Hexcitations

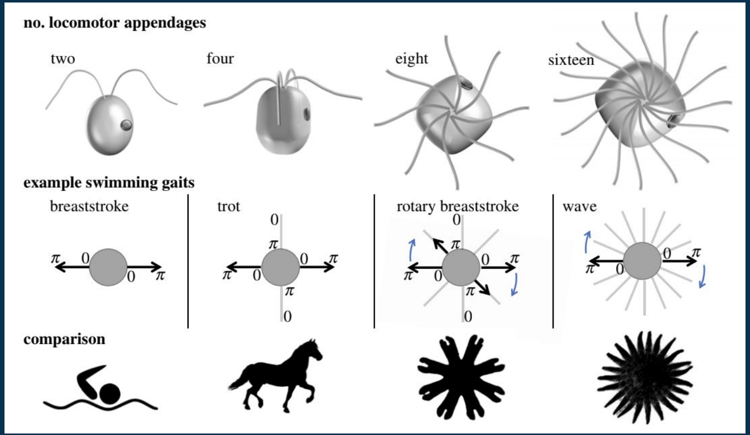
Andrew Mombay, Kaya Bruno, Helen Primis, Sarah Nguyen

Project Partner: Jack-William Barotta

Introduction

Active Particle & Collective Motion

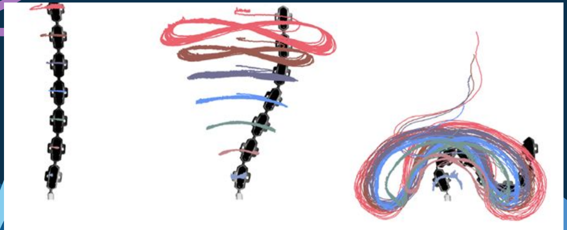
What makes a particle active? An active particle is one that takes energy from its environment in order to move autonomously. When active particles interact with one another, they have the ability to exhibit collective motion. Collective motion behaviors are seen across nature, including colonies of bacteria, schools of fish, and swarms of birds.

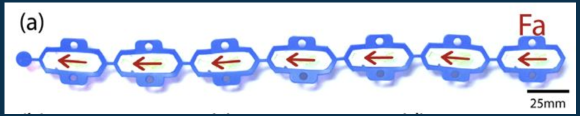


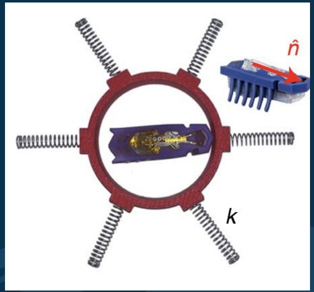
An application of collection motion includes investigation into the mechanics of animal appendages. One study models locomotive patterns of animals by analyzing microalgae, demonstrating that the locomotive functions of the animal's appendages are linked to a central pattern generator, and that active particle are driven by autonomous collective movements without the direct influence of a central force. In this way, animal appendages can be modeled as an active filament.

HexBugs as Active Particles

Hexbugs can serve as macroscopic representations of the microscopic particles referenced in the relevant literature on active particles. Taking into account that the inertia of our active particles can affect the collective movement of the system, coupled ODEs can model the aggregate motion upon linking the discrete Hexbugs together. We can design different elastic linkages by varying length and thickness to discover different modes of motion. Past researches have coupled the hexbugs with combinations of springs, rubber, soft plastic, and different styles of cages to provide support for the connections. Cilia-like filaments and webbed particles provide a foundation for our exploration into the world of active particles







Our Goals

Our project aims to study active particles and observe the collective behaviors emergent upon elastic coupling. We look to create an accurate model for an adjustable linked filament to mimic cilia-like behaviors by utilizing Hexbugs and 3D-printed Hexbug-like particles. We will create both a battery based and direct voltage sourced filament, observing it's motion when one end is fixed in place.

For further biological application, we look to model the appendage-based movement of a jellyfish by attaching multiple active filaments to a central body. In particular, we model ctenophores, or comb jellies, which employ bands of cilia to swim. As such, the design of these active filaments allows for the controlled study via a tabletop robotic setup that exhibits behaviors/observation of both cilia behavior at the microscale and ctenophores behavior at the macroscale.

Furthermore, the modular design of the linkages between Hexbug particles will allow us to explore how different orientations and combinations of filament power and length affect the interplay between the filament elasticity and activity.

Key Questions

- ★ How can we build a cilia-like filament by employing regular or 3D-printable hexbugs and mechatronic principles?
- ★ How do different orientations of hexbug coupling influence their collective actuation?
- ★ How does torsional stiffness influence synchronization and sustained oscillations in elastic networks?
- ★ Can we achieve biomimicry of cilia with our model?

Design Phases

Experimental Approach

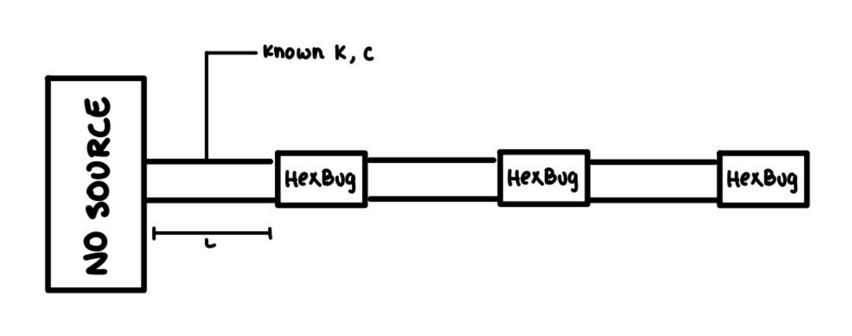
PHASE I: We utilize Hexbug Nanos from the Harris lab as the macroscopic active particles within the filament. We provide a linkage point made from a silicone mold between each discrete bug. The filament will be attached to one fixed non-powered source, such as a cilia attached to the parent body.

