UNIVERSITY OF CALIFORNIA

Los Angeles

Dynamic modeling of untethered soft flagellated locomotion in viscous fluids and granular media

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mechanical Engineering

by

Yayun Du

2022

ABSTRACT OF THE DISSERTATION

Dynamic modeling of untethered soft flagellated locomotion in viscous fluids and granular media

by

Yayun Du
Doctor of Philosophy in Mechanical Engineering
University of California, Los Angeles, 2022
Professor Mohammad Khalid Jawed, Chair

Bacteria, one of the most common microorganisms in nature have been discovered to be propelled by rotating one or more slender rod-like filaments, called flagella. Flagellar propulsion results from a complicated fluid-structure interaction (FSI), between the structural flexbility of flagella and the hydrydynamic forces generated by the surrounding flow. This FSI can result in geometrically nonlinear deformation and structural instability. The former generally happens in compliant structures, such as soft robots, but is challenging to model and simulate while the latter is conventionally avoided because they normally causes structural failure. However, bacteria were recently discovered to utilize structural instability to change their movement direction, and this mechanism is called “buckling-to-turn" [1]. This mechanism can be used to control the moving direction of microrobots that can potentially revolutionize in-vivo targeted drug delivery or minimally invasive surgeries [2]. Nonetheless, there is no released work that developed soft robots that are able to replicate this mechanism. This is due to experimental challenges: flagellum is difficult to be controlled precisely on generic soft robots that are actuated by external fields [3] while self-actuated soft robots have a surprisingly low energy efficiency (mostly below 0.1%) [4].

Recently, the hydrodynamic force model - Resistive Force Theory (RFT) [5]

that was original defined in viscous fluids only accounting for the local force around each segment have been also found valid in granular media (GM) [6], such as sand, chia seeds, snow. This amazing discovery is a milestone to the exploration of locomotion in GM because it reveals a validated theory for movements in granules for the first time, and enables fast simulation of these movements thanks to the simplicity of RFT. Although previous work performed experiments on animals in the nature to comprehend their locomotion, work that employs soft robots as experimental platform to replicate the locomotion for practice is very few because of challenges in both experiments and simulations.

Our work solve the above problems by combining model experiments with state-of-the-art computational tools in computer graphics, and theoretical analysis towards developing predictive physical understanding of untethered flagellar propulsion in viscous fluids and granular media. We scale up the flagellar bacterial propulsion to desktop-scale soft flagellated robots. This allows for systematic experimental exploration of parameter space. In parallel, we conduct numerical simulations using the Discrete Differential Geometry (DDG)-based method, which was originally produced for special effects of the visually dramatic dynamics of slender structures, e.g., hair and fur in the animation industry. We adapt DDG-based simulator into engineering as a predictive computational tool and test it against our experiments. Overall, this dissertation makes four major contributions:

First, we introduce arguably the simplest soft multi-flagellated robot with a single binary control signal, which can move along an arbitrary 2D trajectory near air-fluid interface and at the interface between two fluids. Our work explores the performance of multi-flagellar propulsion near an open boundary as opposed to closed boundaries such as walls, as the former is much less studied but has numerous applications, such as flagellated robots used as baits, and tools for oil spill cleanup, water quality monitoring, and infrastructure inspection. We investigate the performance of the robot versus the number of flagella. In the end, we briefly propose the idea of incorporating machine learning with our fast-running simulator as a handy inverse design tool of flagellated robots.

Secondly, we use the same robot as experimental platform to explore the locomotion in granular media (GM). Numerically, the same DDG-RFT-Stokes' framework is applied again to model the hydrodynamics of locomotion moving through GM. Numerical and experimental results match quantitatively with each other when the number of flagella is two or three, validating the applicability of RFT in GM. However, “stick-slip" or “jamming", i.e., the robot randomly gets stuck at the same position with time passing, happens when the number of flagella turns four or five. The simulator fails to capture this, which proves the limitation of RFT in GM. Moreover, our main finding is that increasing the number of flagella from two to three decreases the speed of the robot. This is kind of counter intuitive, proving the complexity of flexible flagellar locomotion, the competition between the drag and propulsion. This indicates that our simulator is potentially applicable for unknown physics exploration. We find that there is an optimal rotational speed at which maximum efficiency is achieved. This highlights that our validated simulator can be used as a design tool for soft robots. Our third contribution is the implementation of an Euler-Bernoulli beam-based analytical framework that is both simple and capable of capturing the performance of the robot in GM.

Our forth contribution is developing the first untethered underwater robot with a flexible polymeric flagellum that can replicate bacterial “buckling-to-turn" mechanism. Additionally, we show the effect of flagellar geometrical properties on the performance of flagellar propulsion. Moreover, we prove that while bacteria utilize buckling to steer, flagellar buckling is probably not ample for a robust robotic system to follow any 3D prescribed trajectory. As a result, we develop a “mass-transformer" mechanism to make the robot system robust and

be able to reach a destination in the 3D space. Additionally, we are the first to demonstrate that the state-of-the-art continuous hydrodynamic model Regularized Stokeslet Segment (RSS) method can accurately model the hydrodynamic force on a rotating flagellum on an untethered robot (with a rigid head). We develop a numerical framework that incorporates (i) DDG to account for the elasticity of soft flagellum, (ii) RSS for the long term hydrodynamic flow by the rotating helical flagellum, and (iii) Stokes' law for the hydrodynamics induced by a spherical head. Our modular robot design enables researchers to use it as testbeds for studying generic flagellar propulsion. The “mass-transformer" mechanism together with simple flagellar buckling control scheme can be used for developing autonomous underwater robots for exploration and and exploitation of new environments.

The dissertation of Yayun Du is approved.

Jonathan Hopkins

Dennis W. Hong

Tetsuya Iwasaki

Mohammad Khalid Jawed, Committee Chair

University of California, Los Angeles

2022

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my Ph.D. advisor and mentor, Prof. M. Khalid Jawed, for his full support and guidance throughout my research career. Without his encouragement, I would not try faculty hunting at the last year of my Ph.D. and got offers. This dissertation could not have been finished on time either.

Next, I want to greatly thank my committee members, Prof. Iwasaki, Prof. Hong, and Prof. Hopkins, for their advice during my study and job hunting procedure. My great thanks to our collaborators in the projects of agricultural robot and autonomous coating at North Dakota State University: Prof. Rahman, Prof. Quadir, and Ph.D. researchers Afrina, Karan for their support during my stay in Fargo from 2019 to 2021. Sincere thanks to Professor Fisher, Laub, Wang, and Roychowdhury, for their full support during my faculty job search.

I also want to thank my supportive graduate teammates Zicheng and Deming, lovely undergraduate teammates and friends Andrew, Arthur, Bhrugu, Chenda, Guofeng, Jacqueline, Jingyi, Karunesh, Keerthi, Taiki, Wenjie, Yuchen, Zihang, Zihao, and all the members in SCI Lab, for their continued supports. We had so much fun doing research and hanging out together. I would love to show my pure love to my families in the U.S., Christina Flores and Kristine Hakizimana, for their selfless love and support to me, especially during my tough first two-year life at UCLA. My sincere appreciation goes out to all my friends that accompany me and involve in research discussion during my Ph.D., especially Fangjia, Junjie, Shengxin, Xiaoyu, Yi, Yihao, Yuzhen, and the entire MAE staff and Deeona at ECE for your professionalism and effort for boosting my research. Thanks go to Swapnil and Sandeep for pleasant research collaboration and career experience sharing. Thanks to more than seven hundred students for whom I served as TA for helping me improve my ability to deliver science clearly to the

general public.

My deepest appreciation goes to my beloved parents, younger brothers, and boyfriend, Hongbiao, for their unconditional love and care, constant supports, and tremendous understandings.

VITA

2012 - 2016	B.S.E. in Automotive Engineering, Harbin Institute of Technology, China.
2016 - 2018	M.S. in Mechanical Engineering, UCLA, U.S.A.

PUBLICATIONS

[1] **Du, Y.**, Saha, S., Sandha, S., Lovekin, A., Wu, J., Jawed, M. K., Srivastava, M., Siddharth, S., Chowdhary, M., “Neural-Kalman GNSS/INS Navigation for Precision Agriculture", *IEEE International Conference on Robotics and Automation (ICRA)*, 2023 (*Under review*)

[2] Lim, S., **Du, Y.**, Lee, Y., Panda, S., Tong, D., Jawed, M. K., “Modeling, control, and fabrication of robots inspired by flagella and cilia", *Bioinspiration and Biomimetics*, 2022 (*Under review*)

[3] **Du, Y.**, Bansal, K., Palan, E., Quadir, M., Jawed, M. K., “Robotic Painting: Mimicking Human Applicators", Journal of Coatings Technology and Research 2022 (*Revised and resubmitted*)

[4] **Du, Y.**, Miller, A.,$^{\#}$, Jawed, M. K., “Mechanics-based analysis on flagellated robots", *Soft Robotics,* 2022 (*Revised and resubmitted*)

[5] **Du, Y.**, Lam, J.,$^{\#}$, Sachanandani, K.$^{\#}$, Jawed, M. K., “Modeling the locomotion of articulated soft robots in granular medium", *IEEE Robotics and Automa-*

tion Letter (RAL), 2022 [link]

[6] **Du, Y.**,Zhang, G.,#, Tsang D.#, Jawed, M. K., “Deep-CNN based real-time robotic multi-class weed identification", *IEEE International Conference on Robotics and Automation (ICRA)*, 2022 [link, video1, video2]

[7] **Du, Y.**, Mallajosyula, B.#, Sun, D.#, Chen, J.#, Zhao, Z.#, Rahman, M., Quadir, M., Jawed, M. K., “A Low-cost Robot with Autonomous Recharge and Navigation for Weed Control in Fields with Narrow Row Spacing", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Prague, Czech Republic, 2021 (Finalists for **Best Paper Award on Agri-Robotics**, **Best Paper Award on Robot Mechanisms and Design**) [video1, video2]

[8] **Du, Y.**, Miller, A.#, Jawed, M. K., “Simple Flagellated Soft Robot for Locomotion near Air-Liquid Interface", *IEEE International Conference on Soft Robotics (RoboSoft)*, Yale, CT, 2021 [link, video]

[9] **Du, Y.**, Deng, Z. #, Fang, Z.#, Wang, Y.#, Nagata, T.#, Bansal, K., Quadir, M., Jawed, M. K.,“Vision and force based autonomous coating with rollers", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, pp. 9954-9960, 2020 [link, video1, video2]

[10] Qin, L., Huang W., **Du, Y.**, Zheng, L., “Genetic algorithm-based inverse design of elastic gridshells", *Structural and Multidisciplinary Optimization*, 62(5), pp.2691-2707, 2020 [link]

[11] **Du, Y.**, Miller, A.,#, Lovekin, A. #, Jawed, M. K., “Like bacteria: untethered underwater robots exploiting flagellar instability for steering" (*under review)*

CHAPTER 1

Introduction

Animals, plants, and microbes in nature provide engineers with much inspiration for robot designs. In general, their locomotions result from the interactions between their structures and the surrounding fluidic media, e.g., air, water, and blood; hence, they are modeled as fluid-structure interaction (FSI) problems. There are several types of propulsion methods utilized in robots to mimic bio-locomotion, such as flapping propulsion utilized by fish [7, 8, 9] and bats [10, 11], jetting- and paddling-based propulsion used by jellyfish [12, 13], octopuses [14] and cephalopods [15], and rotational and oscillating flagellated propulsion discovered in microorganisms. The former two propulsion approaches have been widely investigated in biomimetic robots, typically macroscale. They are designed to perform autonomous or semi-autonomous daily tasks, such as exploration and exploitation of new environments, search and rescue, and delicate underwater sample collection.

Bacteria, one of the most common prokaryotic germs, typically move through fluids by rotating one or more slender helical structures called flagella [16]. The rotation produces a propelling force and offers locomotion. Bacteria play a useful role in human digestion, vitamin synthesis in the intestine, and pest management, but they can also serve as disease-transmission agents. In the food industry (e.g., fermentation), environmental engineering (e.g., sewage treatment), biotechnology (e.g., antibiotics), and genetic engineering (creation of genetically modified organisms), bacteria have enormous economic values [17, 18]. Examples such as *Escherichia coli* that is frequently linked to disorders and illnesses such as diarrhea, urinary tract infections, respiratory sickness, and pneumonia. However, the vast majority of *E. coli* are not only harmless, but certain *E. coli* can even be utilized as markers for water pollution, meaning that their presence in drinking water indicates contamination despite the fact that they are innocuous. Flagellar propulsion is prevalent in a wide range of bacteria, with 90% of marine bacteria being uni-flagellated. Vibrio cholerae is a uni-flagellated bacterium that causes cholera, which can occasionally be fatal. Salmonella is the bacterium that causes salmonellosis, a type of food poisoning. Every year in the United States, it causes one million foodborne infections, 19,000 hospitalizations, and 380 deaths [17]. Lactobacillus is one of the most prominent friendly bacteria [18, 19]. Some varieties of Lactobacillus are found in the digestive, urinary, and genital systems of humans, and they can be utilized to treat and prevent a range of disorders.

As a joint, the cell body and flagellum are connected by a *hook* that is minuscule in comparison to the flagellum [1]. The existence of the short but flexible hook enables bacterial locomotion by breaking the scallop theorem [20], i.e., the time-reversibility constraint [21]. Additionally, Ref. [1] demonstrates that bacteria achieve “buckling-to-turn" by elongating and compressing the *hook*, as shown in Fig. 1.2(d1)(d2). Flagellar bacteria swimming stays in the low Reynolds number (*Stokes*) regime, where viscous effects dominate over inertia. Due to the simplicity and efficiency of propulsive organelle, i.e., flagella, bacterial locomotion has been cited as “the most efficient machine in the universe"[82] because bacteria by moving 25 - 450 times the body length per seconds [22].

Resistive Force Theory (RFT), the hydrodynamic force model that was original defined in viscous fluids only accounting for the local force around each segment have been also found valid in granular media (GM) [6]. This amazing discovery is a milestone to the exploration of locomotion in GM because it reveals a validated theory for movements in granules for the first time, and enables fast simulation of these movements thanks to the simplicity of RFT. Although previous work performed experiments on animals in the nature to comprehend their locomotion, work that employs soft robots as experimental platform to replicate the locomotion for practice is very few because of challenges in both experiments and simulations.

Motivated by these observations about bacterial locomotion modes and aforementioned work in locomotion in GM, we will present in this Thesis a uniform predictive framework consisting of model experiments, numerical simulations, and an analytical approach for flagellated locomotion in viscous fluid and GM. We will investigate a collection of modes of flagellated locomotion in viscous fluids and GM, e.g., swimming in circles near open boundaries in viscous fluids, increase or decrease of the robot speed based on the number of flagella in GM, and utilizing the buckling instability of the flagellum (“buckling-to-turn") Furthermore, we make efforts to translate into practice by making robotic systems robust and simple to control in Ch. 5.

The objective of this chapter is to offer a background for the Thesis consisting of the theoretical, experimental, and numerical methods that will serve as the basis for the research in the subsequent chapters.

Despite the fact that research about flagellated locomotion dates back to 1600s [23], researchers have been committed to observing the performance of flagellated locomotion and conducting experiments on scaled-up platforms (including mesorobots and macrorobots) to understand the underlying FSI problem. Recent advances in microfabrication make it possible to microfabricate robots with flagella. They are remote-controllable and have the potential to revolutionize clinical *in vivo* targeted therapy by inflicting considerably less tissue damage than conventional medical interventions. Accordingly, the research on macro-scale robots and prototypes that operate using similar mechanisms are being developed to further improve the existing theories and verify them.

This leads to a complex interaction between the geometrically nonlinear deformation in the soft flagellum and the hydrodynamics from the low Reynolds flow, such as tumbling [24], turning [1], and bundling [25]. Recently, there has been significant progress in understanding flagellar propulsion – particularly from a single flagellum – through experiments [1], computation [26], and theory [27]. The models used for this fluid-structure interaction problem are: Resistive Force Theory (RFT) [5], Lighthill's Slender Body Theory (LSBT) [5], and Regularized Stokeslet Segments (RSS) method [28] We use DER-LSBT framework to show that bacteria can exploit buckling in flagellum to precisely control their swimming direction, and then consider DER-RSS theory to perform the bundling behavior between multiple filaments.

The fundamental mechanics of thin structures is another key component of this thesis. The elastic gridshells that described by are of interests.

1.1 Geometric nonlinearity for functionality, not failure

The past several decades have witnessed expedited development of soft robots thanks to the structural compliance and reversibility of soft materials in comparison to rigid materials [29]. Unlike Euler-Bernoulli beams where geometric linear analysis is applied and equations of equilibrium are not updated with the change in geometry, geometric nonlinearity is ubiquitous in nature and engineering. Compliant structures, e.g., soft robots are prone to mechanical instabilities such as buckling that will be covered in Ch. 5. Buckling and geometric nonlinearities are conventionally associated with structural failures and thus were tried to be avoided [30]. Nonetheless, one of emerging themes in mechanics resides in utilizing nonlinearities and mechanical instabilities for mechanical metamaterials, stuctures of desired functionality [31], or locomotion [32, 33]. Recent advances in computational tools and experiments, as well as industrial demands, have led to the development of a vast array of metamaterials. Metamaterials are man-made substances that derive their extraordinary capabilities through mechanical structure instead of chemistry. Notable examples are structures that are foldable and deployable origami [34], stretchable electronics [35], and nonlinear deformation-induced locomotion [36, 37].

The ubiquity of flagellated microorganisms with instinctive sensing, motility, and non-motile functions in nature inspires scientists to investigate the underlying mechanisms and further apply them in practice. This includes *in vitro* and *in vivo* biomedical applications, which were achieved through advancements in manufacturing. Hence, this review reports the design, fabrication, control, and modeling of microrobots, mesorobots, and macrorobots inspired by flagella and cilia. Typically, their design, fabrication, and control are interdependent. As depicted in Figs. 1.3(a)(b), we mainly report synthetic microrobots that are magnetically and optically controlled. We also briefly summarize microrobots that are controlled by other approaches. Additionally, we cover bio-hybrid microswimmers that integrate biological flagella or cilia with artificial substrates, as seen in Fig. 1.3.

The tendency of incorporating geometric nonlinearities and instabilities into systems for locomotion inspires us to place our emphasis on modeling large deformation in locomotive systems. Finite element methods (FEM) can be powerful tools for accurately modeling large deformations, but they are so time-consuming, fluid-involved particularly, that they are even less effective than performing trial-and-error experiments directly. Despite the emergence of an increasing number of machine learning (ML)-assisted FEM methods [38] benefiting from the upsurge in ML, released models are typically data-hungry and only confidently applicable to cases from which experimental data have been collected. In this Thesis, we strive to predictively comprehend and quantitatively capture the nonlinear configurations in fluid-structure interaction related applications. The physical insight and predictive function of the system behav-

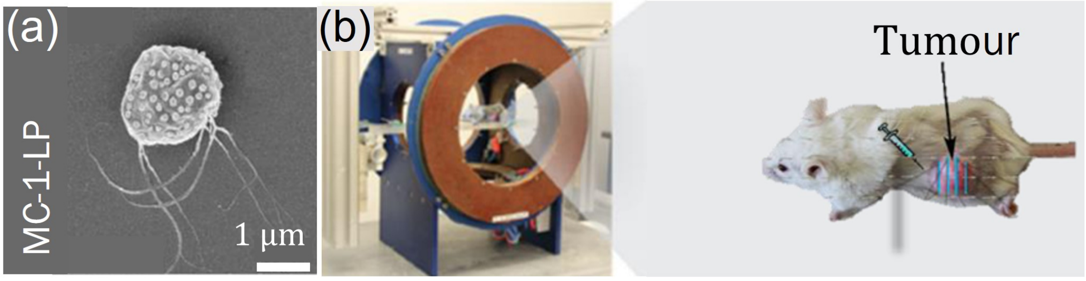

Figure 1.1: Illustration of successful application of micro flagellated robots (flagellated MC-1 cells) for targeted drug delivery (to kill tumors).

ior gained from the computational framework in this Thesis show the significant potential of our numerical tool to be used as efficient design tools for generic soft robots to save time, energy, and costs. Moreover, we also demonstrate the potential of incorporating our fast-running simulator with machine learning techniques as inverse design tools for soft locomotive systems.

1.2 Efficient untethered flexible flagellated underwater/underground robots development and simulations

Robotics is pushing forward the boundary of exploration, by targeting distant planets in space, soils, and the depths of our oceans. Because of the human potential for minuscule robots to revolutionize targeted drug delivery [39], a large number of researchers are either developing macro-robots to understand the underlying mechanics of flagellar locomotion or advancing micro-fabrication in order to minimize and engineer micro-robots. As illustrated in Fig 1.1, one of successful cases in applications of micro flagellated robots to targeted drug delivery is the injection of flagellated MC-1 cells into mice to eliminate tumours. Before diving into bacteria-inspired robots, we take a look at the bacterial structure. A micrograph of a Vibrio cholerae cell is presented in Fig. 1.2(a). In Fig. 1.2(b), a fluorescently labeled experimental image of Escherichia coli is shown. The bacterial cell body is spheroidal in shape with radius on the order of 1μm, and is attached to single (Fig. 1.2(a)) or multiple (Fig. 1.2(b)) flagella. A little hook, in comparison to the flagellum, may serve as a connection between the cell body and the flagellum [1].

Fig. 1.3 summarizes flagellar motility and collage of existing flagella-inspired robots. There are also a number of robots inspired by cilia, which are structurally shorter than flagella. We discuss it to demonstrate that the outcomes of this study are applicable to cilia research. The following content reports the design, fabrication, control, and modeling of microrobots, mesorobots, and macrorobots inspired by flagella. Typically, their design, fabrication, and control are interdependent. Existing magnetic, optical, biohybrid, chemical, and other microbots are shown in Fig. 1.4. The detailed summary of the pros and cons of each actuation method is in Table 1.1 for the ease of reference.

