AFRL-AFOSR-JP-TR-2018-0055

Thermo-Mechanical Modelling of Granular Materials With Non-spherical Particle Shape

Sharen Cummins
COMMONWEALTH SCIENTIFIC AND INUSTRIAL RESEARCH ORGANISATION

06/28/2018
Final Report

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/ IOA
Arlington, Virginia 22203
Air Force Materiel Command

DISTRIBUTION A: Distribution approved for public release.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
28-06-2018	Final	20 Dec 2016 to 19 Dec 2017

4. TITLE AND SUBTITLE
Thermo-Mechanical Modelling of Granular Materials With Non-spherical Particle Shape

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA2386-17-1-0073

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Sharen Cummins

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
COMMONWEALTH SCIENTIFIC AND INUSTRIAL RESEARCH ORGANISATION
LIMEEONE AVE
CABERRA, 2612 AU

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AOARD
UNIT 45002
APO AP 96338-5002

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR IOA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-AFOSR-JP-TR-2018-0055

12. DISTRIBUTION/AVAILABILITY STATEMENT
A DISTRIBUTION UNLIMITED: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
In this grant a new capability to model heat transfer through granular materials with non-spherical shape has been developed. This has involved creating a new body-fitted grid solver designed to simulate intra-particle heat conduction through super-quadric (SQ) particles, coupling this new solver to a traditional Discrete Element Method (DEM) solver (which simulates inter-particle collisions) and implementing these coupled solvers on a hybrid CPU/GPU architecture for maximum computational efficiency. A 14-fold speed-up was noted in using the hybrid CPU/GPU architecture versus running purely on the CPU. This suggests that the use of hybrid CPU/GPU architectures is promising for the simulation of intra-body heat transfer in granular materials with non-spherical shape. Future work will involve applying this capability to study heat conduction in more realistic granular materials which will have impact in a variety of applications ranging from multi-phase reactors, catalytic processes and explosives.

15. SUBJECT TERMS
mechanical loading, granular modelling, thermal loading

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			AHLERS, KRISTOPHER
Unclassified	Unclassified	Unclassified	SAR		19b. TELEPHONE NUMBER *(Include area code)* 315-227-7009

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Final Report for AOARD Grant 16IOA073 "Thermo-Mechanical Modelling of Granular Materials With Non-Spherical Particle Shape"

4th April 2018

PI information: Sharen J. Cummins; sharen.cummins@csiro.au; CSIRO; Data 61; Private Bag 10, Clayton, Victoria, AUSTRALIA, 3168; +61 3 9545 8061.

Period of Performance: 12/20/2016 – 12/20/2017

Abstract:
In this grant a new capability to model heat transfer through granular materials with non-spherical shape has been developed. This has involved creating a new body-fitted grid solver designed to simulate intra-particle heat conduction through super-quadric (SQ) particles, coupling this new solver to a traditional Discrete Element Method (DEM) solver (which simulates inter-particle collisions) and implementing these coupled solvers on a hybrid CPU/GPU architecture for maximum computational efficiency. A 14-fold speed-up was noted in using the hybrid CPU/GPU architecture versus running purely on the CPU. This suggests that the use of hybrid CPU/GPU architectures is promising for the simulation of intra-body heat transfer in granular materials with non-spherical shape. Future work will involve applying this capability to study heat conduction in more realistic granular materials which will have impact in a variety of applications ranging from multi-phase reactors, catalytic processes and explosives.

Introduction:
The goal of this grant was to provide a new capability to simulate heat transfer in realistic granular materials of variable shape. Particle shape is recognized as one of the most important parameters influencing the behavior of granular media. Mixing rates [1], material strength [2] and flow patterns [3] are all strongly affected by particle shape. While there has been significant development in modelling the mechanics of granular media with variable particle shape, the most recent research in granular heat transfer assumes particles have spherical shape which is an unrealistic degree of idealization of the particles. There exists little work on thermo-mechanical simulation of granular materials with more realistic, three-dimensional non-spherical particle shape. This is despite the fact that the simulation of granular systems with both thermal and mechanical behavior has relevance in numerous industrial (e.g blast furnaces and kilns), defense (e.g explosives) and geological applications (e.g heat generation at fault lines). These diverse applications all require a capability to characterize heat transfer through granular media.