PHASE II: We 3D print and construct Hexbugs that can be connected via wiring to a fixed direct voltage source, eliminating the need of the battery powered Hexbug Nanos. This design will be modular to allow for adjustable filament length and attachments.

PHASE III: TIme permitting, we engage with the concept of biomimicry using the DV-source based filament design. We attach multiple appendages to one fixed central body, mimicking a jellyfish, investigating the Hexbug orientations that generate the most thrust considering different lengths and numbers of cilia attached to the beam.

Approach Phase I

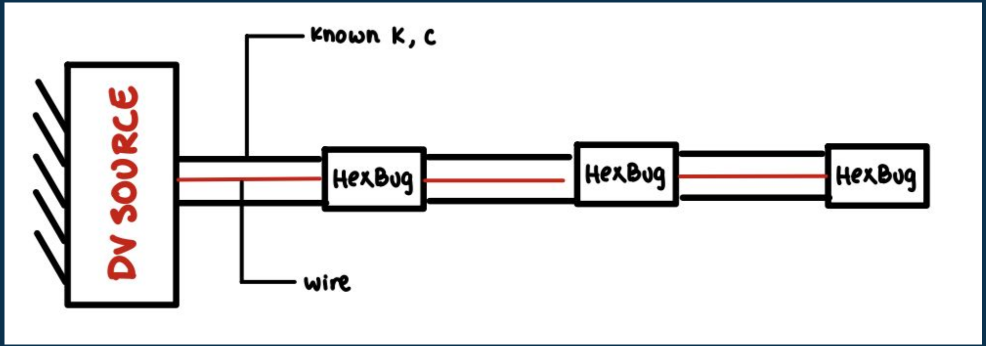
Experimental Workflow: Using the Hexbugs from the Harris Lab and silicone rubber linkages developed in self-designed molds, we will start by making a singular filament. This filament will be attached to a stationary body for us to explore the movements which arise.



Modeling: We use MATLAB image tracking to track the relative displacements and velocities of each particle to the fixed base. The purpose of this iteration is to understand the aggregate movement of the linked active particles when using battery-based Hexbugs and get a strong foundational prototype for future facets.

Approach Phase II

Experimental Workflow: We 3D print and construct Hexbugs that can be connected via wiring to a fixed direct voltage source, utilizing pager motors. This design will be modular to allow for adjustable filament length and attachments.

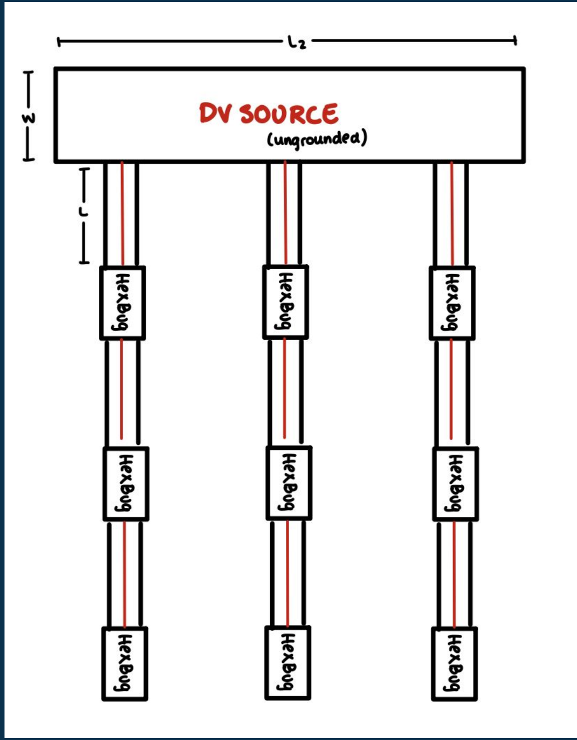


Modeling: We will perform the same data collection set as Phase 1, now employing a DV source based model instead of the battery Hexbugs to see if results are replicable. The nuance of modular design for this phase will allow us to easily change the number of particles attached to the fixed base to discover different aggregated movements, dependent on adjustable features.

Approach Phase III

Experimental Workflow: We engage with the concept of biomimicry using the DV-source based filament design. We attach multiple appendages to one fixed central body, mimicking a jellyfish, investigating the Hexbug orientations that move the fastest considering different lengths and numbers of cilia attached to the beam.

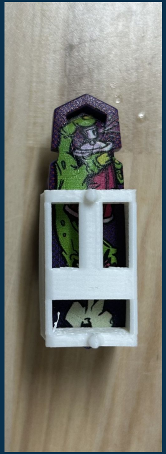
Modeling: We collect mean curvature and polarization data data using particle tracking on MATLAB for varying configurations of the apparatus upon changing shape and length of connection beam, length of filament, number of hexbugs on each filament. The purpose of this iteration is to apply the principles developed in the previous phases to replicate a biological phenomenon.

