

MCSC 6040

Macromolecular crowding directs the motion of small molecules inside cells

Andree Qi
Instructor: Prof. de Haan
Dec.7 2020

Contents



Motivation

What does this model do?



Model Setup

How the model is set up?



Case I: Symmetric Gaussian Crowders



Case II: Asymmetric Crowders



Q & A

01

Motivation

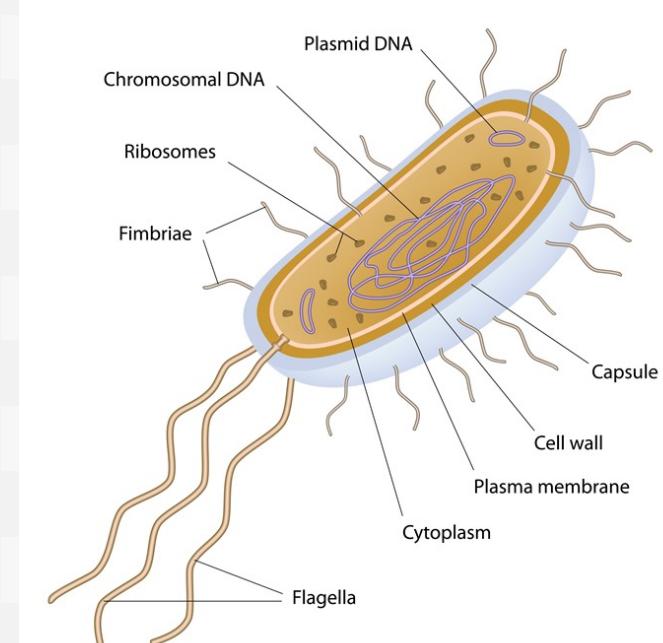
What does this model do?
Why we are interested in this model?

Motivation

Q: How does Macromolecular crowding affect the diffusive motion of small molecules?

If we are inside of a cell:

- There are a lot of Macromolecules inside a cell
 - e.g RNA, ribosomes and enzymes
- These Macromolecule are usually crowded together
 - Macromolecular crowding
- The cell is not a homogeneous environment
 - In prokaryotes, many macromolecules are gathering around the nucleoid
 - Hydrophobic and electrostatic interactions between different macromolecular species may lead to distinct regions of high and low crowder density
- There are small molecules that are also moving inside the cell
 - e.g amino acids and small proteins



02

Model Setup

How the model is set up?

Original Model

- 3D model with boundaries:
 - X range: -10 to 10 Y range: 0 to 1 Z range: 0 to 1
- Macromolecules are stationary and distributed as $\phi(x)$ along the x axis, and uniformly distributed along y and z axis
- Small molecules have Brownian motion and are initially started at x=0, uniformly distributed along y and z axis

□ Global Parameters:

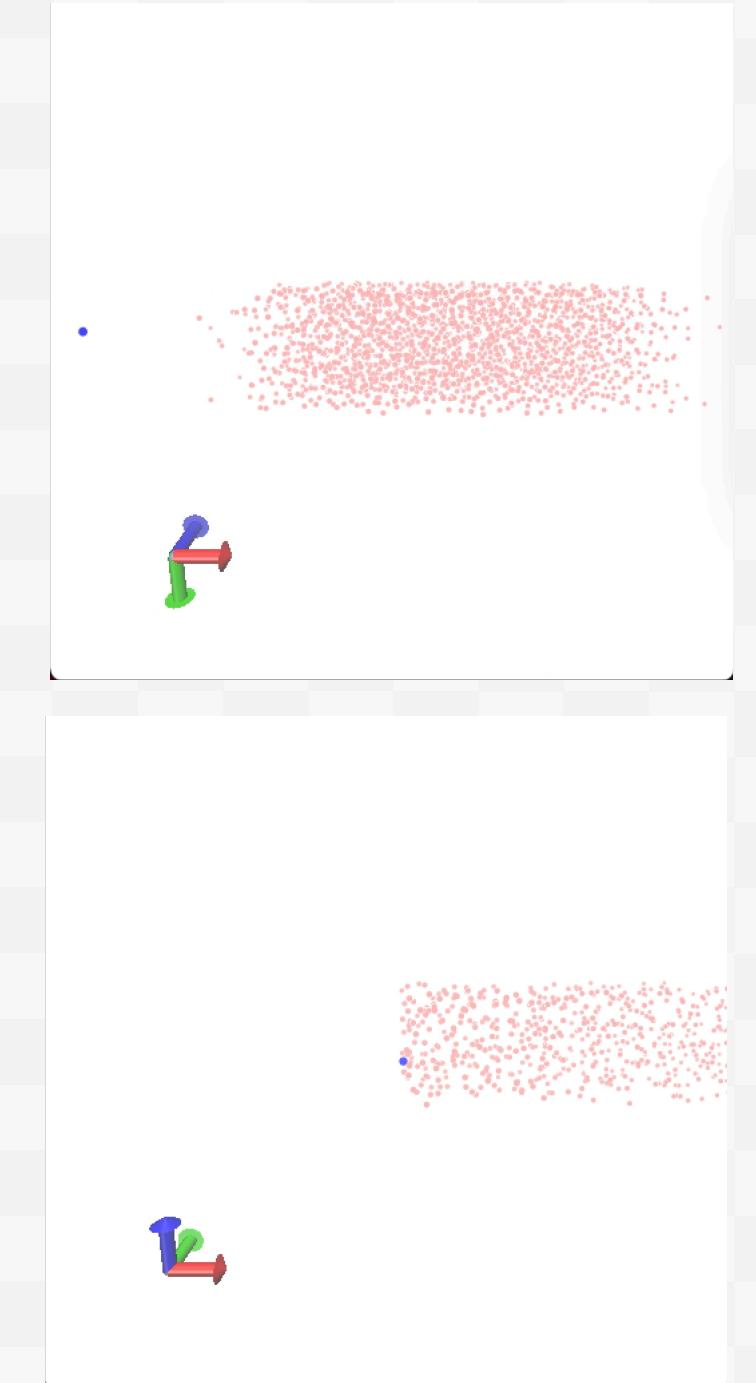
-R=0.05: Radius of Macromolecules

-r=0.005: Radius of small molecules

- ϵ : the Ratio of the molecule radius $\frac{r}{R}$ $k = 0.52$ as the max volume occupied