As we can see, microrobots are primarily intended for medical applications, where design, fabrication, and control are typically coupled with trial-and- error processes. In addition, miniaturization poses fundamental technical challenges, such as power sourcing, precise actuation, multifunctional integration, and post- injection recovery or biodegradation. It is therefore time-consuming, labor-intensive, and prohibitively expensive. The trial-and-error processes are significantly prolonged if the fundamental physics of the movement of microrobots is not adequately understood. As a result, many flagella or cilia-inspired macroscopic robots are used as scaled- up experimental platforms to explore the underlying mechanisms of their locomotion, including hydrodynamics, and optimize the actuation efficiency. Regardless of length scale, the flows surrounding macroscopic robots are governed by Stokes equations so long as the Reynolds number remains low.

In contrast, flagella-inspired macro robots are far less studied and can be categorized based on structural changes, the proximity of a boundary, and the computational models used to simulate the robots. The details (literature re-

Table 1.1: Summary of the different actuation methods for microrobots

Methods	Pros	Cons	Bio-hybrid compatible microorganisms
Magnetic	• Penetrable • Magnetic fields are harmless to human tissues and organs • Low-powered magnetic fields are sufficient • High speed and force output	• Magnetic parts are not biocompatible, and are typically coated with biocompatible materials • Not easily scalable • Challenges in small scale robot due to scaling issues • Bulky magnetic actuation systems that requires cooling	• Magnetospirillum gryphiswaldense [54] • Bovine sperm cell [55] • Marine magnetotactic ovoid [56] • Salmonella typhimurium [57, 58, 59]
Optical	• Easily scalable fabrication methods • Highly biocompatible materials • Ideal for organ-on-chip applications	• Optical radiation has low penetrability (NIR penetration depth: around 1–2 cm) • Requires high power light for actuation in fluids due to attenuation • Not suitable for medical applications due to limited depth	• Chlamydomonas reinhardtii [60]
Chemical	• Does not require external power source and can be self-propelled • Can easily be miniaturized	• Directional control is limited • Many methods still rely on toxic catalysts • Dependent on medium	S. marcescens [61]
Acoustic	• Biocompatible • Penetrable • Dual purpose for single acoustic wave (actuation and imaging)	• Directional control has not yet been achieved • Bulky acoustic actuation system	N/A

view) will be given below. The structural difference includes the quantity and flexibility of the flagella and the presence or absence of a head, the working environment, i.e., granular media or viscous fluids, and the employment of a motor as the actuator. Pictures of meso robots and macro robots can be found in Fig. 1.5. Farma *et al.* investigated the flagella-inspired locomotion in low *Re* fluid in a cylindrical channel to comprehend the motion and development of micro-robots inside capillaries and blood vessels (Fig. 1.5(a)) [62, 63]. The flagellum is composed of rigid steel wire, so it can swim forward but fails to replicate the locomotion mode of bacteria with a flagellum, e.g., “turning-by-buckling" [1].

Additionally, the steel wire tail of the robot prevented it from being neutrally buoyant, so it rested at the wall of the horizontally placed channel; hence, a rigid BC is close. Computationally, it modeled the locomotion through a computationally intensive computational fluid dynamics (CFD) model using off-the-shelf COMSOL software. This study used a previously validated CFD model [69] to study the effects of geometric parameters of the tail and the radial position of the robot in the silicone-oil-filled tube on its swimming performance. The comparison between experimental and simulated results is qualitative rather than quantitative because the exact location and orientation of the robot in the tube were not measured. The same robot structure and magnetic control method were applied in [70] to learn the propulsion of micro-organisms in complex non-Newtonian fluids. Later, Ye *et al.* fabricated submillimeter-scale (mesoscale) swimming robots with multiple flexible straight ABF that are actuated and controlled by rapidly rotating magnetic fields and conducted numerical simulations [3] to predict the swimming performance of the robot and guide its design (Figs. 1.5(c1)-(c3)) [64]. Xu *et al.* designed an experimental setup and scaled-up helical nanobelt swimmers with magnetic coatings on the tail, with and without a head, to compare their rotating propulsion performance [71]. Similar scaled-up helical microswimmers were employed in [63, 65] (1.5(d)) to design a closed-loop control strategy that is robust and accurate to disturbances so that the robot can follow a 3D path. Meanwhile, Jawed *et al.* [72] were the first one to quantify the propulsive force and buckling instability of a flexible helical rod rotating in a low Reynolds number fluid by modeling geometrically nonlinear elastic rods with discrete elastic rods (DER) [73] and the fluid loading with a nonlocal hydrodynamic model [5]. The effect of a rigid boundary was then incorporated into the same framework. A systematic parametric study was conducted to quantify the dependence of the wall effects on the geometric parameters of the helical filament [74]. However, in Jawed's work above, the helix was fixed without a head, and thus the head effect was ignored. His work does not comprise developing a robot. Nevertheless, it still deserves attention as one of the pioneers in adapting a tool introduced by the computer graphics community to perform fast simulations of flexible structures and soft robots. A similar simulation framework could be found in [75], demonstrating that bacteria can exploit flagellar buckling to control their swimming direction, i.e., bacterial "turning-by-buckling" mechanism [1] that is demonstrated in Figs. 1.2(d1)(d2). Lim *et al.* developed a multi-flagellated robot (Figs. 1.5(f1)-(f2)) and applied the DDG method to simulate the structure of the robot, Regularized Stokeslet Segments to model the hydrodynamics on the flagella, as well as a penalty-based method to model the contact between the flagella [68].

Overall, the comprehension of bacterial flagellar locomotion is still relatively shallow due to the following challenges to replicate the locomotion:

- Most soft robots are tethered because of the bulky actuators, such as pump [37], and high-current circuits [76]. Tethered robots have limitations replicating flagellar locomotion in nature. Moreover, robots made from state-of-the-art soft materials such as shape memory alloy and liquid crystal elastomer [77] typically have bi-stable states, meaning that the locomotion between the transition is not controllable;

- To explore flagellar locomotion in viscous fluids, the first challenge to conquer is to make the robot neutrally buoyant while perfectly matching the center of mass and center of buoyancy. However, underwater robotic systems are only stable when the center of mass is located below the center of buoyancy. This requires ingenious design if the robot is going to be translated into practice.

- Mechanical nonlineariy and instability, such as flagellar flexibility is a double-sided sword. Robots with a flexible flagellum can achieve “buckling-to-turn" if the flagellum buckles but cannot realize an efficient locomotion if the flagellum is too soft and thus too easy to buckle.

1.3 Numerical method: discrete differential geometry

Although most soft robots look nothing like a rod, they can still be modeled as a collection of elastic rods by being discretized through discrete differental geometry (DDG). The Discrete Elastic Rods (DER) [73, 78] approach is one of the most prominent applications of DDG in physics-based modeling for capturing the geometrically nonlinear deformation of thin elastic rods, such as curled hairs. The numerical simulations presented in this Thesis employ Discrete Elastic Rods (DER)-based simulator. DER lays the basis on Kirchhoff elastic rod [79]. In the 18th century, Euler and Bernoulli proposed a simple one-dimensional beam model, which marked the appearance of rod theory [80]. In the 19th century, Kirchhoff and Cosserat generalized their work [81, 82]. Recently, researchers in the field of computer graphics devised a rapid numerical framework, Discrete Elastic Rods technique, to conduct the nonlinear dynamics of elastic rods [73, 78]. As this Thesis aims to model the dynamics of untethered soft flagellated locomotion in viscous fluids and granular media, DER-based framework is augmented to include various boundary conditions (§ 4.3.3) and external forces, mainly the hydrodynamic loading from the surrounding viscous fluids and granular drag which will be incorporated in § 5.4.2, § 4.3.3, and § 2.4.3. Note that our application of DER is restricted to rods with circular cross-sections and helical or straight shapes; nevertheless, this approach is applicable to arbitrary (i.e. curved) undeformed configurations and arbitrary cross-sections (i.e. non-circular). The kinematics and elastic energy in a discretized geometry situation are presented here, followed by the formulation of forces and the time marching scheme.

The continuous centerline of a rod is discretized into N nodes: $\mathbf{x_0}, ..., \mathbf{x}_{N-1}$ and in-between $N-1$ edge vectors: $\mathbf{e}^0, ..., \mathbf{e}^{N-2}$ such that $\mathbf{e}^k = \mathbf{x}_{k+1} - \mathbf{x}_k$ and $k = 0, \ldots, N-2.$, as shown in Fig. 1.6. Hereafter, subscripts will be used to express quantities related with nodes, such as $\mathbf{x}_k$, while superscripts will be used to denote values associated with edges, such as $\mathbf{e}^k$. At every edge, $\mathbf{e}^k$, an orthonormal adapted reference frame $\left\{\mathbf{d}_1^k, \mathbf{d}_2^k, \mathbf{t}^k\right\}$ and a material frame $\left\{\mathbf{m}_1^k, \mathbf{m}_2^k, \mathbf{t}^k\right\}$ are attached; these two sets of frames share the same tangent $\mathbf{t}^k = \mathbf{e}^k / |\mathbf{e}^k|$ as one of directors. Referring to Fig. 1.6, at each time step, reference frame is updated through parallel transport (in the time scale). Then, material frame is obtained after applying a scalar twist angle θ^k. This accounts for the physical bending and twisting happening during the adjacent time steps. Please check Ref. [83] for a detailed pedagogical explanation of the DER method. The $4N-1$ degrees of freedom (DOF) vector of the rod consist of positions of nodes and twist angles, i.e., $\mathbf{q} = \left[\mathbf{x}_0, \theta^0, \mathbf{x}_1, ..., \mathbf{x}_{N-2}, \theta^{N-2}, \mathbf{x}_{N-1}\right]$. Subsequently, on the basis of this kinematic representation of an elastic rod, we provide the formulation of elastic energies and forces, and the time marching scheme inside the DER solver.

An elastic rod is essentially modelled as a mass-spring system: each node (and edge) is associated with a lumped mass (and rotational inertia). The associated elastic energies are the linear sum of stretching, bending, and twisting energies. For a rod with an isotropic circular cross section and physical parameters: Young's modulus E, and shear modulus G, stretching, bending, and twisting energies are as follows [73, 78]

$$E_s = \frac{1}{2} \sum_{k=0}^{N-2} EA(\epsilon^k)^2 |\bar{\mathbf{e}}^k|, \tag{1.1a}$$

$$E_b = \frac{1}{2} \sum_{k=0}^{N-1} \frac{EI}{\Delta l_k} \left[(\kappa_k^{(1)} - \bar{\kappa}_k^{(1)})^2 + (\kappa_k^{(2)} - \bar{\kappa}_k^{(2)})^2 \right], \tag{1.1b}$$

$$E_t = \frac{1}{2} \sum_{k=0}^{N-1} \frac{GJ}{\Delta l_k} (\tau_k)^2, \tag{1.1c}$$

where A is the area of cross-section, EI is the bending stiffness, I is the area moment of inertia, GJ is the torsional rigidity, J is the polar moment of inertia, ϵ^k is the stretching strain associated with the k-th edge, $\bar{\mathbf{e}}^k$ is the undeformed edge length of the k-th edge, $\mathbf{e}^k$. At the k-th node, $\kappa_k^{(1)}$ and $\kappa_k^{(2)}$ are the bending curvatures while $\bar{\kappa}_k^{(1)}$ and $\bar{\kappa}_k^{(2)}$ stand for the curvatures in the undeformed configuration, τ_k is the twist, and $\Delta l_k = \left(|\mathbf{e}^k| + |\mathbf{e}^{k+1}| \right)/2$ represents the Voronoi length. Note that strain measures ϵ^k, $\kappa_k^{(1)}$, $\kappa_k^{(2)}$, and τ_k, are functions of the degrees of freedom $\mathbf{q}$. The above formulation can still be used for the case of non-circular cross-section with minor changes, which is detailed in Refs. [73, 78].

At each degree of freedom q_j, the elastic forces (coupled with nodal positions) and elastic moments (coupled with twist angles) are

$$F_j^{\text{int}} = -\frac{\partial}{\partial q_j} \left(E_s + E_b + E_t \right), \tag{1.2}$$

where $j = 0, 1, ..., 4N - 2$ as the dimension of nodal positions is $3N$ and the dimension of twist angles is $N - 1$. Depending on the variables that are known at each time step during simulation execution, the simulator may be semi-implicit or implicit. Implicit approach can make the simulator run stably even under big time steps. Implicit Euler integration is used to solve the following $4N - 1$ equation of motions and update the DOF vector $\mathbf{q}$ and its velocity (time derivative of

DOF) $\mathbf{v} = \dot{\mathbf{q}}$ from time step t_k to $t_{k+1} = t_k + h$ (h is the time step size):

$$\mathbf{M}\Delta\mathbf{q}_{k+1} - h\mathbf{M}\mathbf{v}_k - h^2 \left(\mathbf{F}^{\text{int}}_{k+1} + \mathbf{F}^{\text{ext}}_{k+1}\right) = \mathbf{0} \tag{1.3a}$$

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \Delta\mathbf{q}_{k+1} \tag{1.3b}$$

$$\mathbf{v}_{k+1} = \frac{1}{h}\Delta\mathbf{q}_{k+1}, \tag{1.3c}$$

where $\mathbf{F}^{\text{ext}}$ is the external force vector (e.g., gravity and damping force), $\mathbf{M}$ is the diagonal mass matrix comprised of the lumped masses, and $\dot{()}$ represents derivative with respect to time. The superscript $k+1$ (and k) denotes evaluation of the quantity at time t_{k+1} (and t_k). Newton's method is used to iteratively solve the $4N-1$ sized equation of motions.

In summary, DER is formulated based on the classical Kirchhoff theory, e.g., the elastic energies are given by the curvatures of rod centerline, and the internal elastic forces required by equations of motion are derived from the energies in a discrete format. Our researches adapt the well-established DER simulation to develop discrete differential geometry (DDG) to simulate the dynamic movement of untethered soft flagellated locomotions in granular media and viscous fluids. The flagellated locomotions reported in this Thesis are driven by a rotary motor; hence, robot dynamics is complicated, involving both rotation along the robot's long axis and translational movement. Overall, although the robots look nothing like rods, our DDG-based simulator can still well capture the performance of such complicated systems. These locomotions are fluid-structure interaction problems; we will consider hydrodynamic loading and integrate it to DDG in Ch. 2, 3, 4, and 5.

1.4 Outline of the Thesis

This chapter introduces the novelty of our Thesis or new science that we unveil – utilizing geometric nonlinearity for functionality through efficient untethered flexible flagellated underwater/underground locomotion development, and a well-established numerical tool (DDG) for modelling the mechanics of rods that serve as the foundation for this Thesis. Subsequent chapters present some robots as testbeds to explore the mechanics of locomotions in granular media and viscous fluids, as well as extensions of the DER method to investigate the dynamics in soft robots. The primary contributions of this thesis are as follows:

Ch. 2 displays, for the first time, an untethered robot with multiple soft flagella capable of moving through granular media, where the drag on the flagellum contributes to propulsion. The propulsion results from the battle between the drag on the head and flagella and the propulsion of flagella. One counterintuitive phenomenon we find is that the speed of the robot diminishes as the number of flagella increase, demonstrating the complexity of this system. The drag from the surrounding granules onto flagella is modelled by resistive force theory (RFT). Meanwhile, we develop a fully implicit DDG-based simulation framework that incorporates actuation of systems, e.g., the rotation of motor, through a time-varying undeformed configurations. This differs from the normal simulation approach, in which undeformed configurations at each time step are often unchanged. In this innovative manner, the simulation is entirely implicit, capable of precisely simulating the system's dynamics over a huge time step, and therefore computationally efficient. The high computational efficiency of our validated simulator makes it an ideal tool for simulating soft robots, inverse design and optimization of such systems. Also, the large amount of data generated by the simulator in a short time can be learned by machine learning algorithms to learn the underlying physics in these systems and to function as accurate predictors.

The results of our simulator in Ch. 2 quantitatively match those from experiments, verifying that RFT is powerful to model the hydrodynamic forces on the robot in granular media. In Ch. 3, we further develop an analytical tool on the basis of simple Euler-Bernoulli beam theory. The analytical framework completely resolves the system's dynamics, as the number of equations equals the number of system variables. Our analytical approach yields results that are qualitatively consistent with DDG-based simulation results and experimental data, acting as the gold standard and providing physical context for our DDG-based simulator. Our analytic methodology also suggests that there should be a second design in which the robot has more flagella and runs quicker. Under the guidance of this framework's forecast, we revise our robot design to achieve the prediction. The simplicity of this analytic framework and the effectiveness of its predictive function contrast starkly.

The same robot platform and DDG-based simulation framework are then proved to be applicable to explore the locomotion in viscous fluids (low Reynolds number fluids) in Ch. 4. his chapter employs the aforementioned methods to reveal the science behind why bacteria swim in circles near boundaries. As the mechanics of bacteria swimming near close barriers, such as walls, have already been investigated, we intend to conduct tests near open boundaries. Finally, we determined that the circular motion of bacteria is caused by the unequal distribution of the fluid's viscosity. The computational efficiency of the numerical method enables it to run faster than real-time on a desktop processor, which makes it ideally suited for algorithms that iterate over a wide variety of parameters in order to select a robot design or locomotion strategy. Therefore, at the conclusion of this chapter, we propose how our simulator might work with machine learning models as inverse design tools for robots of this type.

Ch. 5 exhibits a new platform, an untethered underwater robot with a rigid head and a helical soft flagellum that is able to move forward in straight lines when the rotational speed of its motor is below a threshold but turns a large angle when the rotational speed exceeds the threshold. We are the first one to develop and use the arguably simplest robot to replicate the same mechanism happening in bacteria, utilizing structural instability for functionality. A compact head is required due to the fact that rigid electronics inside the robot head and the radius of the head is proportionally to the drag. On the way to achieving this objective, we overcame several common obstacles in the development of underwater robots, including making the robot neutrally buoyant, aligning the center of mass and center of buoyancy for horizontal swimming, and customizing soft PCB board to make the robot head as compact as possible. To make the platform accessible to flagellar locomotion researchers, we present a method to begin from tethered setup, a gimbal-based configuration that is much simpler for experiments while keeping the same number of degrees of freedom as an untethered robot. In parallel, we develop a new simulation framework, DDG to discrete and model the robot structure, Regularized Stokeslet Segments (RSS) to consider long distance hydrodynamics interaction among nodes on the flagellum, and Stokes' law to model forces and torques on a rigid head. This framework is validated against experiments step by step for the first time. The two strengths of RSS, elimination of discontinuity in forces, and its compatibility with large edgelength, improve the computational efficiency of the simulation to another level. Additionally, RSS is the state-of-the-art hydrodynamics model. We build "mass-transformer" to continue pushing the robot's autonomy to its limits, a device that can disperse the mass within the robot based on the orientation measurement embodied by an IMU. The control loop is as brief as feasible due to the seamless embodied perception and the basic yet effective control signals. As a result, the robot can now follow any 3D trajectory, buckling its flagellum for a minor in-plane turn and providing "mass-transformer" control when the robot requires a severe turn, especially invert. Our study sheds light on the mechanics of the bio-locomotion of microorganisms, provides researchers with easy testbeds, motivates the design of novel biomimetic soft robots, and even gives a template for researchers to scale down our concept to microrobots for targeted drug delivery.

Finally, in Ch. 6, we summarize all the findings presented in this dissertation thesis.

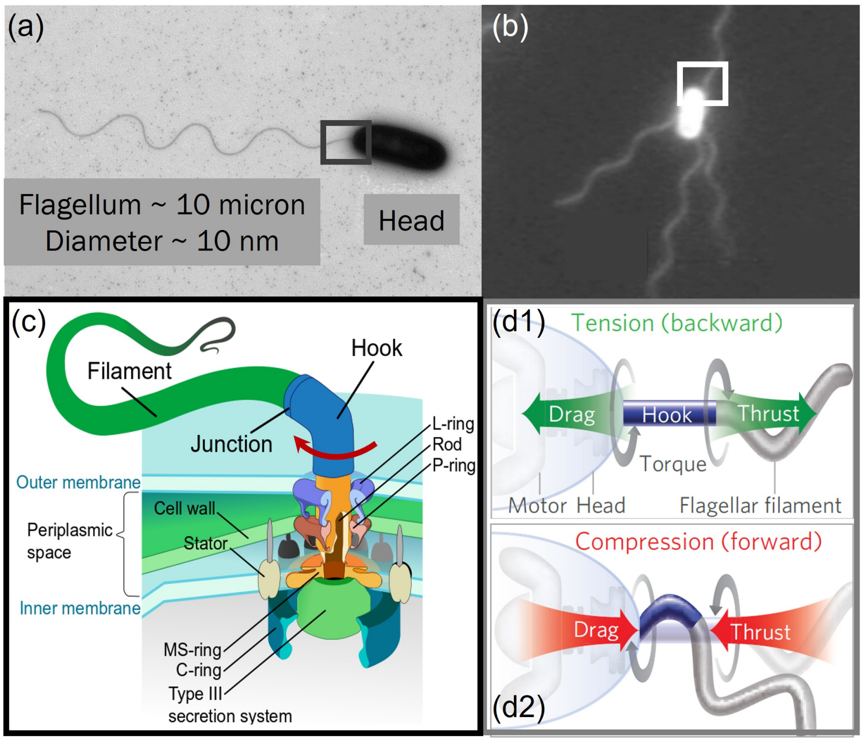

Figure 1.2: Diagram showing the overall structure of flagellated bacteria and bacterial“buckling-to-turn" mechanism. (a) A uni-flagellated bacterium, Vibrio cholerae. Courtesy of Kwangmin Son and Roman Stocker. (b) A multi-flagellated bacterium, Escherichia coli. Immobilized cells illuminated by a mercury arc. Adapted from Ref. [40]. (c) The structure of flagellated bacteria. Schematics (not to scale) of the flagellar filament, hook and rotary motor during backward swimming (d1), when the hook is in tension, and during forward swimming (d2), when the hook is in compression. The structures noted inside the rectangles in (a) and (b) are hooks, which are actuated by the motor embedded in the body and connect the body and slender filaments (flagella) as seen in (c).