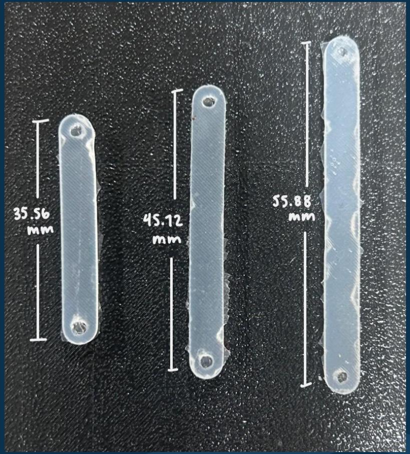


Physical and Numerical Modeling

Physical Modeling

Physical Modeling: To create the filaments, we use linkages made from a silicone mold to join the Hexbugs. This allows for semi-constrained movement along the x and y axes. Different iterations of a 3D-printed cage placed on top of each Hexbug allows the linkage to be affixed via prongs, as well as a show a T-shaped tracking marker to be utilized in MATLAB image processing.



Parameter Estimation

Spring Constant (k): To determine the spring constant, k, of each linkage length, we hung known masses (m) from the linkages and used MATLAB image processing to calculate displacement (x). The spring constant k is given by k = F/x where F=mg.

Young's Modulus (G): Determining the spring constant, k, allows us to calculate the Young's Modulus, G = kL/A where L = length of linkage, and A = cross sectional area of the linkage. The length of the small, medium, and large linkages were determined as 35.56 mm, 45.72 mm, and 55.88 mm respectively. We determined A = 11.91 mm^2.

Torsional Constant (C): After computing the Young's Modulus for each linkage, we then defined the torsional constant C = GJ / L, where the second moment of area $J \approx ab^3 \left(\frac{16}{3} - 3.36 \frac{b}{a} \left(1 - \frac{b^4}{12a^4} \right) \right)$

Elastoactive Parameter (σ): From supplementary material we define the elastoactive parameter σ = FaL/C where Fa is average force of Hexbug while pinned given as 15.7 ± 3.1 mN, The elastoactive parameter represents the ratio between active forces to bending forces. When sigma is small we expect straight line trajectory, and when sigma is large we have limit cycle trajectory for the Hexbugs.

Parameter Estimation

Linkage	Small	Medium	Large
Length L (mm)	35	45	55
Spring Constant k (N/mm)	.0614	.0456	.0362
Young's Modulus G (MPa)	0.215	0.205	0.199
Torsional Stiffness C	0.0023	0.0017	0.0014
Sigma σ	0.515	0.7812	1.0929

Simulations

Simulation Overview

Simulation: Employing equations from Ellen Zheng's 2023 paper on aggregated movement amongst N number of particles. Using Matlab to show how a singular filament with Hexbugs connected facing inward toward the pivot will move in relation to each other.

Data Collection: For each phase of this experiment, we use MATLAB particle tracking to track the relative displacements and velocities of each particle (x, y, theta) to the fixed base. This allows us to consistently measure the tracked values of the Hexbug's movement. We can break down their behavior based on their position, velocity, angle of movement, and general error due to inconsistencies in the hexbugs and the surface. In particular for phase III, we collect mean curvature (final angle - initial angle) and polarization data (orientation of the self-propelling particles) for varying configurations of the apparatus upon changing shape and length of connection beam, length of filament, number of hexbugs on each filament.

Active Particle Movement

- Employing governing equations and supplemental material for the filament based experiments found in a 2023 paper authored by Ellen Zheng to simulate and illustrate the x-y movement and the orientation development of theta
- F_a is given by the nature of the hexbug, L is the length of the hexbug, C is something we need to compute today depending on silicone, tao is something never used by the authors (dimensionless time), gamma is in the paper
- In turn, we will see the change in potential energy equal to the change in work of the system
- Using the summations and dimensionless parameters, we can develop a series of n coupled ODEs to run through a solver and fine the orientation and trajectory of each particle

$$\delta W = -\gamma \sum_{i=1}^{N} (\dot{x}_i \delta x_i + \dot{y}_i \delta y_i) - F^a \sum_{i=1}^{N} (\cos\theta_i \delta x_i + \sin\theta_i \delta y_i)$$

