

## Contents

---

- [Clearing previous data](#)
- [Load the data and start post processing](#)
- [The basic plot for force chains is ready, I need to now plot the streamlines](#)
- [Streamlines plot](#)

```
% Code Summary:  
% This was an initial attempt at finding out force lines. It was physically  
% decent, however there were a few issues regarding the force chains not  
% being properly visible. It was not fully and physically correct as well.  
% Hence, I cleared this, but kept the code stored in case I have to revert  
% back. The code force_chains has the changed versions, however,  
% D2_force_chains contains the physically most accurate code for 2D setup
```

## Clearing previous data

---

```
clc;  
close all;  
clear variables;
```

## Load the data and start post processing

---

```
file = importdata("post1\particles_115300.liggghts", " ", 9);  
data = file.data;  
clear file;  
  
x = data(:, 3); y = data(:, 4); z = data(:, 5);  
zbool = z < 0.05; % Taking only particles in the heap  
  
x = x(zbool); y = y(zbool); z = z(zbool);  
radius = data(1, end-1);  
dp = 2*radius;
```

```
adj = zeros(size(x, 1)); % This creates the adjacency matrix which will store the overlap between particles if any  
xvec = zeros(1, size(x, 1)); % For each particle, this will store the x component of the interaction  
zvec = zeros(1, size(x, 1));  
  
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;  
            continue  
        end  
  
        adj(j, i) = adj(i, j);  
        % xvec(i, j) = x(i) - x(j);  
    end  
end  
  
for i=1:size(adj, 1)  
    rows = find(adj(:, i) ~= 0);
```

```

if isempty(rows)
    continue
end
xvec(i) = mean(x(i) - x(rows)); % Vectorially summed
zvec(i) = mean(z(i) - z(rows));
end

network = graph(adj);

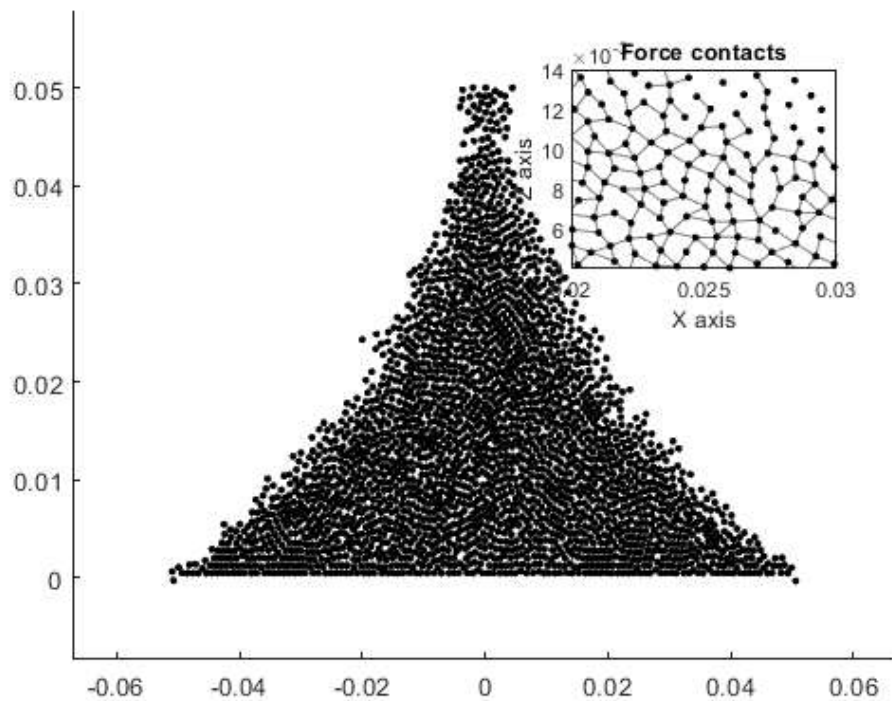
figure(1);
scatter(x, z, 2, "red")
hold on
mainPlot = plot(network, "NodeColor", [0 0 0], "EdgeColor", [0 0 0], 'XData', x, 'YData', z);
% axis equal

insetPosition = [0.6, 0.6, 0.25, 0.25]; % 0.6*xsize
insetAxes = axes('Position', insetPosition);
copyobj(mainPlot, insetAxes);

xlim(insetAxes, [0.020, 0.020 + 0.010]);
ylim(insetAxes, [0.004, 0.004 + 0.010]);
set(insetAxes, 'Box', 'on');
title(insetAxes, 'Zoomed Inset');

xlabel("X axis")
ylabel("Z axis")
title("Force contacts")

```



```

xbins = linspace(min(x), max(x), 10);
zbins = linspace(0, max(z), 10);
node_value = zeros(size(xbins, 2)-1, size(zbins, 2)-1);
u = zeros(size(xbins, 2)-1, size(zbins, 2)-1);