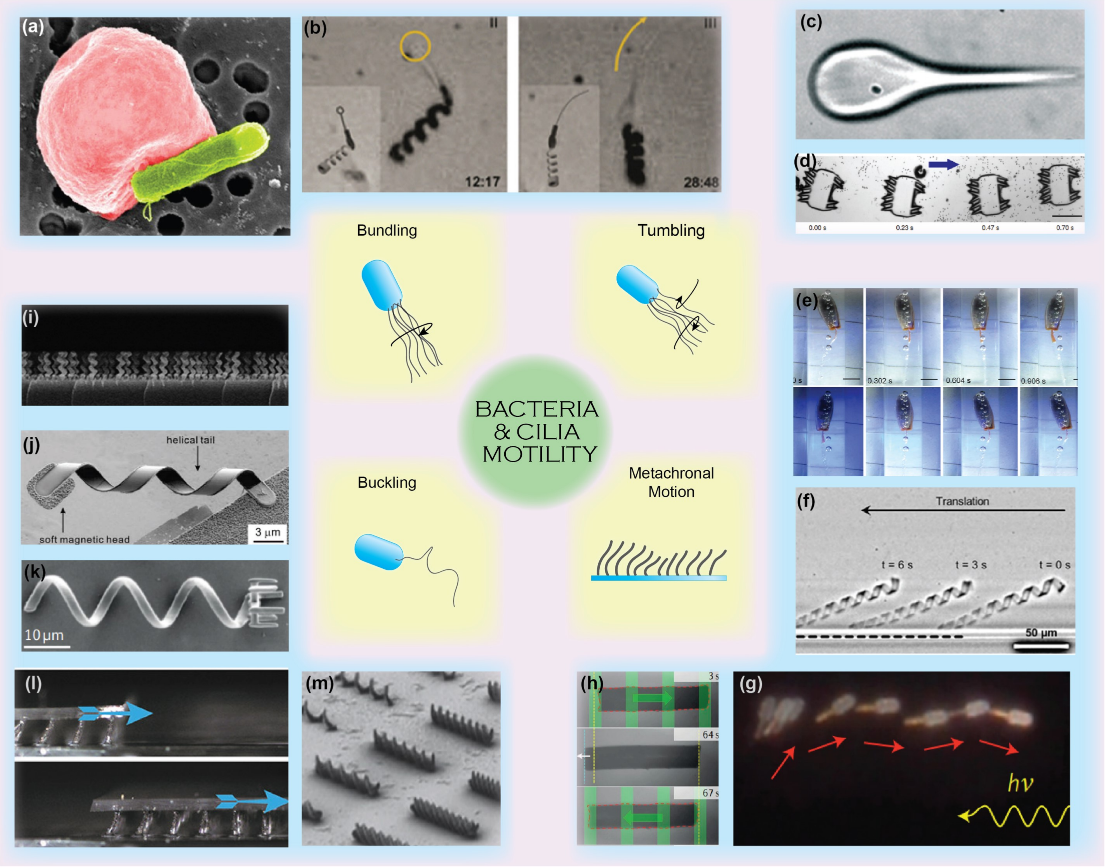

Figure 1.3: Illustration of representative flagella, cilia motility and collage of existing flagella, cilia-inspired robots. Bundling, tumbling and buckling phenomena are also often found in uni-polar arrangements of flagella as depicted (lophotrichous, monotrichous) yet also prevalent with microorganisms with multipolar arrangements of flagella (amphitrichous, perithichous). (a)-(b) Bio-hybrid microrobots. (a) RBC microswimmer with an attached bacterium [41] (b) Magnetic helices coupled with live sperm cells [42] (c)-(d) Acoustically actuated microrobot. (c) Design of acoustically actuated microswimmer [43]. (d) Controlled translation motion of acoustically actuated microrobot [44]. (e)-(f) Optically actuated microrobots. (e) Swing of the robot flagellum, actuated through light-driven liquid-crystal film [45]. (f) The directional motion of a helical microstructure [46]. (g) Positive phototaxis of Janus nano trees [47]. (h) Back and forth swimming of a cylindrical microrobot [48]. (i)-(m) Examples of magnetically actuated microrobots. (i) Side view SEM image of magnetic nano propellers. [49]. (j) SEM image of Artificial Bacterial Flagella, with a diameter of $2.8\mu m$ [50]. (k) Helical micromachine with a micro holder [51]. (l) Walking capability of ciliated metachronal robot [52]. (m) Microscopic images of fabricated helical, single twist-type, double twist-type actuators [53] .

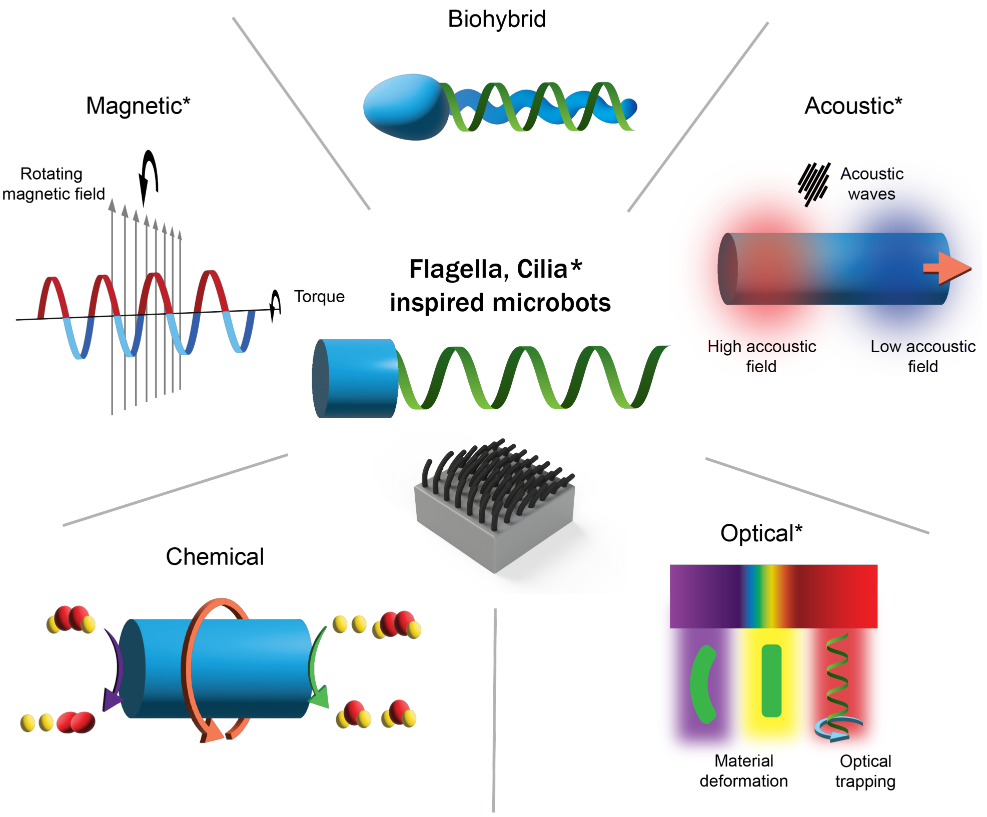

Figure 1.4: Schematic of different actuation methods for flagella-inspired microrobots. The asterisk (*) indicates actuation methods that are applicable to cilia (not the focus of our paper).

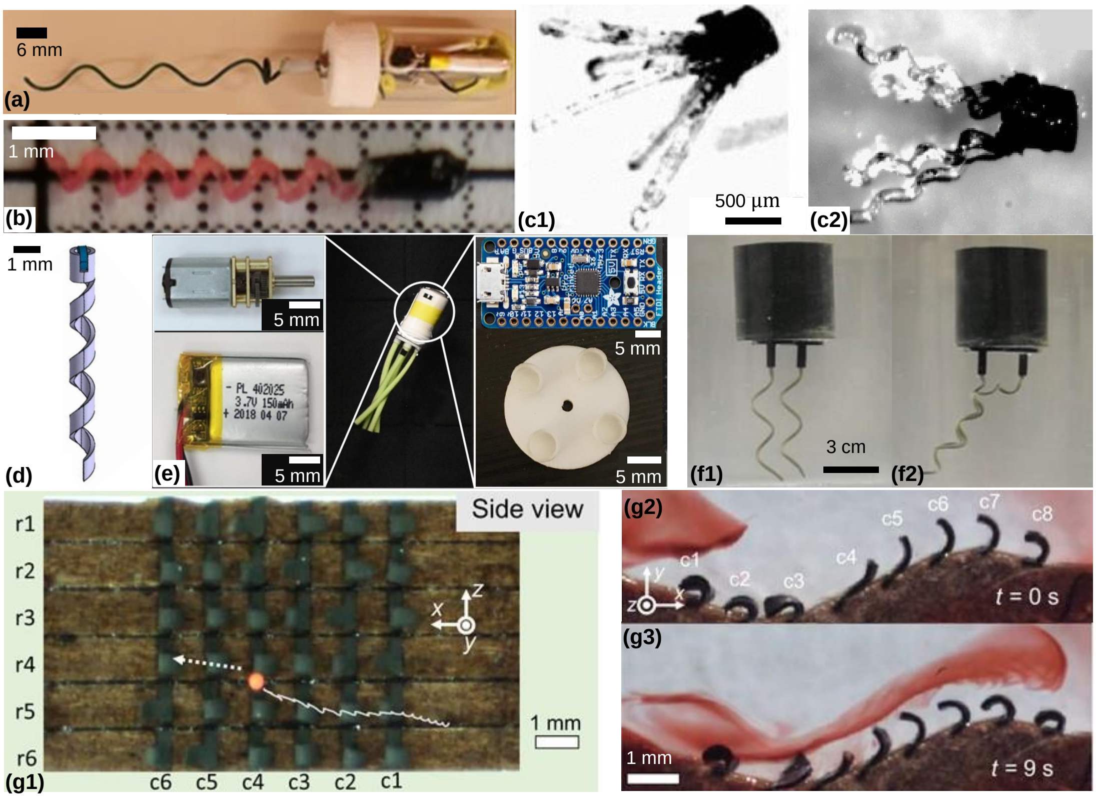

Figure 1.5: Schematic showing released flagella-inspired meso robots and macro robots that seek to understand the fundamental fluid-structure interaction physics and physiology of microorganisms with flagella or cilia. (a)A robot with a rigid metal helical tail with three waves with a 3 mm amplitude [62]; (b)3D printed swimmers [63]; (c1)-(c3)Optical microscope images of fabricated miniature swimming robots with different designs. Robots with a cylindrical body of (c1)500 μm (D)×600 μm (H) and flagella of 120 μm (W)×100 μm (B)×1.5 mm (L) made of polymer ST-1087 with a Young's modulous $E = 9.8$MPa (BJB Enterprises), (c2)500 μm (D)×600 μm (H), and sinusoidal flagella made of ST-1087 ($E = 9.8$ MPa) [64];(d)A scaled-up helical microswimmer with 14 mm in length and 1 mm in diameter [65]; (e)Compositive top view of a robot with four straight, flexible Vinyl Polysiloxane (VPS) flagella in a viscous fluid - glycerin. Inside the robot head, there is a DC geared motor, two batteries (from up to down on the left side of the robot), a microcontroller, and a 3D printed circular disc connecting the flagella to the rotating motor shaft that protruded from the robot head [66]. Inside the head of the robot moving in granular media, two batteries and a motor (same as in (e)) are contained. Chrome steel bearing balls are added to increase the moment of inertia of the head. Nuts are attached to the outer surface of the robot head to increase the friction [67]; The video snapshot of a macro robot propelled by rotating two elastic VPS flagella with the same handedness in glycerin when they (f1) rotate at the same rotational speed in the same direction and do not bundle and (f2) rotate in the same direction but with different rotational speeds and bundle as illustrated in Fig. 1.3 [68].

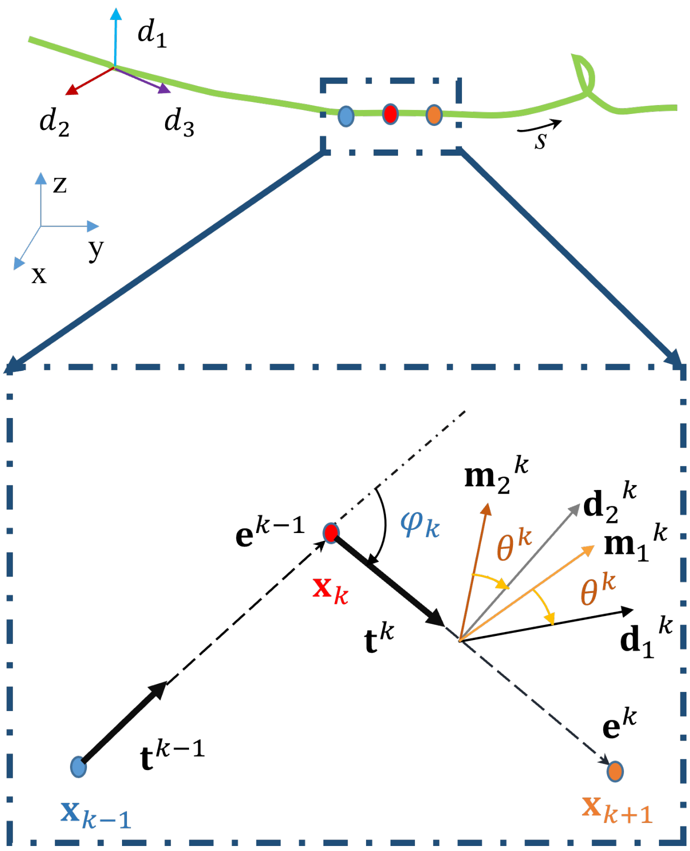

Figure 1.6: Schematic diagram of the centerline of one elastic rod (at the top) that is discretized into a collection of elastic rods with Discrete Elastic Rods method (at the bottom). The rod is discretized into N vertices (with coordinates $\mathbf{x}_0$ to $\mathbf{x}_{N-1}$) and in-between $N-1$ edges $\mathbf{e}^0$ to $\mathbf{e}^{N-2}$ (dashed arrows). At edge k, a reference frame $\left\{\mathbf{d}_1^k, \mathbf{d}_2^k, \mathbf{t}^k\right\}$ and material frame $\left\{\mathbf{m}_1^k, \mathbf{m}_2^k, \mathbf{t}^k\right\}$ are assigned.

CHAPTER 2

Modeling the locomotion of articulated soft robots in granular medium

We introduce a numerical tool for modeling articulated soft robots that couples discrete differential geometry-based simulation of elastic rods, our model for the articulated structure, and other external forces. Parallel to simulations, we build an untethered robot testbed, in the granular medium, comprised of multiple flexible flagella that are rotated about an axis by a motor. Drag from the granules causes the flagella to deform and the deformed shape generates a net forward propulsion. External drag depends on the flagellar shape, while the change in flagellar shape is the result of the competition between the external loading and elastic forces. We find reasonable quantitative agreement between experiments and simulations. Owing to a rod-based kinematic representation of the robot, the simulation can run faster than real-time in some cases, and, therefore, we can use it as a design tool for this class of soft robots. We find that there is an optimal rotational speed at which maximum efficiency is achieved. Moreover, both experiments and simulations show that increasing the number of flagella from two to three decreases the speed of the robot. This indicates that our simulator is potentially applicable for unknown physics exploration. We also gain insight into the mechanics of granular medium - while resistive force theory can successfully describe the propulsion at low number of flagella, it fails when more flagella are added to the robot.

We describe the motivation behind our research in § 2.1 followed by a literature review in § 2.2. Next, we describe the experimental design in § 2.3 including the robot design, the procedure of choosing the granular medium, and the process of data analysis. Then, we present the DDG-based numerical framework in § 2.4 depending on the geometry of our soft flagellated robot, after which we point out the novelty of our framework. Results and discussion are then given in § 2.5. In the end, summary of our work and the outlook is expressed in § 2.6. Most of the content in this chapter has appeared in Ref. [67].

2.1 Motivation

Soft robots and continuum robots inspired by nature that mimic echinoderms, bacteria, and fish, are primarily composed of intrinsically soft matter and fluids, enabling them to deform elastically into reversible shapes [37, 84, 85]. Their modeling and control are particularly challenging due to the geometric nonlinearity induced by the structural flexibility and the nontrivial coupling among elasticity, contact, and other external forces such as hydrodynamic and magnetic forces. Depending on the mechanics of the medium, locomotion can face unique physical constraints, e.g. at fluid flow with low Reynolds number (viscous forces dominate inertia), scallop theorem states that a swimmer with time reversible motion cannot achieve propulsion [21]. Over the past two decades, numerous investigations have been conducted on natural and artificial locomotions in the marine environment [86, 87]. As a result, the interplay between the environment and aquatic and aerial locomotion (swimming and flying) is well studied. To comprehend the hydrodynamics, Navier-Stokes equations with boundary conditions must be solved.

Compared with well studied underwater locomotion [86, 87], the mechanisms behind underground locomotion are far less understood. Slender flexible animals have evolved to apply various locomotion modes depending on their physiology and environmental factors [88, 89, 90]. Unlike the Navier-Stokes equations for fluids, no validated theories for locomotions on or inside terrestrial surfaces exist until recently, when the granular flow is shown to be functionally equivalent to low Reynolds fluid [91]. Flagellar propulsion, widely studied since 1955 [92] for application in low Reynolds fluids, is effective in granular media (GM) as well [93]. This builds a remarkable connection between the microscopic world of bacteria [94] and meter-sized snakes in sand. Real-time simulators for soft robots interacting with granules are also nonexistent. A comprehensive understanding of locomotion of soft bodies in GM can lead to novel design of robots for application in hazardous terrain, e.g. search and reconnaissance through debris and underground environmental monitoring. Moreover, the design and control of soft robots usually require painstaking trials due to the limitations of current simulators. Previous work on soft robot modeling has focused on the Finite Element Method [95], voxel-based discretization [96], and modeling of slender soft appendages with Cosserat rod theory [87] (including the piecewise constant strain method [97], piecewise variable strain method, differential kinematics [98], the constant curvature model, and piecewise constant curvature model [99]). A real-time simulator that can conveniently incorporate geometry and external forces will accelerate robot design and explore unknown physics in a complicated environment. Moreover, a simple-to-deploy testbed can greatly benefit theoretical and computational verification.

In this chapter, we draw inspiration from the bacterial locomotion and introduce a palm-sized untethered robot comprised of $n \geq 2$ naturally straight elastic rods and a rigid head with an embedded motor and battery. As shown in Fig. 2.1, the rotation of these *tails* generates drag from the GM, deforming the soft material. The tails provide a net propulsive force as a result of their nonlinear deformation. This net propulsion is only feasible in flexible structures; propulsion is zero in the case of *rigid* straight tails. We introduce a numerical method for simulating the dynamics of a collection of Kirchhoff elastic rods [79] subjected to viscous drag described by Resistive Force Theory (RFT) [92]. This computational tool is used to simulate the multi-limbed robot and quantitatively compared against experiments. We conduct parametric studies on the speed of the robot as a function of the number of tails and rotational speed, and determine the optimal rotational speed for maximum efficiency. We test the applicability of RFT to GM and indicate regimes in which it can fail.

2.2 Literature review

Simulating the dynamics of soft robots is complex and slow because of the numerous degrees of freedom and nonlinear material properties. Modeling soft robot locomotion can be divided into two components: (1) external loading on the flexible structure from the surrounding medium and (2) articulated slender bodies composed of multiple thin elastic rods.

Model of external loading from GM: GM, such as sand, soil, muddy sediments, and other mechanically unstable terrestrial substrates, display solid-like behavior in bulk and fluid-like behavior when disturbed. A major challenge of modeling the nonlinear dynamics of soft robots in GM is modeling the external forces on thin filaments. Modeling the motion of soft robots in purely fluidic medium is possible by solving Navier-Stokes hydrodynamics in the presence of moving boundary conditions. However, the computational cost is prohibitive for application in design and control of soft robots. For rods – mechanical structures with one dimension much larger than the other two – moving in low Reynolds flow, RFT is widely used to connect the hydrodynamic force from viscous environment and the velocity along the rod's centerline [92, 100]. Despite differences in the physical mechanisms involved, a solid friction analog to RFT in viscous fluid has been successfully applied in the context of GM to describe the undulatory motion of sand lizards and snakes [101, 102]. Several studies have shown that the frictional forces perpendicular to the body per unit length are

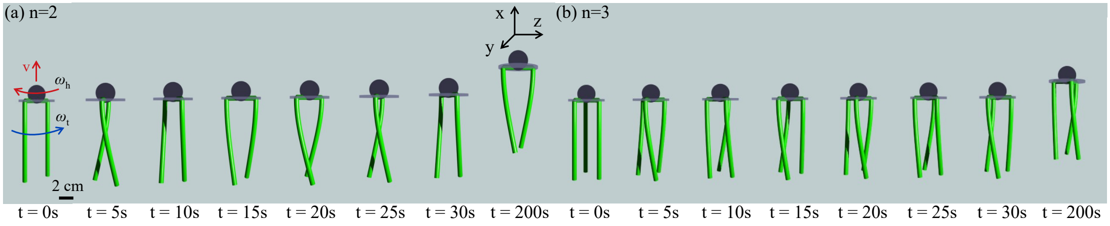

Figure 2.1: Snapshots from simulation. The shape of a robot with (a) $n = 2$ tails (Rotational speed of the motor $\omega_T = 100.00$ rpm, head rotational speed $\omega_h = 95.47$ rpm, tail rotational speed $\omega_t = 4.53$ rpm) and (b) $n = 3$ tails ($\omega_T = 100.00$ rpm, $\omega_h = 97.49$ rpm, $\omega_t = 2.51$ rpm) between $t = 0$ and $t = 200$ seconds. The speed of the robot is $v = 0.22$ mm/s (and $v = 0.13$ mm/s) for $n = 2$ (and $n = 3$).

greater than those along the body [6, 93].