$$\sigma = F^a \ell / C$$

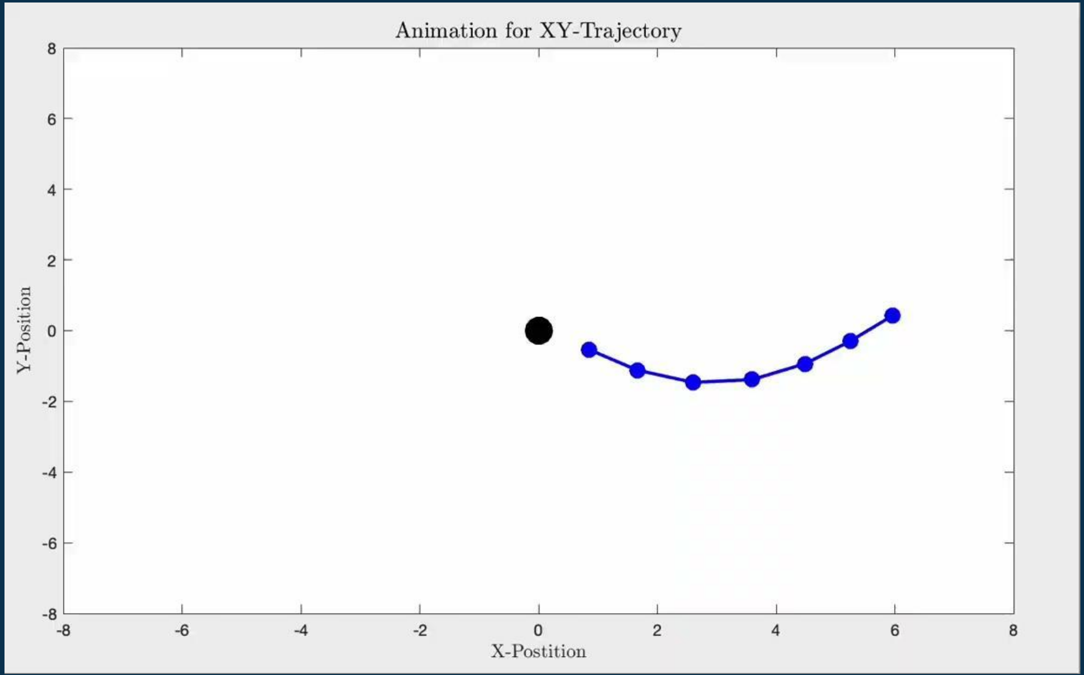
$$\tau = \gamma l^2 / C,$$

$$U = \frac{C}{2}\theta_1^2 + \frac{C}{2}\sum_{i=1}^{N-1} (\theta_i - \theta_{i+1})^2,$$

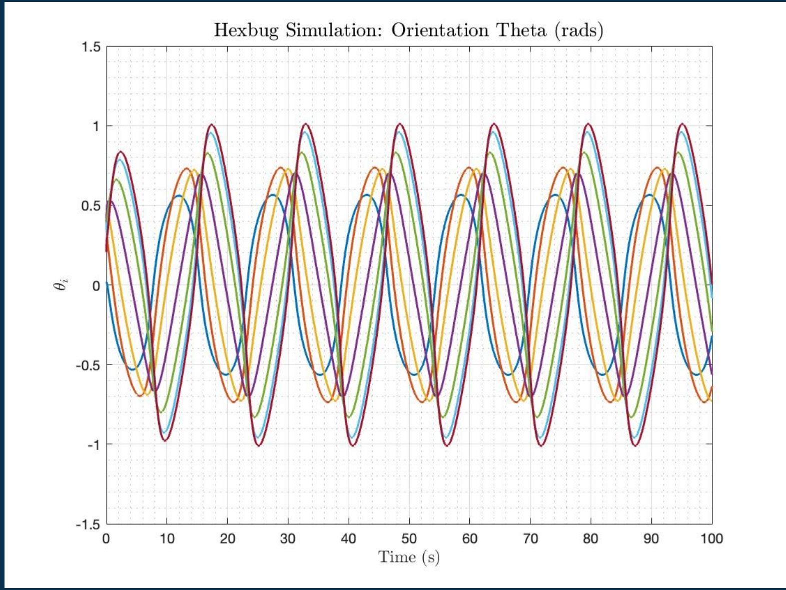
Active Particle Movement

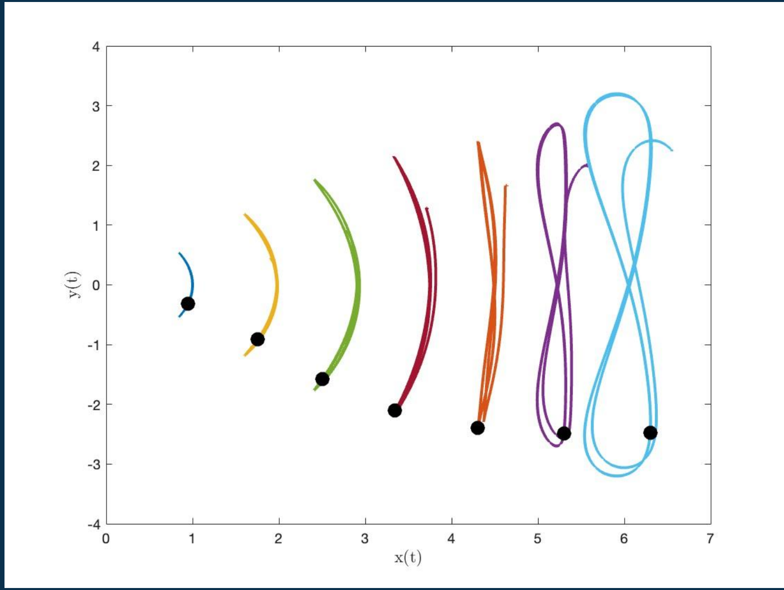
- Forces are in x, energy equation is in theta
  - Thus, the simulation script takes forcing equations and translates them all into theta to graph orientation
  - Then uses trigonometric relationships to graph XY trajectory back in those dimensions
  - If we have too much noise, the phenomena would not be visible
- Of Note: This simulation does not take noise
  - Demonstrates through experimental data that the more noise present in the trial, there is a decreasing nature of aggregated movement
  - If we were to account for noise, we no longer would be able to employ ODE45 in our Matlab simulation to solve our coupled equations
  - In turn, we could develop our own solver in Matlab or Python to account for the random nature of noise and see how it affects our preliminary simulation results

Simulation Animation



Simulation Plotting

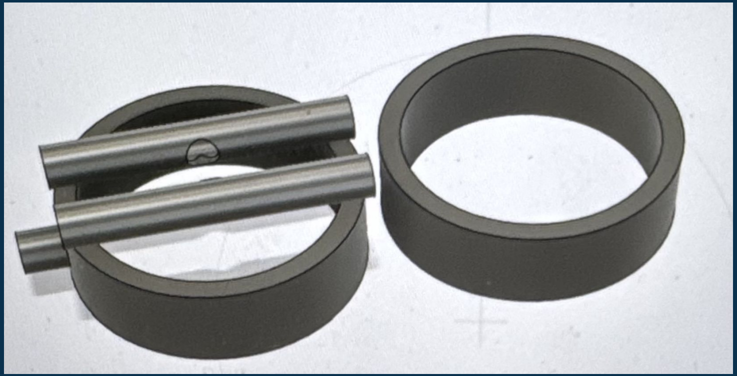




Initial Findings

Iteration I

Iteration I: Initially, we were going to attach the HexBugs using flexible PVC tubing, capturing each in a 3D printed shell before linking them together. However, the cages were too heavy and constrained the movement of the Hexbugs to the point where velocity data collection was not feasible. Additionally, the PVC tubing was too stiff, further constraining observable movement.