The specific aims of this research were:
1) to develop a new computational tool (solver) designed to simulate internal particle heat conduction through super-quadric (SQ) particles using a body-fitted coordinate finite difference method,
2) to couple this new solver to CSIRO's traditional Discrete Element Method (DEM) [4] solver and
3) to implement these coupled solvers on a hybrid CPU/GPU architecture in order to address the need for large computational resources (both processing and memory).

Capabilities Developed:

1. The Intra-Body Diffusion (IBD) Solver:

The first requirement of the project was to develop a generic heat/species diffusion solver that can be applied to a body-fitted grid – in this project the grid is fitted to a super-quadric (SQ) particle. A body-fitted coordinate finite difference method of the type described in [5] was developed which from here on in is termed the Intra-Body Diffusion (IBD) solver. In this approach, a grid fitted to a SQ particle is transformed to a rectangular (computational) grid which is used in the diffusion computations. The diffusion equation, initial and boundary conditions (BCs) are re-formulated from the co-ordinate (physical) frame of the SQ particle (x_1, x_2, x_3) to the co-ordinate (computational) frame of the rectangular grid $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$ and solved. The heat diffusion equation in physical space is

$$\rho c_p \frac{\partial T}{\partial t} = \sum_{m=1}^{3} \sum_{n=1}^{3} \frac{\partial}{\partial x_m} \left(K_{mn} \frac{\partial T}{\partial x_n} \right)$$

where ρ is density, c_p is specific heat capacity, T is temperature and K_{mn} is the conductivity tensor.

This is transformed to the following equation in computational space

$$\rho c_p \frac{\partial T}{\partial t} = \sum_{m=1}^{3} \sum_{n=1}^{3} \frac{\partial}{\partial \varepsilon_m} \left(\widehat{K}_{mn} \frac{\partial T}{\partial \varepsilon_n} \right) \quad (1)$$

where the conductivity tensor is

$$\widehat{K}_{mn} = \sum_{p=1}^{3} \sum_{q=1}^{3} \frac{1}{J} \beta_{pm} \beta_{qn} K_{pq}$$

where β_{ij} is the *ij*th cofactor of the Jacobian transformation matrix

$$J = \begin{vmatrix} \frac{\partial x_1}{\partial \varepsilon_1} & \frac{\partial x_2}{\partial \varepsilon_1} & \frac{\partial x_3}{\partial \varepsilon_1} \\ \frac{\partial x_1}{\partial \varepsilon_2} & \frac{\partial x_2}{\partial \varepsilon_2} & \frac{\partial x_3}{\partial \varepsilon_2} \\ \frac{\partial x_1}{\partial \varepsilon_3} & \frac{\partial x_2}{\partial \varepsilon_3} & \frac{\partial x_3}{\partial \varepsilon_3} \end{vmatrix}.$$

The shape of a SQ particle can be expressed algebraically as

$$\left(\frac{x_1}{a}\right)^m + \left(\frac{x_2}{b}\right)^m + \left(\frac{x_3}{c}\right)^m = 1 \quad (2)$$

where m is the SQ sharpness factor and a, b, c are the semi-axes lengths of the SQ particle. Consequently, the following generalisation of a cubed-sphere approach [6] is used to map a point from the SQ physical space to computational space

$$x_1 = \left(\frac{\varepsilon_1}{a}\right)\left[1 - \frac{1}{2}\left(\left(\frac{\varepsilon_2}{b}\right)^m + \left(\frac{\varepsilon_3}{c}\right)^m\right) + \frac{1}{3}\left(\frac{\varepsilon_2}{b}\right)^m\left(\frac{\varepsilon_3}{c}\right)^m\right]^{1/m}$$
$$x_2 = \left(\frac{\varepsilon_2}{b}\right)\left[1 - \frac{1}{2}\left(\left(\frac{\varepsilon_1}{a}\right)^m + \left(\frac{\varepsilon_3}{c}\right)^m\right) + \frac{1}{3}\left(\frac{\varepsilon_1}{a}\right)^m\left(\frac{\varepsilon_3}{c}\right)^m\right]^{1/m}$$
$$x_3 = \left(\frac{\varepsilon_3}{c}\right)\left[1 - \frac{1}{2}\left(\left(\frac{\varepsilon_1}{a}\right)^m + \left(\frac{\varepsilon_2}{b}\right)^m\right) + \frac{1}{3}\left(\frac{\varepsilon_1}{a}\right)^m\left(\frac{\varepsilon_2}{b}\right)^m\right]^{1/m}$$