- Macromolecules will interact with small molecules
- Small molecules will not interact with each other
- Number of small molecules: 10^5 .
- Number of Macromolecules: Calculated as

$$\square N_{macro} = \int_{-10}^{10} \phi(x) dx \int_0^1 dy \int_0^1 dz / V_{macro}$$



Simplified Model

- **3D model** with boundaries: **2D Model**
 - X range: -10 to 10 Y range: 0 to 1 Z range: 0 to 1

- **Macromolecules** are stationary and distributed as $\phi(x)$ along the **x axis**, and uniformly distributed along **y** and **z axis**
- **Small molecules** have **Brownian motion** and are initially started at **x=0**, uniformly distributed along **y** and **z axis**
- **Global Parameters:**

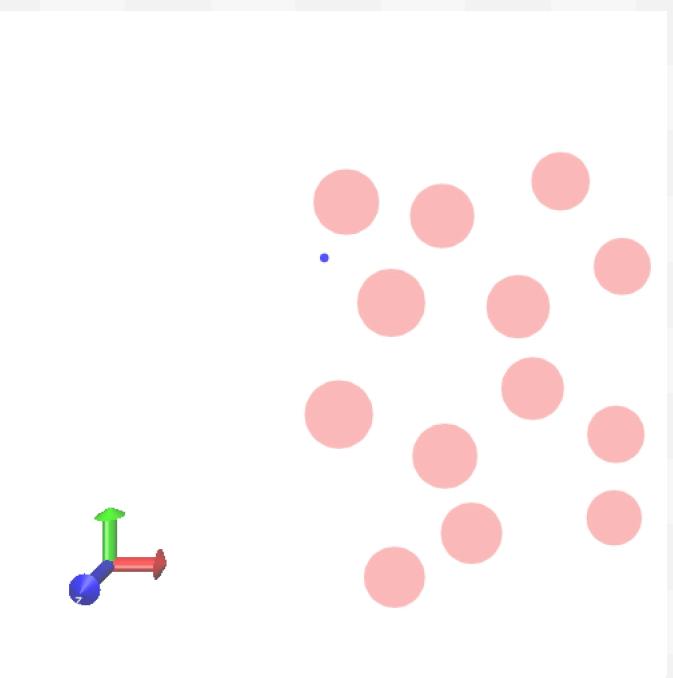
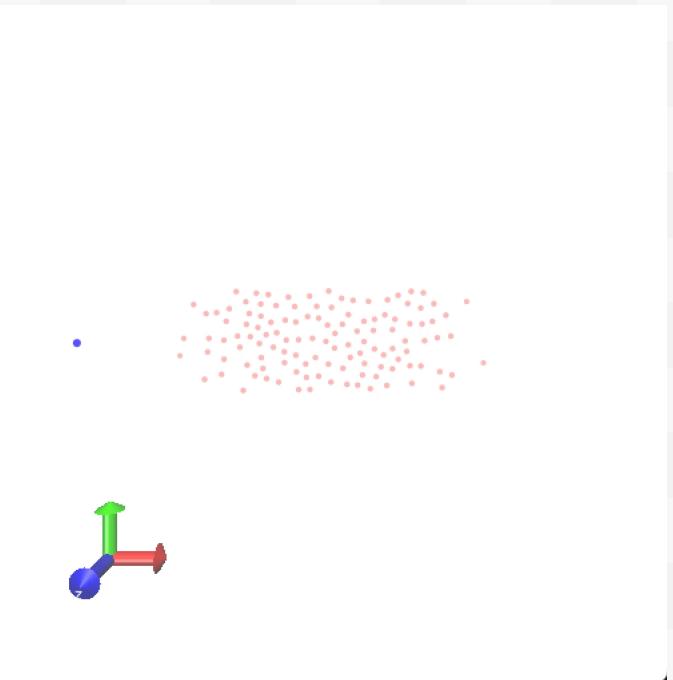
-R=0.05: Radius of Macromolecules

-r=0.005: Radius of small molecules

- ϵ : the Ratio of the molecule radius $\frac{r}{R}$ k = 0.52 as the max volume occupied

- **Macromolecules will interact with small molecules**
- **Small molecules will not interact with each other**
- **Number of small molecules:** 10^5 $\rightarrow 10^4$
- **Number of Macromolecules:** Calculated as

$$\square N_{macro} = \int_{-10}^{10} \phi(x) dx \int_0^1 dy / A_{macro}$$

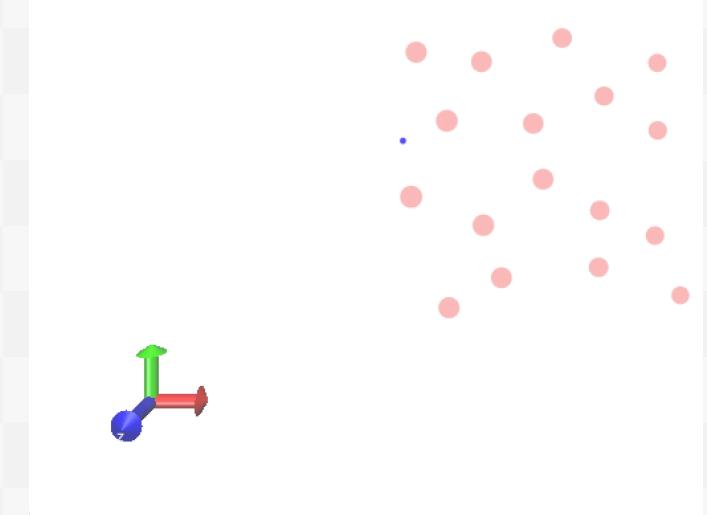
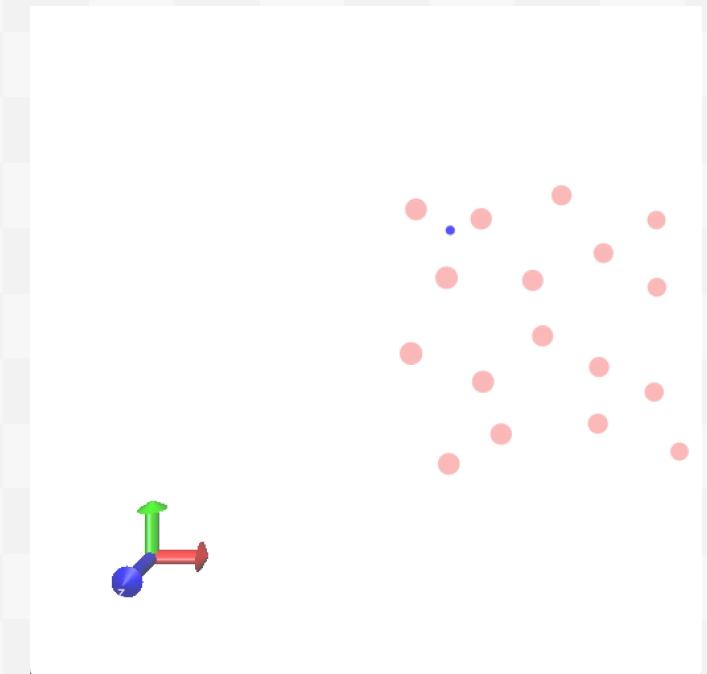


