Internship Presentation

Ojas Phadake, Dr. Aqib, Prof. Nott

June 28, 2024

Introduction

My topic title was:
'Understanding the plastic flow inside a grain heap during the process of deposition'.
1) What are granular materials?
It is a collection of solid particles or grains, such that most of the particles are in contact with at least some of their neighboring particles.

They are a collection of **discrete, rigid and microscopic grains**. This means:

- **Discrete:** Large enough that Brownian motion or thermal fluctuations are not important
- **Rigid:** Young's modulus » Pressure. This means there is no deformation.
- **Microscopic Grains:** Their particle diameters d_p are more than 100μm

Granular materials are when diameter is $> 100\mu$m

Introduction (cont'd)

2) What is different about them?

1. We typically use a method called DEM
2. Pressure Head Independence (Exponential saturation)
3. Force Chains (Stress transfer along specific directions)
4. Material of suspension (dry, saturated)
5. Angle of Repose

3) On DEM and it's applications:

Applications of **DEM** occur in pharmaceutical tableting, packaging and flow simulations, landslide prediction, and impact analysis.
DEM allows a more detailed study of the **micro-dynamics of powder flows** than is often possible using physical experiments.

Introduction (cont'd)

My project involves understanding the various factors such as **force chains, force lines, stress relationships, velocity,** etc. I did was the post-processing of LIGGGHTS data using MATLAB and also performed simulations. My task was to validate the experiments done by Aqib Sir numerically and also understand the theory myself.

The geometry which we studied extensively has been described as below:

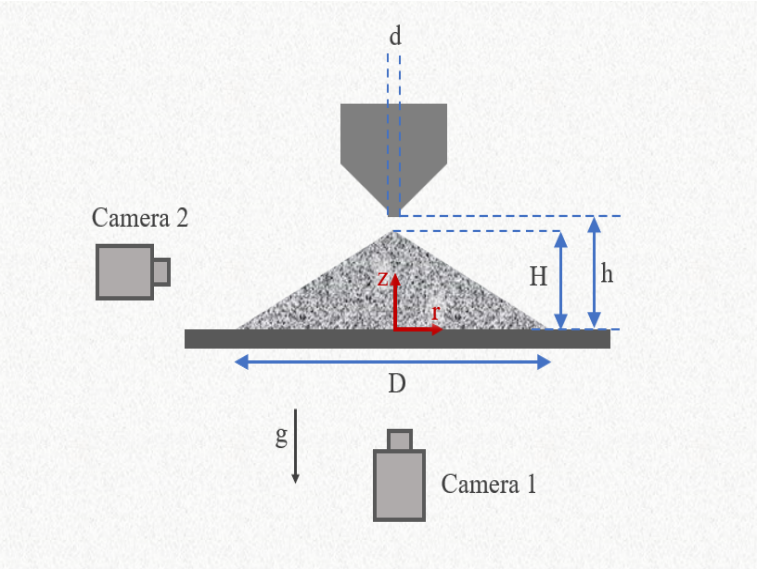

Figure: Experimental Setup Source: Dr. Aqib's presentation

Week 1

- Installed *LAMMPS*
  - Ran the basic *flow* examples in it
  - Familiarised myself with the documentation
- Installed *OVITO*
  - Uncommented the *dump* commands in the LAMMPS Script
  - Visualised the data in OVITO
- Downloaded *LIGGGHTS* on my computer, but could not run the heavy computation on my laptop.
- Familiarised with debugging scripts, changing commands and understanding their impacts, transferring the *post* data to local machine

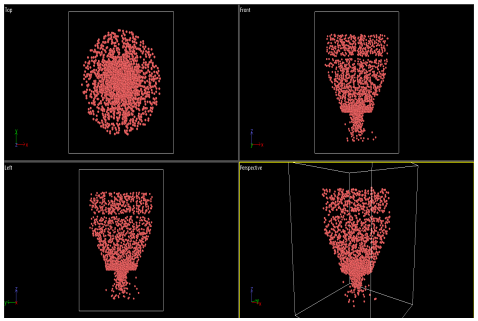

Figure: The first simulation visualised

Week 2

1) Modelling Methods

Multiple particle-particle interaction methods exist:

- Hookean Model
- Hertzian Model

They can be modelled using a spring and dashpot model, or a series-parallel combination of it. I also understood that we model deformation of particles using *overlap*. In the **Hookean model** the normal force is modelled as:

$$\mathbf{F}_n = -(k_n \delta_n \mathbf{n} + C_n \mathbf{v}_n)$$

It is through the stiffness that particles *see* each other elastically.

Week 2 (cont'd)

2) An initial attempt at post-processing

I initially read the data line-by-line, then understood that that is very slow. Finally used the function *importdata* to read the dump data fast.

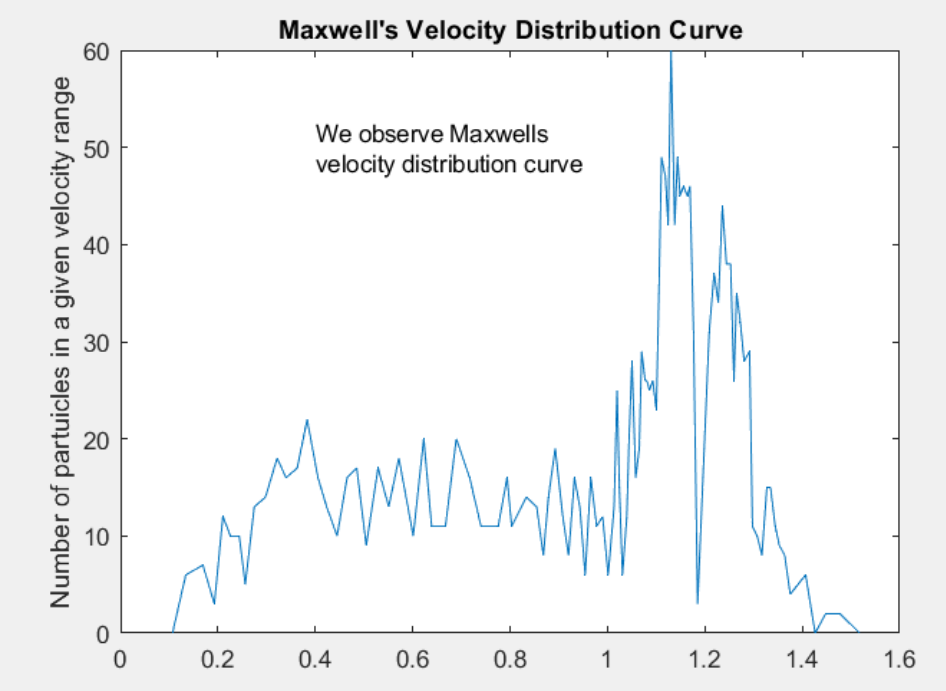

Figure: Attempt at replicating Maxwell's velocity distribution curve

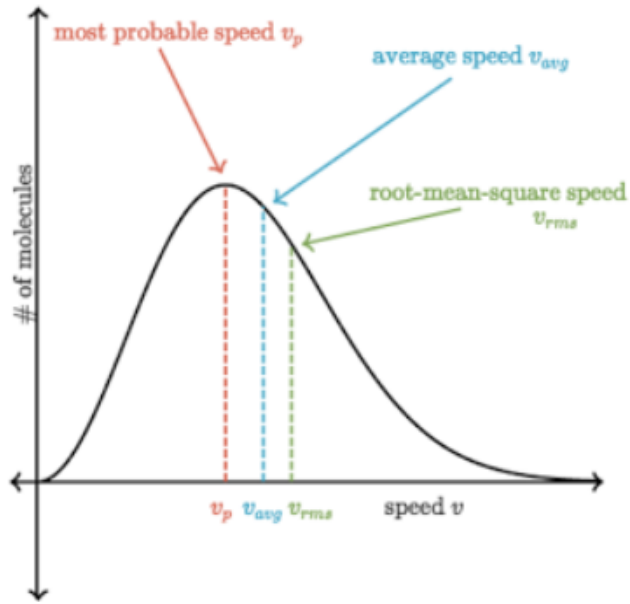

Figure: Actual curve in ideal gases(Source: Khan Academy)