These expressions are analytically differentiated to form the Jacobian transformation matrix. The transformed heat diffusion equations in Eqn. (1) were discretised using a finite difference formulation. Centred differences were used in the calculation of the spatial derivatives $\frac{\partial T}{\partial \varepsilon_n}$ in Eqn. (1) and first-order Euler time-stepping used for the time derivatives $\frac{\partial T}{\partial t}$.

The IBD solver was used to simulate heat conduction in variously shaped SQ particles with both Dirichlet (isothermal) BCs and/or homogenous Neumann (adiabatic) BCs for the temperature field. For all cases, the diffusivity was set to $\alpha = \frac{K}{\rho c_p} = 1$. Figs. 2-4 below show results of the IBD solver applied to SQ shaped particles with different sharpness factors (m) and semi-axes lengths (a, b, c), initial and boundary conditions. Fig. 2 shows the evolving temperature field in a slice through a spherical-shaped SQ particle ($m = 2, a = b = c = 1$) with initial condition $T = 1K$ and Dirichlet (isothermal) boundary condition $T = 0K$ applied over the sphere surface. Fig. 2 shows the temperature field reducing smoothly and symmetrically in the interior as heat is transferred from the warm interior to the cooler surface. Fig. 3 shows the evolving temperature field in a slice through a blocky SQ particle ($m = 6, a = b = c = 1$) with initial condition $T = 0K$, Dirichlet boundary condition $T = 1K$ applied to the left and right edges and homogenous Neumann BCs applied elsewhere on the surface. Heat energy diffuses smoothly inward from the left and right edges. Finally, Fig. 4 shows an elliptical-shaped SQ particle ($m = 4, a = b = 2, c = 1$) with initial condition $T = 0K$, Dirichlet BCs $T = 1K$ applied to the corners and homogenous Neumann BCs applied elsewhere on the surface. Heat energy smoothly diffuses inward from the corners.

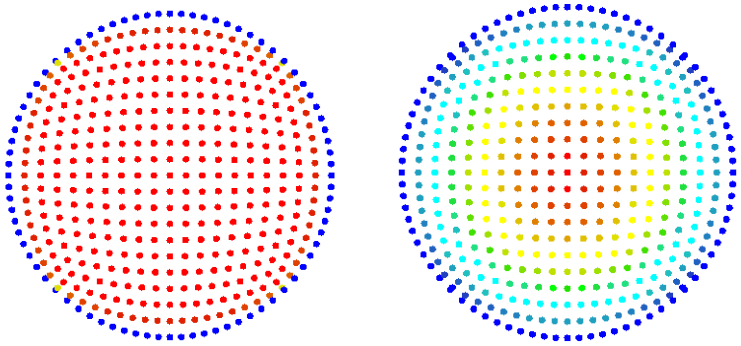

Figure 2. Temperature evolution through the central slice of a spherical-shaped SQ particle ($m = 2, a = b = c = 1$) using the IBD solver. Dirichlet temperature BCs are applied on the entire surface.

