# Simulating Active Dumbbells using Langevin Dynamics

S Shri Hari

December 8, 2021

# Active Matter: Introduction

## Definition:

- Aggregate of units that consume energy to move / exert mechanical forces.

## Examples:

- Bacterial Colonies
- Self Assembling biomolecules
- School of Fish, Flock of Birds

## Key Features:

- Non-trivial behaviour exhibited when in large numbers / densities
- Example: Self-Organization, Phase Separation, etc.

# Active Matter: Example 1

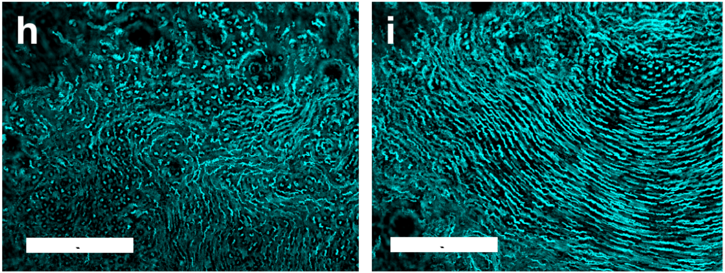



Figure: Internal Flow observed in high-density phases of active swimmers

Video Demonstrating Internal Flow Structure of Swimmers at high local concentrations

# Active Matter: Example 2

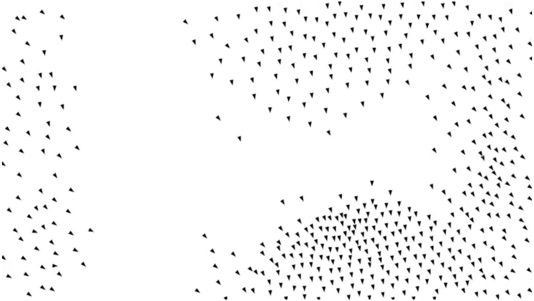


Figure: Boids is an artificial life program, developed by Craig Reynolds in 1986, which simulates the flocking behaviour of birds. The name "boid" corresponds to a shortened version of "bird-oid object", which refers to a bird-like object.

Credits: 'From Python to NumPy' by Nicolas P. Rougier
Link to Video

# Motivation

## The Experiment:

- "Cooperation in a fluid swarm of fuel-free micro-swimmers" by MYB Zion, A Modin, Y Caba, PM Chaikin (2021). arXiv: 2012.15087

## Key Features

- Constructed particles that are active but 'fuel-free'. Dynamics of Particles unaffected by local fuel concentrations.
- Study focused on behaviour exhibited by aggregates
  - Homogeneous system of Active Particles
  - Heterogenous system: Mixture of Active and Passive
- Homogeneous System: Motility Induced Phase Separation (MIPS)
- Heterogeneous System: A new phase - 'Corralling'; Passive Particles compressed to HCP by active particles