Mechanics of articulated elastic rod structures: As with bacteria, the head and tails of our untethered articulated robot rotate in opposite directions [94]. The external force induced by GM can result in geometrically nonlinear deformation of tails, as displayed in Fig. 2.1. This coupling between the structural deformation and the forces from the GM in the context of an articulated soft robot is yet to be addressed in the literature. Notable prior works investigated the force on thin *rigid* rods in viscous fluid [100] or GM [93, 86]. Here, we use Discrete Elastic Rods (DER) [73, 78, 83] to capture the nonlinear deformation of thin elastic rods in the presence of external forces. The accuracy of DER has been established several times through prior works [103, 72, 104]. Previous studies combined DER with hydrodynamic models in viscous fluids to investigate the deformation and instability of a *single* helical elastic rod [72, 74]. All of these studies considered only a single elastic rod that is deforming because of hydrodynamic forces. Recently, we developed a model of multi-flagellated robots operating near the air-fluid interface [66].

A wide variety of soft robots can be modeled as a network of elastic rods, optionally connected to rigid bodies. Structures comprised of multiple elastic rods, e.g. elastic gridshell (also known as Cosserat net) [104] and flexible rod mesh [105], have also been modeled with DER. The multi-rod gridshell simulator [104] used stiff springs at the joints between two rods to impose constraints and computed the spring forces explicitly. This requires a smaller time step compared with an implicit approach and ignores the coupling of twisting and bending modes [105] between two rods at the joints. In this study, we present an algorithm that treats all the elastic and external forces implicitly in a network of rods and accounts for the presence of a rigid head. We demonstrate that a seemingly complex robot can be kinematically represented by a network of rods; this rod-based presentation can be used to leverage the computational efficiency of cutting edge tools like DER.

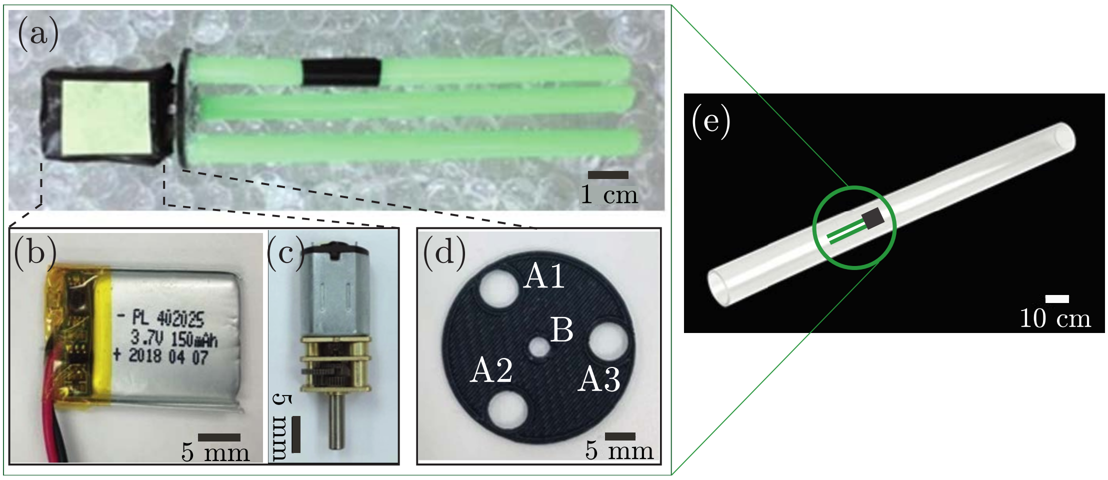

Figure 2.2: The compositive view of the experimental setup. (a) The robot with $n = 3$ tails. The head is comprised of (b) a battery and (c) a motor. (d) A circular disc with a circular array of holes (holes A1-A3) for gluing the tails with Gorilla glue super glue, gel, and a centered hole (hole B) for inserting the motor shaft. (e) The robot is placed inside a cylindrical tube full of granular medium (transparent water beads in this work).

2.3 Experimental design

The primary purpose of our experiments is to investigate the motion of a robot propelled by multiple rotating elastic rods. We place emphasis on the conditions and restrictions pertinent to RFT's application to model the drag from granular media. Keeping this goal in mind, we sequentially describe the design of the robot, fabrication, experimental setup, and data analysis in the following. Effective soft flagella-propelled locomotion in GM needs a complex interplay between the robot and GM, requiring back-and-forth iterations of robot design and GM selection. This section will detail the final robot design, GM, and the induced locomotion experiments.

2.3.1 Robot design

One of the prominent obstacles for light-weight small soft-robot design is the compromise among its geometrical size, weight, and actuation force. A characteristic process of robot design usually includes geometry design, actuator design, and microcontroller design [106, 107] which are first, second, last step, respectively. However, our robot started with function design. It originally consisted of a PCB board on which one microcontroller, two regulators, one Bluetooth wireless interface, one IMU and one servo motor, but then the whole robot body was too heavy for the soft tails to actuate because of the low actuation efficiency of soft tails. Therefore, some simplifications were made by taking the geometrical size and weight into consideration and the robot we are using now is shown in Fig. 2.2.

Fig. 2.2(a) shows a photograph of the soft robot which is a small, lightweight (14 cm, 35 g) structure actuated by n number of soft elastic tails that are made of Vinyl Polysiloxane (ZHERMACK Elite Double 32). It includes (1) a head with two 3.7V 200mAh rechargeable 502025 LiPo batteries (Fig. 2.2(b)) and one DC geared motor (uxcell) with 3V nominal voltage (Fig. 2.2(c)), 0.35W nominal power and 0.55A stall current, (2) multiple elastic tails, and (3) one 3D-printed plate ((Fig. 2.2(d))) to hold those tails. The elastic tails are fabricated straight using the molding and casting technique in [108]. The PVC tube mold is affixed to a straight steel bar to hold the shape completely straight.

The changeable inner and outer diameters of PVC tube molds make the scale-up or scale-down of our robot platform effortless. Inside the head, two batteries are connected in parallel, making the entire structure symmetric. We design the robot head as cuboidal to increase its ability to fluidize the GM in front of it. It is empirically verified that a robot with a spherical head moves much slower than one with a cuboidal head, as slippage occurs more frequently. The tails are inserted and glued (using Gorilla super glue, gel) into a circular array of holes on a 3D-printed plate and are driven by a motor via the shaft protruding from the robot head. We vary the number of tails to explore its effect on the translational speed, v, of our robot. The control parameter is the rotational speed of the tails relative to the head, ω_T. To modify it, we build robots with different motors but identical other components. The motor's rotational speed decreases as the voltage supplied drops. To ensure that the motor rotates at a constant speed, we fully charge the batteries before and after each 10-minute experiment. During data analysis, we count the number of rotations with time and verify that the methodology outlined above ensures a constant rotational speed throughout each experimental trial. Moreover, the size and weight of all motors are almost the same, $13-15$g and $(1.5-1.7)\times 1.2\times 1.0$ cm even though they provide different rotational speeds. Since the size of batteries are $27\times 20\times 5.1$ mm, decided by which the outer dimension is instead of the motor size. When necessary, we wrap electrical tape around the motor to account for the minor differences in size and weight among different motors.

2.3.2 Granular medium & locomotion experiments

We choose water crystal beads as the GM to test the locomotion due to their transparency. The robot can be seen from outside the medium and its movement is recorded using a conventional digital camera (Nikon D3400) with a frame rate of 29.98fps. The diameter of the beads in dry state is 2.5 mm, which increases to $d_b = 9.4 \pm 0.4$ mm after fully absorbing water. The size of beads is determined by the amount of time they are placed inside water and reversible after dehydration. Due to this property, the water crystal beads can also be used to investigate the performance of the robot, efficiency to be mentioned in Section 2.5.5 for example, related to the granular configuration, such as size, density and homogeneity. When performing experiments, we use the beads fully absorbing water to keep their size consistent. Before experiments are carried out, we dry their surfaces to decrease the possibility of slippage between the GM and the robot. The volume fraction [6] – the ratio between the solid volume and the occupied volume – is $\approx (1/6\,\pi d_b^3)/d_b^3 = 0.52$. The volume fraction is stated to control the response of GM to intrusion [6], and we will discuss how it might be related to the "stick slip" in Section 2.5. As illustrated in Fig. 2.2(a), the diameter of the beads is on the same order of magnitude as the diameter of the tails. RFT is intended for grains considerably smaller than the size of the robot; our choice of rather large grains is to test the limits of RFT.

Before deciding to use water beads, we tried packed foam beads (radius is 4.58 mm, density is $7.27\ \mathrm{kg/m^3}$), chia seeds, oatmeal, and white beads (radius is 1 mm, density is $15.35\ \mathrm{kg/m^3}$). Packed foam beads are too airy so they are not able to provide enough propulsion for our robot while chia seeds are too slippery, oatmeal is too dense so the friction it generates is larger than the propulsive force it provides. White beads are neither too dense nor too airy so our robot can move smoothly inside those beads, however, it is hard for us to capture where the robot is. To solve this problem, we put an LED on the robot head but the light is blocked when the thickness of granular medium is too large. Also, beads around are illuminated by the LED so it is still difficult to find the exact location of our robot. Infrared camera is tried by us to help locate the robot but it does not work for lights from LED. Therefore, water beads finally come to our mind. An interesting question to ask regarding the choice of granular medium talked above is: It seems that what we need is a kind of clear beads to locate the robot, why water beads are not the first choice for us? If you look into the definition of RFT, you will be aware that RFT was originally defined in low Reynolds number of fluid. Consequently, the radius of granular medium should be smaller than the radius of robot tails. However, from our description above, it is clear that the radius of water beads is on the same order and even larger than robot tails. As a matter of fact, this can help us extend the applications of RFT if we can verify that RFT can be applied to explaining our experiments. More details about experiment results will be discussed in Section 2.5.

In our case, we initially put our soft robot in a 2 m $\times 0.5$ m $\times 0.5$ m tank filled with the 9.4 mm diameter water beads but it is shown that the robot moved forward in a straight line but slowly in the tank. Therefore, We run our soft robot in a cylindrical transparent tube with the length of 121.92 cm and radius of 5.3 cm, as shown in Fig. 2.2(e), filled with the same size water beads. However, we fill the water beads up to 76cm rather than completely filling the tube in order to allow the robot to have easier mobility. While attempting to allow for freer mobility for the robot, we encounter the issue that the robot is not able to be fully covered by the water beads. To counter this issue, we place one end of the tube at a sloped angle (1.32°). By solving this problem, we find ourselves with another dilemma. The distribution of granular medium is not even under gravity. To fix this, we try our best to shift the water beads evenly before every experiment trial. The robot is initially positioned at one end near the center of the cross-section of the tube, meant to cancel the wall effect. Since the robot is placed at the center of the tube, surrounded by compact granules against the tube wall, the drag-induced lift mentioned in [109] is suppressed. Hence, the rotation of tails propels the robot forward in a roughly straight line through the GM. A bright yellow marker is attached to the black colored head and a black marker is attached to one of the green colored elastic tails in order to count the rotational speed of the robot head (ω_h) and tail (ω_t).

2.3.3 Data analyses

To capture the position of the robot, we used *ffmpeg* to transform videos recorded to pictures every 10 or 30 seconds. Then, MATLAB was utilized to read those pictures in which the head positions were tracked manually by us and draw the track of points as the track went along. Also, we counted the rotation speeds of robot head and tail per minute from the video and calculated the average of head and tail rotation speed respectively.

2.4 Numerical model description

Following experiments, we develop a simulator in Sections 2.4(A-E) to simulate the movement of the robot, with DDG simulating the structure by incorporating RFT for the drag force on the flaglum and Stokes's law for the force and torque on the robot head. The material and geometric parameters of the robot are given in Section 2.4(F).

2.4.1 Kinematics

Referring to Fig. 2.3(a), the first step in modelling the robot is to represent it as a "stick figure". A number of nodes (circles in Fig. 2.3) are located along the stick figure. Fig. 2.3(b) shows the nodes at the "joint" between the head and tails ($n = 2$ in the figure). Node $\mathbf{x}_a$ is unique since it is connected to $n + 1$

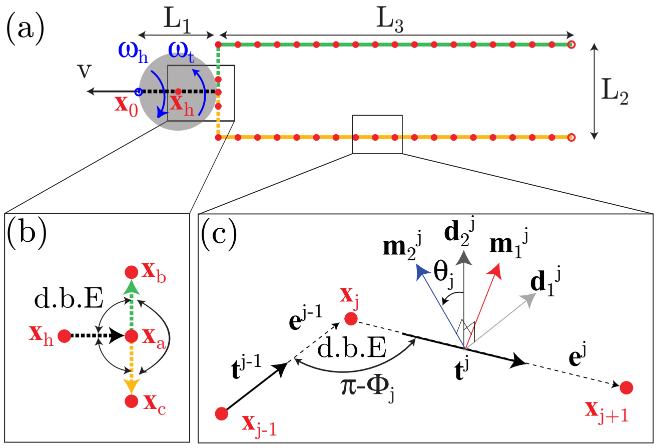

Figure 2.3: Schematic of the discrete representation of a robot with $n = 2$ tails. (a) Geometric parameters of the robot in undeformed state. Here, $L_1 = 2a$ is the diameter of the robot head, L_2 is the diameter of the disc connecting the head and the tails, and L_3 is the length of each tail. Dashed lines represent rigid structure whereas solid lines correspond to flexible structure. Node $\mathbf{x}_h$ represents the location of the head. (b) A close-up of the “joint" node $\mathbf{x}_a$ that connects the head with tails. In this figure, d.b.E. indicates discrete bending and twisting energy between adjacent edges,i.e., $\mathbf{x}_h\mathbf{x}_a$, $\mathbf{x}_a\mathbf{x}_c$, and $\mathbf{x}_a\mathbf{x}_b$. This is the only node that is connected to more than two nodes. (c) A close-up of three nodes, $\mathbf{x}_{j-1}, \mathbf{x}_j$, and $\mathbf{x}_{j+1}$, and two edges, $\mathbf{e}^{j-1} = \mathbf{x}_j - \mathbf{x}_{j-1}$ and $\mathbf{e}^j = \mathbf{x}_{j+1} - \mathbf{x}_j$. The turning angle from edge $\mathbf{e}^{j-1}$ to $\mathbf{e}^j$ is ϕ_j. The reference frame on $\mathbf{e}^j$ is $\left\{\mathbf{d}_1^j, \mathbf{d}_2^j, \mathbf{t}^j\right\}$ and the material frame is $\left\{\mathbf{m}_1^j, \mathbf{m}_2^j, \mathbf{t}^j\right\}$. The twist angle on edge $\mathbf{e}^j$ is θ^j.

nodes. All other nodes are connected to two nodes or a single node in case of terminal nodes (open circles in Fig. 2.3(a)). As illustrated in Fig. 2.3(c), a node $\mathbf{x}_j$ is typically connected with two nodes $\mathbf{x}_{j-1}$ and $\mathbf{x}_{j+1}$. Details of concepts “edge" (e.g. $\mathbf{e}^j = \mathbf{x}_{j+1} - \mathbf{x}_j$), reference frame $\left\{\mathbf{d}_1^j, \mathbf{d}_2^j, \mathbf{t}^j\right\}$,and material frame $\left\{\mathbf{m}_1^j, \mathbf{m}_2^j, \mathbf{t}^j\right\}$ are given in our previous work [66]. Reference frame is initialized at time $t = 0$ and then updated at each time step of the simulation using time-parallel transport. Detailed DER can be found [73, 78, 83]. A scalar quantity, θ^j, is necessary per edge to obtain the material frame from the reference frame as outlined in Fig. 2.3(c). Angle θ^j is the “twist angle". We follow the convention of using subscripts to denote node-based quantities and superscripts for edge-based quantities.

The locations of the nodes, $\mathbf{x}_j$ ($0 \leq j < N$ where N is the number of nodes), and the twist angles, θ^j ($0 \leq j < N_e$ where N_e is the number of edges), completely describe the configuration of the robot. For the robot studied in this chapter, $N_e = N - 1$ (see Fig. 2.3(a)). The DOF vector for the robot is $\mathbf{q} = \left[\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N-1}, \theta^0, \theta^1, \ldots, \theta^{N_e-1}\right]^T$, where the superscript ^{T} denotes transpose. If a robot has N nodes, the size of $\mathbf{q}$ is ndof $= 3N + N_e$. Since the robot deforms with time, the DOF vector is a function of time, i.e. $\mathbf{q} \equiv \mathbf{q}(t)$. Knowing the configuration of the robot at $t = 0$ (i.e. $\mathbf{q}(0)$ is known), the task at hand is to compute $\mathbf{q}(t)$.

2.4.2 Macroscopic strains & Elastic energies

At time $t = 0$, the robot is undeformed with zero strains and the DOF vector is $\mathbf{q}(0) \equiv \bar{\mathbf{q}}$; hereafter, $\bar{(\ )}$ represents evaluation of a quantity in its undeformed configuration. While the undeformed and initial configurations of the system studied here are identical, this is not a required assumption for the simulation scheme. Axial stretch, curvature, and twist are the macroscopic strains along the structure.

The axial stretch, ϵ^j, in the j-th edge is

$$\epsilon^j = \frac{\|\mathbf{e}^j\|}{\|\bar{\mathbf{e}}^j\|} - 1. \tag{2.1}$$

Curvature binormal is a vector representing the turn whose proof is given in Fig. 3.2 and Eqs. 3.6 - 3.11 in [83]:

$$(\kappa \mathbf{b})_j = \frac{2\mathbf{e}^{j-1} \times \mathbf{e}^j}{\|\mathbf{e}^{j-1}\|\|\mathbf{e}^j\| + \mathbf{e}^{j-1} \cdot \mathbf{e}^j}. \tag{2.2}$$

It turns out that $\|(\kappa \mathbf{b})_j\| = 2 \tan\left(\frac{\phi_j}{2}\right)$, where ϕ_j (in Fig. 2.3(c)) is the turning angle. No curvature is associated with the terminal nodes. The curvature of the osculating circle passing through $\mathbf{x}_{j-1}, \mathbf{x}_j$, and $\mathbf{x}_{j+1}$ is $\|(\kappa \mathbf{b})_j\|/\Delta l$ where

$\Delta l = \|\mathbf{e}^j\| = \|\mathbf{e}^{j-1}\|$. The scalar curvatures along the first and second material directors are

$$\kappa_j^{(1)} = \frac{1}{2}(\mathbf{m}_2^{j-1} + \mathbf{m}_2^j) \cdot (\kappa \mathbf{b})_j, \tag{2.3a}$$
$$\kappa_j^{(2)} = \frac{1}{2}(\mathbf{m}_1^{j-1} + \mathbf{m}_1^j) \cdot (\kappa \mathbf{b})_j. \tag{2.3b}$$

Associated with every curvature is a twist that represents the rotation of the material frame from one edge to the next. Eqs. 2.3a and 2.3b and following equations were derived by [78] and a pedagogical exposition is available in [83]. In Fig. 2.3(c), the twist at the j-th node is

$$\tau_j = \theta^j - \theta^{j-1} + \Delta m_{j,\text{ref}}, \tag{2.4}$$

where $\Delta m_{j,\text{ref}}$ is the reference twist, i.e. the twist of the reference frame as it moves from the $(j-1)$-th edge to the j-th edge [73].

The procedure to calculating this reference twist is discussed next. The first director of the reference frame, $\mathbf{d}_1^{j-1}$, is *parallel transported* from the $(j-1)$-th edge to the j-th edge to get $\mathbf{d}_{\text{tmp}}$. Parallel transport is the process of moving the reference director from one edge to the next without twist; it involves the following steps.

$$\begin{aligned}
\mathbf{b} &= \mathbf{t}^{j-1} \times \mathbf{t}^j, \\
\hat{\mathbf{b}} &= \frac{\mathbf{b}}{|\mathbf{b}|}, \\
\mathbf{n}_1 &= \mathbf{t}^{j-1} \times \hat{\mathbf{b}}, \\
\mathbf{n}_2 &= \mathbf{t}^j \times \hat{\mathbf{b}}, \\
\mathbf{d}_{\text{tmp}} &= (\mathbf{d}_1^{j-1} \cdot \mathbf{t}^{j-1})\mathbf{t}^j + (\mathbf{d}_1^{j-1} \cdot \mathbf{n}_1)\mathbf{n}_2 + (\mathbf{d}_1^{j-1} \cdot \hat{\mathbf{b}})\hat{\mathbf{b}},
\end{aligned}$$

where $\mathbf{t}^{j-1}$ and $\mathbf{t}^j$ are the tangents on the $(j-1)$-th and j-th edges, respectively.

The reference twist, $\Delta m_{j,\text{ref}}$, is the signed angle from $\mathbf{d}_{\text{tmp}}$ to $\mathbf{d}_1^j$ about $\mathbf{t}^j$.

The total elastic energy of the structure is the linear sum of stretching E_s, bending E_b, and twisting E_t energies such that

$$E_{\text{elastic}} = E_s + E_b + E_t, \tag{2.5}$$

where $E_s, E_b,$ and E_t are the quadratic functions of strains

$$E_s = \sum \frac{1}{2} EA \left(\epsilon^j\right)^2 \|\bar{\mathbf{e}}^j\|, \tag{2.6}$$

$$E_{\text{b}} = \sum \frac{1}{2} \frac{EI}{\Delta l_j} \left[\left(\kappa_j^{(1)} - \bar{\kappa}_j^{(1)}\right)^2 + \left(\kappa_j^{(2)} - \bar{\kappa}_j^{(2)}\right)^2 \right], \tag{2.7}$$

$$E_t = \sum \frac{1}{2} \frac{GJ}{\Delta l_j} \left(\tau_j - \bar{\tau}_j\right)^2. \tag{2.8}$$

Here, $\sum$ in Eq. 2.6 represents summation over all the edges and curvatures in Eqs. 2.7 and 2.8. E is the Young's modulus, $A = \pi r_0^2$ is the cross-sectional area, and r_0 is the cross-sectional radius. $EI = \frac{\pi}{4} E r_0^4$ is the bending stiffness. G is the shear modulus and $GJ = \frac{\pi}{2} G r_0^2$ is the twisting stiffness. To model rigid components of the robot, the bending stiffness is assumed to be large enough so that the curvatures at the rigid nodes remain almost constant throughout the simulation. For edges that are located on rigid parts (i.e. head and disc denoted by dashed lines in Fig. 2.3(a)), the stiffness parameters EA, EI and GJ are set sufficiently large to ensure negligible deformation. Equations of motion are statements of the balance of forces. The internal forces in the robotic structure arises from the elasticity of the material. In this paper, the rigid components (e.g. the head and disc indicated by dashed lines in Fig. 2.3(a)) are assumed to be elastic with high elastic stiffness so that their deformation is negligible compared with the deformation in the flexible tails. The next sections discuss the strains in the structure, the elastic energies associated with these strains, and the elastic and external forces in order. The Voronoi length $\Delta l_j = \frac{1}{2}\left(\|\bar{\mathbf{e}}^{j-1}\| + \|\bar{\mathbf{e}^{j}}\|\right)$ is associated with the j-th node. $\bar{\kappa}_j^{(1)}$ and $\bar{\kappa}_j^{(2)}$ are the material curvatures in undeformed configuration, and where $\bar{\tau}_j$ is the undeformed twist along the centerline, This stiffness is assumed to be sufficiently large for the rigid components.