Iteration II

Iteration II: Considering the restricted movement of the Hexbugs due to the weight of the cages and stiffness of linkages, we looked to amend our setup by connecting the Hexbugs directly and using string as the linkage to increase mobility. From this, we were able to plot orientation vs time and X-Y trajectories.

Tracker Code
One Bug with No Cage

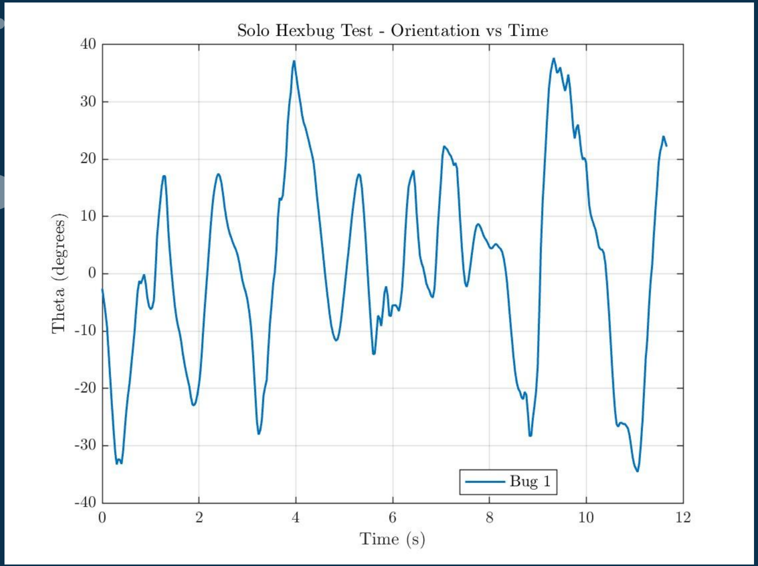
Takeaways from Plots:

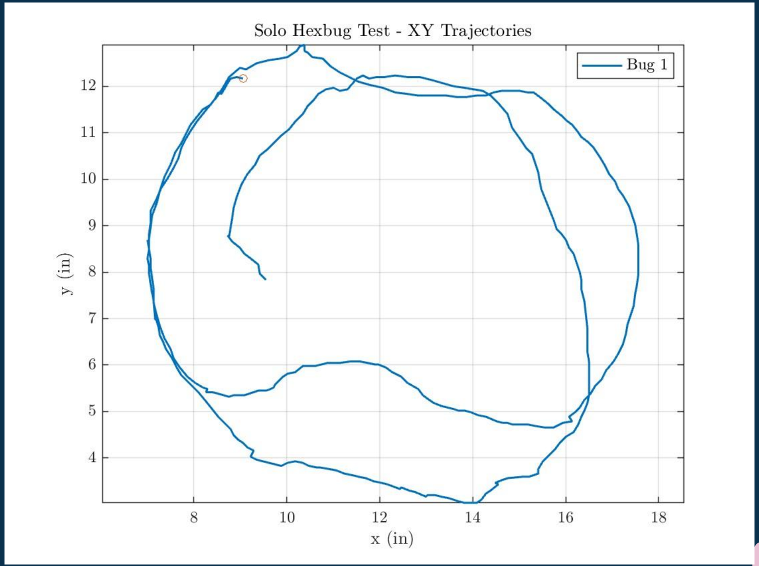
- Properly able to track trajectory
- Encountering sensitivity in the theta direction
- Issues with periodicity

Next moves:

- Add smoothing to the theta function in the code and account for periodic rotations of the bug
- Design cages with orientation-tracking design

Iteration II - One Bug Plotting





Iteration III

Iteration III: Upon identifying issues with initial tracker code, we designed a lighter cage mounted on top of the Hexbug, with “T” on it to try and get Matlab to pick up the orientation better. The cage also enabled us to attach the string connections closer to the center of the bug and near where the bulk of the mass is.

Tracker Code
Two Bugs with Cages

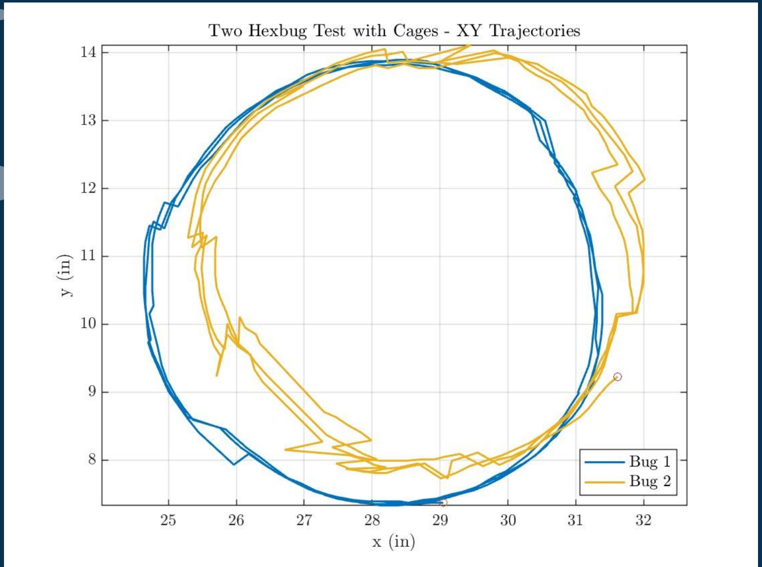
Takeaways from Plots (Next Slide):