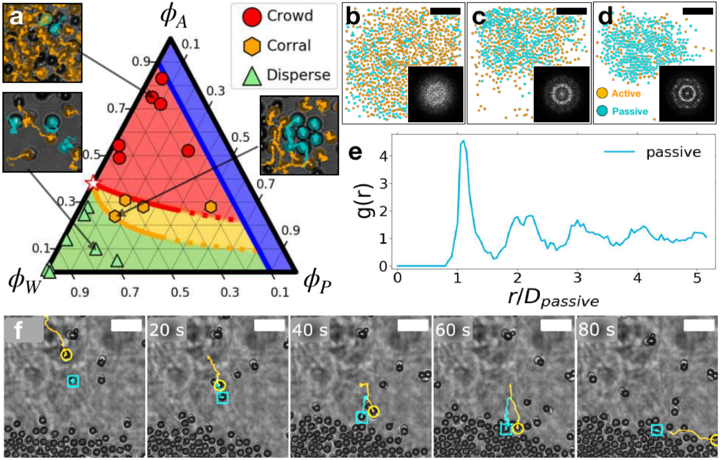



Figure 4: A swarm of active swimmers cooperatively corrals passive particles. **a** a ternary phase diagram of active ($\phi_A$) passive ($\phi_P$) and the surrounding water bath ($\phi_W$) depicts the resulting dynamics of a mixture of passive particles and active swimmers, given their initial area fraction. Green triangles show that at low overall-particle area fraction (large $\phi_W$), the active swimmers quickly disperse. Red circles show a region where the active particles crowd once activated and form a persistent dense phase. Between the red and the green regions, $1 - 4\phi_P < \phi_A/\phi^* < 1 - \frac{\sqrt{12}}{\pi}\phi_P$, a yellow region emerges, where orange hexagons signify experiments where active particles spontaneously corral the passive particles ($\phi^*$ is denoted by a star). Red and orange curves are theoretical predictions for the onset of the corralling state extrapolated for the small $\phi_P$ limit. Thumbnails show snapshots from corresponding points on the diagram (orange and cyan label active and passive particles respectively). Blue region indicates an overall particle area fraction greater than 0.91 (hexagonal packing). **b-d** snapshots taken at time 0 mins (**b**) 5 mins (**c**) and 30 minutes (**d**) show the global dynamics of the active particles (orange) as they corall the passive particles (cyan) (see Supplementary Video 6). Insets are the structure factor, $S(\vec{q})$, of the passive particles, showing the evolution towards hexagonal order in their packed arrangement. **e** the pair correlation function has peaks at 1, 2, 3, and 4 diameter, with $1 - \sqrt{3}$ splitting around 2 indicative of ordered hexagonal packing. **f** snapshots at 20 seconds intervals show the corralling process which proceeds through entrainment events, where an active particle (orange) deposits a passive particle (cyan) at the dense region. After deposition, the active particle leaves (see Supplementary Video 2). Scale bars: **b-d** 30 $\mu$m; **f**: 10 $\mu$m.

# The Experiment: Key Details

## Particles:

- Passive: Spherical liquid droplets of n-dodecane in oil-water emulsion
- Active: Light Absorbing Particle coupled to passive sphere; 'Peanut' Shaped
- Medium: Water; Particles exhibit Brownian Motion

## Activity

- Energy for active particles derived from light
- Temperature gradient formed by light absorbing particles heating water surrounding it when exposed to light; Gradient verly localized to the particle
- Propulsion along the axis due to thermo-capillary effect
- Continuous relation between flux and activity

# Relevant Study: Active Dumbbells

- "Dynamics of a homogeneous active dumbbell system" by A Suma, G Gonnella, G Laghezza, A Lamura, A Mossa... - Physical Review E, 2014. Phys. Rev. E 90, 052130
- A Theoretical + Computational study about the system of active dumbbells
- Behaviour observed similar to previous experiment's study on Homogeneous System

# Simulating the System

- Need to write own code from scratch to implement the driving forces
- Reason: Lack of support to assign forces dynamically to each particle in LAMMPS
- Objective of the Project: To simulate the heterogeneous system and observe/study the 'coralling' phase

# The Model: Potentials

- Weeks-Chandler-Anderson(WCA) Potential

$$V_{WCA}(r) = \begin{cases} V_{LJ}(r) - V_{LJ}(r_c) & r < r_c \\ 0 & r > r_c \end{cases} \tag{1}$$

$$V_{LJ} = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \quad ; \quad r_c = 2^{1/6}\sigma \tag{2}$$

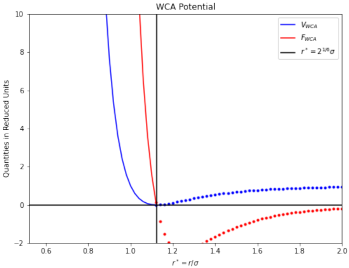
where $\sigma$ is the diameter of the spherical particles

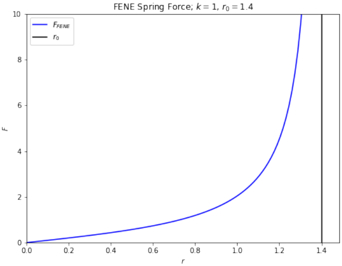
- Spring: Finite Extensible Nonlinear Elastic (FENE)

$$F_{FENE}(\mathrm{r}) = \frac{-k\mathrm{r}}{1 - (r/r_0)^2} \tag{3}$$

where $r_0$ represents the maximum spring length

# Plot: Potentials



# The Model: Particles

- Passive: Spherical
- Active: Dumbbells formed by attaching spring between two spheres
- Activity: Force with fixed magnitude acting on the two spheres of the dumbbell directed along the axis