Week 2 (cont'd)

3) Plane and Line Extraction Codes

- *fix* file was given to me, containing $x, y, z, v_x, v_y, v_z, f_x, f_y, f_z, \tau_{xx}, \tau_{yy}, \tau_{zz}, \tau_{xy}, \tau_{xz}, \tau_{yz}$ as bin data and I had to extract the data of a specific plane. (I chose the $yplane == min(y(y>0))$ plane)
- Contour plot was an important learning curve as I got to play around with the *unique* function.
- I was also able to extract the data on a single line along Y midplane.
- As the system is axisymmetric, I performed **spatial averaging** in r and z directions to summarise the entire system in a half-plane data thus leading to accurate averaging.
- I also learned how to create a time series data and analyse it.

Week 2 (cont'd)

4) Physically accurate plots obtained from codes

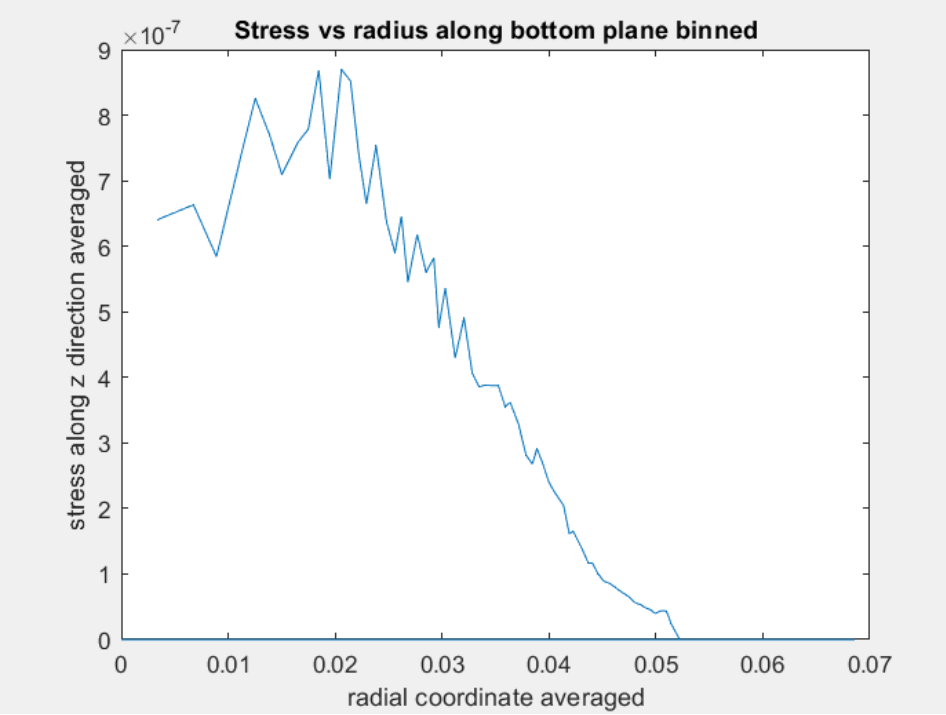

Figure: τ_{zz} Stress radially plotted

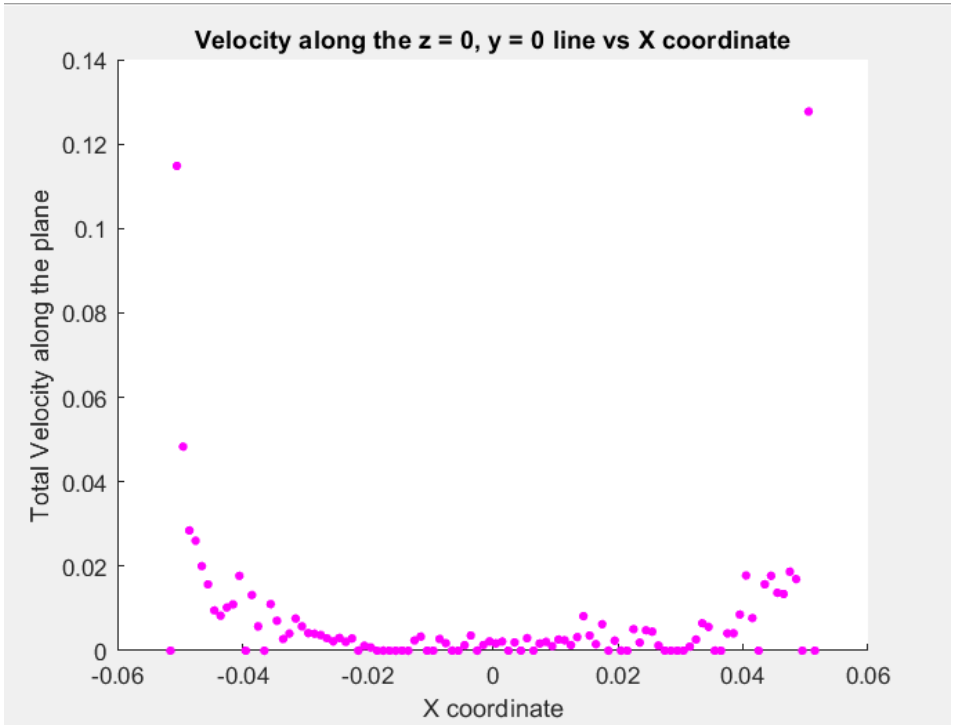

Figure: Total velocity along bottomplane's midline

We can see the presence of a stress dip, and also that the stress maxima arises at $0.33r/R$ radial length.

Week 2 (cont'd)

5) Binning

- Understood the concept of binning
- Engineered a code to create meshes, appropriate bins of desired size and calculate contact and streaming stresses and volume fraction of that bin.
- Created a short movie in MATLAB of time-series data to visualise granular temperature against time variation.
- Here is the movie of total velocity at the bottom plane.

Week 3

- Wrote toy codes for calculating average coordination number and the angle of repose.
