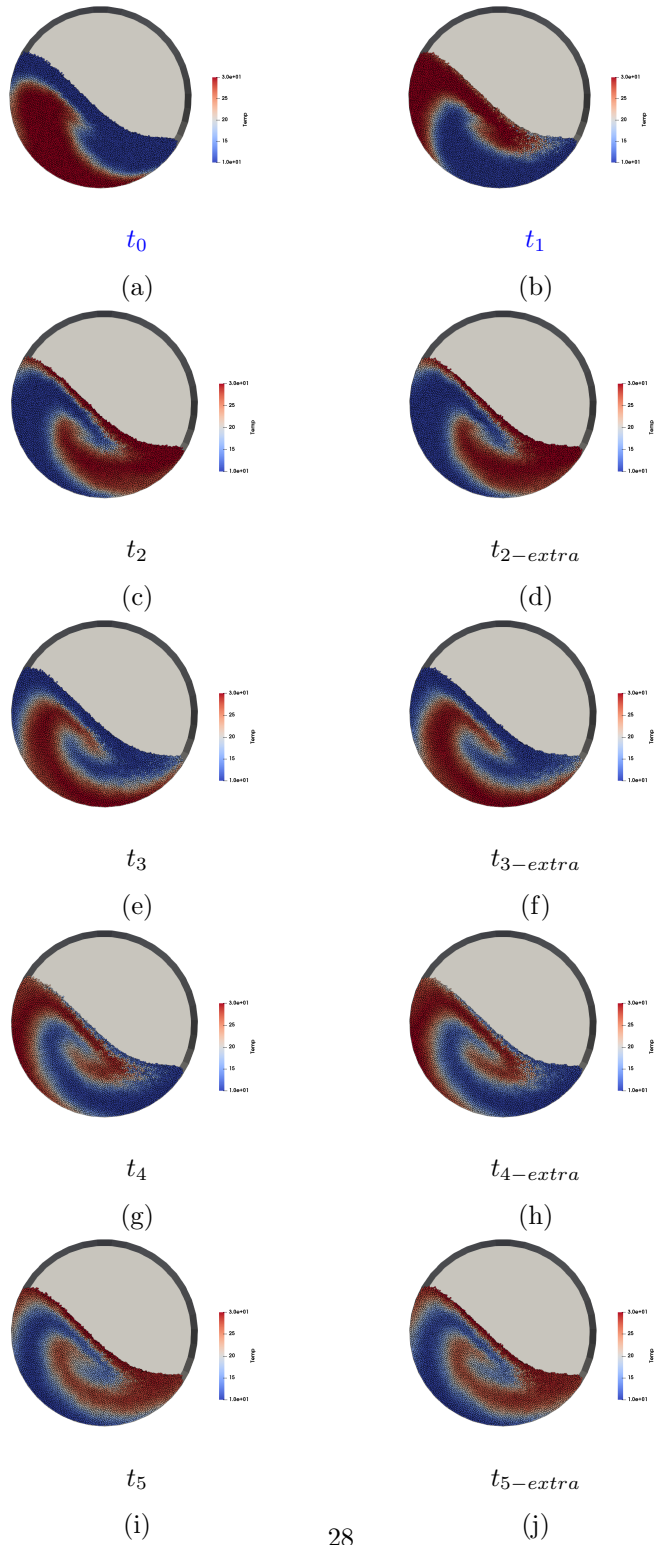


Figure 14: Extrapolation every quarter rotation.