Extended Model

- Small particles are **Bacteria**
- Macromolecules are **Obstacles/Food**

- $P_{\text{run}} = 0.1$
- $P_{\text{tumble}} = 0.01$

- Each bacterium has Gaussian-randomized velocity
- When a bacterium are meet the **Food source**, the angle it changes will be **correlated** with the previous
 - $d\theta = \frac{\pi}{4}(2 * \text{rand} - 1)$
 - The velocity change will be random
- When a bacterium meets the **Obstacles**, the angle and velocity it changed will **purely random**



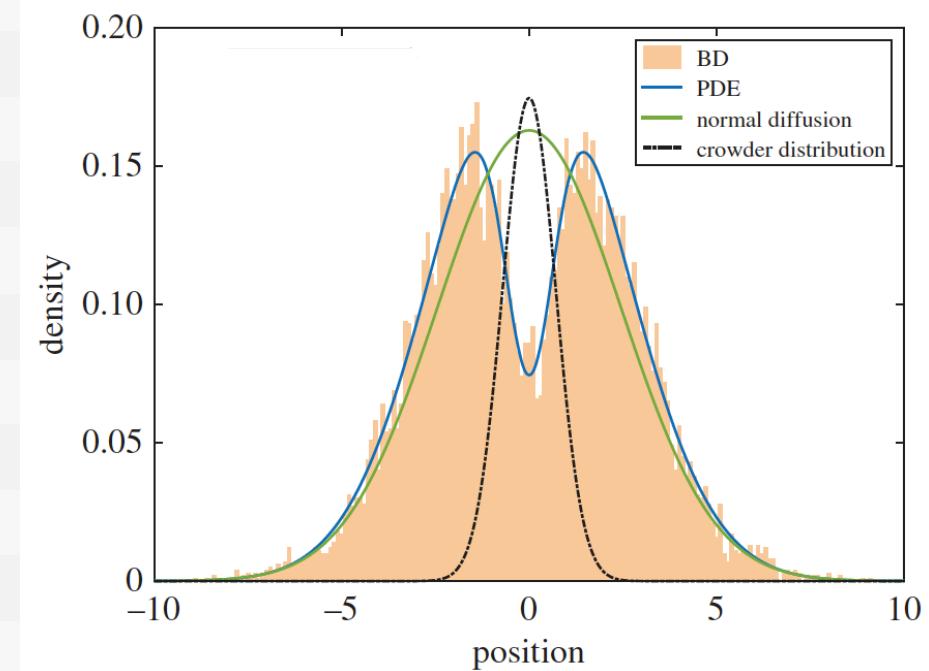
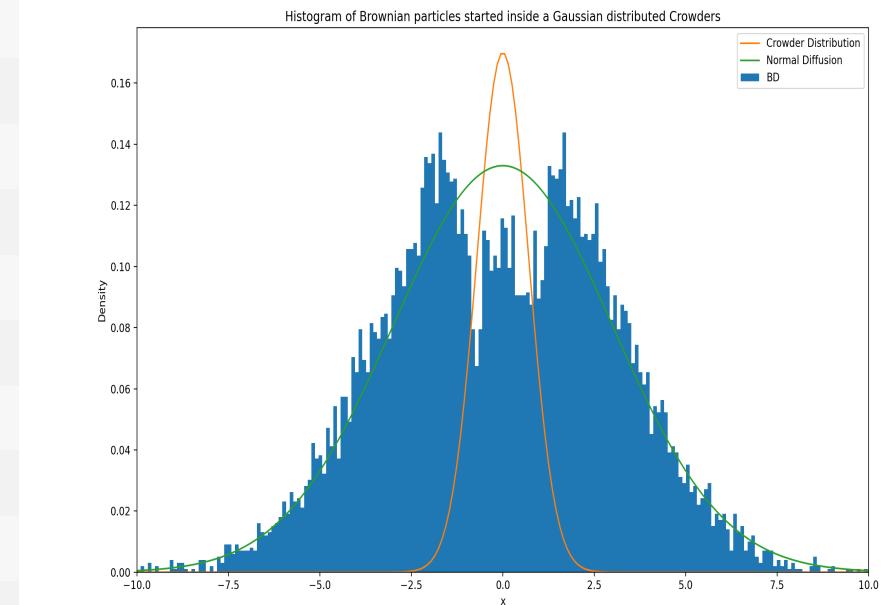
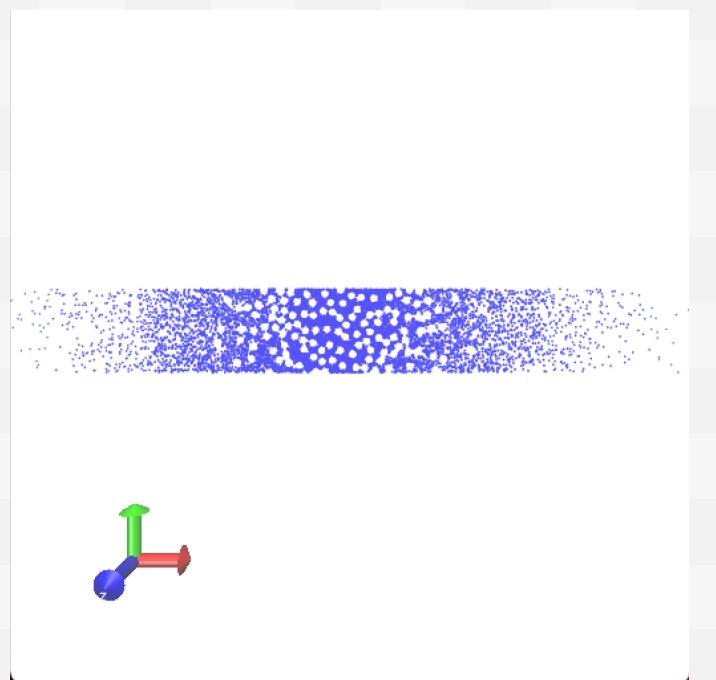
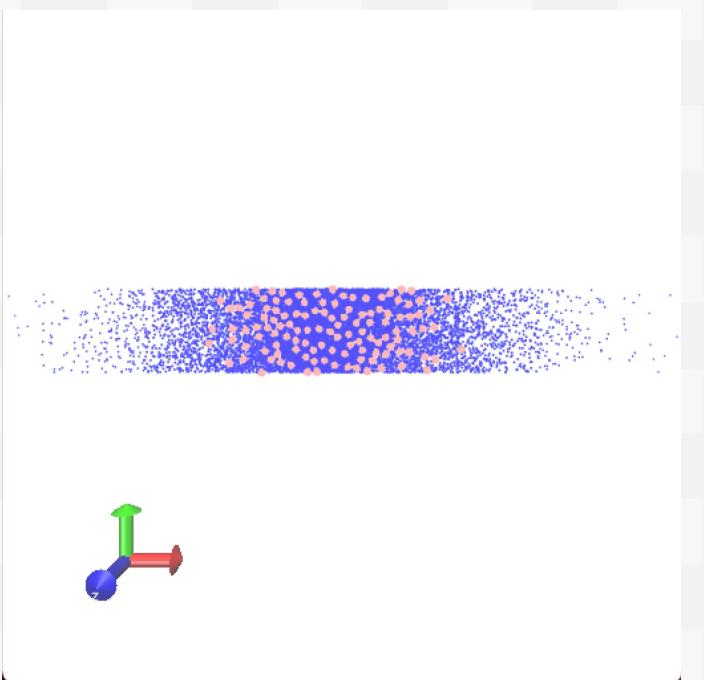
03