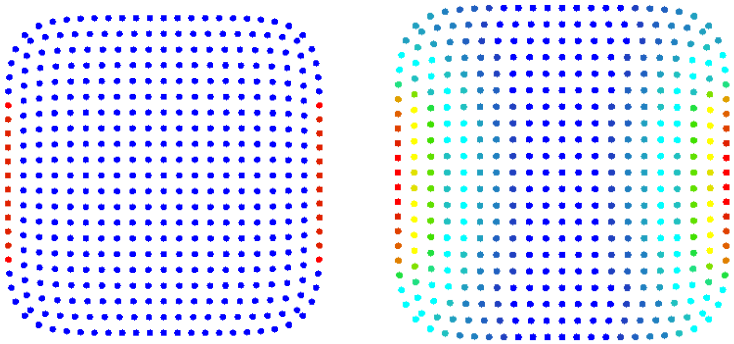

Figure 3. Temperature evolution through the central slice of a blocky-shaped SQ particle ($m = 6, a = b = c = 1$) using the IBD solver. Dirichlet BCs are applied on the left and right edges and homogenous Neumann BCs are applied elsewhere on the surface.

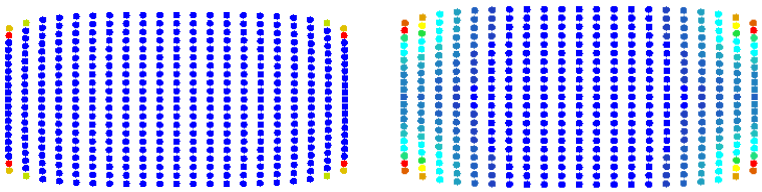

Figure 4. Temperature evolution through the central slice of an elliptical-shaped SQ particle ($m = 4, a = b = 2, c = 1$) using the IBD solver. Dirichlet BCs are applied on the corners of the particle and homogenous Neumann BCs are applied elsewhere on the surface.

2. Coupling the IBD and DEM Solvers:

Having developed the IBD solver (which calculates the intra-particle heat transfer), the next requirement was to couple this IBD solver to CSIRO's traditional DEM solver (which calculates the inter-particle collisional forces and fluxes). This coupling requirement involved two tasks:
(a) Design of the appropriate software within the DEM solver to allow for dynamically changing numbers of SQ particles requiring IBD solutions during a DEM simulation, AND
(b) Development of a heat transfer strategy between colliding SQ particles that each have a body-fitted grid.

The first task (a) is essential because a typical industrial scale DEM simulation will have particles entering and exiting the domain (due to inflows and outflows of granular material). In addition a DEM particle is often involved in multiple collisions which will start and stop at different times and occur at different locations on the surface of the particle. This means that the number of IBD solvers executing will continually change as will the number of boundary source terms per IBD solver (a boundary source term is calculated for each collision between particle pairs and must be continually updated as the collision proceeds). The interface between the IBD and DEM solvers developed during this work permits a dynamically changing number of IBD solvers per time-step and a dynamically changing number of boundary source terms per IBD solver. To achieve this an 'IBD map' was created which stores, for each DEM particle requiring an IBD solution, the memory location of the DEM particle, the memory location of its corresponding IBD solver and the

boundary conditions to be applied to the IBD solver. Routines were then developed to create an IBD solver, to kill an IBD solver and to add/remove boundary sources from an IBD solver.

The second task (b) is needed in order to provide accurate temperature boundary conditions to each IBD solver whose associated particle is involved in a collision. Fig. 5 provides a schematic of two particles, each with an IBD solver, colliding. The contact point is shown by the dark square and the neighboring grid nodes are shown by dark circles. Referring to Fig. 5 this task essentially involves providing accurate temperature boundary conditions at the neighbor grid nodes to the contact (e.g T_A and T_B for the left particle in Fig 1).

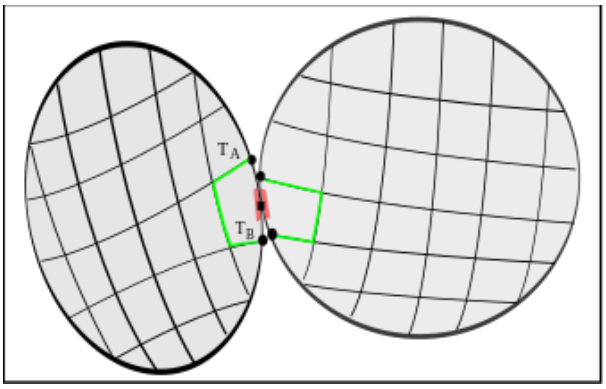

Figure 5. Collision between two SQ particles each with an IBD solver. The contact point is indicated by the dark square and the neighbouring grid nodes in each IBD solver are shown by the dark circles. The proposed heat transfer strategy calculates temperature boundary conditions at the neighbouring grid nodes T_A and T_B.