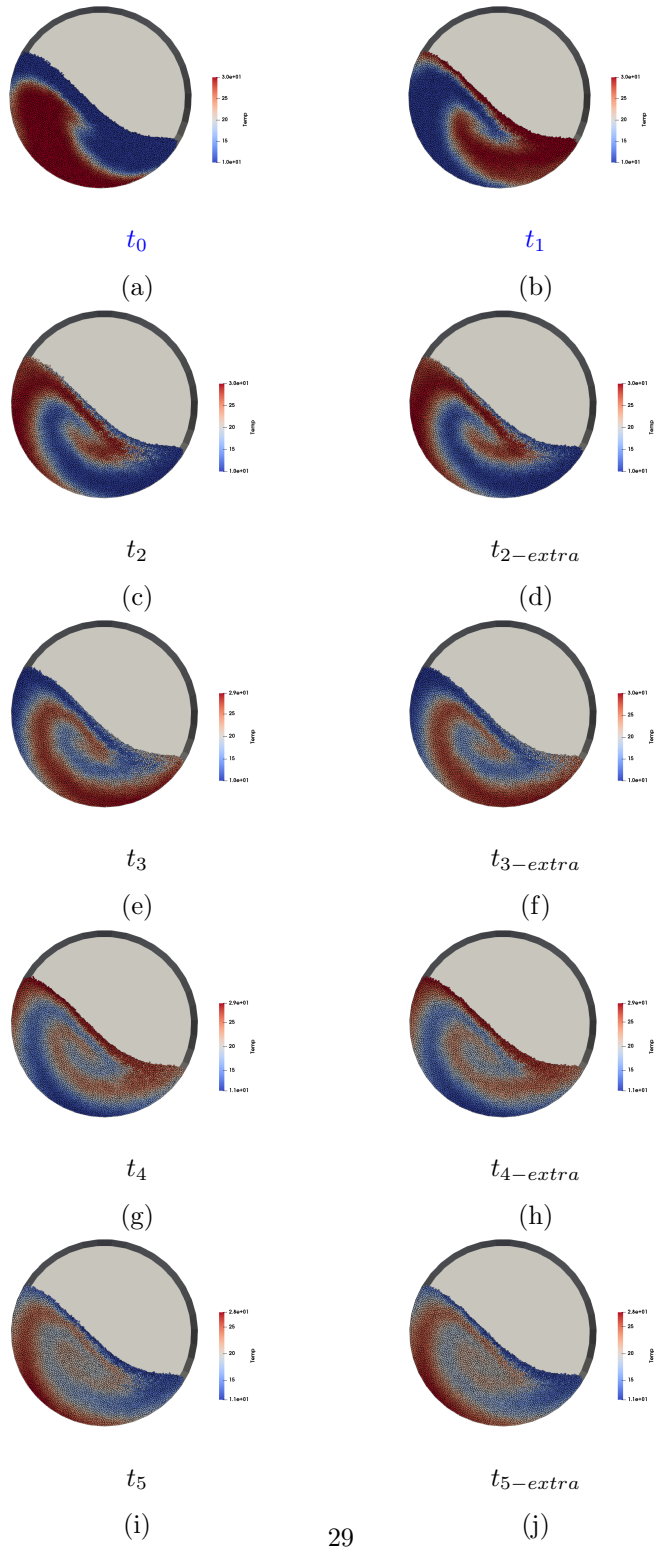


Figure 15: Extrapolation every half rotation.

Table 8: Comparative table for different pairing period with  $n = 10$  and  $\Delta t_c^{max} = 10$ .

| Extrapolation period | Collision file size (Mb) | Time for one rotation(s) |
|----------------------|--------------------------|--------------------------|
| quarter rotation     | 560                      | 13                       |
| half rotation        | 1200                     | 12                       |
| full rotation        | 2800                     | 10                       |

#### 4. Conclusion

370 To understand the behavior of heat transfer due to particle-particle inter-  
actions in dry granular media, DEM simulations were performed. Those sim-  
ulations require high computational time indicating the potency of developing  
new methods to speed them up. The newly developed algorithm has proved its  
efficacy and precision with a low computational time and cost. The results ob-  
375 tained show that from one period of DEM simulations, information for longer  
periods can be revealed. The extrapolated findings agree with the complete  
simulation results with less than 3% error and with a computational gain factor  
of more than 100. This work is promising and serves as a foundation for future  
researches on the topic as the extrapolation for a coupled granular flow simula-  
380 tion using CFD-DEM while taking into account the influence of the interstitial  
fluid (convective heat transfer).