```

```

v = zeros(size(xbins, 2)-1, size(zbins, 2)-1);

for i=1:size(xbins, 2)-1
    for j=1:size(zbins, 2)-1

        rows = find(x > xbins(i) & x < xbins(i+1) & z > zbins(j) & z < zbins(j+1));
        if (isempty(rows))
            continue
        end

        node_value(i, j) = mean(mean(adj(:, rows))); % Averaging the amount of overlap
        xdir = mean(xvec(1, rows));
        zdir = mean(zvec(1, rows));

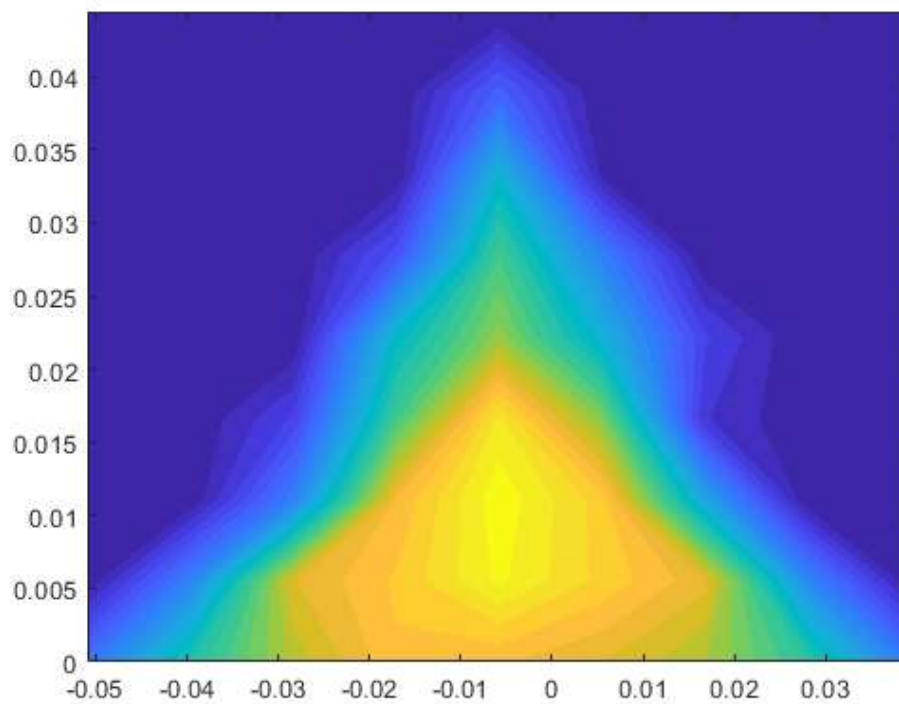
        if(xdir == 0 || zdir == 0)
            continue
        end

        u(i, j) = node_value(i, j)*(xdir)/sqrt(xdir^2 + zdir^2); % u will contain the x component
        v(i, j) = node_value(i, j)*-abs(zdir)/sqrt(xdir^2 + zdir^2); % v will contain the z component

    end
end

[X, Z] = meshgrid(xbins(1:end-1), zbins(1:end-1));
figure(2)
levels = 0:1e-10:max(max(node_value));
shading flat
contourf(X, Z, node_value', 500, 'LineColor', 'none', 'LevelList', levels)

```



The basic plot for force chains is ready, I need to now plot the streamlines

```

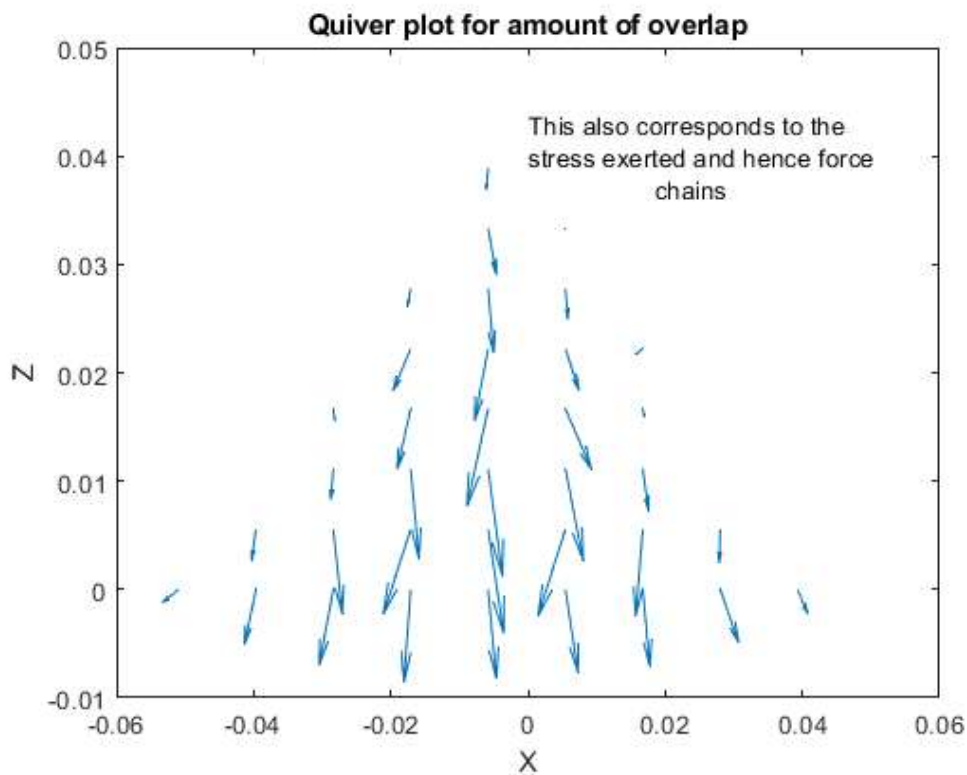
figure(3);
quiver(X, Z, u', v')

```

```

xlabel("X")
ylabel("Z")
title("Quiver plot for amount of overlap")
str = {'This also corresponds to the ', 'stress exerted and hence force', '
chains'};
text(0, 0.04, str);

```



## Streamlines plot

```

% xstart = xbins; % for 10 representative bins
xstart = linspace(min(x), max(x), 50); % for say a higher number of bins, eg 50

[startX,startY] = meshgrid(xstart, zbins(1));

figure(4);
axis equal
lineobj = streamline(X, Z, (u'), abs(v'), startX, startY);

hold on
A = scatter(x, z, 5, "red", "o", "MarkerEdgeColor","flat" );
hold off
xlabel("X")
ylabel("Z")
title("Streamlines plot")

```

Streamlines plot

