Simulating Particle Agglomeration with a GPU

Elijah Andrews, supervised by Professor John Shrimpton

238 AM

Aim and Objectives

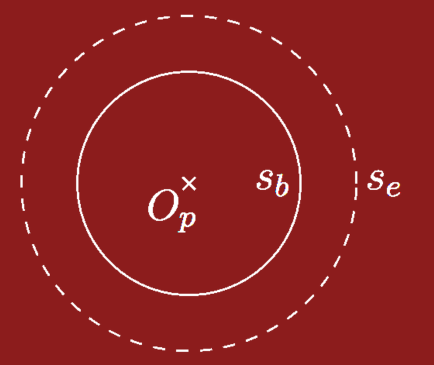
The aim of this project is to analyse how particle agglomerates behave by developing a simulation to use the Discrete Element Method on a GPU. To achieve this:

- An initial simulation was developed to understand the methodologies required.
- An advanced simulation was developed for a GPU.
- Analysis was performed on particles simulated to form agglomerates.

Simulations can take a lot of time to run so a GPU is used to accelerate the simulation. CPUs perform lots of different calculations in sequence whereas GPUs perform the same calculation on a lot of data in parallel. This means GPUs are well suited to speeding up simulations where the same calculations are run many times on lots of data. This project uses OpenCL, a system for interfacing with GPUs that works on many platforms.

The Discrete Element Method (DEM) is used to simulate particle behaviour. The DEM models particles as individual elements. Forces on the particles are modelled as spring/dashpot arrangements where the spring compression is the overlap distance between particle surfaces.

In this project particles are assumed to be spherical, without any rotation, and to have no effect on the fluid they are in. The forces considered in this project are contact forces (normal and tangential), a drag force, and a cohesion force between particles.



This diagram (left) shows how a particle is defined.

- O_p is the particle origin
- s_b is the body surface (solid)
- s_e is the effect surface (dashed)

Normal Contact Force

The normal contact force is a spring-dashpot arrangement based on the overlap of the particle body surface using a simple linear model.

Cohesion Force

The cohesion force is a linear attraction force between particles proportional to the overlap distance of the particle effect surface.

Drag Force

The drag force uses a Stokes drag model which assumes very low Reynolds number.

Tangential Contact Force

The tangential contact force uses a complex friction model with static and dynamic friction regimes.

The DEM Algorithm

The DEM algorithm is usually split into four sections. These sections are outlined below and shown in the flowchart (right).

Each section is reduced into individual kernels (blocks of code) to be passed over the data. For example, there is one kernel that runs on each collision to resolve it and another that runs on each particle to iterate it.

Simulation Setup

This is where all of the geometry and initial conditions are set.

Broad Phase Collision Detection

Broad phase collision detection is designed to reduce the number of collisions that have to be resolved by determining which collisions could happen. In this project the 'spatial zoning' algorithm is used. Spatial zoning is explained further in the next column.

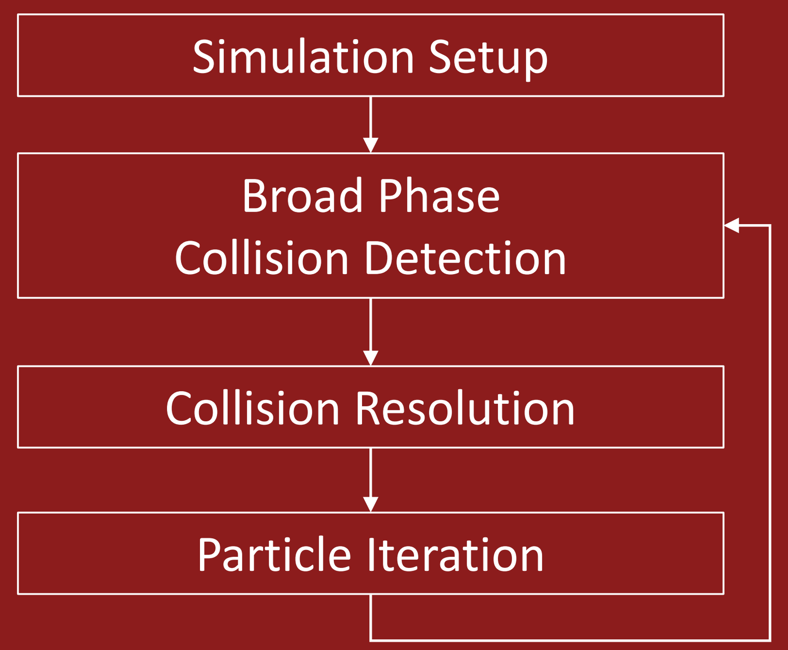
Collision Resolution

Each possible collision is resolved by calculating the forces on each particle.