To calculate these temperatures, the following strategy was formulated. When two bodies collide heat is exchanged across the contact surface area A. Assuming heat flows from body 1 to body 2, the heat flow rate across the surface area out of body 1 equals the heat flow rate into body 2, ie

$$\overrightarrow{q_1} \cdot \overrightarrow{n_1} A = -K_1 \frac{\partial T}{\partial n_1} A = \overrightarrow{q_2} \cdot \overrightarrow{n_2} A = -K_2 \frac{\partial T}{\partial n_2} A \qquad (3)$$

In order to calculate the heat flow rate accurately, the normal temperature gradients $\frac{\partial T}{\partial n_1}$, $\frac{\partial T}{\partial n_2}$ must be accurate. This requires knowing the temperature at the contact point and the temperature distribution in the interior of the particle near the contact point. This is challenging because an estimate of the contact temperature and temperature distribution is known only at the previous time-step and only at discrete grid points within the body. This may not accurately represent/capture the sharp temperature changes that occur near the contact as heat is transferred. The assumption was therefore made that within a small distance ΔS from the contact, the temperature distribution is represented by the analytic solution of one-dimensional heat conduction in a finite slab (of width ΔS),

$$T(x,t) = T_c(t) - (T_c(t) - T_s)\left(\frac{x}{\Delta S} + \frac{2}{\pi}\sum_{n=1}^{\infty} sin\left(\frac{n\pi x}{\Delta S}\right) exp\left(-\left(\frac{n\pi x}{\Delta S}\right)^2 \alpha t\right)\right) \quad 0 < x < \Delta S \qquad (4)$$

Here $T_c(t)$ is the temperature at the contact point and T_s is the temperature at the edge of the slab at the start of the contact. The width of the slab ΔS is assumed much smaller than the IBD grid

cell size and is shown by the small red regions in Fig. 5. An approximation to the normal temperature gradient can be calculated using a finite difference discretisation of Eqn. (4),

$$\frac{\partial T}{\partial x}\Big|_{x=0} = \frac{(T(\Delta x,t)-T_c)}{\Delta x}$$

where Δx is chosen to be a fraction of the diffusion length scale, so that a finite difference approximation for the gradient of the 1D heat conduction profile in Eqn. (4) is valid. Specifically

$$\frac{\Delta x}{2\sqrt{\alpha t}} = 0.25.$$

Substituting Δx into Eqn. (3) provides the following estimate of the contact temperature $T_c(t)$,

$$T_c(t) = \frac{\left(K_2 T_2(\Delta x,t)+K_1 T_1(\Delta x,t)\right)}{(K_1+K_2)}$$

To update the neighbouring grid point temperatures (ie T_A and T_B in Fig. 5) during the collision it is assumed a continuous heat point exists at the contact which delivers heat radially outward at a rate $Q(t)$ during the contact. This heat source applied at the contact point for body 1 at time t delivers the following heat energy

$$Q_1(t) = \overrightarrow{q_1}\cdot\overrightarrow{n_1}A \approx -K_1\,\frac{\left(T_1(\Delta x,t)-T_c(t)\right)}{\Delta x}A$$

The temperature change at a radial distance r from the heat source is then assumed to obey the one-dimensional unsteady solution for a continuous heat point source

$$dT(r,t) = \frac{1}{8\rho c_p(\alpha\pi)^{3/2}}\int_0^t \frac{Q(t)}{(t-\tau)^{3/2}} exp\left(-\frac{r^2}{4\alpha(t-\tau)}\right) d\tau \qquad (5)$$

Eqn. (5) is then used to update the neighboring grid point temperatures (e.g T_A and T_B in Fig. 5) at each time-step during the collision. These updated temperatures are applied as Dirichlet temperature boundary conditions in the IBD solver. Future work will involve implementing this strategy in the DEM solver in order to finalise the fully coupled DEM-IBD solver.

