BE/APh 161: Physical Biology of the Cell, Winter 2025
Homework #8
Due at the start of lecture, 2:30 PM, March 5, 2025.

Problem 8.1 (The Vicsek model, 100 pts)**.**
As we have discussed in lecture, the Vicsek model (Vicsek, et al., *PRL*, **75**, 1226–1229, 1995) was in many ways the start of the theory of active matter. The model consists of a collection of individual agents called "boids," sometimes referred to as self-propelled particles (SPPs). Like Vicsek, we will consider boids on a two-dimensional plane. Consider a single boid i. At time point t, the boid moves with a velocity v_0 along a unit vector $\mathbf{u}_i(t+\tau) = (\cos\theta_i(t+\tau), \sin\theta_i(t+\tau))^\mathsf{T}$. The vector $\mathbf{u}_i(t+\tau)$ is uniquely defined by the angle $\theta_i(t+\tau) \in (-\pi, \pi]$. The angle $\theta_i(t+\tau)$ is determined by averaging the value of $\theta_i(t)$ for every boid (including boid i itself) that is within a distance r from boid i, plus some added noise. This happens every time interval τ. Stated mathematically, the position $\mathbf{x}_i$ of boid k evolves according to

$$\theta_i(t+\tau) = \arctan\left(\frac{\langle \sin\theta(t)\rangle_{r_i}}{\langle \cos\theta(t)\rangle_{r_i}}\right) + \eta\,\zeta_{i,t}, \tag{8.1}$$

$$\mathbf{u}_i(t+\tau) = \begin{pmatrix}\cos\theta_i(t+\tau)\\ \sin\theta_i(t+\tau)\end{pmatrix}, \tag{8.2}$$

$$\mathbf{x}_i(t+\tau) = \mathbf{x}_i(t) + v_0\tau\mathbf{u}_i(t+\tau). \tag{8.3}$$

The $\eta\,\zeta_{i,t}$ term give the noise. Here, $\zeta_{i,t}$ is a random number drawn from a uniform distribution on the interval $[-\pi, \pi]$. As such, maximal noise is achieved when $\eta = 1$. The notation $\langle\cdot\rangle_{r_i}$ denotes an average over all boids within a distance r of boid i, including boid i.

The parameters for this system are the velocity v_0, the interaction radius r, the noise strength η, the time between reorientations τ, the total size of the system L (we assume an $L \times L$ 2D surface), and the density of boids, $\rho = N/L^2$, where N is the number of boids. We can choose our spatial units to be such that $r = 1$ and our time units such that $\tau = 1$. We are then left with parameters v_0, η, ρ, and L. Note that if we are trying to avoid finite size effects, we take L to be large, leaving only v_0, η, and ρ as parameters.

Throughout this problem, we will consider $r = 1$ and $\tau = 1$. Like Vicsek, we will set $v_0 = 0.03$, but you are welcome to play with that parameter if you want to explore further. We will primiarly be considering different values of the noise η and of the density ρ of boids. We use **periodic boundary conditions**. This means that if a boid moves rightward from position $x = L$, it will move to a position ear $x = 0$, and similarly in the y-direction. It also means that a boid at position $x = L$ can have neighbors near position $x = 0$. Periodic boundary conditions are employed in an effort to mitigate boundary effects.

We will typically leave N fixed and vary L to adjust the density. You should use a minimum value of $N = 100$, but you should use larger N if your code is fast enough to handle it. I found that $N = 1000$ worked well, and I could do all calculations in under an hour.

You may use AI tools to complete parts (b) and (c) under the following conditions.

- You clearly indicate which AI tool(s) you use.
- You clearly delineate what is your own, what comes directly from AI, and what came from AI but was then edited by you.
- You include all prompts you gave.

a) The expression

$$\arctan\left(\frac{\langle \sin\theta(t)\rangle_{r_i}}{\langle \cos\theta(t)\rangle_{r_i}}\right) \tag{8.4}$$

in equation (8.1) is meant to compute the average of the orientations of the neighboring boids. Why are we computing it this way instead of simply averaging all of the neighboring θ values directly?

b) Write code to simulate the Viscek model. You should have a function that takes as input N, η, v_0, and L, as well as any other inputs you see fit (e.g., obviously the number of steps you want to run the simulation should be an input), and returns the x, y, and θ values over time for all of the boids.

Subsequent parts of this homework will be posted soon.