Case I: Symmetric Gaussian Crowders

- Compare to the figure in the paper
- Compare with the extension model

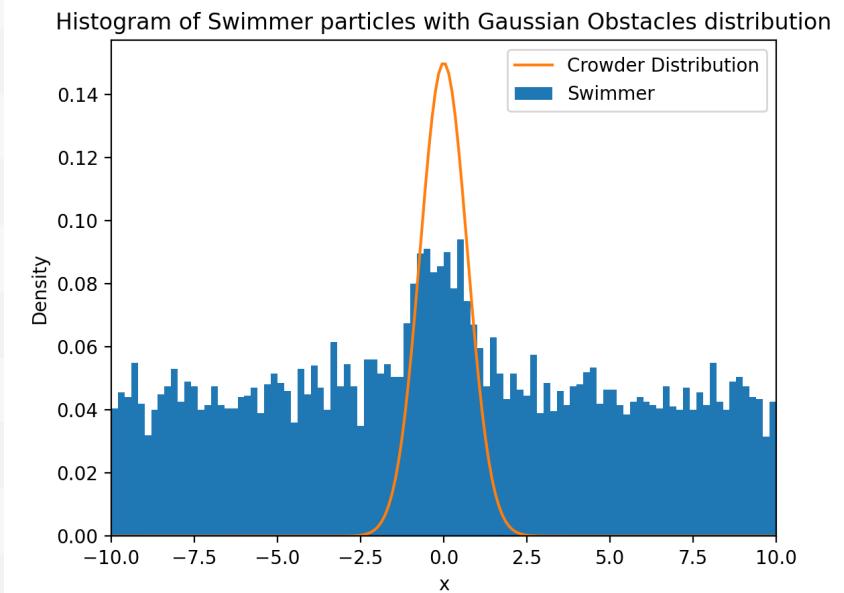
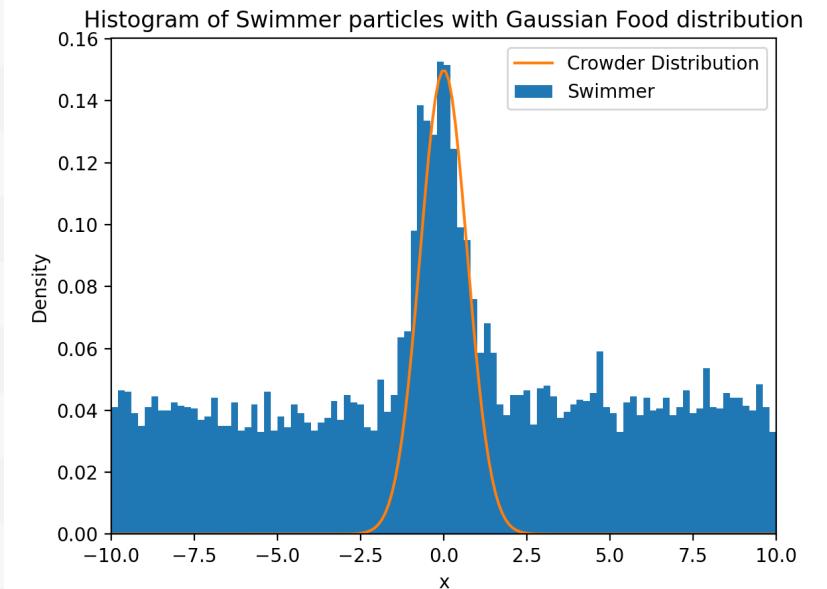
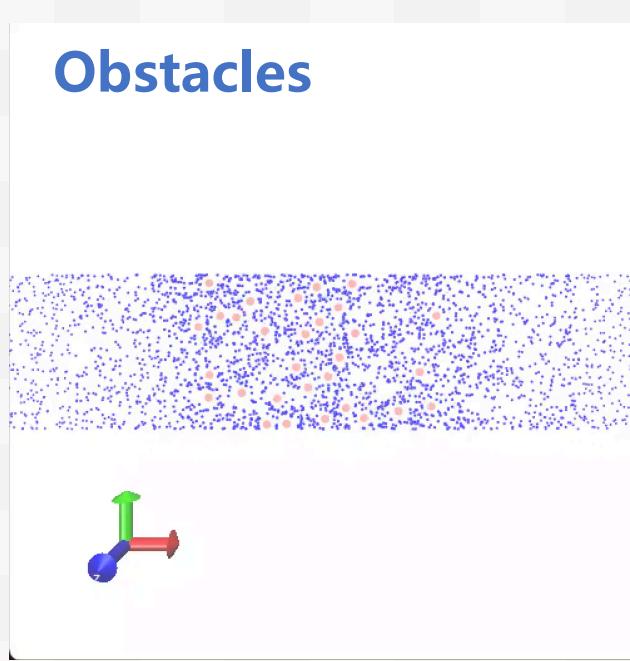
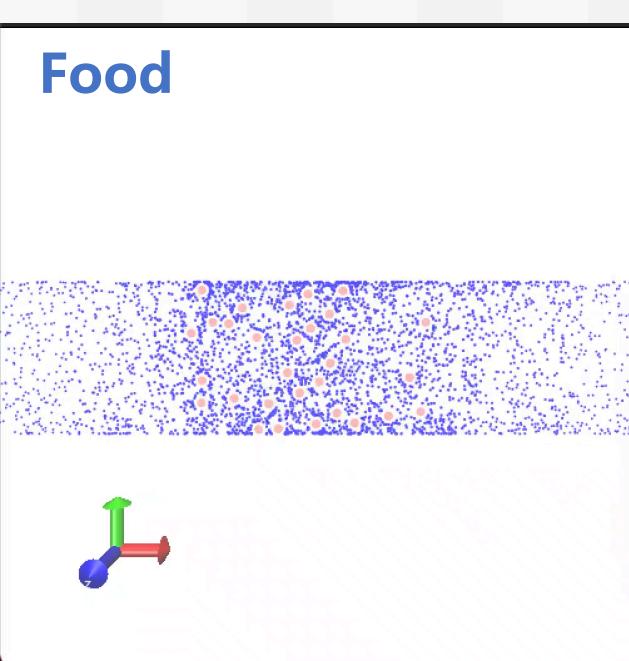
Replicated Result