- Obtained the coordination number i.e. the number of nearest neighbours to be **4.733** which is physically correct, and repose angle as **33.3 degrees** using appropriate criterion to figure out radius and height of the heap.
- Understood how PIVLab works. It is a peak of the correlation of multiple particles and their velocities.

Week 4

Force Chains

Experiments on static beds of granular materials show that the stresses are transmitted along preferred directions, which are called force chains or stress chains.

- My task was to find out the force chains in such a granular heap
- I used *overlap* between 2 particles as the measure of the amount of force.
- Despite the force equation being $\mathrm{F}_n = -(k_n\delta_n + C_n\mathrm{v}_n)$, we can ignore C_n as it is very small and that leads to $F_n \ \alpha \ \delta$
- I have modelled them using the exact same method and this was also done by Dr. Ravi Gautam in his thesis.
- I however take note that because he was studying a steady state system, hence some inaccuracies will be there

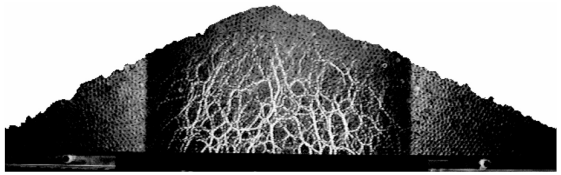

Figure: A heap formed by pouring photoelastic disks of two sizes from a funnel onto a plate. Reproduced from Vanel et al., 1999

Week 4 (cont'd)

My task was to try and predict the force chains accurately in a 3D heap. My initial try was much faster than the traditional method. I used the MATLAB matrix operations to a much better effect, needing matrices of a size 99.71% smaller than the traditional way.