3. Extending the IBD/DEM Coupled Solver to Hybrid CPU/GPU Architectures :

An important part of the grant was to investigate the use of GPUs to perform the IBD solver calculations, leaving the DEM (inter-particle) collisional calculations for the CPU. A non-trivial DEM simulation will typically have millions of interacting particles at each time-step. When combined with the IBD solver, each of these particles will have a body-fitted grid, typically of size $32 \times 32 \times 32$ grid cells. Assuming double-precision temperature data is stored at each of these grid points, a coupled DEM-IBD simulation applied to one million particles would require ~ 260 Gb of memory with one million IBD solutions calculated per time-step. So these solutions need to be fast and need to fit in the available memory.

The IBD diffusion solve is expected to suite the GPU architecture. Once temperature BCs are calculated in the DEM solver (see Eqn. (5)), the diffusion solve in the particle interior is independent of any other information exterior to the particle domain. This operation has a high

degree of inherent parallelism which is naturally suited to the GPU architecture. In contrast, the traditional DEM solver, which solves for millions of dynamic collisions between particles does not have this high degree of parallelism. Typical DEM simulations also require significant memory, more so than what is available on the GPU. A simplified schematic of this hybrid CPU/GPU implementation is shown in Fig. 6.

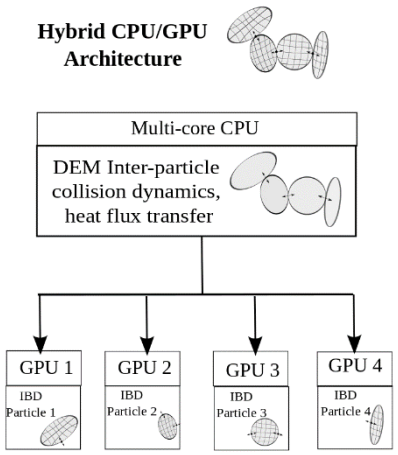

Figure 6. Coupled DEM-IBD solver implemented on a hybrid CPU/GPU architecture.

In the design of the coupled DEM/IBD solver implementation we accounted for the following factors:

(a) Cross-language interoperability. The traditional DEM solver is written in F90. The most convenient (readily usable and available) language to implement our GPU solvers with is OpenCL [7]. OpenCL is the open standard for parallel programming of heterogenous systems and has standardised C and C++ APIs. We used ISO C bindings to communicate data between the F90 DEM solver and the OpenCL GPU solvers. This meant that the data types transferred between the F90 and OpenCL solvers had to be standard integers and floats/doubles rather than complex structures and derived data types.

(b) Minimization overheads in transferring data between DEM and IBD solvers. Fig 5 shows that the only information an IBD solver needs from the DEM solver is the temperature data at the grid nodes nearest to the contact. So the data transferred between the DEM and IBD solvers for each colliding particle needs only to be composed of the neighbouring grid node indices and the temperature data at these grid nodes. Two vector structures were built, one to store the neighbouring grid node indices and one to store the temperature data at these grid nodes. For each collisional time-step, these two vectors were assembled in the DEM solver and unpacked into temperature boundary conditions for each IBD solver. The IBD solvers were then integrated forward in time with these temperature boundary conditions, providing updated internal particle temperature fields. The updated temperatures at the grid nodes neighbouring the contacts were then packed back into the temperature vector which was returned to the DEM solver in preparation for the next collisional time-step. As these vector structures are composed of integer and floats they are easily transferred between the F90 DEM solver and the OpenCL IBD solver using ISO C bindings [8].