Particle Iteration

Particles are iterated using various numerical methods. Velocity is iterated with a backwards Euler method and position is iterated with trapezoidal integration.

Algorithm Flowchart

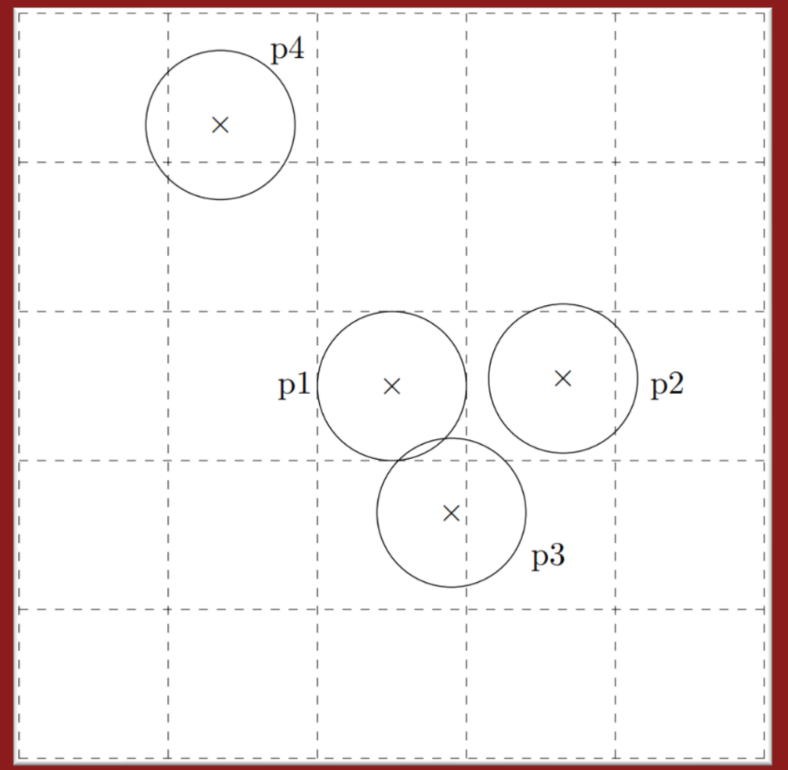


Spatial Zoning

Spatial zoning separates the domain into control volumes. Particles are sorted into control volumes. Particles in neighbouring, or the same, control volume could be colliding and so are resolved. Control volumes have to be at least as big as the biggest particle to ensure that all possible collisions are found.

The number of operations required in this approach is approximately proportional to the number of particles in the simulation.

Spatial Zoning Example



This diagram shows an example of how particles could be positioned in control volumes (shown by the dashed grid).

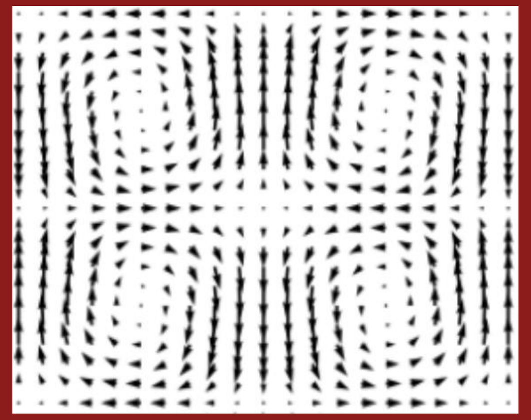
The broad phase collision detection would determine that p1, p2, and p3 could be colliding.

Collision resolution would show that only p1 and p3 are colliding. Collision forces would then be calculated.