The method was as follows:

```
for i=1:length(x)
    x1 = x(i); y1 = y(i); z1 = z(i);
    xfluc = x1 - x; yfluc = y1 - y; zfluc = z1 - z;
    dist = sqrt(xfluc.^2 + yfluc.^2 + zfluc.^2);
    bool = dist<dp;
    contact_particles = find(bool==1);
    % Store i and contact particles and dist in a matrix
end
```

However, on talking to Ravi Sir, I decided to follow the traditional path of *adjacency matrices* for ease of computation.

Force Chain Initial Attempts

An adjacency matrix is a $num_{particles} * num_{particles}$ matrix, and it stores the overlap (if any) in the columns. As each particle has 3-8 contacts usually, hence most of the space is filled with 0, and it is a *sparse matrix*.

Attaching a few plots for the 3D heap below:

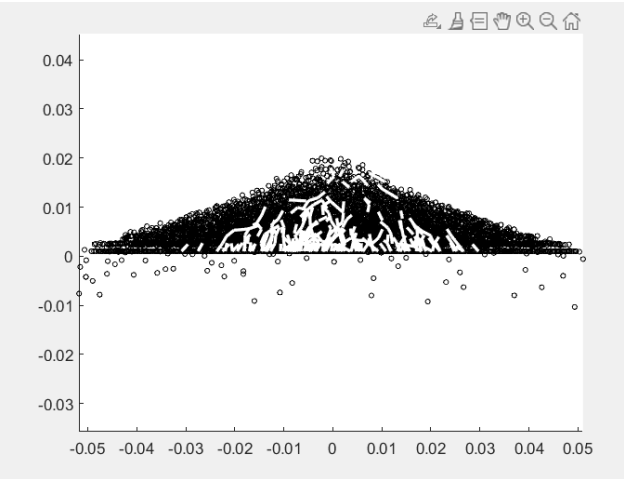

Figure: All particles and highest contacts

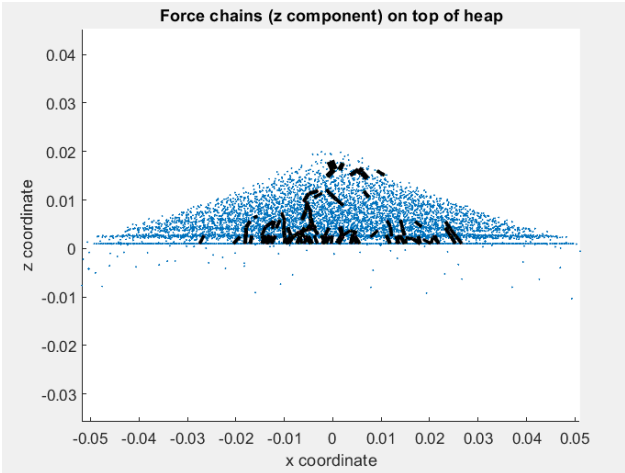

Figure: Another initial attempt

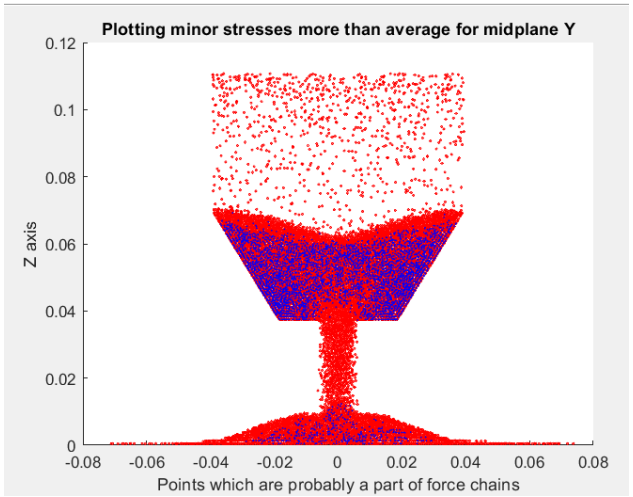

Figure: Force hotspots which have highest load

Further on Force Chains

A force chain is a "**quasilinear particle assembly** where stress is concentrated". The proposed minimum number of particles required for an **"assembly"** is three.