- Gaussian distributed Macromolecules
- All small particles are started at $x=0$
- Diffusion Rate **drops** at the Macromolecular Crowding
 - Peak in the middle—**trapped** small particles
- Macromolecule directs small molecule to go to the less dense locations



Extended Model

- With Food source at Gaussian Distributed in the middle, Bacteria will **stick** to them
- With only obstacles in the middle, they are also **sticky but less sticky** as the food source



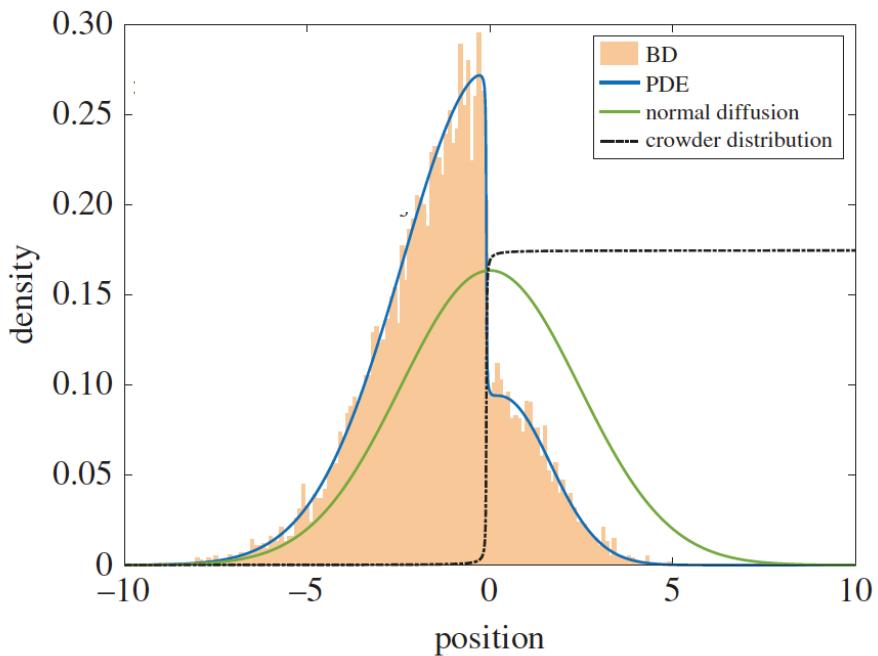
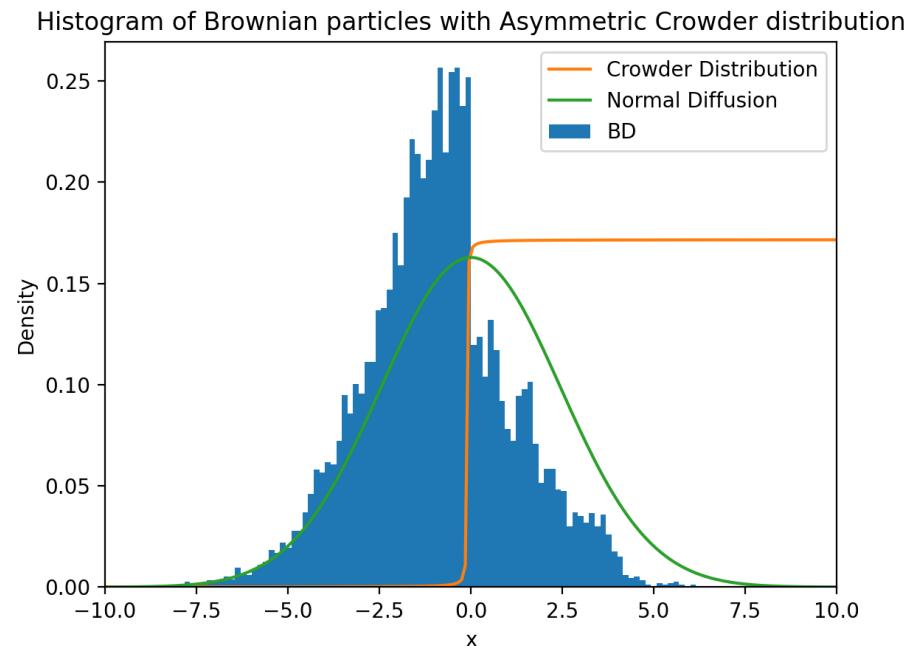
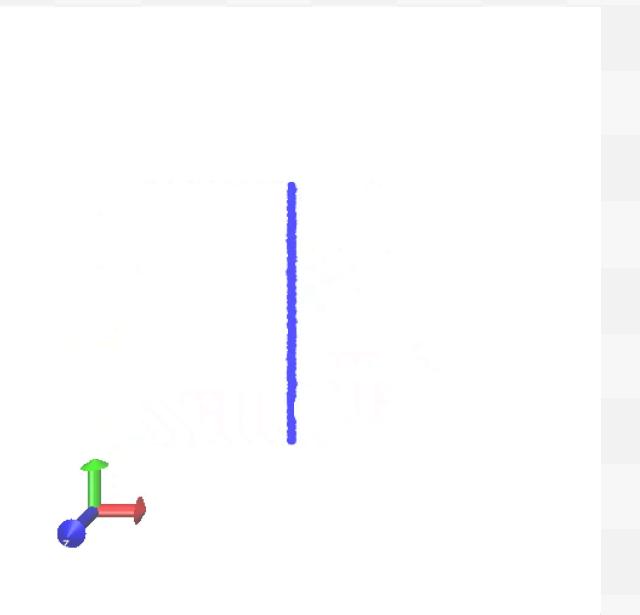
04

