PhD Diary

Nathan Hughes

November 30, 2018

Contents

1 Diffusion of Chitin

- Useful link
- Chitin Structure Possibly

1.1 Simple Structure

http://www.chemspider.com/Chemical-Structure.22563.html?rid=cdc5ab75-0c3b-4c17-890e-ceebf59b0942&page_num=0

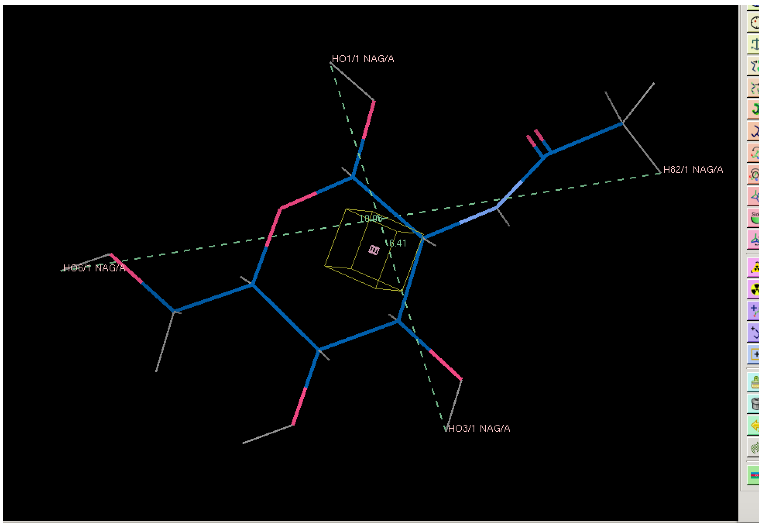

Figure 1: N-Acetyl-b-D-glucosamine

1.2 **TODO** Reapply this with more accurate structure

A decent structure can be found in (Li et al., 2013)

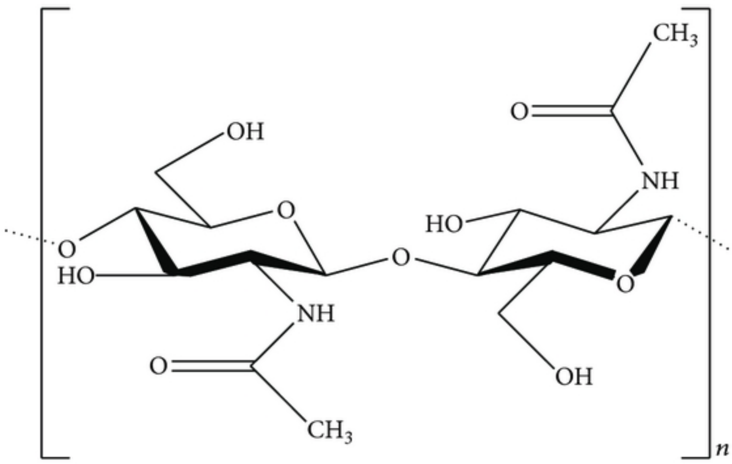

Figure 2: Chitin Molecule

1.3 Stokes-Einstein Equation

For diffusion of spherical particles through a liquid with low Reynolds number

$$D = \frac{kT}{6\pi\mu r} \tag{1}$$

Where:

- D is the diffusion constant
- μ is the *mobility*, or the ratio of the particles terminal drift velocity
- k is Boltzmann's constant
- T is the absolute temperature
- r is the radius of the spherical particle

Values for these could be:

- $\mu = 8.90 \times 10^{-4} Pa$ at 25°C
  - Water Viscosity Table
- $k = 1.38 \times 10^{-23}$
- T = 25°C = 298.15 K
- r = 5.4nm

Thus

$$D = \frac{1.38e^{-23} \times 298.15}{6\pi \times 8.9e^{-4} \times 5.4e^{-10}} \tag{2}$$

$$D \approx 4.54181050564094e^{-10} m^2/s \tag{3}$$

```
from math import pi
D = (1.38e-23 * 298.15)/(6*pi * 8.9e-4 * 5.4e-10)
```