(c) Efficient memory swapping between CPU and GPU. Analysis of the specifications of the

GPU cluster to be used revealed that the available GPU memory would restrict problem sizes to ~15,000 particles. However, the longer term requirement for this work is to be able to run the IBD solver for millions of particles. In order to be able to access the available memory on the CPU **and** exploit the fast GPU processors an efficient parallel process to swap memory between the GPU and CPU while simultaneously calculating IBD solutions on the GPU was developed. In this process each GPU has a set of buffers that each holds data for a small number of IBD solvers at any one time. Each of these buffers has a corresponding buffer held in 'pinned' CPU host memory. Data transfers between GPU and pinned CPU memory are fast because the GPU device can access the pinned memory directly without additional processing/checking from the CPU. In the memory swapping method, IBD solver data is continuously swapped between the CPU source buffer and the pinned CPU buffers. Data arriving in the pinned host CPU buffers is then transferred to the GPU buffers and computed. Once the computation is completed it is transferred back to the pinned CPU buffers and swapped back to the CPU source buffer. The processes of swapping memory between the CPU source buffer and the pinned CPU host buffer and performing GPU calculations is parallelised using multiple command queues executed with OpenMP.

A schematic of the coupled DEM/IBD implementation using this memory swapping is shown in Fig. 7.

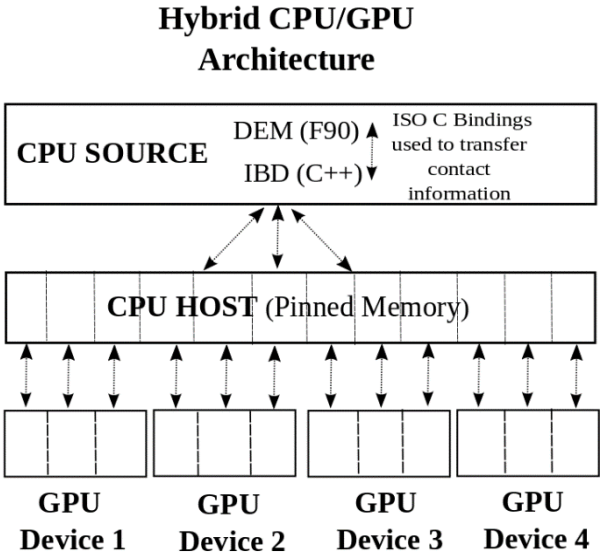

Figure 7. Schematic showing typical data transfers in the memory-swapping method of the hybrid CPU/GPU implementation. Packets of IBD solvers are transferred between CPU source and pinned CPU host memory (parallelised using OpenMp). These packets are then transferred between pinned CPU host memory and the GPU devices. GPU computation and data transfer occur in parallel to ensure the memory swapping method is as efficient as possible.

The performance of the coupled DEM/IBD solver on the hybrid CPU/GPU architecture was tested for problem sizes between $N = 10^4$ particles and $N = 10^6$ particles, with each IBD solver having grid size $32 \times 32 \times 32$ nodes. The GPU machine used in this work has 28 CPU cores, 256 Gb memory and 4 Nvidia P100 GPUs. The maximum problem size that can fit in the available 256 Gb of CPU memory is $N = 10^6$ particles (assuming each particle has an IBD solver of grid size $32 \times 32 \times 32$).

As the collisional heat transfer strategy proposed in Section 2 is still to be implemented and

assessed, testing a fully coupled DEM/IBD solver is not yet possible. So for the purpose of this performance test, each SQ particle is assumed to have 10 contact points with specified neighbour grid node temperatures. The average time taken per SQ particle to run one DEM/IBD iteration was measured where a DEM/IBD iteration involves transferring temperature data from the DEM solver to the IBD solver, running the IBD solver for one time-step and transferring the updated surface temperature data from the IBD solver back to the DEM solver. The performance was compared using the CPU only (ie no GPUs used to perform the IBD solver calculations) and using one, two, three and four GPUs to perform the IBD solver calculations. Fig. 8 shows the timing results on a log-log scale. The use of all four GPUs results in a speed-up of ~14 compared to using the CPU only. This speed-up is relatively consistent with the number of particles. A significant speed-up is noted between using CPU only and using one GPU (speed-up ~ 9). The speed-up between using one and two GPUs is ~ 2 which is almost linear. However, the speed-up reduces as the number of GPUs increases such that the speed-up between using one GPU and four GPUs is ~2.7. It is suspected that this reduction in speed-up is due to the transferal of data between the temperature vector buffers and the IBD solvers (which is serial and occurs on the CPU). Further work will involve more detailed profiling of this performance.

