

Contents

- [Clearing previous data](#)
- [Data loading](#)
- [Load the data](#)

```
% Code Summary:  
% Even I don't understand what physical significance does binning hold. the  
% array CN seems very arbitrary. However, coordination as an array is  
% perfectly accurate and correct. sum(coordination)/length(x) gives me  
% 4.733 which is a physically correct value as well.
```

Clearing previous data

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

Data loading

```
file = importdata("post\particles_1000000.liggghts", " ", 9);  
data = file.data;  
clear file;
```

Load the data

```
x = data(:, 3);  
y = data(:, 4);  
z = data(:, 5);  
radius = data(1, end-1);  
dp = 2*radius;  
  
bins = linspace(min(z), max(z), 100);  
CN = zeros(1, 100);  
num = zeros(1, 100);  
dist_particles = [];  
  
for i=1:99  
    rows = find(z > bins(i) & z < bins(i+1));  
    num(i) = length(rows);  
end  
coordination = zeros(1, 20);  
  
for r=1:length(x)  
    x1 = x(r); y1 = y(r); z1 = z(r);  
    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);  
    coordination(sum(bool)) = coordination(sum(bool))+1;  
    for j=1:length(contact_particles)  
        if(r == contact_particles(j))
```

```

        continue
    else
        dist_particles = [dist_particles; [r, contact_particles(j), -1*(dist(contact_particles(j)) - dp)]];
    end
end

bin = bins(1:end-1);
A = find(z1<bin);
if (isempty(A))
    continue
end
CN(A(1)) = CN(A(1)) + length(contact_particles);

end

for i=1:20
    coordination(i) = i*coordination(i);
end

for i=1:100
    if(num(i) ~= 0)
        CN(i) = CN(i)/num(i);
    else
        CN(i) = 0;
    end
end
end

```

```

figure(1)
plot(CN)
xlabel("Discretized bin along Z axis")
ylabel("Coordination number of that bin")
title("Coordination Number variation along Z")
inf = "Mean CN: " + sum(coordination)/length(x);
str = {'Maxima are attained almost on base', 'and above orifice plane', inf};
text(50, 10, str)

```