1.4 Applying Stokes-Einstein Equation to diffusion model

1.4.1 Estimating for 2D area

```
import numpy as np
from numpy import pi

def diffuse_2D(nx, ny, dx, dy, nt, D, dt, prevState=None, prevIter=None):
    dx2 = dx**2
    dy2 = dy**2
    u = np.zeros((nx, ny))
    mid_x = int(nx/2)
    mid_y = int(ny/2)

    u = prevState.copy()
    start = prevIter

    for n in range(start, nt):
        un = u.copy()  # Update previous values
        u[1:-1, 1:-1] = un[1:-1, 1:-1] + D * \
            (((un[2:, 1:-1] - 2 * un[1:-1, 1:-1] + un[:-2, 1:-1])/dx2) +
             ((un[1:-1, 2:] - 2 * un[1:-1, 1:-1] + un[1:-1, :-2])/dy2))
    return un

# Number of x,y positions
ny = 160
nx = 270

# Change in X & Y
dx, dy = 1, 1
# Number of timesteps to calculate until
nt = 6
# Max time state to reach
max_t = 60*60 # 1 hr
# Diffusion constant in mm2/s
diff = lambda x: ((1.38e-23 * 298.15)/(6*pi * 8.9e-4 * x) * 1000)
D = diff(5.4e-10)

# for visualisation we resize the mm to
# pixel ratio and apply it to the diffusion constant
# 42 px to mm
D = D/42

dt = 1  # change in time = 1 second

nts = [str(nt) for nt in np.linspace(1, max_t, nt, dtype=int)]

# Calc initial state
prevState = np.zeros((nx, ny))
# 1 mm2 zone
prevState[60:60+21, 70:70+21] = 1
states = [prevState]

for idx, n in enumerate(nts):
    if idx is 0:
        continue
    prevIter = int(nts[idx-1])
    prevState = diffuse_2D(nx, ny, dx, dy, int(n), D, dt, prevState=prevState, prevIter=prevIter)
    states.append(prevState)
```

1.5 Diffusion of Chitin in 60 seconds (minimum threshold of > 1)

Essentially, if there was no cell barriers how far could **at least some** chitin diffuse in a minute?

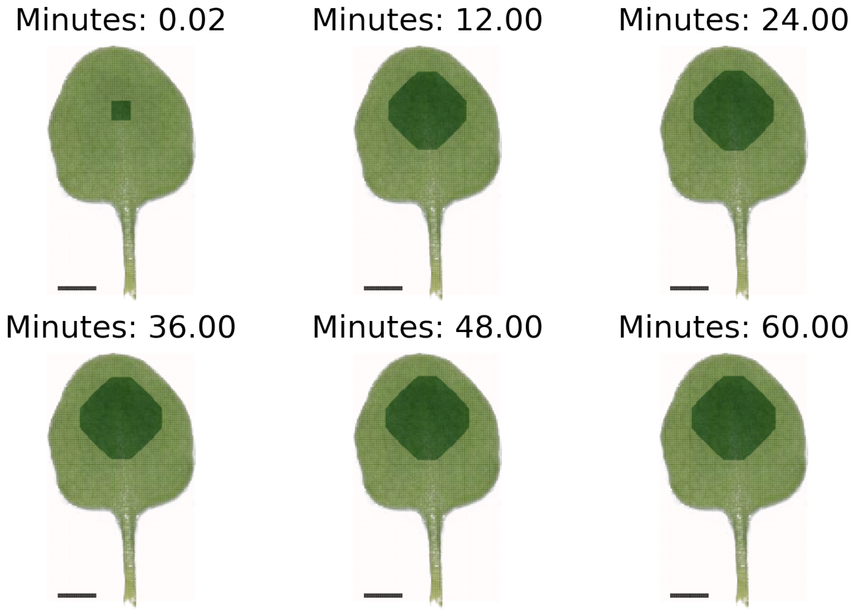

Figure 3: Applying Einstein-Stokes to Diffusion Models, a 0.5mm^2 zone is introduced at TS:1.

1.6 Question

The question now becomes: "What quantity of Chitin molecules is required to be of interest?"

2 Musings

2.1 On Cell Walls

Does it make sense to consider each cell as a container that fills a certain amount before "spilling-over"

3 Verification of equations

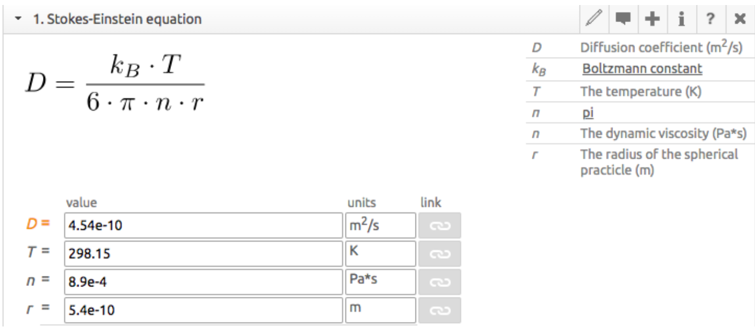

Figure 4: fxSolver applied to same values produces similar value for D

References

Xiaosong Li, Min Min, Nan Du, Ying Gu, Tomas Hode, Mark Naylor, Dianjun Chen, Robert E. Nordquist, and Wei R. Chen. Chitin, Chitosan, and Glycated Chitosan Regulate Immune Responses: The Novel Adjuvants for Cancer Vaccine. https://www.hindawi.com/journals/jir/2013/387023/, 2013.