1. Initially, I implemented adjacency matrices using a cutoff, however it was arbitrary and couldn't get meaningful results.
2. I tried finding the chains for a single plane.
3. I also implemented the paper by Peters, Muthuswamy (2005). In the paper, they used **most compressive principal stresses** (eigenvalues of the τ tensor) for a 2D Geometry and used those minor stresses above average to outline the force chains.
4. I also used a similar method, with $\theta < 45°$. The obtained plots has been shown beside. Here, θ is the angle between two in-contact particles. This is what is meant by "quasilinear".

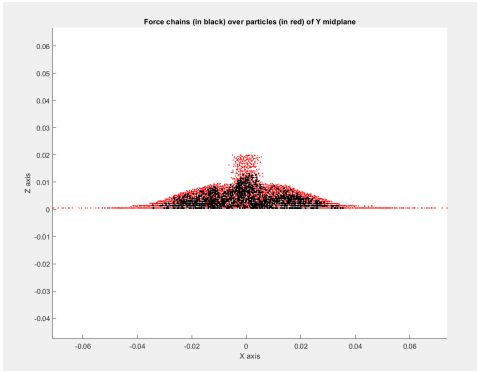

Figure: Nodes having above average minor stress

Discussion with Ravi Sir

- He explained that force lines represent the coarse-grained direction of the grain contact forces
- After binning the vectorial components of the contact forces inside that bin should be added for all such bins.
- Streamlines for such bins when plotted will give us force lines.
- However, even they are crude and when such averages for multiple realizations are done, that will give the correct averaged force lines.
- As the geometry I was studying was 3D and the $num_{particles}$ were around 10^6, it will take many days to actually get meaningful and stastically accurate results.
- Hence, Aqib Sir and I decided that we should simulate the same in wedge-shaped hoppers (2D) geometry. It has x going from -0.051 to 0.051, y going from -0.0081 to 0.0082, and z going from -0.001 to 0.13.
- This allows us to get much faster results for the same geometry.

Week 5

Brief Code for the 2D simulation method

```
for i=1:size(adj, 1)
    for j=i+1:size(adj, 2)
        adj(i, j) = dp - sqrt( (x(i) - x(j))^2 + (y(i) -
            y(j))^2 + (z(i) - z(j))^2 );

        if(adj(i, j) < 0)
            adj(i, j) = 0; % particles do not touch
            continue
        end

        adj(j, i) = adj(i, j);

        xvec(i, j) = x(i) - x(j); xvec(j, i) = x(j) - x(i);
        zvec(i, j) = z(i) - z(j); zvec(j, i) = z(j) - z(i);
    end
end
```

Week 5 (cont'd)

I used the following matrices to calculate the vector components of force on each particle due to contact.

$$x_{mat} = adj.\sqrt{\frac{xvec^2}{xvec^2 + zvec^2}}.sgn(xvec); z_{mat} = adj.\sqrt{\frac{zvec^2}{xvec^2 + zvec^2}}.sgn(zvec); \tag{1}$$

The $sgn(x)$ ensures that the summation is **geometric** and **not algebraic**. Summing up all the values column-wise will give us the vector addition of all the forces in the x and z directions respectively.