The material of tails is nearly incompressible (i.e. Poisson's ratio $\nu = 0.5$) and therefore $G = E/3$. Each internal node of a single elastic rod is associated with a discrete bending and twisting energy. However, a "joint" node ($\mathbf{x}_a$ in Fig. 2.3(b)) has multiple associated discrete bending (d.B.E. in Fig. 2.3(b)) and twisting energies. This observation is important during the implementation of the simulation algorithm.

2.4.3 External forces using Resistive Force Theory

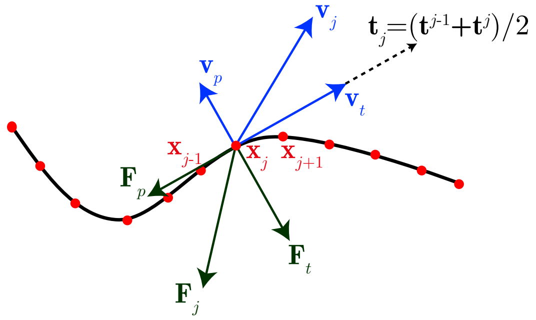

Figure 2.4: Schematic representation of RFT.

In Fig. 2.4, we schematically represent a slender rod in the discrete setting moving in the GM. The velocity, $\mathbf{v}_j \equiv \dot{\mathbf{x}}_j$ at point $\mathbf{x}_j$ can be decomposed into two parts: the parallel term $\mathbf{v}_t = (\mathbf{v}_j \cdot \mathbf{t}_j)\mathbf{t}_j$ and the perpendicular term $\mathbf{v}_p = \mathbf{v}_j - \mathbf{v}_t$, where the tangent at the j-th node $\mathbf{t}_j = \frac{1}{2}(\mathbf{t}^{j-1} + \mathbf{t}^j)$ is the average of the tangents along the two associated edges. The external force on the flagellar j-th node is $\mathbf{F}_j = \mathbf{F}_t + \mathbf{F}_p$, where the tangential and perpendicular forces that resist $\mathbf{v}_t$ and $\mathbf{v}_p$ are

$$\mathbf{F}_t = -\eta_t \mathbf{v}_t \Delta l_j, \tag{2.9a}$$

$$\mathbf{F}_p = -\eta_p \mathbf{v}_p \Delta l_j, \tag{2.9b}$$

the drag coefficients along the tangential and perpendicular directions [92] are

$$\eta_t = 2\pi\mu / \left[\log(\frac{2L}{r_0}) - \frac{1}{2}\right], \tag{2.10a}$$

$$\eta_p = 4\pi\mu / \left[\log(\frac{2L}{r_0}) + \frac{1}{2}\right], \tag{2.10b}$$

μ is the constant used to quantify the body-granule friction coefficient, and L is the tail length($L = L_3$ in Fig. 2.3(a)).

The head rotates and translates as the robot moves. The rotational speed of the head (ω_h in Fig. 2.3) can be extracted from the time derivative of the twist angle, θ^h, of the edge connecting $\mathbf{x}_0$ and $\mathbf{x}_h$, i.e. $\omega_h \equiv \dot{\theta}^h$. The velocity of the head is $\mathbf{v}_h \equiv \dot{\mathbf{x}}_h$. If the head is spherical with radius a, the viscous drag on it according to Stokes's law is $\mathbf{F}_h = -6\pi\mu a \mathbf{v}_h$ and an external torque on the edge is $T_h = -8\pi\mu a^3 \omega_h$. In our case, the robot head shape is not a sphere so we use numerical coefficients, C_1 and C_2, to account for the shape. As a result, the drag and torque are updated as follows

$$\mathbf{F}_h = -(6\pi C_1)\mu a \mathbf{v}_h, \tag{2.11}$$

$$T_h = -(8\pi C_2)\mu a^3 \omega_h. \tag{2.12}$$

Overall, the GM is characterized by parameters C_1, C_2, and μ. If any of the physical properties of the medium, e.g., grain size, changes, these parameters will need to be updated.

2.4.4 Simulation loop, equations of motion

In the simulation scheme, time is discretized into small time steps (Δt) and the DOF vector, $\mathbf{q}$, is updated at each time step. The equation of motion at the i-th DOF to march from $t = t_k$ to $t = t_{k+1} = t_k + \Delta t$ is

$$f_i \equiv \frac{m_i}{\Delta t}\left[\frac{q_i(t_{k+1}) - q_i(t_k)}{\Delta t} - \dot{q}_i(t_k)\right] + \frac{\partial E_{\text{elastic}}}{\partial q_i} - f_i^{\text{ext}} = 0, \tag{2.13}$$

where $i = 1, \ldots, \mathrm{ndof}$, the *old* DOF $q_i(t_k)$ and velocity $\dot{q}_i(t_k)$ are known, E_{elastic} is the elastic energy evaluated at $q_i(t_{k+1})$, f_i^{ext} is the external force (or moment for twist angles) on the i-th DOF, and m_i is the lumped mass at each DOF. Since the dynamics of the system is dominated by viscosity with negligible influence of inertia, the results presented in this paper do not vary with the mass parameters as long as low Reynolds number is maintained. Note that Eq. 2.13 is simply a statement of “mass times acceleration = elastic force + external force" at the i-th DOF. Eq. 2.13 represents a system of ndof equations that has to be solved to obtain the *new* DOF $q_i(t_{k+1})$. Once the new DOF is obtained, the new velocity is simply $\dot{q}_i(t_{k+1}) = \left(q_i(t_{k+1}) - q_i(t_k)\right)/\Delta t$.

Newton-Raphson method is used to solve the equations of motion. This involves solving the linear system $\mathbb{J}\Delta\mathbf{q} = \mathbf{f}$, where $\mathbf{f}$ is a vector of size ndof, the i-th component of this vector can be computed from Eq. 2.13, and $\mathbb{J}$ is a square Jacobian matrix for Eq. 2.13. The (i, j)-th component $(i, j = 1, \ldots, \mathrm{ndof})$ of the Jacobian is

$$\mathbb{J}_{ij} = \frac{\partial f_i}{\partial \xi_j} = \mathbb{J}_{ij}^{\text{inertia}} + \mathbb{J}_{ij}^{\text{elastic}} + \mathbb{J}_{ij}^{\text{ext}}, \tag{2.14}$$

where

$$\mathbb{J}_{ij}^{\text{inertia}} = \frac{m_i}{\Delta t^2}\delta_{ij}, \tag{2.15}$$

$$\mathbb{J}_{ij}^{\text{elastic}} = \frac{\partial^2 E_{\text{elastic}}}{\partial q_i \partial q_j}, \tag{2.16}$$

$$\mathbb{J}_{ij}^{\text{ext}} = -\frac{\partial f_i^{\text{ext}}}{\partial q_j}. \tag{2.17}$$

Here, δ_{ij} represents Kronecker delta. Evaluation of the gradient of the elastic energy ($\frac{\partial E_{\text{elastic}}}{\partial q_i}$) as well as its Hessian ($\frac{\partial^2 E_{\text{elastic}}}{\partial q_i \partial q_j}$) are well documented in [83, 78].

Algorithm 1 Discrete Simulation of Robots

```
Require: $\mathbf{q}(t_k), \dot{\mathbf{q}}(t_k)$    ▷ DOFs and velocities at $t = t_j$
Require: $\left(\mathbf{d}_1^j(t_k), \mathbf{d}_2^j(t_k), \mathbf{t}^j(t_k)\right)$, $0 \le j < N_e$    ▷ Reference frame at $t = t_k$
Ensure: $\mathbf{q}(t_{k+1}), \dot{\mathbf{q}}(t_{k+1})$    ▷ DOFs and velocities at $t = t_{k+1}$
Ensure: $\left(\mathbf{d}_1^j(t_{k+1}), \mathbf{d}_2^j(t_{k+1}), \mathbf{t}^j(t_{k+1})\right)$, $0 \le j < N_e$    ▷ Reference frame at $t = t_{k+1}$

function DISCRETE SIMULATION OF ROBOTS( $\mathbf{q}(t_k), \dot{\mathbf{q}}(t_k), \left(\mathbf{a}_1^j(t_k), \mathbf{a}_2^j(t_k), \mathbf{t}^j(t_k)\right)$ )
    $\bar{\tau}_h(t_k) \leftarrow \omega_T t_k$    ▷ Actuation using Eq. 3.44
    Guess: $\mathbf{q}^{(1)} \leftarrow \mathbf{q}(t_k)$
    $n \leftarrow 1$
    while error > tolerance do    ▷ Newton-Raphson iterations
        Compute reference frame $\left(\mathbf{d}_1^j, \mathbf{d}_2^j, \mathbf{t}^j\right)^{(n)}$    ▷ Parallel transport $\mathbf{d}_1^j(t_k)$
    and $\mathbf{d}_2^j(t_k)$ from ...
            ▷ ... $\mathbf{t}^j(t_k)$ to tangent on $j$-th edge in $\mathbf{q}^{(n)}$ to get $\left(\mathbf{d}_1^j\right)^{(n)}$ and $\left(\mathbf{d}_2^j\right)^{(n)}$
        Compute reference twist $\Delta m_{j,\text{ref}}^{(n)}$ at each internal node
        Compute material frame $\left(\mathbf{m}_1^j, \mathbf{m}_2^j, \mathbf{t}^j\right)^{(n)}$    ▷ Eq. 3.30
        Compute $\mathbf{f}$ and $\mathbb{J}$    ▷ Eqs. 2.13 and 2.14
        $\Delta\mathbf{q} \leftarrow \mathbb{J} \backslash \mathbf{f}$
        $\mathbf{q}^{(n+1)} \leftarrow \mathbf{q}^{(n)} - \Delta\mathbf{q}$    ▷ Update DOFs
        error ← sum ( abs ( f ) )
        $n \leftarrow n + 1$
    end while

    $\mathbf{q}(t_{k+1}) \leftarrow \mathbf{q}^{(n)}$
    $\dot{\mathbf{q}}(t_{k+1}) \leftarrow \frac{\mathbf{q}(t_{k+1}) - \mathbf{q}(t_k)}{\Delta t}$
    $\left(\mathbf{d}_1^j(t_{k+1}), \mathbf{d}_2^j(t_{k+1}), \mathbf{t}^j(t_{k+1})\right) \leftarrow \left(\mathbf{d}_1^j, \mathbf{d}_2^j, \mathbf{t}^j\right)^{(n)}$
    return $\mathbf{q}(t_{k+1}), \dot{\mathbf{q}}(t_{k+1}), \left(\mathbf{d}_1^j(t_{k+1}), \mathbf{d}_2^j(t_{k+1}), \mathbf{t}^j(t_{k+1})\right)$
end function
```

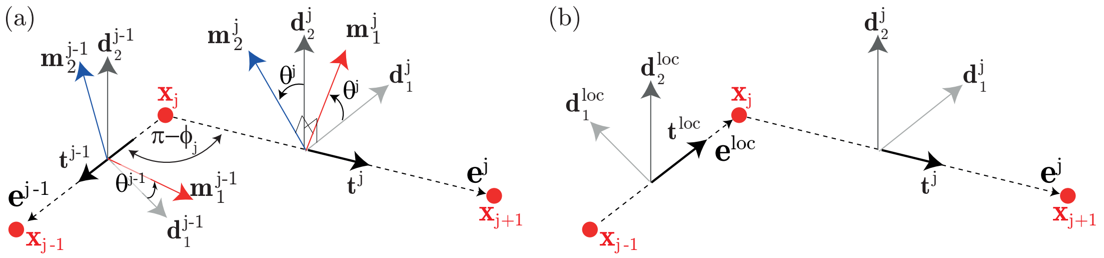

Figure 2.5: (a) Discrete bending and twisting energy is located at $\mathbf{x}_j$. Both the edges are pointing away from $\mathbf{x}_j$. (b) One of the edges ($\mathbf{e}^{j-1}$ in this case) is flipped to slightly modify the kinematic representation. This representation can be used to compute the gradient and Hessian of the bending and twisting energies following analytical expressions available in the literature [78].

2.4.5 Main contributions & observations in the algorithm

The overall computational framework is summarized in the Algorithm above. An important contribution of this study is the observation that the actuation (e.g. rotation of motor) can be readily accounted for in the above framework by updating the undeformed configurations with time. Typically, undeformed configuration of a structure is fixed and assumed to be invariant through the simulation. The strains in undeformed configuration (e.g. $\bar{\kappa}_j^{(1)}, \bar{\kappa}_j^{(2)}, \bar{\tau}_j$) are used in calculation of elastic energies, their gradient (i.e. elastic forces), and Hessian. However, in case of this robot, the rotation of the motor causes the undeformed twist at the head node ($\mathbf{x}_h$) to vary with time. If the rotational speed of the motor is ω_T, we assume that the undeformed twist at the head node is $\bar{\tau}_h(t_k) = \omega_T t_k$. As this is a torque-free system, the head and tails of the robot will get actuated at t_0 and start rotating. This results in rotations of the head (ω_h) and the tails (ω_t) along opposite directions such that $|\omega_T| = |\omega_h| + |\omega_t|$. The total rotational speed, ω_T, is a control parameter in this study.

The most computationally expensive part of the algorithm is solving the linear system. Observing and exploiting the sparsity of the Jacobian matrix, $\mathbb{J}$, is important to reduce computing time. Referring to Fig. 2.3(c), the entire structure is modelled as a series of stretching (e.g. one stretching spring is between $\mathbf{x}_j$ and $\mathbf{x}_{j+1}$) and bending-twisting springs (e.g. one bending-twisting spring is constructed by $\mathbf{x}_{j-1}, \mathbf{x}_j$, and $\mathbf{x}_{j+1}$). The stretching energy of each spring (Eq. 2.6) depends only on six DOFs (nodal coordinates of two nodes). For the stretching spring on edge $\mathbf{e}^j$, these DOFs are $\mathbf{x}_j$ and $\mathbf{x}_{j+1}$. The gradient vector $\left(\frac{\partial}{\partial \mathbf{q}}\left[\frac{1}{2}EA\left(\epsilon^j\right)^2\|\bar{\mathbf{e}}^j\|\right]\right)$ has only six non-zero terms and the Hessian matrix $\left(\frac{\partial^2}{\partial \mathbf{q}\partial \mathbf{q}}\left[\frac{1}{2}EA\left(\epsilon^j\right)^2\|\bar{\mathbf{e}}^j\|\right]\right)$ has only 6×6 non-zero terms. The bending and twisting energies of each spring (Eqs. 2.7 - 2.8) depend only on eleven DOFs, i.e. $\mathbf{x}_{j-1}, \theta^{j-1}, \mathbf{x}_j, \theta^j$, and $\mathbf{x}_{j+1}$ in case of the spring located at $\mathbf{x}_j$ in Fig. 2.3(c). The gradient vector and the Hessian matrix of these two energies therefore have only eleven and 11×11 non-zero terms. The full expressions for the gradient and Hessian terms are released [73, 83]; software implementation is also available [110][1]. The simulation requires the gradient of external forces (Eq. 2.17) expressed in Eqs. 2.9, 2.11, and 2.12. Their gradients with respect to the DOFs can be trivially obtained. Note that $\mathbb{J}_{ij}^{\text{ext}}$ is sparse. Since the expressions of all the Jacobian terms can be analytically evaluated and incorporated into the software, the simulation can use implicit method. In comparison with the explicit method, the implicit method typically can converge at larger Δt and requires less computation time. This is our second contribution.

Unlike the banded Jacobian in simulating a single elastic rod [78], the Jacobian here is only sparse but not banded due to the “joint" node. Another difference is the implementation of the gradient and Hessian of bending and twisting energies. As in Fig. 2.3, the gradient and Hessian in [78] assume that the tangent $\mathbf{t}^{j-1}$ points towards $\mathbf{x}_j$ and the second tangent $\mathbf{t}^j$ points forward from $\mathbf{x}_j$. The assumption does not always hold in this paper. For example, as represented in Fig. 2.3(b), we can have cases where both tangents (dashed ar-

[1] https://github.com/QuantuMope/imc-der

rows) point away from $\mathbf{x}_a$, the “joint" node. In this case, we can simply flip the first tangent ($\mathbf{t}^{\text{loc}} = -\mathbf{t}^{j-1}$) and use $\left\{\mathbf{d}_1^{\text{loc}} = -\mathbf{d}_1^{j-1}, \mathbf{d}_2^{\text{loc}} = \mathbf{d}_2^j, \mathbf{t}^{\text{loc}} = -\mathbf{t}^{j-1}\right\}$ as the “local" reference frame on edge $\mathbf{e}^{\text{loc}} = \mathbf{x}_j - \mathbf{x}_{j-1}$. The reference frame on the other edge $\mathbf{e}^j$ remains unchanged as $\left\{\mathbf{d}_1^j, \mathbf{d}_2^j, \mathbf{t}^j\right\}$. Flipping the edge also implies that the twist angle on $\mathbf{e}^{\text{loc}}$ in this local frame is $\theta^{\text{loc}} = -\theta^{j-1}$. This local representation in Fig. 2.3(b) can be used to compute the gradient and Hessian of the bending and twisting energies at $\mathbf{x}_j$ with respect to $\left\{\mathbf{x}_{j-1}, \theta^{\text{loc}}, \mathbf{x}_j, \theta^j, \mathbf{x}_{j+1}\right\}$ following the analytical expressions in [78]. Prior to including these gradient and Hessian terms in f (Eq. 2.13) and $\mathbb{J}$ (Eq. 2.14), we have to be mindful that $\frac{\partial}{\partial \theta^{j-1}}() = -\frac{\partial}{\partial \theta^{\text{loc}}}()$.

2.4.6 Physical parameters

The material and geometric parameters of the robot during experiments are: Young’s modulus $E = 1.2 \times 10^6$ N/m^2, Poisson’s ratio $\nu = 0.5$, density of the robot is 1000kg/m^3 (this is used to compute m_i in Eq. 2.13), and cross-sectional radius of tails $r_0 = 3.2$ mm. The length of each flagellum is $L_3 = 0.111$ m, radius of the robot head is $a = 0.02$ m, and the diameter of 3D-printed circular disc is $L_2 = 0.04$ m. Time step is $\Delta t = 10^{-2}$ s and the length of each edge on tails (in undeformed state) is $\|\bar{\mathbf{e}}^j\| = 4.11$ mm. We performed convergence studies to ensure that the size of temporal and spatial discretization ($\Delta t, \|\bar{\mathbf{e}}^j\|$) has negligible effect on the simulation results. The parameters μ, C_1, and C_2 will be fitted later in Section 2.5.3.

2.5 Results and discussion

MATLAB reads images extracted from the recorded videos in Section 2.3.2 in sequence to track the positions of the robot, s, versus corresponding time points, t, and the speed of the robot $v = s/t$. The number of turns the head and tails of the robot rotate (N_h, N_t) are counted brute force, and used to calculate

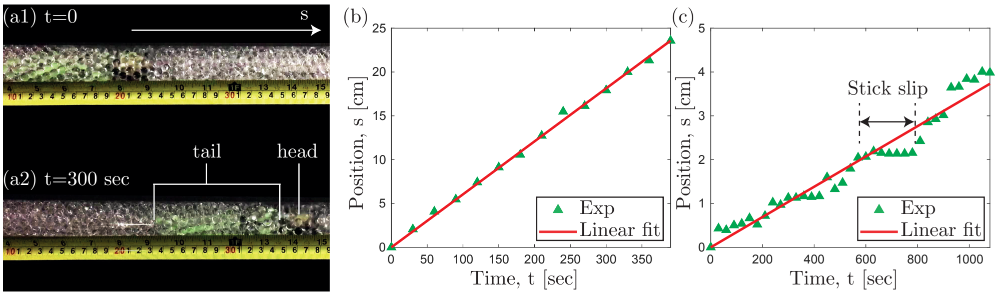

Figure 2.6: Position of the robot with time. (a1-a2) Snapshots from experiments showing the location of a robot with $n = 2$ and $\omega_{\text{total}} = 250$ rpm at time $t = 0$ and $t = 300$ s. (b) Position, s, of the same robot as a function of time, t. Solid line corresponds to the linear fit $s = v\,t$ where v is the speed. (c) Position vs. time of a robot with $n = 4$ and $\omega_{\text{total}} = 208$ rpm, where stick-slip dynamics is prominent.

$\omega_h = N_h/t$, $\omega_t = N_t/t$. Recall from Fig. 2.1 that the motor embedded in the head generates a rotational speed, ω_T. The head and tails rotate in opposite directions with rotational speeds of ω_h and ω_t such that $\omega_T = \omega_h + \omega_t$ (ω_T, ω_h, and ω_t are all non-negative). The total rotational speed, ω_T, is considered as a control parameter in our study. The rotation of the tails generates an axial propulsive force (see Fig. 2.3). The entire system uses this propulsive force to move forward at a speed of v.