#### Acronyms

**CFD** Computational Fluid Dynamics 3, 30

**CFL** Courant Friedrichs Lewy 11

385 **CPU** Central Processing Unit 3, 21, 24

**DEM** Discrete Element Method 1-8, 10, 11, 16, 17, 21-24, 30

**FEM** Finite Element Method 2

**LIGGGHTS** LAMMPS Improved for General Granular and Granular Heat  
Transfer Simulations 5

390 **RAM** Random Access Memory 21, 27

**SPH** Smoothed Particle Hydrodynamics 3

**TDEM** Thermal Discrete Element Method 2

### Nomenclature

$A_p$  Particle's cross sectional area (m<sup>2</sup>) 6

395  $\varepsilon_{abs}$  Absolute error 16, 17

$\alpha$  Constant 25, 27

$a_{ij}$  Contact area (m<sup>2</sup>) 5–7, 9, 11, 21, 22

$Biot^*$  Quasi-Biot number for particle–particle conduction 6

$C_i$  Particle's specific thermal capacity J/(Kg.K) 6, 11, 13, 15

400  $\Delta t_c$  Real Collision Duration (s) 9, 10, 22, 23

$\Delta t_{critical}$  Critical duration (s) 13

$\Delta t_{DEM}$  DEM timestep (s) 22

$\varepsilon$  Error 23

$H_{ij}$  Conductance between the particles (K/W) 5

405  $i$  Particle i 5–7, 9, 11, 13

$j$  Particle j 5–7, 9, 11, 13



$k$  Particle  $k$  13  
 $K_i$  Thermal conductivity of particle  $i$  W/(K.m) 5  
 $K_{ij}$  Thermal conductivity W/(K.m) 5, 6, 11, 15  
410  $K_j$  Thermal conductivity of particle  $j$  W/(K.m) 5  
  
 $l$  Particle  $l$  13  
  
 $m$  Duration of a single collision in seconds 9–11, 13, 21–23  
  
 $n$  Sampling factor 9, 16, 22–24, 30  
  
 $\omega$  Rotational speed (s/revolution) 15, 24, 25  
415  $\Delta_n$  Overlap distance of two particles (m) 6  
  
 $\vec{x}$  Position vector for DEM particles 7  
  
 $Q_i$  Particle's heat flux (W) 6  
 $Q_{ij}$  Heat flux due to particle-particle collision (W) 5, 13  
  
 $r^*$  Mean particle radii (m) 15  
420  $r_c$  Contact radius between two particles (m) 6  
  
 $r_p$  Particle radius (m) 6  
  
 $\varepsilon_{rel}$  Relative error 16, 17  
  
 $\rho$  Particle's density (Kg/m<sup>3</sup>) 6, 11, 13, 15  
  
 $f_{ext}$  Extrapolation function 7, 10, 13  
425  $T$  Temperature (K) 15  
  
 $t_0$  First instant of the extrapolation period (s) 8–10, 12, 16, 17, 25–29

- $t_1$  Final instant of the extrapolation period (s) 7–10, 12, 16, 17, 21, 24–29
- $t_2$  Instant of the first extrapolation step (s) 7, 11, 12
- $\Delta t_c^{max}$  Imposed maximum collision duration (s) 8–10, 16, 22, 30
- 430  $\overline{T}_{cold}$  Average temperature of the cold particles (K) 16, 17
- $\overline{T}_{DEM}$  Average temperature of the particles for the DEM results (K) 16
- $\overline{T}_{extra}$  Average temperature of the particles for the extrapolated results (K) 16,  
17
- $\overline{T}_{hot}$  Average temperature of the hot particles (K) 16, 17
- 435  $t_{init}$  Initial time (s) 9, 14
- $t_{max}$  Maximum collision duration (s) 5
- $\tau$  Extrapolation period 16, 21, 24, 25
- $T_i$  Temperature of the particle i (K) 5, 6, 13
- $T_j$  Temperature of the particle j (K) 5, 6, 13
- 440  $v$  Number of voronoi cells 16
- $V_i$  Particle's volume (m<sup>3</sup>) 6, 11, 13

## References

- Bauer, H., Matic, J., Evans, R. C., Gryczke, A., Ketterhagen, S. K., W., & Khinast, J. (2022). Determining local residence time distributions in twin-screw extruder elements via smoothed particle hydrodynamics. *Chemical Engineering Science*, 247, 117029.
- 445 Beaulieu, C., Vidal, D., Yari, B., Chaouki, J., & Bertrand, F. (2021). Impact of surface roughness on heat transfer through spherical particle packed beds. *Chemical Engineering Science*, 231, 116256.