Final Code

Make **BINS**.
for bin in **BINS**:
    find which particles lie inside the bin
    Add up all x and z components of such particles
    Store these values
end

Week 5 (Results)

The above code was used to find the *force lines* of multiple geometries.
Initial geometry was of **1**) Small Orifice and the second one (to validate) was of **2**) Raining in heaps.
The results are as below:

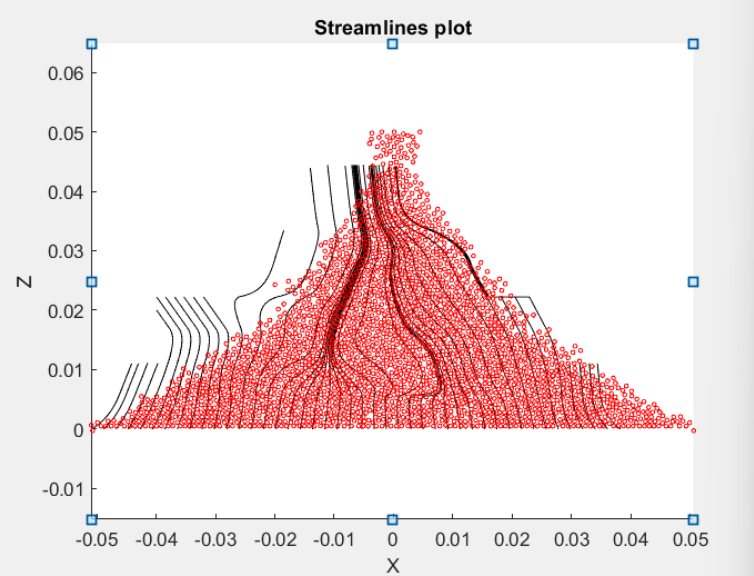

Figure: Narrow Orifice heap

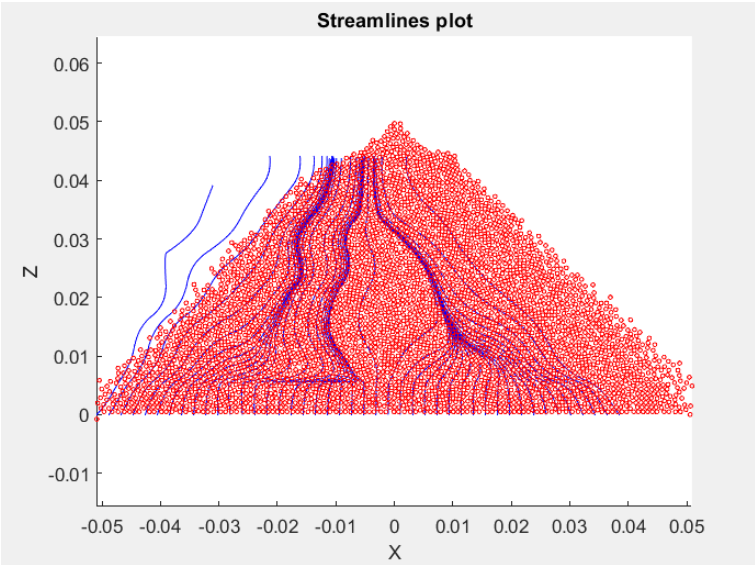

Figure: Heap formed due to granular rain

Few more plots...