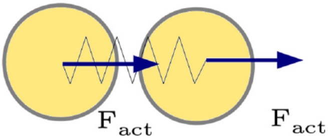



Figure: Diagram representing Active Dumbbell

# Equations of Motion: Langevin Dynamics

- As particles are in a medium, either the particles in it or their effect on other particles need to be simulated
- Langevin Dynamics allows us to simulate two key effects of the medium:
  - Resistance to Movement i.e Viscosity; $\propto$ v($t$)
  - Random force due to fluctuations in the medium's density: $\xi(t)$
- Equations of Motion:

$$\mathrm{d}x_i\,(t) = v_i(t)\,\mathrm{d}t$$

$$\mathrm{d}v_i\,(t) = m_i^{-1}F_i(x(t)) - \gamma_i v_i(t)\,\mathrm{d}t + \sqrt{2k_B T\gamma_i m_i^{-1}}\,\mathrm{d}W_i\,(t)$$

# Langevin Equations (Continued)

- $W(t)$ - Wiener Process
  - $W(0) = 0$
  - Increments are independent: $W(t+u) - W(t)$ is independent for all past values $W(s)$ for all $u, t > 0, s \leq t$;
  - Increments are from normal distribution: $W(t+u) - W(t) \sim \mathcal{N}(0, u)$
  - $W$ is continuous in $t$
  - Coefficient of last term derived from this condition
    - $\langle v^2(t) \rangle_{eq} = \frac{k_B T}{M}$
- Integration Scheme provided by: "Second-order integrators for Langevin equations with holonomic constraints" by E Vanden-Eijnden, G Ciccotti - Chemical physics letters, 2006. https://www.sciencedirect.com/science/article/pii/S0009261406011092
- Modified Velocity Verlet Scheme

# Integrating Equations of Motion

- Changing Notation: $f_i(x) = m_i^{-1} F_i(x)$, $\sigma = \sqrt{2k_B T \gamma_i m_i^{-1}}$

$$\mathrm{d}x_i(t) = v_i(t)\,\mathrm{d}t$$
$$\mathrm{d}v_i(t) = f_i(x(t)) - \gamma_i v_i(t)\,\mathrm{d}t + \sigma_i\,\mathrm{d}W_i(t)$$

$$\begin{aligned} v^{n+1/2} &= v^n + \frac{1}{2}hf(x^n) - \frac{1}{2}h\gamma v^n + \frac{1}{2}\sqrt{h}\sigma\xi^n \\ &\quad - \frac{1}{8}h^2\gamma(f(x^n) - \gamma v^n) - \frac{1}{4}h^{3/2}\gamma\sigma\left(\frac{1}{2}\xi^n + \frac{1}{\sqrt{3}}\eta^n\right) \end{aligned} \tag{4}$$

$$x^{n+1} = x^n + hv^{n+1/2} + \frac{1}{2\sqrt{3}}h^{3/2}\sigma\eta^n \tag{5}$$

$$\begin{aligned} v^{n+1} &= v^{n+1/2} + \frac{1}{2}hf(x^{n+1}) - \frac{1}{2}h\gamma v^{n+1/2} + \frac{1}{2}\sqrt{h}\sigma\xi^n \\ &\quad - \frac{1}{8}h^2\gamma\left(f(x^{n+1}) - \gamma v^{n+1/2}\right) - \frac{1}{4}h^{3/2}\gamma\sigma\left(\frac{1}{2}\xi^n + \frac{1}{\sqrt{3}}\eta^n\right) \end{aligned} \tag{6}$$

- Note that setting $\gamma = \sigma = 0$ results in Velocity Verlet

# Implementation Details:

## Algorithms Implemented:

- Velocity Verlet
- Optimization: Verlet Neighbour List
- Thermostat: Langevin Equations
- Potential: Truncated and Shifted Lennard-Jones Potential

## Code Details:

- Language: Cython (C + Python)
- Packages: NumPy + SciPy, matplotlib (for Plots and Animations)