Running Simulations

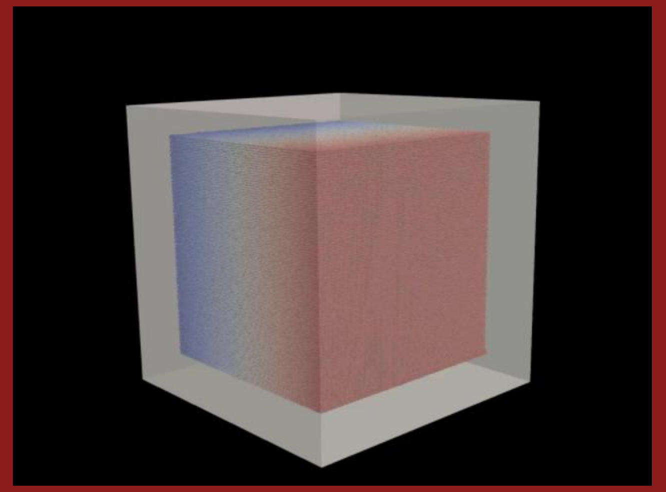
Simulations are limited by the types of geometry available. In this project only spherical particles and axis-aligned walls are available.

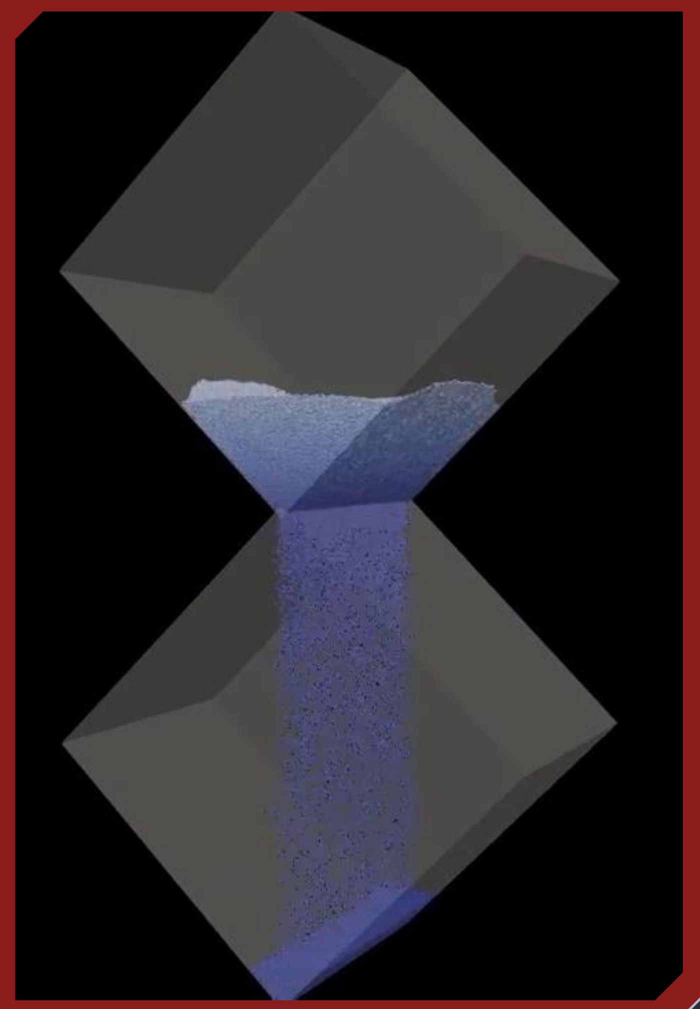
Simulations in this project can have different gravity and flow fields. This allows more varied simulations to be run. For example, the gravity vector can be changed so that more interesting wall geometries can be considered such as the hourglass in the animation (right, bottom). This also allows for interesting flow fields to be considered, such as the Taylor-Green Vortex flow used for agglomeration analysis.