Case II: Asymmetric Crowders

- Compare to the figure in the paper
- Compare with the extension model

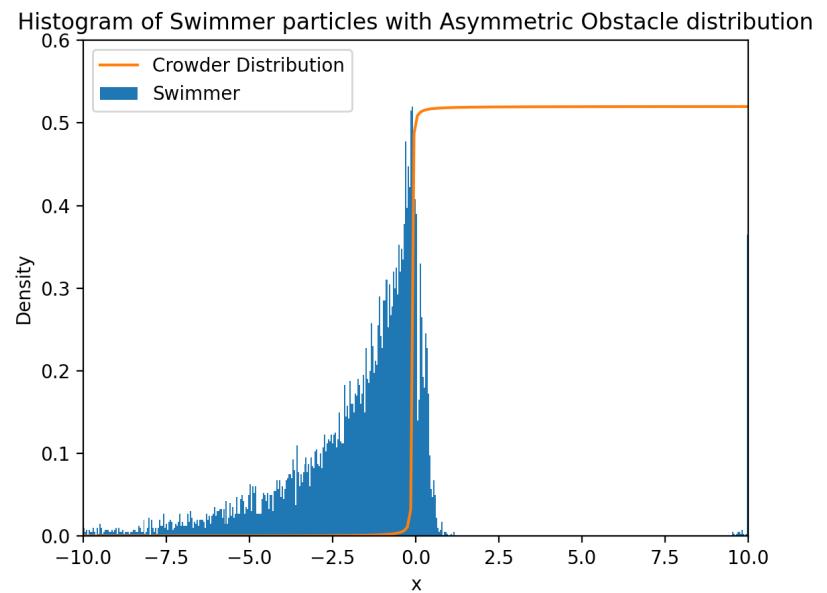
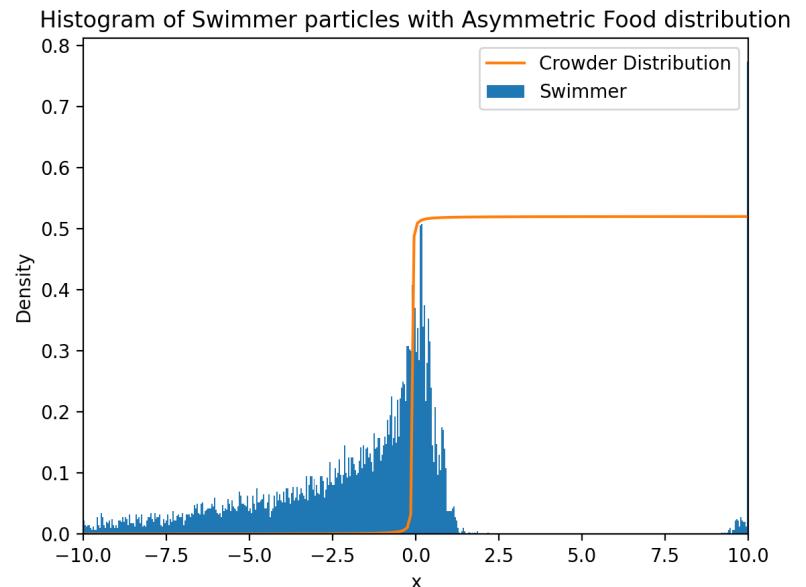
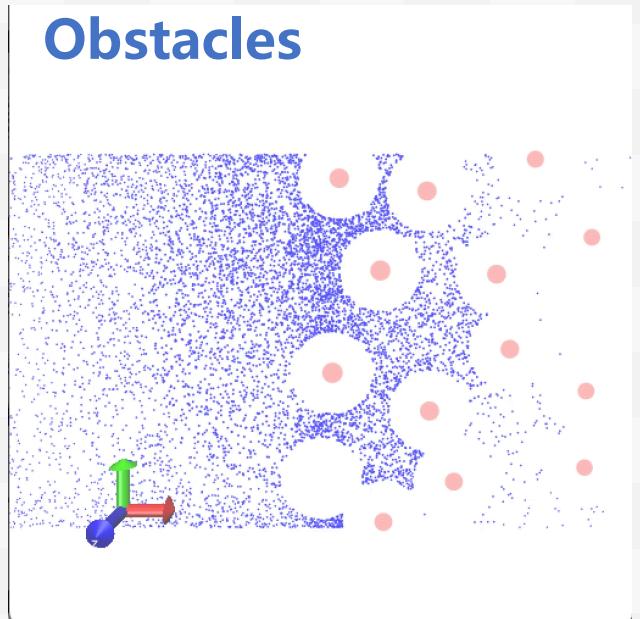
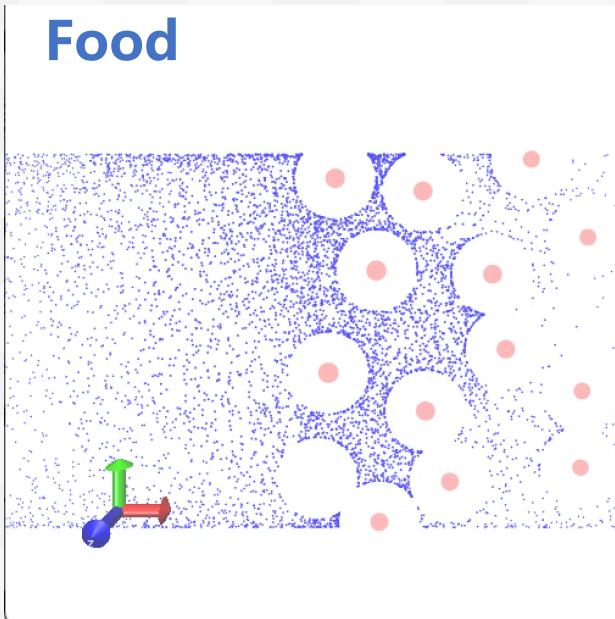
Replicated Result