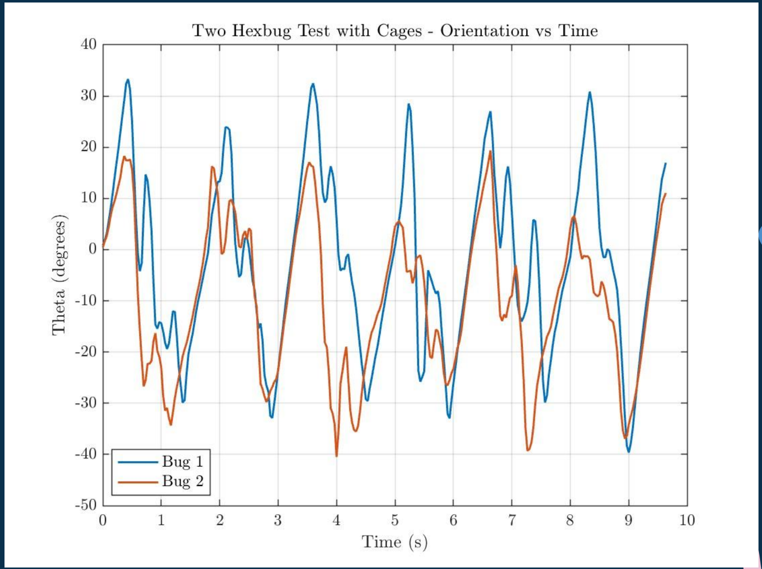
- Proof of concept that the XY tracking can account for multiple bugs in close proximity to each other
- Theta is still struggling with the periodic nature of the bug movement

Next Moves:

- Adapt theta tracker to not set a resetting zero line, which is why we were getting the recurring jumps in theta
- Make theta a summation of the orientation to see a linear relationship

Iteration III - Two Bugs Plotting

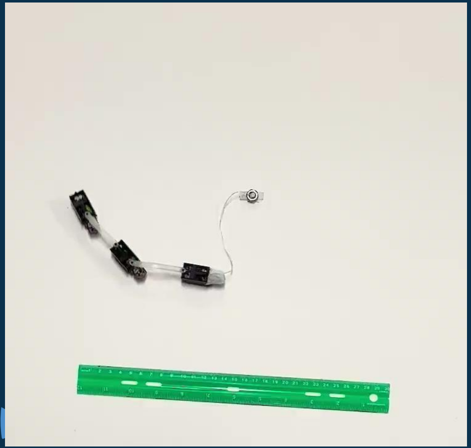




Iteration IV

Iteration IV: We amended the linkages to be made from silicone molds. For the code, fixed up the theta orientation function with Jack and printed new cages in a dark contrasting color to get the grayscale helper function to work as intended. Also used reference image function in Matlab for one of the cages Hexbugs to aid in tracking.

Tracker Code
Three Bugs with New Cages



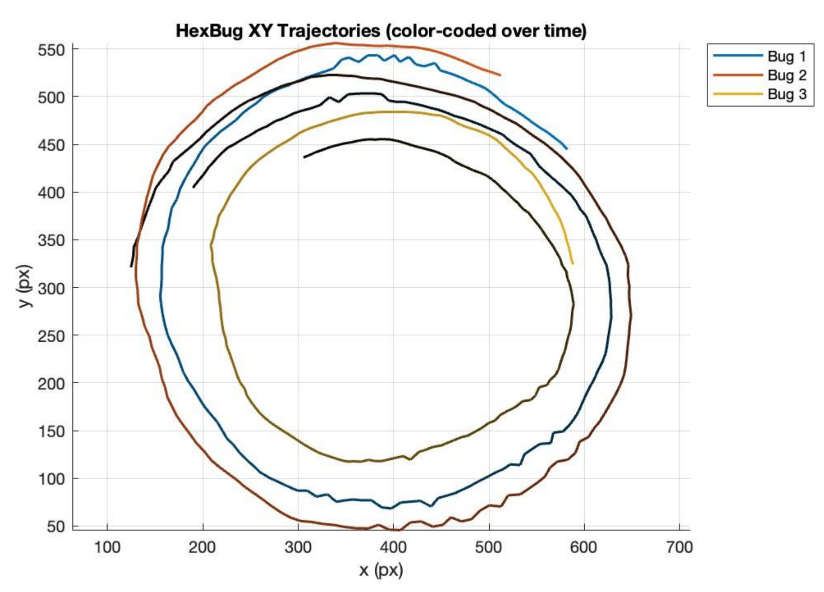
Takeaways from Plots:

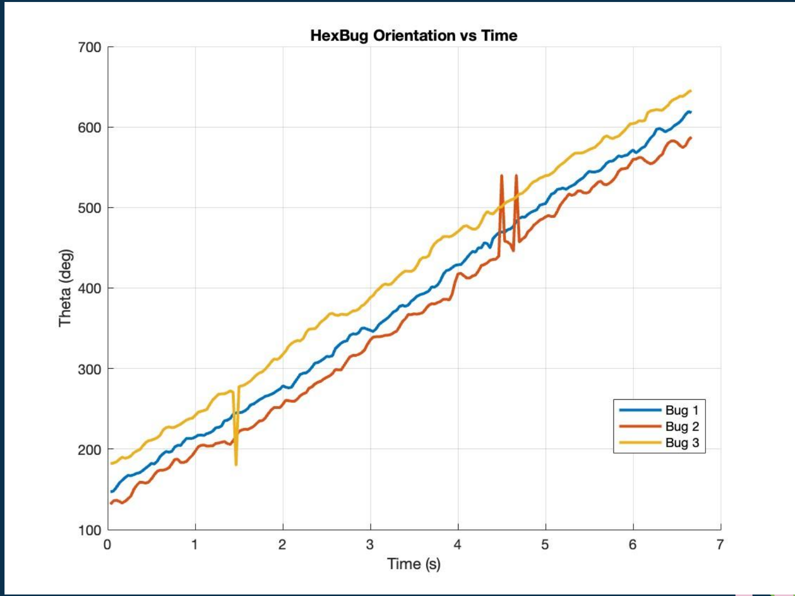
- Properly able to track trajectory and theta

Next moves:

- Develop shorter/stiffer linkages in an attempt to mimic the jellyfish-like phenomenon
  - So the linkages will not fold over when all running inward toward a pivot point
- Re-incorporate scaling through ruler

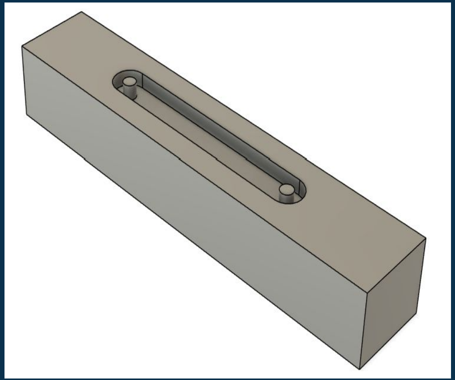
Iteration IV - Three Bugs Plotting

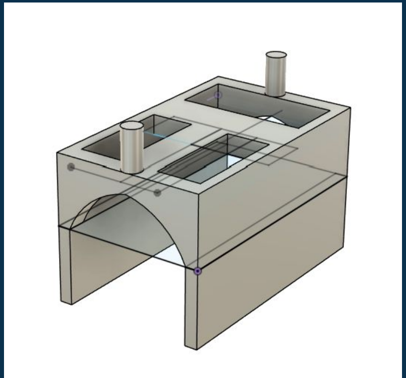




Iteration V

Iteration V: Additional efforts were made to revise the design to increase the fit of the individual hexbug cages and different lengths of silicone linkages were made to begin testing the optimal length. Additionally, development of a multi-filament system was begun to test for future iterations.





What We Learned and Next Steps

Preliminary Findings Summary

- ★ Evolved the physical setup of the experiment through iteration of the cages and linkages
  - Developed molds for silicone which can be easily modified for whatever σ value we desire for a certain trial
    - Where σ determines whether the chain acts as a rigid line (high σ) or an elastic filament (small σ)
- ★ Easily adjustable cages and linkages allowed us to explore what types of experimental setups are feasible to achieve particular data collections
- ★ Having the connection point of the Hexbugs above the surface on which they move helps to reduce friction and show increased movement
- ★ Data tracking is sensitive but achievable by employing Matlab helper functions and color contrast between bugs and background
  - Keeping code modular allows us to change number of particles as we see fit

New Guiding Questions

- ★ What is the best way to consistently track our Hexbug's orientation and position without experiencing spikes or dips?
  - ○ Is this a design change or a code change in order?
- ★ What will happen if we try to account for noise inherent to the bugs and the experimental setup?
  - ○ What does creating our own solver function in Matlab look like?
- ★ While changing the silicone molds is easy, the curing time is long; is there a way for us to use our numerical modeling more efficiently in order to cut down on the wait time between multiple linkage iterations?
- ★ Can we make our cages even more elevated off of the Hexbugs to make sure there is no skewing or interference with the legs?

Roadblocks and Challenges

- ★ Design for homemade Hexbugs and utilizing a DV source for power
  - Honing the 3D printable design for our accessible version of Hexbugs so that they do not just die once the battery dies (as seen with the store bought ones)
  - Goal of trying to achieve Brownian-style motion in our bugs even though they are made of different materials than our current ones
  - Getting circuitry to work, especially by troubleshooting the switches
- ★ Matlab struggling to totally orient the bug as it makes minor movements
  - Could pose problems when we have smaller theta values due to code sensitivities

Moving Forward and Next Steps

Design:

- ★ Continuing to explore the balancing relationship between collective activity amongst particles and flexibility in the linkages
  - ○ Honing design choices (i.e. thickness, length, cages, attachments, etc) to discover different phenomena
- ★ Debugging circuitry for the homemade Hexbug
  - ○ Playing around with dimensions in order to mimic similar movement to what we had seen
  - ○ Getting switches and pager motors in line and functioning with the coin batteries
- ★ Potentially exploring anti-friction design choices (i.e. ball bearings, air-hockey table)

Tracking:

- ★ As we iterate into our own Hexbug design, we need to ensure that tracking capabilities are still functioning even though the "particles" are different
- ★ Continue to hone smoothness in XY and theta plots

Timeline

★ **11/7:** Assemble phase II model with 3D printed Hexbugs connected to direct voltage source via wiring. Conduct experimental trials of different numbers of Hexbugs in single filament

★ **11/14:** Continue phase II experimental trials, and analyze experimental data

★ **11/21:** Begin phase III model with multiple filaments attached to one fixed central body, mimicking a jellyfish.