2.5.1 Threshold angular speed to move

Experiments reveal that there is a threshold below which the robot stays still ($v = 0$) and above which it starts to move. This is explicable in terms of granular mechanics. Granular materials can behave solidly or fluidly. A threshold angular speed is necessary to convert the medium's behavior from solid to fluid. This threshold ω_T in our experiments is ≈ 50 rpm and thus there are no data points at $\omega_T \lesssim 50$ rpm. When $\omega_T \lesssim 50$ rpm, the tails and head still rotate relative to one another; however, the robot does not change its location. In this study, the maximum total angular speed is $\omega_T \approx 250$ rpm and the aforementioned regime ($\omega_T \approx 50$) is a relatively small part of the the overall parameter space. In simulations, we use RFT that does not consider this threshold. Nonetheless, the simulation can capture the motion of the robot when $v > 0$. It is straightforward to include this threshold in Section 2.4.4 by imposing boundary conditions on the head. However, this will introduce new fitting parameters without significantly improving the overall predictive ability of the simulation.

DC motors (uxcell), whose rated rotation speed with no load are 50 rpm, 75 rpm, 100 rpm, 150 rpm, 200 rpm and 250 rpm, are used. More off-the-shelf DC geared motors with the same nominal voltage, power and stall current are also available and they were tested in experiments. Their rated rotational speed is 19 rpm, 25 rpm, 35 rpm respectively but they do not show up in Fig 2.7 as they are below the threshold, not able to propel the robot to move. Note that the rated rotational speed of motors are very likely different from the actual rotational speed during experiments because of the deformation of tails due to uncontrollable load difference. For example, the rated rotational speed of the motor used in Figure 2.6 is 250 rpm with no load but ω_T turned out to be 208 rpm when the robot was equipped with four tails.

2.5.2 Speed of the robot

We use the speed of the robot, v, along the axial direction as the primary performance metric of the robot. This parameter will be used in subsequent sections to study the effect of the total angular speed, ω_T, and the number of tails, n. The efficiency of the robot, η, will also be defined related with the speed, v. During experiments, we used a digital camera to capture videos of the motion of the robot. Figs 2.6(a1) and (a2) show two snapshots of a robot with $n = 2$ tails and total rotational speed $\omega_T = 250$ rpm at $t = 0$ and $t = 300$ sec. The green tails were marked with black markers, and the black head was marked with a bright yellow marker. Aided by the transparency of the GM and the markers on the robot, these videos were processed to extract the position of the robot, s, as a function of time. Fig. 2.6(b) presents the position of the robot as a function of time. Closed triangles denote data from experiments, and the solid line represents a linear fit of the form $s = vt$. We observe that the robot moves at an almost constant velocity of $v \sim 0.6$ mm/s. This is expected from a solid body moving inside a medium governed by RFT.

Fig. 2.6(c) shows the position of a robot with $n = 4$ tails and rotational speed $\omega_T = 208$ rpm. The motion of the robot is now qualitatively different from the one presented in Fig. 2.6(b). The robot continuously moves forward in general but intermittently stays at the same position. This phenomenon is reminiscent of stick-slip – sudden motion that occurs when two multiple bodies are sliding past one another. At a larger number of tails (e.g., $n = 4$ and $n = 5$), experimental observations indicate that the GM can get jammed (i.e., increase in viscosity) and the robot frequently gets stuck. Interestingly, our experiments (see Fig. 2.6(b)) indicate the robot can resolve the jamming on its own through rotation (i.e., creating disturbance) for a few seconds. The periodic jamming or stick-slip cannot be captured by RFT and we do not include this behavior in our simulations. We focus only on robots with $n = 2$ and $n = 3$ tails that move at a constant speed with time. Nonetheless, this indicates room for expanding the theories for locomotion inside GM beyond RFT. Integrating such theories that describe the viscosity as a function of the robot configuration and time into the algorithm in Section 2.4.4 should be relatively trivial.

Granular media can present solid behavior in bulk through contact forces and fluidlike features when disturbed. It is asserted that volume fraction of sand varies between 0.57 and 0.64 depending on the history and a higher volume fraction means flowable-prone while a lower value means harder for the granular media to be intruded [6]. We measured its volume fraction, 0.52, meaning that it is easier for the beads to consolidate rather than flow, making the happening possibility of “stick slip” higher. The robot can be mainly di-

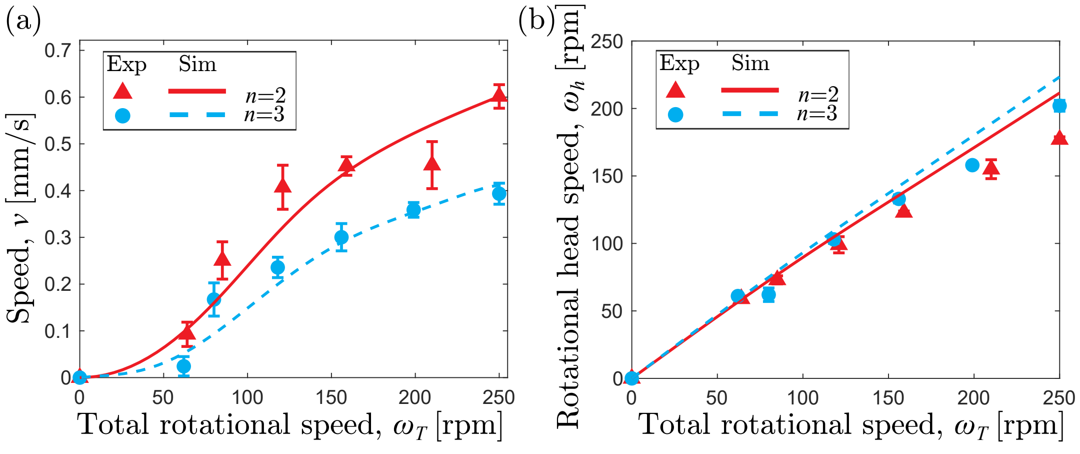

Figure 2.7: Comparison between experiment data and simulation results for the relationship between(a) total rotation speed of head and tail and robot moving speed; (b)total rotation speed of head and tail and rotation speed of head. The red triangles and blue circles with error bars are experiment data when the tail number is 2 and 3 respectively. The solid red line is the simulated outcome associated with the fitting parameters, C_1, C_2, and μ whereas the dashed blue line represents the simulation result predicted by the same fitting parameters.

vided into two parts, head and tails. Inspection from experiments showed when the robot was stuck, its head rotation slowed down or stopped, helping with the consolidation of the granular medium surrounding the head. During the “stuck” phase, and in preparation for escaping, the robot tails rotate faster, providing larger propulsion, as well as redistribute the beads. More and more beads around the tails contact and fluidize those ones adjoining the head, resulting in speeding up internal stress release and robot escaping in the meanwhile. This explanation can also be verified by experiment data in Fig. 2.6 (c). As you can see, the slop, robot moving speed, is dramatically larger after the “stick slip” than before.

2.5.3 Parameters fitting for simulations

We now move on to numerical simulations (details in Section 2.4) to model the locomotion and deformation of the robot.

Recall from Eqs. 2.11 and 2.12 that C_1 and C_2 are fitting parameters to account for the shape and surface roughness of the head. In addition, μ is the 3rd fitting parameter. As detailed next, experimental data with a 2-tailed robot ($n = 2$) are used to estimate C_1, C_2, and μ. Simulations are performed with these parameters for $n = 3$; simulation results are then compared against experiments for validation of the fitting process.

Figs. 2.7(a) and (b) present the speed of the robot, v, and the rotational speed of the head, ω_h, respectively, as functions of the total rotational speed, ω_T. The data for $n = 2$ and $n = 3$ are shown in the figures. The data (v vs. ω_T and ω_h vs. ω_T) for $n = 2$ are used to obtain the best fit values of the fitting parameters: $C_1 = 2.420, C_2 = 0.039$, and $\mu = 6.828$. These parameters are then used to simulate the locomotion of a robot with three tails. In Fig. 2.7(a), speed vs. total rotational speed data show good agreement between experiments and simulations. Fig. 2.7(b) shows the rotational speed of the head as a function of total rotational speed and we find that, in both experiments and simulations, a robot with $n = 3$ has a slightly larger head rotational speed than the one with $n = 2$.

The slight mismatch between the experimental and simulation data can be partially attributed to the assumptions made in the model. The fluid model assumes that the drag force exerted by the GM can be expressed using RFT. The structure model assumes that the tails are infinitesimally thin elastic rods. The drag force and torque on the head are assumed to be linearly proportional to its velocity and angular speed, respectively. Moreover, invariably there are experimental errors, e.g. structural defects introduced during fabrication. Nonetheless, the reasonably good agreement between experiments and simulations support the validity of RFT in this case.

2.5.4 Speed vs. number of tails

A counterintuitive discovery from Fig. 2.7(a) is that at a fixed value of the rotational speed of the motor, ω_T, the robot with 2 tails moves faster than the one with 3 tails. Additionally, the speed vs. total rotational speed curve is nonlinear. These observations point to the large structural deformation and strong coupling between the head and tails. As the number of tails, n, increases, the rotational speed of the head, ω_h, increases (at a fixed value of ω_T). Since $\omega_T = \omega_h + \omega_t$, this implies that the rotational speed of the tails, ω_t, decreases as n increases. The propulsive force generated by each tail (denoted as f_t) therefore also decreases. However, two additional factors to be considered to understand the overall speed, v, of the robot. First, the total propulsive force available is $n\, f_t$ and even though increasing n reduces f_t, it may (or may not) ultimately increase $n\, f_t$. Second, the total propulsive force is spent to overcome the drag on the head and the tail. As n increases, the amount of propulsive force spent on moving the tails forward also increases, and the propulsive force budgeted for the head decreases. All of these factors above combined dictate the dependence between the robot speed and the number of tails. Furthermore, our simulator predicts that for a small parameter space, an intersection exists between the relationship between the robot speed and the total rotational speed of the motor, such that the robot with more flagella can run faster than the robot with fewer flagella without experiencing “stick slip".

In the experiments presented herein, the set of physical parameters are chosen in such a way that the speed decreases with the number of tails. However, this is not universally true for this system. For example, consider a robot with $C_2 \to \infty$ in which the head never rotates (i.e. $\omega_h = 0$) and the rotational speed of the tail is always equal to the total rotational speed. Thus, f_t is a function of only ω_T (rather than n). Then, the total propulsive force, $n\, f_t$, increases with n

(assuming ω_T is fixed) and the speed of the robot is also expected to increase.

An absorbing thing found by us is when the number of robot tails is the same, velocity of the robot keeps increasing with the growth of total rotational speed of head and tail. Nonetheless, robot moving velocity increases sublinearly when the total rotation speed is rising up, which might be explained by the fact that our robot tails are soft so the whole system encounters high geometrical and mechanical nonlinearity.

2.5.5 Efficiency

The efficiency, η, of the robot is defined as the ratio of propulsive force to propulsive torque. Since η is a non-dimensional quantity, we choose the radius of the head, a, as the length scale. The expression for η is

$$\eta = \frac{|\mathbf{F}_h|}{|\mathbf{T}_h|} a = \frac{6\pi C_1 \mu a^2 v}{8\pi C_2 \mu a^3 \omega_h}, \tag{2.18}$$

where $|\cdot|$ denotes absolute value and expressions for $\mathbf{F}_h$ and $\mathbf{T}_h$ can be found in Eqs. 2.11 and 2.12, respectively.

The numerator represents the drag force exerted on the robot by the medium, whilst the denominator gives the overall torque generated by the rotation of the motor. We use the simulator to predict the variation of efficiency, η, with the rotational speed, ω_T, as exhibited in Fig. 2.8. The efficiency of a robot with $n = 2$ is non-monotonic and peaks at $\omega_T \approx 150$ rpm. At this optimal rotational speed, the robot moves the farthest per unit motor torque. Such a clear presence of an optimal rotational speed in the operating range of the motor highlights the need of a numerical simulator that can be used to design robots. Moreover, for $\omega_T \lesssim 200$rpm, the efficiency of a two-tailed robot is greater than the one of a three-tailed robot. Beyond $\omega_T \gtrsim 200$rpm, the three-tailed robot outperforms the two-tailed robot. These findings underline the high degree of nonlinearity in the functional dependence between the efficiency and the physical parameters (e.g. n and ω_T).

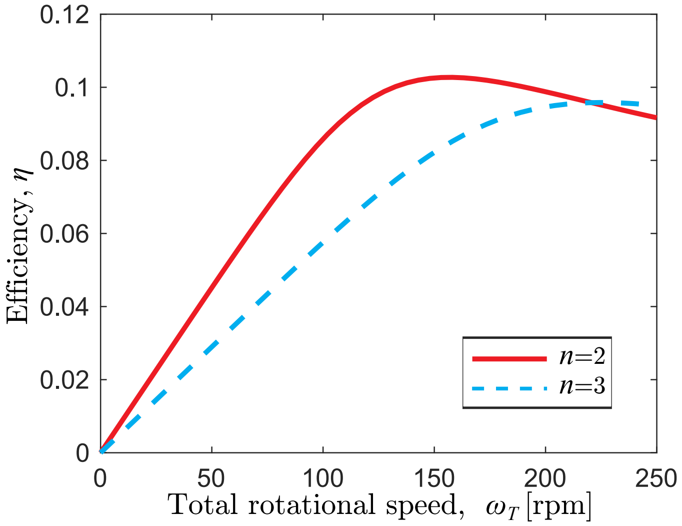

Figure 2.8: Variation of torque utilization efficiency versus the total rotational speed of robot predicted by our simulator.

2.6 Summary and Outlook

In this work, a discrete differential geometry(DDG)-based simulation framework was introduced where the robot is discretized into a number of mass-spring systems, with discrete elastic (bending, twisting, stretching) energies associated with each spring. The total elastic energy of the robot is the sum of all the discrete elastic energies. At each DOF, the sum of elastic force (i.e. negative gradient of the elastic energy) and external force is equal to the lumped mass times acceleration of that DOF. The actuation of the robot (i.e. rotational speed of the motor) is represented by a time varying natural strain. This approach allows us to simulate the shape of the robot in a fully implicit manner.

We explore the physics of locomotion in GM with our simulation tool and an articulated robot testbed with multiple elastic tails. RFT – originally developed to model the hydrodynamics of low Reynolds fluid flow – was used to model the external force exerted by the GM. This force can be easily integrated into the simulation framework. Comparison between experiments and simulations showed that RFT is reasonably valid for flagellated robots discussed here. However, when the number of tails is large, “stick-slip" was observed and the underlying assumption behind RFT was no longer valid. Note that other types of external forces, such as gravity, hydrodynamics, and magnetic forces, can be easily incorporated to our simulation framework.

The simulation tool, supported by experiments, shed light on the highly nonlinear functional dependence between the performance of the robot (e.g. speed or efficiency) and the relevant physical parameters (e.g. number of tails). Some counter-intuitive observations include the inverse relation between the speed and number of tails of the robot. This happens when the robot is in a tube with different titled angles, e.g. 8°, 90°, the results of which are not displayed here because of the page limit. The non-monotonic dependence of efficiency on the rotational speed of the motor highlighted the necessity of a design tool for optimal control of the robot. The computational speed of the simulator can be exploited to run parametric studies and identify the optimal design and control of general articulated soft robots. The simulation tool, supported by experiments, shed light on the highly nonlinear functional dependence between the performance of the robot (e.g. speed or efficiency) and the relevant physical parameters (e.g. the number of tails).

One of the most exciting aspects of this is that our simulator is able to simulate the dynamics of any articulated soft robots in real time and generate plentiful data in a short time, which makes it possible for us to develop data-driven method to assist people working on robotic design. For the experimental part, we generalize a fast-fabricated method of building a low-cost functional articulated soft robot testbed comprised of flexible filaments to help robot design optimization. Researchers also benefit from investigating on movements in granular media or low Reynolds number fluid through timely verification of theories or their ideas. In the near future, it may be possible to attach skin-like sensors to the flagella and cameras to the head to detect their rotation and the relative speed of the granular media relative to the flagellum. These signals can then be utilized to enhance the RFT model by adding the slippage-related component and the performance of the granular medium during the movement of the robot. Thus, the granular media do not need to be transparent in order to locate the robot and visualize its movement.

CHAPTER 3

Mechanics-based Analysis on Flagellated Robots

We explore the locomotion of soft robots in granular medium resulting from the elastic deformation of slender rods. A low-cost, rapidly fabricable robot inspired by the physiological structure of bacteria is presented. Multiple elastic rods, our model for flagella, are rotated at one end by the motor, and they deform due to granular drag, propelling the robot forward. The external drag is determined by flagellar shape, while the shape changes due to the competition between external loading and elastic forces. In this coupled fluid-structure interaction problem, interestingly, we observe that – depending on the physical parameters of the system – increasing the number of flagella can decrease (design 1) or increase (design 2) speed of the robot. This nonlinearity between propulsion and physical parameters motivates us to analyze its mechanics using theory, numerical simulation, and experiments. We present a simple Euler-Bernoulli beam theory-based analytical framework that is capable of qualitatively capturing both designs. Theoretical and experimental results match quantitatively under small flagellar deformation. To account for the geometrically nonlinear deformation often encountered in soft robots and microbes, we implement a simulation framework that incorporates discrete differential geometry-based simulations of flagella, a resistive force theory-based model for drag, and a modified Stoke's law for the hydrodynamics of the head. Comparison with experimental data indicates that simulations can quantitatively predict the robotic motion in both designs. Overall, the theoretical and numerical tools presented can shed light on the design and control of this class of articulated robots in granular or fluid media.

We introduce the motivation in § 3.1 and then do the literature review in § 3.2. Next, we state and formulate the definition of the problem in § 3.3, following which is the methodology we use in § 3.4 to solve the problem. § 3.4 mainly is comprised of two big parts: beam theory-based analysis of the untethered soft-flagellated propulsion given in § 3 and DER-based numerical simulation of flagellar locomotion of two different robot designs in § 3.4.2. Next, we detail the experimental design in § 3.5. Subsequently, we extensively illustrate our findings from experiments and their comparison against theoretical analyses and numerical simulations in § 3.6. After that, this chapter is concluded in § 3.7 Eventually, the summary and outlook are presented in § 3.8. The content of this chapter has appeared in Ref. [111].

3.1 Motivation

The motivation of our research is to investigate the mechanics of locomotion in GM so the inspiration is similar to Ch. 2. The deeper inspiration of this chapter relies on the result of Ch. 2. After figuring out how to use a fully-implicit DER-based framework to simulate the performance of the locomotion in GM, we discovered that it is possible to use a simple Euler-Bernoulli beam theory-based framework to analytically express the relationship between measurable system performance, e.g., the speed of the robot, and the control variables, e.g., the rotational speed of the motor to which flagella are attached.

3.2 Literature review

Apart from animals capable of crawling, digging [112], slithering [101], swimming [113], and gliding [89, 114] in environments, bacteria, one major group of microorganisms, also inspire the development of novel and efficient robots. 90% of marine bacteria[115] achieve efficient locomotion in a fluid through the rotation of a flagellum (a slender filament), exploiting the anisotropic drag to produce thrust and violating the constraints of the scallop theorem[21]. Flagellar locomotion results from a non-trivial coupling between the geometrically nonlinear deformation in the flagellum and mechanics of the surrounding medium, posing a challenging fluid-structure interaction (FSI) problem.

Theoretical study about flagellar propulsion dates back to 1955 when Taylor [116] first analyzed the swimming of microscopic organisms. Over the last two decades, numerous studies have studied flagellar propulsion in low Reynolds fluids through experiments[40, 117, 118, 1, 86, 119], computation[120, 121, 26], and theory[122, 123, 27, 124]. Recent investigations[125, 66] have modelled the flagellum as a Kirchhoff elastic rod[126], and coupled to the fluid with hydrodynamic forces[5]. Jawed investigated the dynamics of a helical elastic flagellum rotating in a viscous fluid[72] and near a rigid boundary[74]. However, the role of the head and the flow generated by its motion and coupling with the flagellum-induced flow are ignored. Huang performed simulations that utilized flagellar buckling to change moving direction on a robot composed of a mass-point head and uni-flagellum[75]. Our previous work[66] established an untethered articulated robot that was composed of a rigid head and multiple soft flagella and used discrete differential geometry to simulate the flagella and resistive force theory[5] (RFT) to model the interaction force between the fluid and flagella. Simulation and experimental results agreed well quantitatively. Our robot also demonstrated the same behavior as bacteria, ii.e., head and flagella rotating in the opposite directions[127] and circling when near the air-fluid boundary.

RFT, initially used for analyzing the movements of microscopic organisms in viscous fluids, is proved to apply to animal and robot locomotion on and within granular media (GM) [6]. This builds an intimate link between the microscopic bacterial world and meter-sized animals in sand. The most recent studies [93, 128], on rotational intruders moving through GM conclude on the empirical feasibility of applying modified and entirely empirical RFT, e.g., granular RFT [128]. These experimental studies demonstrate that RFT is simple yet very effective in granular materials considering the complicated constitutive features of granular matter such as nonlinearity and nonlocality. However, the intruders in these works are *rigid*, inspiring our further study on the geometrical design and efficiency study on robots with *soft* flagella in GM. Previously, our other work modeled the locomotion of articulated soft robots in GM[125]. Leading design and control parameters of our untether robots are the rotational speed of the embedded motor, the number of flagella, geometrical parameters of flagella such as radius and length. Some counter-intuitive observations include the inverse relationship between the speed and the number of flagella of the robot in the representative setup (as shown in Fig.3.1(**a1-a5**)). This verifies that the flagellar locomotion is the result of a complicated coupling between granular mechanics and deformable bodies, a complex FSI problem. In summary, while there exist works[86, 75, 66] that explore flagellar locomotion considering the effect of the head, systematic simulations are required for more quantitative predictions, e.g., under what design and control parameters the maximal efficiency of actuation is achieved. Furthermore, although a comprehensive simulation framework is offered in our earlier work[125], elaborate equations suffocate and make it difficult for designers to understand and predict the movement of the system directly. Subsequently, a reduced model capable of qualitatively capturing the relationship between the performance of flagellar locomotion in GM versus the design space is required, hence avoiding the cumbersome trial and error design process.

Here, we employ a structural robot design similar to that used in our previous work[125]: a palm-sized untethered robot composed of $n \geq 2$ naturally straight elastic rods and a rigid head housing a geared DC motor and batteries.