- Uniformly distributed Macromolecules at $x > 0$
- All small particles are started at $x = 0$
- Diffusion Rate dramatically **dropped** when $x > 0$
 - Decrease the diffusion Coefficient
- Macromolecule directs small molecule to go to the $-x$ direction: empty places



Extended Model

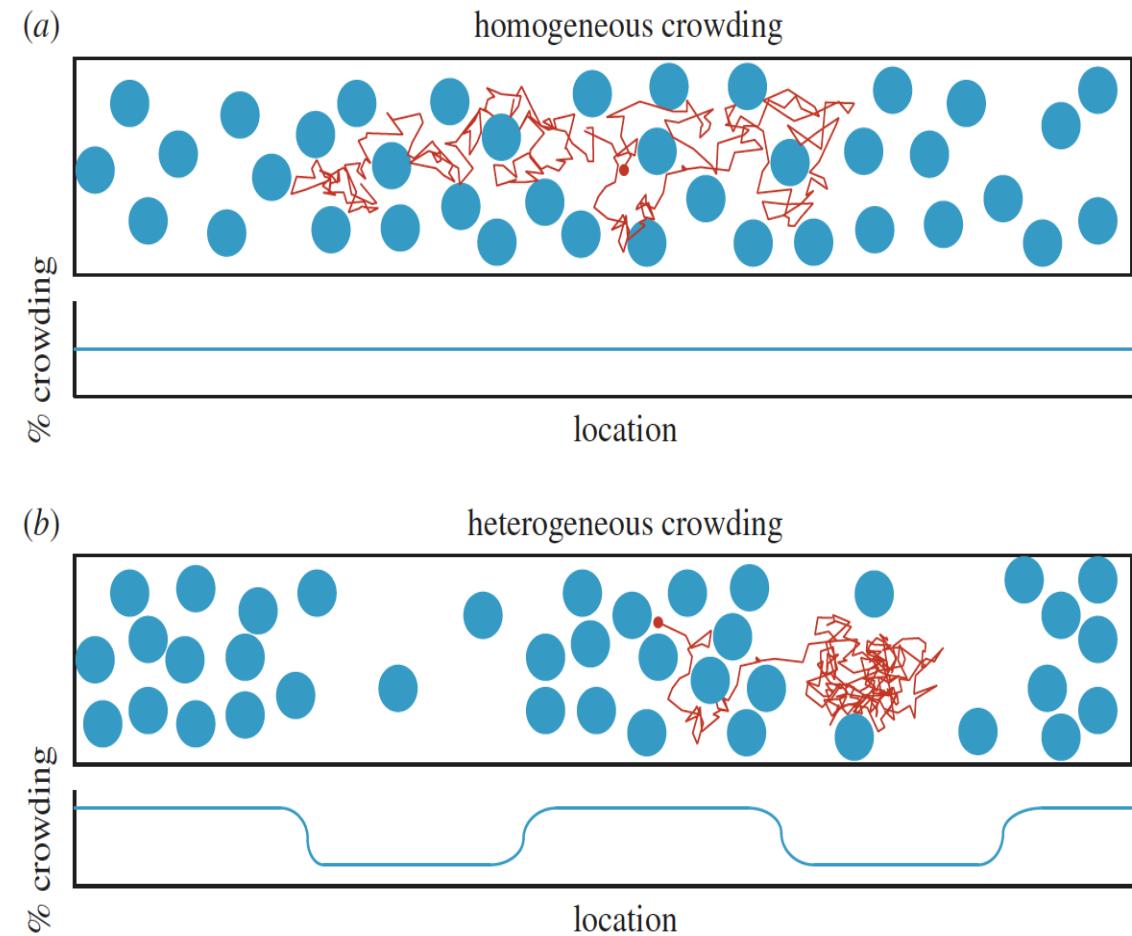
- With Food source at the $+x$ direction, more particles tends to go to the $+x$ direction and got **stick** to the food source quickly.
- With pure obstacles at the $+x$ direction, particles also prefer **empty spaces** and do not like entering the crowding region like BD.