- 450 Bednarek, X., Martin, S., Ndiaye, A., Peres, V., & Bonnefoy, O. (2019). Extrapolation of dem simulations to large time scale. application to the mixing of powder in a conical screw mixer. *Chemical Engineering Science*, *197*, 223–234.
- Bertrand, F., Leclaire, L. A., & Levecque, G. (2005). Dem-based models for the  
455 mixing of granular materials. *Chemical Engineering Science*, *60*, 2517–2531.
- Chaudhuri, B., Muzzio, F., & Tomassone, M. (2006). Modeling of heat transfer in granular flow in rotating vessels. *Chemical Engineering Science*, *61*, 6348 – 6360. doi:<https://doi.org/10.1016/j.ces.2006.05.034>.
- Cundall, P., & Strack, O. D. (1979). A discrete numerical model for granular  
460 assemblies. *Geotechnique*, *29*, 47–65.
- De, T., Chakraborty, J., Kumar, J., Tripathi, A., Sen, M., & Ketterhagen, W. (2022). A particle location based multi-level coarse-graining technique for discrete element method (dem) simulation. *Powder Technology*, *398*, 117058.
- Doucet, J., Hudon, N., Bertrand, F., & Chaouki, J. (2008). Modeling of the mix-  
465 ing of monodisperse particles using a stationary dem-based markov process. *Computers & Chemical Engineering*, *32*, 1334–1341.
- Feng, Y., Han, K., Li, C., & Owen, D. (2008). Discrete thermal element modelling of heat conduction in particle systems: Basic formulations. *Journal of Computational Physics*, *227*, 5072–5089.
- 470 Feng, Y., Han, K., & Owen, D. (2009). Discrete thermal element modelling of heat conduction in particle systems: pipe-network model and transient analysis. *Powder Technology*, *193*, 248–256.
- Fry, A. M., Umbanhowar, P. B., Ottino, J. M., & Lueptow, R. M. (2019). Diffusion, mixing, and segregation in confined granular flows. *AIChE Journal*,  
475 *65*, 875–881.

- Gui, N., Yan, J., Xu, W., Ge, L., Wu, D., Ji, Z., Gao, J., Jiang, S., & Yang, X. (2013). Dem simulation and analysis of particle mixing and heat conduction in a rotating drum. *Chemical Engineering Science*, *97*, 225–234.
- 480 Hartmanshenn, C., Khinast., J., Papageorgiou, C., Mitchell, C., Quon, J., & Glasser, B. (2019). Heat transfer of dry granular materials in a bladed mixer: Effect of thermal properties and agitation rate. *AIChE Journal*, *66*. doi:<https://doi.org/10.1002/aic.16861>.
- 485 Herz, F., Mitov, I., Specht, E., & Stanev, R. (2012). Influence of operational parameters and material properties on the contact heat transfer in rotary kilns. *International Journal of Heat and Mass Transfer*, *55*, 7941–7948.
- Huang, A., & Kuo, H. (2018). Cfd simulation of particle segregation in a rotating drum. *Advanced Powder Technology*, *29*, 3368–3374. doi:<https://doi.org/10.1016/j.appt.2018.09.019>.
- 490 Kiani-Oshtorjani, M., & Jalali, P. (2019). Thermal discrete element method for transient heat conduction in granular packing under compressive forces. *International Journal of Heat and Mass Transfer*, *145*, 118753.
- Kiani-Oshtorjani, M., Kiani-Oshtorjani, M., Mikkola, A., & Jalali, P. (2022). Conjugate heat transfer in isolated granular clusters with interstitial fluid using lattice boltzmann method. *International Journal of Heat and Mass*  
495 *Transfer*, *187*, 122539.
- Kloss, C., Goniva, C., Hager, A., Amberger, S., & Pirker, S. (2012). Models, algorithms and validation for opensource dem and cfd-dem. *Progress in Computational Fluid Dynamics, An International Journal*, *12*, 140.
- 500 Lichtenegger, T. (2020). Fast eulerian-lagrangian simulations of moving particle beds under pseudo-steady-state conditions. *Powder Technology*, *362*, 474 – 485. URL: <http://www.sciencedirect.com/science/article/pii/S0032591019309428>. doi:<https://doi.org/10.1016/j.powtec.2019.10.113>.