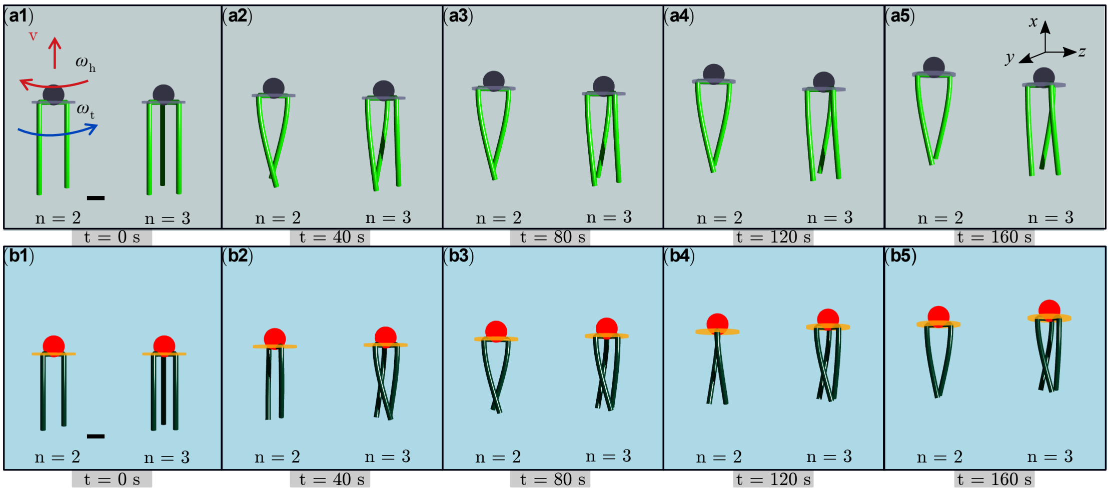

Figure 3.1: Snapshots from simulation. The first row depicts the shape and movement of the robot in design 1 where the robot speed decreases as the number of flagella increases from, $n = 2$, to $n = 3$ in panels (**a1-a5**). In contrast, the second row illustrates the shape and movement of the robot in design 2 where as the number of flagella, n, increases from two to three in panels (**b1-b5**), the speed of the robot increases. In (**a1-a5**), the total rotational speed of the motor $\omega_T = 100.00$ rpm, head rotational speed $\omega_h = 95.47$ rpm, and flagellar rotational speed $\omega_t = 4.53$ rpm while for $n = 3$ in (**a1-a5**), $\omega_T = 100.00$ rpm, $\omega_h = 97.49$ rpm, and $\omega_t = 2.51$ rpm. In (**b1-b5**) and $n = 2$, $\omega_T = 100.00$ rpm, $\omega_h = 44.06$ rpm, and $\omega_t = 55.94$ rpm while $\omega_T = 100.00$ rpm, $\omega_h = 56.98$ rpm, and $\omega_t = 43.02$ rpm for $n = 3$ in (**b1-b5**). Black bar is 2 cm. Physical parameters are available in Sec.3.4.3.

As illustrated in Fig. 3.1, the rotation of flagella induces the drag force from the surrounding GM because of their flexibility. Flagellar deformation provides a net propulsive force forward, which is zero in the case of straight *rigid* flagella. Meanwhile, we implement two numerical frameworks, both of which employ Resistive Force Theory (RFT) to model the drag on a collection of multiple elastic rods and modified Stoke's law to calculate the drag force and moment on the head. In contrast, one uses Euler-Bernoulli beam theory, and the other applies discrete differential geometry (DDG) to simulate the flagellated robot structure. When the motor rotates slowly, e.g., $\omega_{\mathrm{T}} \leq 10$ rpm, the deformation of the robot flagella is linear, and the resulting propulsive force calculated using DDG simulation is nearly identical to that obtained from Euler-Bernoulli beam theory. However, the beam theory fails to precisely capture the nonlinear deformation of the robot flagella at high motor speeds, e.g., $\omega_T = 250$ rpm, DDG-based framework is still capable of accurately representing the performance of the robot in experiments. Due to the rod-based kinematic representation of the robot, this computational tool is used to simulate flagellated robots even faster than real-time. Additionally, our prior study uncovered a counterintuitive phenomenon, the speed of the robot falls as the number of flagella rises (design 1), which is captured by both the beam theory and DDG-based frameworks. However, the beam theory-based framework predicts the existence of design 2, in which the speed of the robot increases as the number of flagella grows. This prediction is successfully confirmed experimentally after we modify the robot design and radius of the GM. In summary, the simple overall Euler-Bernoulli beam theory-based framework can qualitatively capture both designs of the complicated locomotion in GM, the intricate interaction between the robot and GM. In contrast, the DDG-based framework is capable of quantitatively simulating the complex locomotion in GM.

3.3 Problem Statement

In this work, we analyze flagellated locomotion in GM, a typical FSI problem, using a mechanics-based approach. As a result, both the experimental and theoretical aspects of the problem involve two primary components, the GM (fluid part) and the robot (structure part). A GM must be selected to complement the architecture of the untethered flagellated robot. Internal friction between granules prevents the robot head from fluidizing the granules in front and propelling forward, whereas insufficient friction results in ineffective flagellar propulsion. Internal friction during locomotion is correlated with the phase transition of GM from solid to fluid, which is governed by temperature and volume fraction (VF)[129] - the ratio of solid to occupied volume. The less effort required to modify the VF of GM, the better for investigating the locomotion in GM. Apart from the volume fraction of the GM, other physically controllable parameters include the shape and material friction of the head, as well as the number, radius, length, and stiffness of the flagella and the rotational speed of the motor. Among them, the number, n, and the rotational speed of the motor, ω_T, are the most precisely trackable with the least effort owing to the design simplicity of the robot and experiments. Their effect on the translational speed of the robot, v, is explored and will be displayed in Sections 3.6.1 and 3.6.3.1.

The entire robot structure consists mainly of active multiple soft elastic flagella and a rigid passively actuated head, all of which rotate along the long axis of the robot. The case where the flagella are fixed at a 3D-printed plate with uniform external drag forces applied is analogous to the case of a cantilever beam with a uniformly distributed load. In what follows, first, we outline the Euler-Bernoulli beam theory-based mechanics analysis of the untethered robot, including (1) the external loading from the GM onto flexible flagella and (2) the drag force on the head. Beam theory qualitatively captures the two cases in which increasing n can either accelerate or retard the speed, v. Whichever happens, in reality, is determined by the intricate balance of the competition between the external loading and elastic forces. According to experimental evidence, the result is closely related to the robot head and motor speed design, ω_T. Second, we introduce a numerical model of the robot in which the robot structure is represented by a network of Kirchhoff's rods[126]. Finally, we present experiments conducted to quantify the propulsive speed of this class of flagellated robots in Section 3.5.

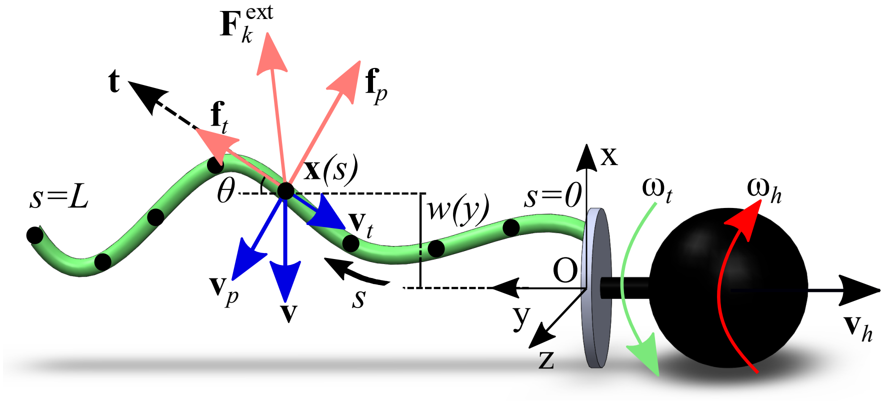

Figure 3.2: Schematic showing the velocities and induced resistive forces on one flagellum rotating at a constant rotational speed ω_t on the robot moving forward at a velocity of $\mathbf{v}_h$. The coordinate system $(O-x-y-z)$ is fixed at the center of the 3D-printed plate; the flagellum starts at $s=0$, propagates along the x-direction until the end $(s=L)$, and is divided into a series of nodes. Each node (numbered based on differential geometry) is characterized by its tangential direction $\mathbf{t}$ and its velocity $\mathbf{v}$ direction; k-th node (with coordinate $\mathbf{x}(s)$) experiences a force $\mathbf{F}_k^{\text{ext}}$ that propels the robot forward (along $-y$-direction). The flagellum rotates at an angular speed of ω_t, clockwise as viewed from the $+y$-axis, while the head rotates in the opposite direction at an angular speed of ω_h.

3.4 Methodology

3.4.1 Beam theory based analysis of propulsion

Benefiting from the elasticity property, the soft flagella of the robot bend out of the long axis, and the drag force generated by the surrounding granules produces a component in the -y direction as displayed in Fig. 3.2, propelling the robot forward.

3.4.1.1 Clamped flagellum fixed in space

For illustration, one flagellum of the robot in a discrete setting moving in the GM is displayed in Fig. 3.2. Hereafter, unless otherwise stated, all the parameters, e.g., forces, are associated with one flagellum. Consider a node $\mathbf{x}(s)$ along the flagellum, where $s \in [0, L]$ is the arc-length parameter and L is the length of the flagellum. At the steady state, the magnitude of the velocity vector at $\mathbf{x}(s)$ is defined as

$$v = \|\mathbf{v}\| = \|\omega_t \times R_d \mathbf{r}\| = \omega_t R_\text{d}, \tag{3.1}$$

where $\mathbf{v}$ is along $-y$ axis, ω_t stands for the rotational speed of flagella, $\mathbf{r}$ is the radial unit vector, and R_d is the radius of 3D printed plate holding soft flagella. The velocity, $\mathbf{v}$ at this point can be decomposed into two parts: the parallel term $\mathbf{v}_t = (\mathbf{v} \cdot \mathbf{t})\mathbf{t} = v_t \mathbf{t}$ and the perpendicular term $\mathbf{v}_p = \mathbf{v} - \mathbf{v}_t$, where $v_t = -v \sin\theta\ \mathbf{t}$ and $\mathbf{t}$ represents the tangent at the point, whose direction is shown in Fig. 3.2. Inspired by Texier *et al.* [93], the drag force exerted on the flagellum by the GM is modeled by resistive force theory (RFT) and is dependent on the rotational speed of the motor, ω_t,and thus, $\mathbf{v}$. The drag is broken down into components normal and tangential to the segment axis, with two corresponding force coefficients, and its dependence on the shape of the object, e.g. the diameter, is demonstrated[130]. Additionally, the work[131] considers the effect of the geometry of the soft filament while modeling the local drag. As a result, the tangential and perpendicular force constants (force per unit length) from the medium that resist $\mathbf{v}_t$ and $\mathbf{v}_p$ are

$$\mathbf{f}_t = -\eta_t \mathbf{v}_t, \tag{3.2a}$$

$$\mathbf{f}_p = -\eta_p \mathbf{v}_p, \tag{3.2b}$$

where the drag coefficients, one along the tangential direction and one along the perpendicular direction are

$$\eta_t = \frac{2\pi\mu}{\log(\frac{2L}{r_0}) - \frac{1}{2}} \tag{3.3a}$$

$$\eta_p = \frac{4\pi\mu}{\log(\frac{2L}{r_0}) + \frac{1}{2}}, \tag{3.3b}$$

where μ is the coefficient constant that quantifies the robot-granule friction as a function of the granule size and the inertia and surface friction of the head, and L and r_0 are the length and radius of each flagellum, respectively. The drag force constant of external forces at node $\mathbf{x}(s)$ is

$$\mathbf{F}^{\text{ext}} = \mathbf{f}_t + \mathbf{f}_p. \tag{3.4}$$

Note that $\mathbf{F}^{\text{ext}}$ depends on the granular pressure and the velocity of nodes on flagella. Then, the component of $\mathbf{F}^{\text{ext}}$ along the x-axis (i.e., antiparallel to velocity) is

$$p_t = -\eta_t v_t \sin\theta - \eta_p \left(-v - v_t \sin\theta\right), \tag{3.5}$$

where

$$\sin\theta = \frac{\mathrm{d}w}{\sqrt{\mathrm{d}y^2 + \mathrm{d}w^2}}, \tag{3.6}$$

and

$$\cos\theta = \frac{\mathrm{d}y}{\sqrt{\mathrm{d}y^2 + \mathrm{d}w^2}}. \tag{3.7}$$

The transverse displacement of a flagellum, which is modeled as a beam is $w(s)$. Assuming a small deflection, $w(s) \approx w(y)$ where $w(y)$ represents the flagellar deflection along the y-axis. We ignore the effects of head translation in this section; they will be discussed in the following section. The drag force can be split into y- and x-components; the former produces propulsive force, while the latter produces a torque that rotates the head. Given our definition of the direction of y-axis, the force constant of propulsive force is as follows along $-y$:

$$q = \left(\eta_p - \eta_t\right) \omega_t R_{\mathrm{d}} \sin\theta\cos\theta. \tag{3.8}$$

Without ignoring any higher order terms, i.e. $\sin^2\theta$, the x-axis force constant is formulated:

$$p_t = EI\frac{\mathrm{d}^4 w}{\mathrm{d}y^4} = (\eta_t - \eta_p)v\sin^2\theta + \eta_p v, \tag{3.9}$$

where EI is the bending stiffness of the beam (i.e. the flagellum).

3.4.1.2 Clamped flagellum moving at constant speed

During the robotic movement, in addition to the flagella, its head is also rotating and translating. Assuming the robot moves along the y-axis at speed v_h, we rewrite its velocity as

$$\mathbf{v} = -\omega_t R_\mathrm{d}\hat{x} - v_h\hat{y}, \tag{3.10}$$

where $\hat{x}, \hat{y}$ are the unit vectors in the direction of the x-axis and y-axis, respectively. The force constant of viscous drag force is:

$$\mathbf{f}_h = -\eta_t v_h^t\mathbf{t} - \eta_p\left(\mathbf{v}_h - v_h^t\mathbf{t}\right), \tag{3.11}$$

and its component along y-axis is

$$q_h = (\eta_p - \eta_t)\, v_h\cos^2\theta - \eta_p v_h. \tag{3.12}$$

Adding q_h to Eq. 3.8, we get the final drag force

$$\begin{aligned} q = (\eta_p - \eta_t)\left(\omega_t R_\mathrm{d}\sin\theta\cos\theta + v_h\cos^2\theta\right) \\ -\eta_p v_h. \end{aligned} \tag{3.13}$$

We integrate this across the length of the flagella to compute the total propulsive force. Its component along the x-axis is

$$p_h = (\eta_t - \eta_p)\, v_h\sin\theta\cos\theta. \tag{3.14}$$

Adding this solution to Eq. 3.9, we get the following expression for the force constant along x-axis:

$$p = p_t + p_h = EI\frac{\mathrm{d}^4 w}{\mathrm{d}y^4}, \tag{3.15}$$

$$= \eta_p v + \frac{(\eta_t - \eta_p)\left(v\left(\frac{\mathrm{d}w}{\mathrm{d}y}\right)^2 + v_h\left(\frac{\mathrm{d}w}{\mathrm{d}y}\right)\right)}{\left(\frac{\mathrm{d}w}{\mathrm{d}y}\right)^2 + 1}. \tag{3.16}$$

We used the `bvpfcn` function in MATLAB to solve this nonlinear fourth order differential equations with boundary conditions

$$w(0) = 0, w'(0) = 0, w''(L) = 0, w'''(L) = 0, \tag{3.17}$$

where the prime ($'$) indicates differentiation with respect to y. Since the head-induced effect is taken into account and no approximations about the flagellar deflection are made during the calculation, this design is referred to as *nonlinear beam (NLB)* regime. However, by making some simplifying assumptions, we can achieve a more straightforward, closed-form solution. If we assume that the flagellum is slightly deflected to be treated as an Euler-Bernoulli beam, then $\sin^2\theta \approx 0$ and $\cos^2\theta \approx 1$. This enables us to simplify the solution and achieve linear approximation, referred to as *linear beam (LB)* regime. The curvature in Euler-Bernoulli beam theory is approximated as $\frac{\mathrm{d}^2 w}{\mathrm{d}y^2}$ and $\left(\frac{\mathrm{d}w}{\mathrm{d}y}\right)^2$ terms are assumed to be negligible compared with 1. The boundary conditions on the beam are Eq. 3.17. The deflection of a *LB* is, therefore, calculated as follows

$$w(y) = \frac{p_t y^2(6L^2 - 4Ly + y^2)}{24EI}, \tag{3.18}$$

and p_t here is obtained as follows by omitting the $\sin^2\theta$ term in Eq. 3.9:

$$p_t = \eta_p v = \eta_p \omega_t R_\mathrm{d}. \tag{3.19}$$

Hence, the simplified total force along x-axis is

$$F_x = \eta_p \omega_t R_\text{d} L. \tag{3.20}$$

The linear propulsive force is likewise simplified as

$$q = (\eta_p - \eta_t)\, \omega_t R_\text{d} w'(y) - \eta_t v_h, \tag{3.21}$$

and the total propulsive force is

$$F_p = (\eta_p - \eta_t)\, \omega_t R_\text{d} w(L) - \eta_t v_h L. \tag{3.22}$$

Note that we can neglect the head velocity in these equations because it has been experimentally proved to be small. The deflection of *LB*, *NLB*, and *NLB w/o head (nonlinear beam without head)* is compared in Fig. 3.7 and Section 3.6.4. In addition, the head experiences a resistive force due to its translation and rotation. The viscous drags due to its translation and rotation are

$$\mathbf{F}_p = -C_1 (6\pi\mu R_\text{h} \mathbf{v}_h), \tag{3.23}$$

$$T_p = -C_2 (8\pi\mu R_\text{h}^3 \omega_h), \tag{3.24}$$

respectively, where $6\pi\mu R_\text{h}\mathbf{v}_h$ is the drag force and $8\pi\mu R_\text{h}^3\omega_h$ is the torque on a perfectly spherical object in a low Reynolds number fluid according to Stokes' law and $\mathbf{v}_h = \dot{\mathbf{x}}_1$ is the velocity of the head, and $\omega_h = \dot{\theta}^0$ is the rotational speed of the head. The coefficients C_1 and C_2 are included because our robot head is not a sphere; given the difficulty of measuring C_1, C_2, and μ during experiments, they are used as fitting parameters (see Section 3.6.2) to match the experimental and simulation results in Section 3.6. Furthermore, this flagellated robot system is balanced in terms of both force and torque. As a result, the propulsive force generated by the flagella should be equal to the drag force on the head, $F_p = \|\mathbf{F}_p\|$, and the torque actuating the rotational movement of n flagella should also be equal to the moment on the head. Using $F_p = \int_{y=0}^{y=L} q(y)\mathrm{d}y$, or the corresponding linear approximation in Eq. 3.22, we can compute the velocity:

$$v_h = \frac{nF_p}{C_1 6\pi\mu R_\mathrm{h}}, \tag{3.25}$$

where R_h is the radius of the robot head. Next, we calculate the rotation speed of the head, ω_h, based on the torque balance. To compute this, we need the torque generated by F_x, which is given by

$$T_p = nR_\mathrm{d}F_x, \tag{3.26}$$

where $F_x = \int_{y=0}^{y=L} p_t(y)\mathrm{d}y$, or with the linear approximation from Eq. 3.20. Given that the torque generated by the flagella (in Eq. 3.26) must be equal and in the opposite direction of the torque on the head in Eq. 3.24, we can compute the rotational speed of the head:

$$\omega_h = \frac{nR_\mathrm{d}F_x}{C_2 8\pi\mu R_\mathrm{h}^3}. \tag{3.27}$$

Plug Eq. 3.6 into Eq. 3.9, we obtain p_t as a function of only y and ω_t, denoted as $p_t = f(y, \omega_t)$ and thus $F_x = f(\omega_t)$. Then, from Eq. 3.27, we find that ω_h is a function of ω_t whereas the remaining parameters are constants. In addition, one of control parameters in experiments is the rotational speed of the motor ω_T:

$$\omega_T = \omega_h + \omega_t. \tag{3.28}$$

Note that ω_h and ω_t here are scalars, but the rotational directions of the head and flagella of the robot are opposite. In conclusion, we have two equations for two unknown variables ω_t and ω_h, allowing us to solve the propulsion of the

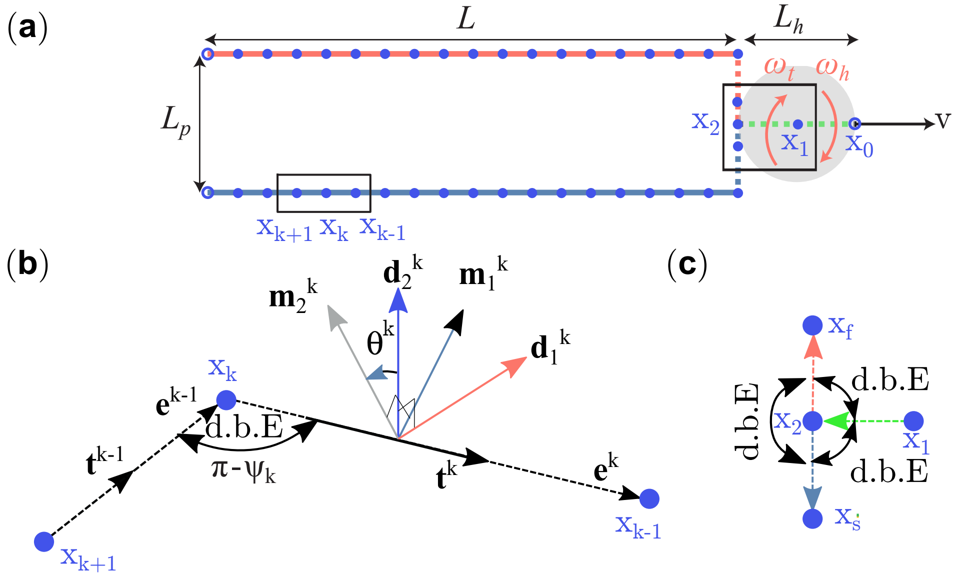

Figure 3.3: Discrete schematic diagram of a robot with two flagella. (a) Geometry of the robot in its undeformed state. Here, $L_h = 2R_\text{h}$ is the diameter of the robot head, L_p is the diameter of the 3D-printed plate bridging the head and flagella, and L is the length of each flagellum. Dashed lines represent the rigid structure, while solid lines represent the elastic structure. The node $\mathbf{x}_1$ specifies the center of the head, i.e. its location. (**b**) A general close-up view of three adjacent nodes, $\mathbf{x}_{k-1}, \mathbf{x}_k$, and $\mathbf{x}_{k+1}$, and two edges, $\mathbf{e}^{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}$ and $\mathbf{e}^k = \mathbf{x}_{k+1} - \mathbf{x}_k$. The turning angle, ψ_k, between the two edges results in bending energy and the rotation of the material frame from one edge to the next results in twisting energy. The reference frame on $\mathbf{e}^k$ is $\left\{\mathbf{d}_1^k, \mathbf{d}_2^k, \mathbf{t}^k\right\}$ and the material frame is $\left\{\mathbf{m}_1^k, \mathbf{m}_2^k, \mathbf{t}^k\right\}$. The twist angle from the first material director associated with $\mathbf{t}^{k-1}$ to the next material director associated with $\mathbf{t}^k$ is θ^k. (c) A close-up of the “T"-shape joint node $\mathbf{x}_2$ that connects the head to the flagella. d.B.E. indicates discrete bending and twisting energy. Only the joint node, $\mathbf{x}_2$, is connected to more than two nodes.

entire system. From these two variables, all other parameters, such as v_h and F_p (by combining Eqs. 3.22 and 3.25) can be calculated. We emphasize that the overall framework includes not only the flagellar propulsion based on beam theory but also the head motion based on Stokes' law.