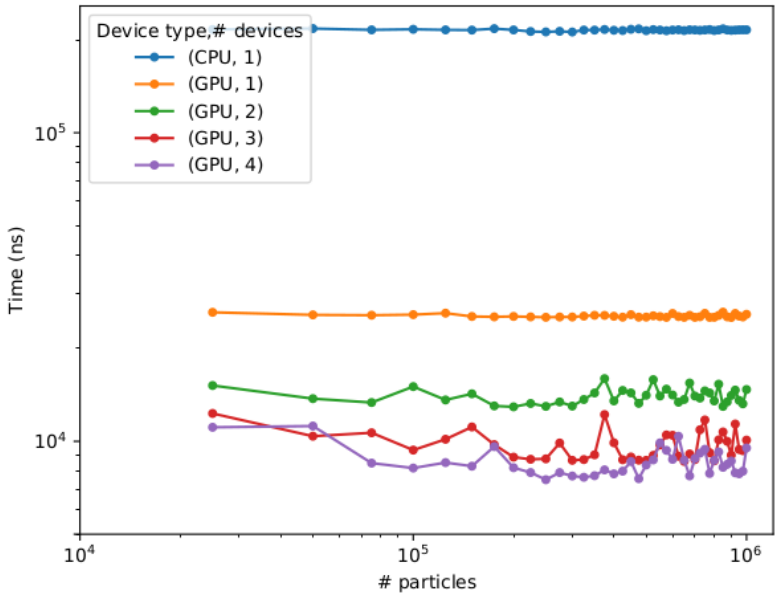

Figure 8. Timing comparison for the coupled DEM/IBD solver on the hybrid GPU/CPU architecture for problem sizes between $N = 10^4$ particles and $N = 10^6$ particles.

Future Work and Discussion:

Significant progress has been made towards achieving a capability to model intra-body heat transfer within SQ particles in DEM simulations. The work has entailed the following advances:

(a) Development of a body-fitted diffusion solver (IBD solver) able to simulate heat diffusion within SQ particles of variable shape.
(b) Design and implementation of the coupling between the IBD solver and CSIRO's DEM solver that is able to be executed on hybrid CPU/GPU architectures.
(c) Development of a memory-swapping method to allow the coupled DEM/IBD solver to be run efficiently for large problem sizes (ie for problem sizes that are limited only by the available CPU memory).

(d) Proposed strategy to model heat transfer between arbitrarily-shaped particles with body-fitted grids.

A 14-fold speed-up was noted in using the hybrid CPU/GPU implementation to run the coupled DEM/IBD solver versus running purely on the CPU. This suggests that use of hybrid CPU/GPU architectures is promising for the simulation of intra-body heat transfer in granular materials but further profiling and analysis is needed. The next stage of work will involve implementing and testing the proposed heat transfer strategy between colliding particles which will complete the development of the fully coupled DEM/IBD solver. The capability will then be applied to study heat conduction in more realistic granular materials.

References:

[1] P.W. Cleary, G. Metcalfe, K. Liffman, How well do discrete element granular flow models capture the essentials of mixing and segregation processes, Applied Mathematical Modelling, 22 (1998) 995-1008

[2] F. Debroux, P.W. Cleary, Characterising the angles of failure and repose of avalanching granular materials using the discrete element method, Proceedings of 6th World Congress of Chemical Engineering (2001), paper 1537

[3] P.W. Cleary, The effect of particle shape on simple shear flows, Powder Technology (2007)

[4] P.A. Cundall, O.D.L. Strack, Discrete numerical model for granular assemblies, Geotechnique 29 (1979) 47-65

[5] M. Koo, D. I. Leap, Modeling three dimensional groundwater flows by the body-fitted coordinate (BFC) method: I. Stationary boundary problems, 30 (1998) 217-239

[6] Mapping a Cube to a Sphere. http://mathproofs.blogspot.com.au/2005/07/mapping-cube-to-sphere.html

[7] OpenCL overview. https://www.khronos.org/opencl

[8] ISO C Binding. https://wiki.scorec.rpi.edu/wiki/Iso_c_binding