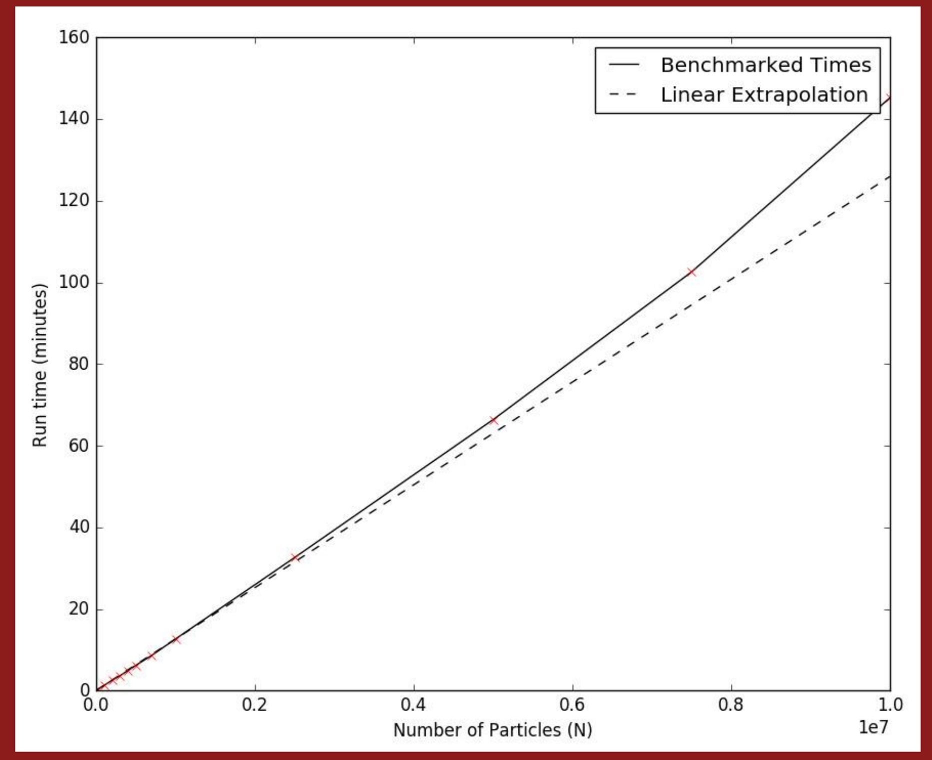


A Taylor-Green Vortex flow field in the z-y plane.

The vortices are exactly bounded by a cubic domain and bring particles together well. This is good for agglomeration analysis because the rate of collision is higher than in many other flow fields.







This graph shows how the time it takes to run a simulation varies with the number of particles (N). The dashed line is a linear extrapolation through N=0 and N=1e4. The solid line is the measured times. A simulation without broad phase collision detection would be $O(N^2)$ because every particle must be resolved with every other particle. This shows that the simulation design allows it to run in almost O(N) time.

The overall performance of this simulation was within an order of magnitude of much more advanced simulations. This was achieved by reducing usage of expensive operations and making as many operations parallel as possible.

Agglomeration Analysis

The objective of this analysis is to find how agglomerates are affected by changes in simulation properties. The Stokes number (Stk) and the Stickiness number (Sy) are investigated in this analysis.