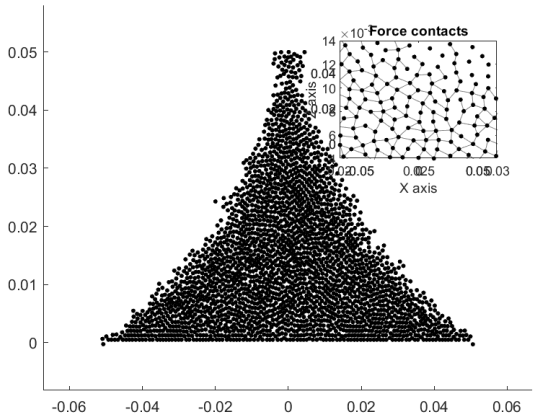

Figure: Contact network

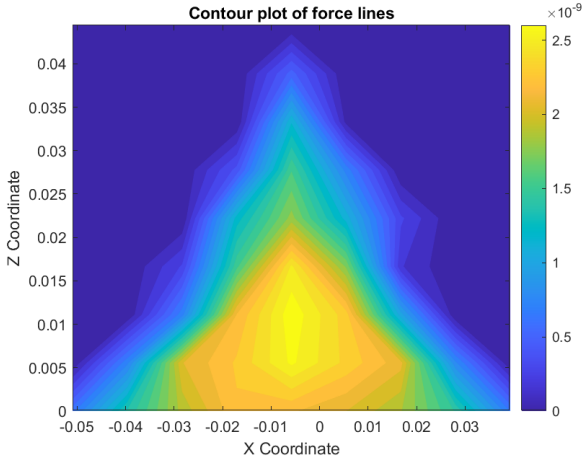

Figure: Contour plot of force lines and the amount of overlap

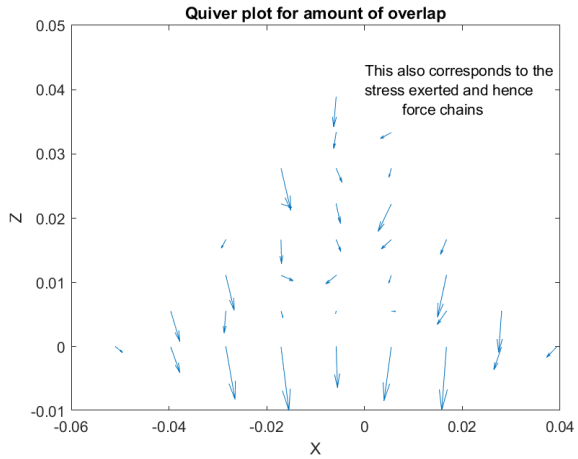

Figure: Quiver plot for contact forces in $-z$ direction

Future Work | Week 6

- **Force Chains** will also give information relevant to the experiment. I have to try working on it in the future so that accurate chains can be obtained.
- Relating the stresses with the outward heap velocity is important to gain a more deeper understanding on the subject. That will help in explaining the *outward motion of glass beads* as shown in Aqib Sir's experiments.
- **Omid Ejtehadi, Aashish K. Gupta (2024)** have released an open source tool for finding force chains in C++ and Python as well. I will try to implement that as well and see if it gives accurate results matching with my plots and force fields.

References I

K P Krishnaraj, P. R. Nott (2020) | Coherent Force Chains in Disordered Granular Materials Emerge from a Percolation of Quasilinear Clusters, published in *Phys Rev Lett.*

K P Krishnaraj, P. R. Nott (2023) | The stress in static granular media under gravity, published in *J. Fluid Mech.*

O. Ejtehadi, A. Gupta, S. Khajepor, S. Haeri (2024) | Force-chain finder: A software tool for the recursive detection of force-chains in granular materials via minor principal stress, published in *Computer Physics Communications.*

Cundall, P.A. and Strack O.D.L. (1979) | A Discrete Numerical Model for Granular Assemblies, published in *Geotechnique, 29, 47-65.*

K. Rao, P. R. Nott (2016) | An Introduction to Granular Flow

J. F. Peters, M. Muthuswamy, J. Wibowo, & A. Tordesillas (2005) | Characterization of force chains in granular material, published in *Phys. Rev..*

R. Gautam (2023) | Micromechanical understanding of the macroscopic features of dense granular materials, *PhD thesis*

H.A. Janssen (1895) | Experiments of Corn Pressure in Silo Cells

Thank You

- **Prof. Prabhu**, for accepting me as an intern under his supervision
- **Dr. Aquib**, for guiding me throughout and testing me with interesting problems
- **Gautam**, **Sanyogita** and **Neha Ma'am**, for the numerous discussions and helping me whenever I couldn't understand something and wanted their help
- **Megavarshini** and **Abeena**, for helping me gain 2 new friends in them