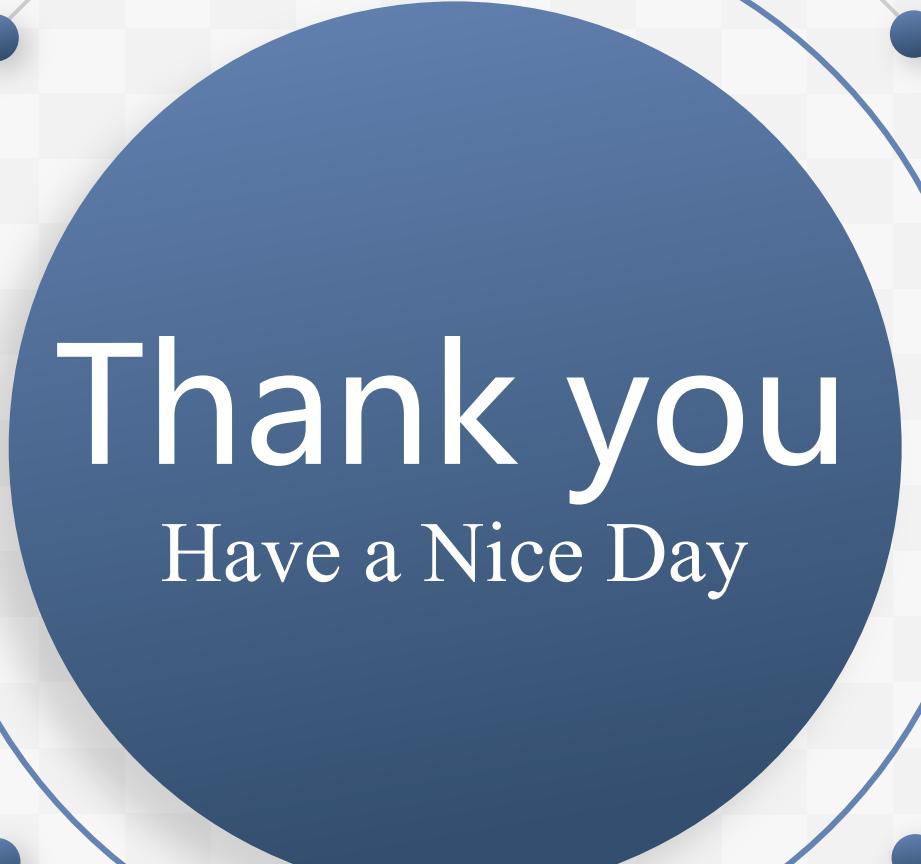


Summary

- For particles in **Brownian Motion**:
 - Heterogeneous Macromolecular crowding will direct particles to diffuse to a **more spacious** location

- For bacteria **Swimmer Motion**:
 - When food source is present, bacteria will always **stick** to the food source.
 - When there is only obstacles, it depends on the initial position of bacteria.
 - Inside the crowders: **a little Sticky**
 - Outside of the crowders: go to **more spacious places**





Thank you
Have a Nice Day

```
83 |           distance = sqrt((x1-x11)**2.0 + (y1-y11)**2.0)

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

Now we are at 2.713312. The difference is: 16595.000000 seconds
Now we are at 2.723326. The difference is: 16655.000000 seconds
Now we are at 2.733340. The difference is: 16715.000000 seconds
Now we are at 2.743353. The difference is: 16774.000000 seconds
Now we are at 2.753366. The difference is: 16834.000000 seconds
Now we are at 2.763379. The difference is: 16894.000000 seconds
Now we are at 2.773393. The difference is: 16953.000000 seconds
Now we are at 2.783407. The difference is: 17012.000000 seconds
Now we are at 2.793421. The difference is: 17072.000000 seconds
Now we are at 2.803435. The difference is: 17131.000000 seconds
Now we are at 2.813448. The difference is: 17191.000000 seconds
Now we are at 2.823462. The difference is: 17250.000000 seconds
Now we are at 2.833475. The difference is: 17308.000000 seconds
Now we are at 2.843488. The difference is: 17367.000000 seconds
Now we are at 2.853503. The difference is: 17429.000000 seconds
Now we are at 2.863516. The difference is: 17489.000000 seconds
Now we are at 2.873530. The difference is: 17548.000000 seconds
Now we are at 2.883543. The difference is: 17607.000000 seconds
Now we are at 2.893557. The difference is: 17671.000000 seconds
Now we are at 2.903571. The difference is: 17738.000000 seconds
Now we are at 2.913584. The difference is: 17798.000000 seconds
Now we are at 2.923598. The difference is: 17857.000000 seconds
Now we are at 2.933611. The difference is: 17993.000000 seconds
Now we are at 2.943625. The difference is: 17968.000000 seconds
Now we are at 2.953638. The difference is: 18028.000000 seconds
Now we are at 2.963651. The difference is: 18087.000000 seconds
Now we are at 2.973665. The difference is: 18147.000000 seconds
Now we are at 2.983679. The difference is: 18206.000000 seconds
Now we are at 2.993693. The difference is: 18267.000000 seconds
```

05

Q & A

Any Questions, Comments?

Code Highlight

```
//initial condition for small particles
for (i = 0; i < Nsp; i++)
{
    flag = 1;
    while (flag != 0)

    for (j = 0; j < Nmp; j++)
    {
        distance = sqrt(pow(propX - X[j], 2.0) + pow(propY - Y[j], 2.0));

    else //if too close or hit the boundary then change its running direction and velocity
    {
        ran1 = (float)rand() / RAND_MAX;
        theta[i] += dtheta * (2 * ran1 - 1.0);
        ran2 = (float)rand() / RAND_MAX;
        gr1 = fabs(sqrt(-2 * log(ran1)) * cos(2 * 3.14159265 * ran2));
        v[i] = sqrt(2 / dt) * gr1;
    }
}
}
}

if (flag == 0)
{
    x[i] = propX;
    y[i] = propY;
}
```

Initial Position of Small Particles

Need to check with Macromolecules to make sure it is not overlapped

Trap or move

-Calculate the distance to decide whether to move or not

Bacteria meet Food

-The new angle is correlated with the previous angle

<https://discord.com/channels/753995864602705921/753996208091168871/7>