PHY 607 Project 3: Modeling Biophysical Behaviors with Monte Carlo Markov Chain Methods in 2D

Benjamin Maves and Julia Rice

October 22, 2024

1 Introduction

1.1 The Problem - Evolutionary Behavior of N Biological Particles

In this paper, we show how we used a Monte Carlo Markov Chain to simulate the collective behavior of self-propelling "rod" shape particles over a given number of iterations. Such a simulation is extremely useful for modeling active matter and other various biophysical phenomenon. For example, many bacteria are known to exhibit "autoaggregation," where the bacteria tend to form small clumps via chemical and/or mechanical interactions between particles.

1.2 Algorithm Overview

To emulate this natural behavior, we programmed two forms of interactions that can occur between particles:

1. **Orientation:** Each particle wants to align itself with neighboring particles.
2. **Attraction:** Each particle experiences an attractive force from neighboring particles.

These interactions are accounted for in the way we calculate the energy. In our "compute_energy" function, the energy of a configuration is found with

$$E = -\sum_{i<j}^{N} \alpha \cos^2(\theta_i - \theta_j) - \frac{\beta}{1+d^2}$$

where α is a parameter determining the strength of alignment between two particles i and j, and β controls how strongly particles i and j are attracted to each other based on the distance d between them. Altering either of these parameter individually will change the energy, but by different amounts. In our simulations we set $\alpha = 1.0$ and $\beta = 25.0$.

We then iteratively evolved a system of N particles using a Monte Carlo Markov Chain with Metropolis Hastings Criterion, employing Bayesian inference. The states (position and orientation) of the particles are our parameters, both initialized from random uniform distributions.

1.2.1 Principles of Bayesian Inference

We used Bayes' formula to find the posterior probability of a proposed state being accepted in our MCMC. According to Bayes,

$$P(\theta|data) = \frac{P(data|\theta)P(\theta)}{P(data)}$$

where $P(\theta|data)$ is the **posterior probability**, $P(data|\theta)$ is the **likelihood**, and $P(\theta)$ is the **prior probability**. We are not running our simulation with a pre-collected set of data (we don't have the $P(data)$ part of the formula), but we still employ the principles of Bayesian inference when evolving our system with random initialization.

1.2.2 Priors, Likelihoods, and Metropolis-Hastings Criteria

We establish the collective evolution of our particles to be predisposed against a larger spread in the "compute_system_prior" function. In this function, the mean of all particles' positions is used to calculate a "spread," given by

$$S = \sum_{i=1}^{N} \|x_i - \bar{x}\|$$

We implement this in our Metropolis-Hastings criteria such that the prior probability for a given particle configuration is $P = e^{-S}$. The likelihood is then calculated in the "compute_liklihood" function. We favor configurations with a lower energy such that $L = \frac{-E}{T}$, where L is the likelihood, E is the energy calculated via our "compute_energy" function, and T is the temperature of the system. The likelihood for a given configuration is then used to find the posterior probability for proposed states in the "metropolis_step_with_prior" function:

$$PosteriorProb. = e^{\frac{-E}{T}} \times e^{-S}$$

This is used in our MH acceptance criteria, where the acceptance ratio of a proposed step in the simulation is defined by

$$P_{accept} = \frac{P_{OldPosterior}}{P_{NewPosterior}}$$

.

More explicitly, we calculate the change in energy including the prior:

$$\Delta E = (E_{proposed} - E_{old}) + (Prior_{new} - Prior_{old})$$

Then, we accept this new energy configuration if it is either negative **or** if a randomly drawn percentage (a number between 0 and 1) is less than $e^{\frac{-\Delta E}{T}}$ (which is equivalent to our acceptance probability).

1.2.3 Burn-In Period

To improve our results, we include a burn-in period in our simulation using the parameter "burn_in_fraction," defined in our "run_mcmc_with_prior" function. This parameter, which can be a number between 0 and 1, indicates what fraction of our results after beginning the simulation are discarded as the burn-in period.

2 Validation

2.1 Expectations and Results

The First and Second laws of thermodynamics dictate that a system with many bodies and a large number of degrees of freedom will tend towards the lowest energy configuration. Consequently, we should see our system evolve to a configuration where the particles clump together and take on similar alignments.

In Figure 1, we see the distribution of particles before and after running our MCMC. While particles before are randomly distributed in position and orientation, we see that particles after have aggregated and slightly more aligned. The system has evolved into a lower energy and thus more energetically favorable state. This can also be seen though the evolution of the total system energy over each MCMC step. Figure 2 shows the evolution of the total energy over N steps and shows that the energy drops into a more favorable state.

2.2 Convergence

The system after sufficient steps should converge to a stable equilibrium solution. After reaching this equilibrium we expect little fluctuation in energy or average particle position or orientation. While at small step counts energy and particle orientations and positions are highly variable, with a high step count total energy reaches a steady state where energy is constant. Figure 2 shows that after sufficient steps, ≈ 4000, energy becomes constant. The system reaches a steady state.