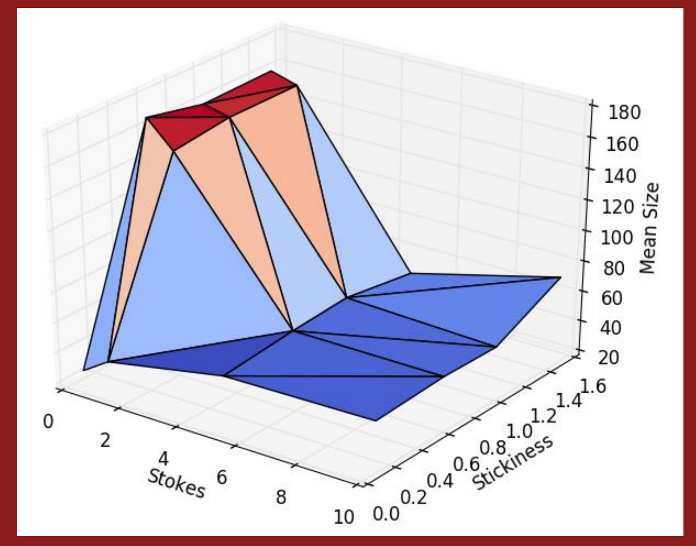
Stokes Number

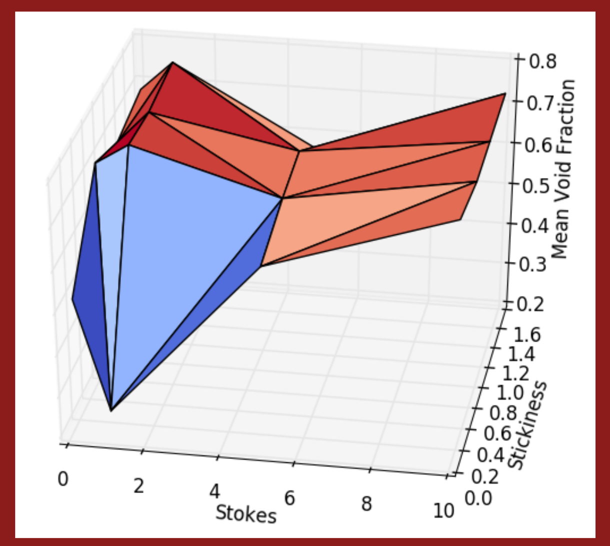
The Stokes number is a measure of how the particles react to the fluid flow. A low Stokes number means that the particles closely follow the flow. A high Stokes number means that the particles are relatively unaffected by the flow.

Stickiness Number

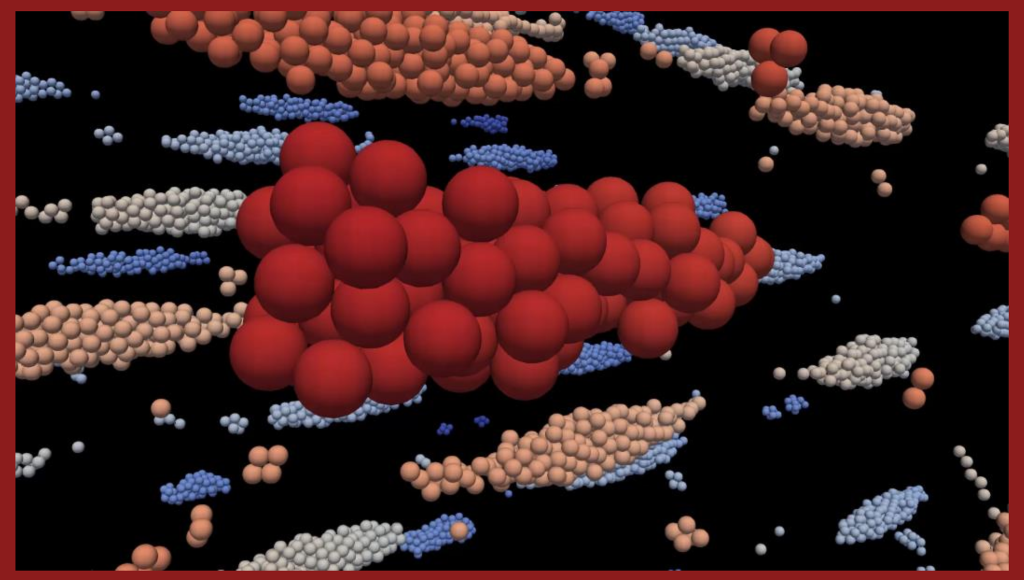
The Stickiness number is a measure of how likely particles are to remain stuck together after a collision. A low Stickiness number means that particles are unlikely to stick. A high Stickiness number means that particles are likely to stick.

Two agglomerate properties were measured: the mean agglomerate size, and the agglomerate void fraction. The void fraction is given by how much of a bounding sphere of an agglomerate is filled with particles. The results are in these graphs (below). The large spikes at low Stokes numbers are caused by the particles immediately getting trapped by the vortices and not being able to escape. These results form a good basis for future analysis of similar phenomena. A focus on the transitional area between Stk=0 to Stk=1 and Sy=0 and Sy=0.5 would be most interesting.





Individual agglomerates are formed and broken up throughout a simulation. However, the overall simulation comes to a statistically steady state, where properties like mean size tend towards a value.



This animation shows how an agglomerate is broken up and reformed constantly in a simulation. Although the agglomerate itself is changing, its statistical properties (e.g. size) are not.

Conclusion

This project was successful. All of the objectives were met and the analysis that was performed can be used by future researchers to understand the general behaviour of particles with varying Stokes and Stickiness numbers. The simulation tool can be used for general particle simulation and can also be further developed to improve its performance and capabilities.