# Testing Accuracy: Simulating LJ Fluid

- Benchmark outlined by: "The Lennard-Jones equation of state revisited" by JK Johnson, JA Zollweg, KE Gubbins - Molecular Physics, 1993. https://doi.org/10.1080/00268979300100411
- Involves simulating $N = 864$ particles with $r_c = 4.0\sigma$ as the cutoff radius of LJ potential
- Test Condition: Pass when potential energy within range given in paper
- Benchmarked for only a few paramters: $T^* = 6$ with $\rho^*$ ranging from 0.1 to 1.25
- Future Plan: Benchmark using: "Efficient Computation of Entropy and Other Thermodynamic Properties for Two-Dimensional Systems Using Two-Phase Thermodynamic Model" by SS Pannir Sivajothi, ST Lin, PK Maiti - The Journal of Physical Chemistry B, 2018. DOI: 10.1021/acs.jpcb.8b07147

# 3D LJ Fluid Simulation

# Testing Accuracy: Langevin Dynamics

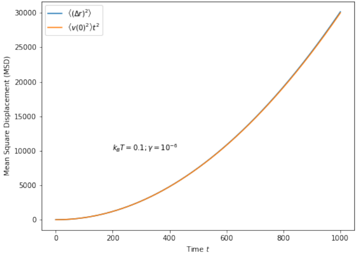
- Simple test using relation between mean square displacement (MSD) vs Time for different parameters.

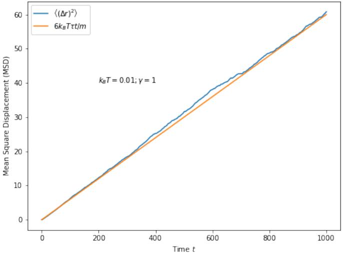
$$\left\langle r^2(t)\right\rangle = v^2(0)\tau^2\left(1-e^{-t/\tau}\right)^2 + \frac{6k_BT}{m}\tau t - \frac{3k_BT}{m}\tau^2\left(1-e^{-t/\tau}\right)\left(3-e^{-t/\tau}\right)$$

where $\tau = \gamma^{-1}$ is the relaxation time of the Brownian Motion

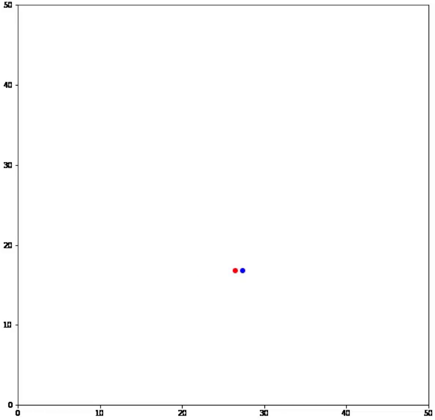
- Note that the above applies to 3D system where $Nk_BT = 3K_BT$
- For small time scales: $\left\langle r^2(t \ll \tau)\right\rangle = v^2(0)t^2$
- For large time scales: $\left\langle r^2(t \gg \tau)\right\rangle = 6k_BT\tau t/m$

# Testing Accuracy: Langevin Dynamics - Plots



# Simulating Active Particles: In Isolation




Link to Video

# Simulating Active Particles:

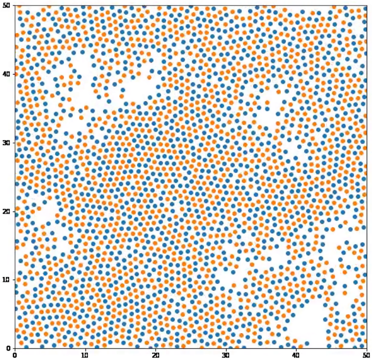



Figure: Simulation of 2500 Active dumbbells in 50X50 Box; $F = 1$ and $T = 0.001$

Link to Video

# Simulating Active Matter: Issues

- To observe Phase Separation and internal flow, the number of particles required to simulate is very large, resulting in long calculation time and heavy memory requirements.
- The code is single threaded.
  - Multithreading via OpenMP is simple for NumPy (BLAS Operations), but needs to be explicit for Cython
- Need to implement efficient algorithm for calculating LJ Interaction.