★ **11/28:** Conduct phase III experimental trials, investigating the Hexbug orientations that move the fastest considering different lengths and numbers of filaments attached to the beam. Work on technical report

★ **12/5:** Complete open source Github, including CAD designs, code, raw and processed data as well as technical report

★ **12/12:** Finalize final presentation slides, practice presentations

★ **12/13:** Final presentation of deliverable, and final group evaluations

Strategy

R3 - Reuse

Currently, in our preliminary stages, we have decided to use silicone to link our hexbugs together. We came upon this idea when talking to Jack about possible materials to link them together and he said that he had leftover silicone from an old project. Through collaboration, we found out that Andrew also had some leftover silicone from a different project, allowing us to use this material in our current iteration. Throughout the project we will continue to collaborate, as we were able to put old material to use that we would have not thought of otherwise.

R6 - Remanufacture

We have manufactured our 3D printed cage to allow for multiple types of linkages between the hexbugs. Because we would want to analyze the effects of different lengths and types of linkages, the pegs on the cage allow for flexibility in the types of linkages we will able to test. This allows for longevity in the project. Also the cages are able to be easily attached and removed if a hexbug dies. If we decide to re-engineer our cage, we will continue to think of ways that will allow us to find multiple purposes of our material, to reduce as much waste as possible.

Proposed Budget

Item	# Units	Cost
PLA Filament (Black)	1 spool	N/A
Pager Motors	20 (1 pack)	$14.29
Wiring, 20 AWG	N/A	N/A
PVC Tubing	10ft	$8.99
Ball Bearings (6705-2RS, 25*32*4 mm)	20 (2 packs)	$29.58
Power Supply	1	N/A
Microswitches	150 (1 pack)	$6.99
Ball Bearings (6705-2RS, 25*32*4 mm)	16 (2 packs)	$23.38
Coin Batteries	100 (1 pack)	$9.99
Coin Battery Holder	20 (4 packs)	$31.96
Total Cost		$125.18

Sources

Baconnier, Paul, Olivier Dauchot, Vincent Demery, et al. “Self Aligning Polar Active Matter.” Reviews of Modern Physics 97, no. No. 1 (2025): 1–26.

Barona Balda, Angelo, Aykut Argun, Agnese Callegari, and Giovanni Volpe. “Playing with Active Matter.” The American Journal of Physics 92 (November 2024): 847–58.

Levay, Sara, Axel Katona, Hartmut Lowen, Raul Cruz Hidalgo, and Iker Zuriguel. “Cluster Dynamics in Macroscopic Photoactive Particles.” Physical Review Letters 135, no. 098301 (2025): 1–7.

Miles, Jason G., and Nicholas A. Battista. “Naut Your Everyday Jellyfish Model: Exploring How Tentacles and Oral Arms Impact Locomotion.” Fluids Journal 169, no. 4 (2019): 1–43.

Sinaasappel, R., K.R. Prathyusha, Harry Tuazon, et al. Collecting Particles in Confined Spaces by Active Filamento Matter. May 9, 2025, 1–27.

Wan KY. 2019 Synchrony and symmetry-breaking in active flagellar coordination. Phil. Trans. R. Soc. B 375: 20190393.

Zheng, Ellen, Brandenbourger, Martin, et al. “Self-Oscillation and Synchronization Transitions in Elastoactive Structures.” Physical Review Letters (April 2023): 1-3.

3D Printable Hexbug Design and Instructions, Blog Link.

AI Usage

Tracker Code and Embedded Image Processing - Our tracking code bridges two source codes: one from Fluid Dynamics ENGN 810 for scale image processing and the other from our project partner, Jack-William Barotta's, research on the rotation and tracking of spinners. ChatGPT was used to bridge these codes and supplement debugging.

- Prompts used:
  - Conceptually explain how this tracking code for spinners can be translated to particle tracking
    - Provided us with overall guidance as to what we were able to keep in our tracking code and what we could get ride of in the context of our problem
  - Provide helper functions for image contrasting and box tracking
    - Aided in making our video into grayscale to exacerbate differences between the bugs and the platform we were running our trials on. And then adjusted the bounding boxed around the bugs after initialization clicks occurred to better track the bugs as they travel.

AI Usage

Tracker Code and Embedded Image Processing

- Prompts used (cont.)
  - Provide debugging help for mregform function
    - We were running into issues employing what was originally from Jack's code for spinners into our scenario for the hexbugs. Making great contrast in colors helped, but AI was able to add parts of the script to tighten orientation tracking through this function
  - Make plots prettier, please
    - Provided small code for using latex for aces, larger legends, gridlines, and other visually appealing aspects to make our plotting more clear to viewers

AI Usage

Spring Constant Image Processing and Young's Modulus Calculation - When calculating the Spring constant, we examined the deflection of the beam as we added weights across beams of all three lengths. Through this process, we collected a numerous amount of photos, where we decided image processing using Matlab to collect the data and find the spring constant for us. We then took that spring constant data and used Matlab to quickly compute the Young's Modulus

- Prompts used:
    - Create a Matlab script to calculate the spring constant of beams. We have 3 folders, named small, medium, and large, all filled with jpegs that are named the mass in grams. In each of the photos prompt the user to click the beginning and end of the beam, and then in that same photo prompt the user to click 100mm on a ruler to scale for mm/pixel
    - Create a Matlab script to calculate the Young's Modulus given the last code. The cross sectional area is a rectangle.