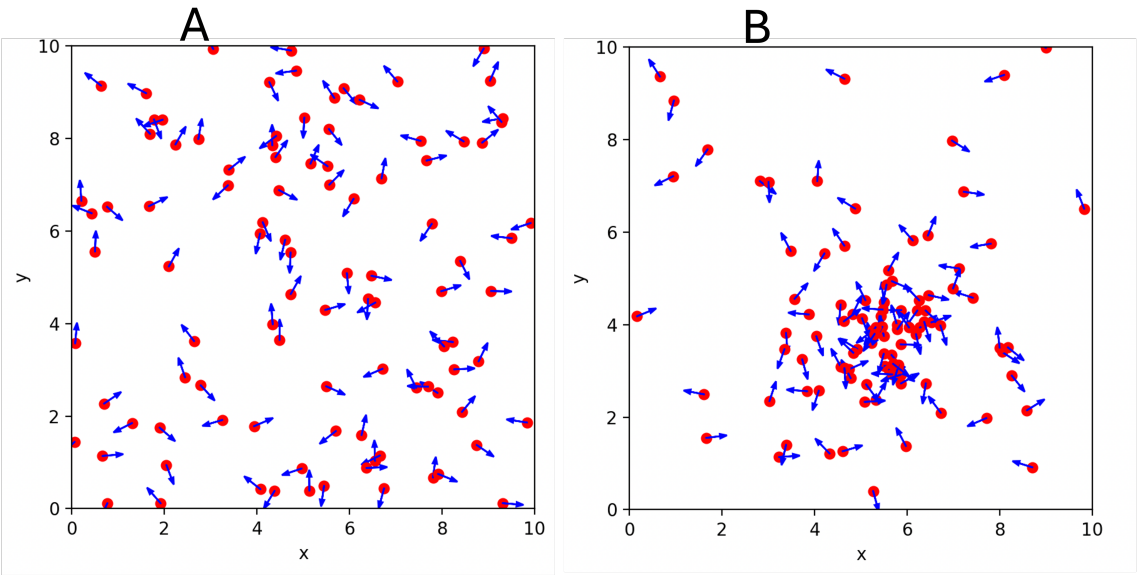

Figure 1: Particle distribution for N = 100 in a 10 x 10 area with $\alpha = 1, \beta = 25$. (A) Initial distribution with uniform random distribution of positions and orientations. (B) Final distribution after 10,000 MCMC steps. Rods are aggregated and have a higher alignment.

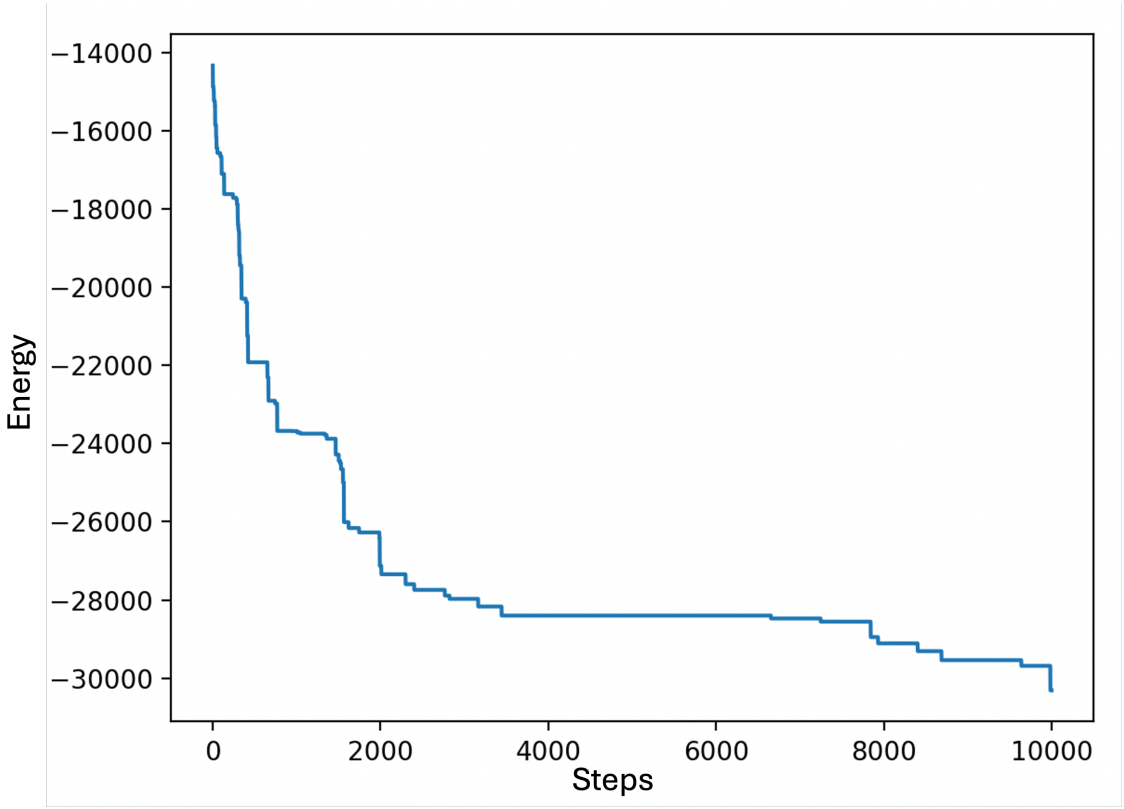

Figure 2: Total energy at each MCMC step for $\alpha = 1, \beta = 25$. The system falls into a more energetically stable state where particles are aggregated and have higher alignment.

To further validate that our system has reached a steady state we compute the autocorrelation between energies at different steps. The autocorrelation is givin as,

$$R_{xx} = E[X_{t_1} X_{t_2}]$$

where E[ ] is the expected value of our given parameter. In our case we use the average energy. For more correlated samples, $R_{xx} \to 1$, and uncorrelated samples go to 0. For $N = 10,000$ steps we get an average autocorrelation between steps of ≈ 0.728. This means that our energies become strongly correlated and have converged to a steady-state value.

3 Conclusion

We have demonstrated our methodology of using a Monte Carlo Markov Chain to simulated behaviors of active rod-shaped particles over many iterations. Our algorithm is useful and robust and accurately models real life behaviors of many active matter systems where aggregation and alignment are favorable such as bacterial systems and some fluid nematic fluid systems. Using a Metropolis-Hastings algorithm with Bayesian inference we clearly converge to a steady state after sufficient iterations which is consistent with real world systems.