- Lichtenegger, T., Kieckhefen, P., Heinrich, S., & Pirker, S. (2019). Dynamics and long-time behavior of gas–solid flows on recurrent-transient backgrounds. *Chemical Engineering Journal*, *364*, 562 – 577. URL: <http://www.sciencedirect.com/science/article/pii/S1385894719301792>. doi:<https://doi.org/10.1016/j.cej.2019.01.161>.
- Lichtenegger, T., Peters, E., Kuipers, J., & Pirker, S. (2016). A recurrence cfd study of heat transfer in a fluidized bed. *Chemical Engineering Journal*, *172*, 310 – 322.
- Lichtenegger, T., & Pirker, S. (2016). Recurrence cfd—a novel approach to simulate multiphase flows with strongly separated time scales. *Chemical Engineering Journal*, *153*, 394 – 410.
- Liu, X., Gui, N., Yang, X., Tu, J., & Jiang, S. (2021). A new discrete element-embedded finite element method for transient deformation, movement and heat transfer in packed bed. *International Journal of Heat and Mass Transfer*, *165*, 120714.
- Mellmann, J. (2001). The transverse motion of solids in rotating cylinders—forms of motion and transition behavior. *Powder Technology*, *118*, 251–270.
- Muzzio, F., Alexander, A., Goodridge, C., Shen, E., Shinbrot, T., Manjunath, K., & Jacob, K. (2004). Solids mixing. *Handbook of Industrial Mixing*, (p. 887–985). doi:<https://doi.org/10.1002/0471451452.ch15>.
- Nguyen, H. T., Cosson, B., Lacrampe, M. F., & Krawczak, P. (2014). Numerical simulation on the flow and heat transfer of polymer powder in rotational molding. *International Journal of Material Forming*, *8*, 423–438.
- Peng, Z., Doroodchi, E., & Moghtaderi, B. (2020). Heat transfer modelling in discrete element method (dem)-based simulations of thermal processes: Theory and model development. *Progress in Energy and Combustion Science*, *79*, 100847.

- Peters, E., Kuipers, J. et al. (2023). A detailed gas-solid fluidized bed comparison study on cfd-dem coarse-graining techniques. *Chemical Engineering Science*, (p. 118441).
- 535 Pirker, S., & Lichtenegger, T. (2018). Efficient time-extrapolation of single- and multiphase simulations by transport based recurrence cfd (rcfd). *Chemical Engineering Science*, 188, 65 – 83. URL: <http://www.sciencedirect.com/science/article/pii/S0009250918302732>. doi:<https://doi.org/10.1016/j.ces.2018.04.059>.
- 540 Saruwatari, M., & Nakamura, H. (2022). Coarse-grained discrete element method of particle behavior and heat transfer in a rotary kiln. *Chemical Engineering Journal*, 428, 130969.
- Siegmann, E., Enzinger, S., Toson, P., Doshi, P., Khinast, J., & Jajcevic, D. (2021). Massively speeding up dem simulations of continuous processes using  
545 a dem extrapolation. *Powder Technology*, 390, 442–455.
- Vargas, W. L., & McCarthy, J. J. (2001). Heat conduction in granular materials. *Aiche Journal*, 47, 1052–1059.
- Wang, S., Luo, K., Hu, C., Lin, J., & Fan, J. (2019). Cfd-dem simulation of heat transfer in fluidized beds: Model verification, validation, and application.  
550 *Chemical Engineering Science*, 197, 280–295.
- Yazdani, E., & Hashemabadi, S. (2020). Three-dimensional heat transfer in a particulate bed in a rotary drum studied via the discrete element method. *Particuology*, 51, 155–162.
- Zhang, H., Zhou, Q., & Zheng, Y. (2011). A multi-scale method for thermal conduction simulation in granular materials. *Computational materials science*,  
555 50, 2750–2758.
- Zhao, S., Zhao, J., & Lai, Y. (2020). Multiscale modeling of thermo-mechanical responses of granular materials: A hierarchical continuum–discrete coupling

approach. *Computer Methods in Applied Mechanics and Engineering*, 367,  
560 113100.

Zhu, H., Zhou, Z., Yang, R., & Yu, A. (2007). Discrete particle simulation of particulate systems: theoretical developments. *Chemical Engineering Science*, 62, 3378–3396.