3.4.2 Numerical method of flagellar locomotion

In this section, we present the numerical simulation framework in which the robot structure is modeled as a network of connected Kirchhoff elastic rods. The elastic energies of this structure are thereby the linear sum of bending, twisting, and stretching energies, the negative gradient of which is the elastic forces on each degree of freedom (DOF). The implicit Euler method is used to solve equations of motion (EOM) in which the external drag forces from the GM on each DOF are modeled through RFT. When ω_T is small, locomotion parameters obtained from this simulation, e.g., velocity, propulsive force, and deflection, are compared to those obtained from analytical equations based on Euler-Bernoulli beam theory described in Section 3.4.1. Furthermore, the simulation results of DDG are shown to match quantitatively with experimental data, validating the method itself.

3.4.2.1 Discrete Differential Geometry (DDG)

Fig 3.3 is a discrete representation of a robot with a rigid head and 3D-printed plate and n straight elastic flagella ($n = 2$ in the figure). The numerical values for physical parameters will be given in Section 3.6.2. Along the schematic discretization, there are a collection of N nodes (circles in Fig. 3.3), $\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_{N-1}$, attached via $N-1$ thin elastic rod vectors, $\mathbf{e}^k = \mathbf{x}_{k+1} - \mathbf{x}_k (k = 0, 1, ..., N-2)$, called edges (the lines between two adjacent nodes). Hereafter, we use subscripts for node-based quantities and superscripts for edge-based quantities. As shown in Fig. 3.3(b), the head is discretized into $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2$, with node $\mathbf{x}_2$ denoting the one at the head-flagella junction. It is a unique node that is connected to $n + 1$ additional nodes. All other nodes are connected to two nodes, or to a single node in case of the last node at the end of each flagellum, denoted by an open circle.

Rather than being just a single vector, each edge (see Fig. 3.3) is also equipped with two corresponding sets of orthonormal reference frames to track its rotation, a reference frame $\left\{\mathbf{d}_1^k, \mathbf{d}_2^k, \mathbf{t}^k\right\}$ and a material frame $\left\{\mathbf{m}_1^k, \mathbf{m}_2^k, \mathbf{t}^k\right\}$; both of them share the tangent $\mathbf{t}^k = \mathbf{e}^k / \|\mathbf{e}^k\|$ as one of the directors ($\| \cdot \|$ represents the Euclidean norm of a vector). The reference frame serves as a frame initialized at time $t = 0$ and updated at each subsequent time step via time-parallel transport, as illustrated in Fig. 3.3(c), the material frame can be calculated in terms of a scalar twist angle, θ^k. The detailed transformation expression between the reference and material frames can be found in Du *et al.*[125] and Jawed *et al.*[83]. For this class of flagellated robots, the DOF vector is constituted of node positions and twist angles formulated as follows

$$\mathbf{q} = \left[\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N-1}, \theta^0, \theta^1, \ldots, \theta^{N_e - 1}\right]^T, \tag{3.29}$$

where N_e is the total number of edges in the entire robot, which in this work is $N - 1$ (see Fig. 3.3(a)), and the superscript ^{T} stands for transpose. $\mathbf{q}$ has a dimension of ndof $= 3 \times N + N_e$. Since $\mathbf{q}$ completely defines the configuration of the robot whose deformation varies with time, the DOF vector is a function of time, i.e. $\mathbf{q} \equiv \mathbf{q}(t)$. At time $t = 0$, the robot is undeformed and the DOF vector is $\mathbf{q}(0) \equiv \bar{\mathbf{q}}$; in the following, $\bar{(\,)}$ represents evaluation of a quantity in the undeformed configuration. The strain calculations coming next all rely on the DOF vector $\mathbf{q}$.

$$\mathbf{m}_1^k = \mathbf{d}_1^k \cos\theta^k + \mathbf{d}_2^k \sin\theta^k \tag{3.30a}$$

$$\mathbf{m}_2^k = -\mathbf{d}_1^k \sin\theta^k + \mathbf{d}_2^k \cos\theta^k \tag{3.30b}$$

Based on this kinematic representation, we will sequentially discuss the formulation of elastic energies and forces, external forces, and simulation loops.

3.4.2.2 Elastic energies and forces

The total elastic energy of a flagellated robot structure is the linear sum of stretching, bending, and twisting energies:

$$E_{\mathrm{E}} = E_s + E_b + E_t, \tag{3.31}$$

where $E_s, E_b,$ and E_t are the stretching, bending, and twisting energies, respectively.

Elastic energies of a structure are associated with the corresponding macroscopic strains, axial stretch, curvature, and twist[132]. The stretching energy associated with each edge is related to the axial stretch of the edge. Axial stretch is the change in length of an edge, normalized by the undeformed length. The axial stretch, ϵ^k, of edge e^k is

$$\epsilon^k = \frac{\|\mathbf{e}^k\|}{\|\bar{\mathbf{e}}^k\|} - 1, \tag{3.32}$$

where $\|\bar{\mathbf{e}}^k\|$ is the undeformed edge length. Given this axial stretch, the stretching energy along edge e^k is

$$E_s^k = \frac{1}{2} EA \left(\epsilon^k\right)^2 \|\bar{\mathbf{e}}^k\| \tag{3.33}$$

where E is the Young's modulus, $A = \pi r_0^2$ is the cross-sectional area of the flagella, and r_0 is the flagellar radius. The total stretching energy of the robot is the sum of individual stretching energies, i.e.

$$E_s = \sum_{k=1}^{N_e} E_s^k. \tag{3.34}$$

For edges on the rigid head and disk, the stretching stiffness EA is set to be sufficiently large to ensure that deformation is negligible.

Bending energy is related to curvature, a node-based quantity that is related to the turning angle, ψ_k, displayed in Fig. 3.3(c). Note that curvature is applied to all nodes except terminal nodes, and the curvature binomial for node x_k is $(\kappa \mathbf{b})_k$ defined as the following vector

$$(\kappa \mathbf{b})_k = \frac{2\mathbf{e}^{k-1} \times \mathbf{e}^k}{\|\mathbf{e}^{k-1}\| \|\mathbf{e}^k\| + \mathbf{e}^{k-1} \cdot \mathbf{e}^k}, \tag{3.35}$$

where $\|(\kappa \mathbf{b})_k\| = 2\tan\left(\frac{\psi_k}{2}\right)$. The scalar curvatures along the first and second material directors, calculated using the curvature binomial, are

$$\kappa_k^{(1)} = \frac{1}{2}(\mathbf{m}_2^{k-1} + \mathbf{m}_2^k) \cdot (\kappa \mathbf{b})_k, \tag{3.36a}$$

$$\kappa_k^{(2)} = \frac{1}{2}(\mathbf{m}_1^{k-1} + \mathbf{m}_1^k) \cdot (\kappa \mathbf{b})_k. \tag{3.36b}$$

The bending energy is then calculated according to the equation

$$E_\text{b} = \sum \frac{1}{2} \frac{EI}{\Delta l_k} \left[\left(\kappa_k^{(1)} - \bar{\kappa}_k^{(1)}\right)^2 + \left(\kappa_k^{(2)} - \bar{\kappa}_k^{(2)}\right)^2 \right], \tag{3.37}$$

where $\sum$ indicates summation over all the discrete bending energies, $\Delta l_k = \frac{1}{2}\left(\|\bar{\mathbf{e}}^{k-1}\| + \|\bar{\mathbf{e}}^k\|\right)$ is the Voronoi length for the k-th node, $\bar{\kappa}_k^{(1)}$ and $\bar{\kappa}_k^{(2)}$ are the material curvatures in the undeformed configuration, and $EI = \frac{\pi}{4} E r_0^4$ is the bending stiffness of the rod. For the rigid robotic head and 3D-printed plate, the value of bending stiffness EI is assumed to be so large that the curvatures at the rigid nodes remain nearly constant throughout the simulation.

Finally, the twisting energy is related to the relative rotation of the material frames between two adjacent edges, i.e. twist. The twist at the k-th node is

$$\tau_k = \theta^k - \theta^{k-1} + \Delta m_{k,\text{ref}}, \tag{3.38}$$

where $\Delta m_{k,\text{ref}}$ is the reference twist, which is the twist of the reference frame as it moves from the $(k-1)$-th edge to the k-th edge. The method by which we calculate this reference twist is detailed at the end of Section 4.2 in our previous work[125].

The twisting energy is then calculated according to the equation

$$E_t = \sum \frac{1}{2} \frac{GJ}{\Delta l_k} \left(\tau_k - \bar{\tau}_k\right)^2, \tag{3.39}$$

where $\bar{\tau}_k$ is the undeformed twist along the centerline, G is the shear modulus, and $GJ = \frac{\pi}{2} G r_0^2$ is the twisting stiffness. For the rigid components, the twisting stiffness is sufficiently large. Additionally, we assume that the material of flagella is nearly incompressible (i.e. Poisson's ratio $\nu = 0.5$), so $G = E/3$.

In summary, at each DOF $\mathbf{q}_k$, the elastic forces (affiliated with nodal position) and elastic moments (affiliated with the twist angles) are

$$F_k^{\text{E}} = -\frac{\partial E_{\text{E}}}{\partial \mathbf{q}_k}, \tag{3.40}$$

where $k = 0, 1, ..., \text{ndof} - 1$.

In a single elastic rod, each internal node is associated with a discrete bending and twisting energy. However, in this paper, the flagellated robot is represented as a network of rods and a "T"-shape joint node. This node is associated with multiple discrete bending and twisting energies(denoted as d.B.E in Fig. 3.3(b)). In order to simulate the dynamics of the robot, EOM for each DOF is required, which includes not only elastic forces but also external forces. Consequently, we illustrate how we model the external forces from the GM onto the robot in the next section.

3.4.2.3 Fully-implicit simulation

In order to simulate the locomotion of the robot, time is discretized into small time steps of size Δt. At each time step t_j, the DOF vector $\mathbf{q}$ is updated. Using the following equations of motion (EOM), the k-th DOF marches from $t = t_j$ to $t = t_{j+1} = t_j + \Delta t$:

$$\frac{m_k}{\Delta t}\left[\frac{q_k(t_{j+1}) - q_k(t_j)}{\Delta t} - \dot{q}_k(t_j)\right] - F_k^{\mathrm{E}} - F_k^{\mathrm{ext}} = 0, \tag{3.41}$$

where $q_k(t_j)$ and $\dot{q}_k(t_j)$ are the known DOF and velocities at the previous time step, respectively, E_{E} is the elastic energy evaluated at t_{j+1}, F_k^{ext} is the external force (or torque for twist angles) on the k-th DOF, and m_k is the lumped mass at the DOF. The external drag force exerted by the GM onto the flagella are detailed as Eqs. 3.2- 3.4 in Section 3.4.1.1.The external drag force and torque onto the head are calculated through Eqs. 3.23 and 3.24. Only the subscript k needs to be added to denote the force/torque at k-th node. Because the dynamics of this system is dominated by drag forces with negligible influence of inertia, the results presented are not mass dependent as long as low Reynolds number is maintained. Eq. 3.41 represents the collection of ndof equations that has to be solved to get the new DOF $q_k(t_{j+1})$. Essentially, this equation is a statement of “mass times acceleration = elastic force + external force" at the k-th DOF. Once $q_k(t_{j+1})$ is updated, the velocity at time t_{j+1} is determined as $\dot{q}_k(t_{j+1}) = (q_k(t_{j+1}) - q_k(t_j))/\Delta t$.

The EOM are solved using the Newton-Raphson method, i.e.

$$\mathbb{J}\Delta\mathbf{q} = \mathbf{f}, \tag{3.42}$$

where $\mathbf{f}$ is a vector of size ndof, the k-th component of this vector can be computed from Eq. 3.41, and $\mathbb{J}$ is a square matrix representing the Jacobian for Eq. 3.41. The (k, i)-th element in the Jacobian matrix is

$$\mathbb{J}_{ki} = \frac{\partial f_k}{\partial \zeta_i} = \frac{m_k}{\Delta t^2}\delta_{ki} + \frac{\partial^2 E_{\mathrm{E}}}{\partial q_k \partial q_i} - \frac{\partial F_k^{\mathrm{ext}}}{\partial q_i}, \tag{3.43}$$

where δ_{ki} represents Kronecker delta, the terms are gradient of inertia, elastic forces, and external forces, respectively, in the order shown in Fig. 3.41. Well-documented evaluation of the gradient of the elastic energy ($\frac{\partial E_{\mathrm{E}}}{\partial q_k}$) as well as its Hessian ($\frac{\partial^2 E_{\mathrm{E}}}{\partial q_k \partial q_i}$) can be found in Jawed *et al.*[83] and Bergou *et al.*[78].

3.4.2.4 Remarks on algorithm

Next, we summarize the novelty of our algorithm for simulating a robot with multiple flagella described above. Note that solving Eq. 3.42 is the most computationally expensive part of the entire simulation procedure. It is crucial to notice the sparsity of the Jacobian matrix, $\mathbb{J}$, and exploit its sparsity during the solution process[133], which helps reduce the computation cost. If the structure to be simulated is a single elastic rod (unlike a network of rods in this paper), the Jacobian is banded, and the time complexity of this algorithm is $O(N)$[78]. However, the Jacobian in this paper is not banded due to the presence of the joint node $\mathbf{x}_2$ in Figs. 3.3(**a**)(**c**). Referring to Fig. 3.3(**b**), the entire structure is a combination of stretching springs (e.g., one stretching spring is between $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$) and bending-twisting springs (e.g., one bending-twisting spring is between $\mathbf{x}_{k-1}, \mathbf{x}_k$, and $\mathbf{x}_{k+1}$). The stretching energy of each spring (Eq. 3.33) only depends on six DOFs (nodal coordinates of two nodes). For the stretching spring on edge $\mathbf{e}^k$, these DOFs are $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$. The gradient vector $\left(\frac{\partial}{\partial \mathbf{q}}\left[\frac{1}{2}EA\left(\epsilon^k\right)^2\|\bar{\mathbf{e}}^k\|\right]\right)$ has only six non-zero terms and the Hessian matrix $\left(\frac{\partial^2}{\partial \mathbf{q}\partial \mathbf{q}}\left[\frac{1}{2}EA\left(\epsilon^k\right)^2\|\bar{\mathbf{e}}^k\|\right]\right)$ has only 6×6 non-zero terms. As for the bending and the twisting energies of each spring (Eqs. 3.37 - 3.39), they are only dependent on eleven DOFs, i.e. $\mathbf{x}_{k-1}, \theta^{k-1}, \mathbf{x}_k, \theta^k$, and $\mathbf{x}_{k+1}$ in case of the spring at $\mathbf{x}_k$ in Fig. 3.3(**b**). As a result, the gradient vector and Hessian matrix corresponding to these two energies contain only eleven and 11×11 non-zero terms, respectively. The complete expressions for the gradient and Hessian terms are available in works[73, 83, 134]; the coding implementation can also be found in open-source repositories[103, 134, 110].

Last but not least, a crucial contribution of this study is the observation that the actuation of the robot, e.g., the rotation of the motor, can be readily accounted for in the framework above by updating the undeformed configurations with time. Normally, the undeformed configuration of a structure is fixed and assumed to remain constant throughout the simulation. The strains in undeformed configuration (e.g., $\bar{\kappa}_k^{(1)}, \bar{\kappa}_k^{(2)}, \bar{\tau}_k$ in Eqs. 3.36a, 3.36b, and 3.39) are used in calculation of elastic energies, their gradient (i.e. elastic forces), and Hessian. However, in this class of robots, the rotation of the motor causes the undeformed twist at the head node ($\mathbf{x}_1$) to vary with time. At each time step t_j, the robot is actuated by updating the undeformed twist of the first node according to the rotational speed of the motor, ω_T:

$$\bar{\tau}_0(t_j) = \omega_T t_j. \tag{3.44}$$

This actuation causes the rotations of the head (ω_h) and the flagella (ω_t) along opposite directions such that $|\omega_T| = |\omega_h| + |\omega_t|$. The total rotational speed, ω_T, is a control parameter in this study while the other one is the number of flagella, n.

3.4.3 Physical parameters

The following session will illustrate how we vary the robot design and experimental setup to realize designs 1 and 2. Remember that design 1 corresponds to the phenomenon that the propulsive speed of the robot decreases as the number of flagella increases, whereas design 2 reverses this phenomenon. Because the robot designs utilized in these two designs are different, we will provide the geometric and material parameters of the robots during experiments as follows. Hereafter, if we do not specify which design the parameters apply to, they apply to both. The number of flagella is $n = 2$ or 3 and each flagellum has a length of $L = 0.111$ m (design 1) and $L = 0.089$ m (design 2), a radius of $r_0 = 3.2$ mm, a density of 1000kg/m^3 (which is used to compute m_i in Eq. 3.41), a Young' modulus $E = 1.2 \times 10^6$ N/m^2, and Poisson' ratio $\nu = 0.5$ (incompressible material). Referring to Fig. 3.3, the radius of the robot head is $R_\text{h} = 0.02$ m (design 1) and $R_\text{h} = 0.015$ m (design 2), and the diameter of 3D-printed circular plate is $L_p = 0.04$ m (design 1) and $L_p = 0.03$ m (design 2). Recall that parameters C_1, C_2 and μ in Eqs. 3.25 and 3.27 are fitting parameters and will be fitted later in Section 3.6.3.1. The time step used in this paper is $\Delta t = 10^{-2}$ s and the length of each edge on flagella (in undeformed state) is $\|\bar{\text{e}}^k\| = 4.11$ mm. Convergence studies were performed to ensure that the size of temporal and spatial discretization (Δt, $\|\bar{\text{e}}^j\|$) has negligible impact on the simulation performance.

3.5 Experimental Design

As the flagellated locomotion is an intricate interplay between the elasticity of the flagella and hydrodynamic loading from the surrounding GM, our experiments considered both the robot design and selection of GM for propulsion. The robot is placed inside a transparent cylindrical tube filled with GM as shown in Fig. 3.4 (**a**). Before finalizing the cylindrical tube as the container, we initially placed the robot at the midline of the height in a rectangular tank of 50×50×40cm or a circular tube with a radius of 25mm, but the robot moved closer and closer to the surface of the GM because of a drag-induced lift[6] though the robot was manufactured as symmetrically as possible.

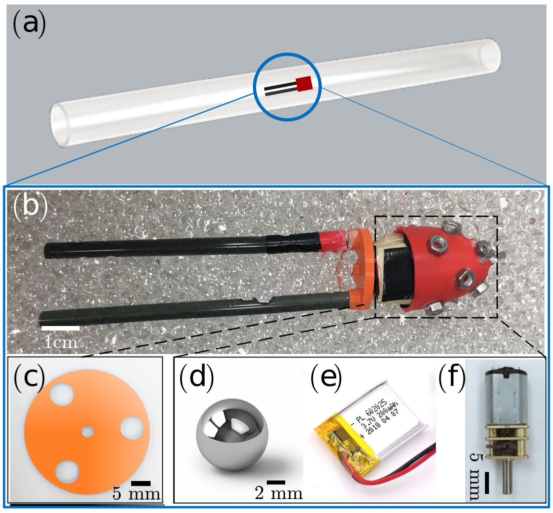

Figure 3.4: The compositive perspective of the experimental system for design 2. The one for design 1 can be found as Fig. 2 in our previous work[125]. Here, (**a**) the robot maneuvers inside the GM, which is contained inside a transparent cylindrical tube. (**b**) the robot is comprised of $n = 2$ flagella and a conic head. (**c**) A circular 3d-printed plate bridges the flagella with the head. The head consists of three components: (**d**) chrome steel bearing balls, (**e**) batteries and (**f**) a motor.

3.5.1 Robot structure

Given that Euler-Bernoulli beam theory in Section 3.4.1 predicts the presence of designs 1 and 2, experiments were performed to realize the prediction. Similar to the compact and lightweight robot used to realize design 1 (see Fig. 2 in our prior work[125]), the photograph in Fig. 3.4(**b**) depicts our robot for design 2 realization, which is propelled by n soft elastic flagella. It consists of four distinct components: multiple straight elastic flagella, Fig. 3.4(**c**) one 3D-printed circular plate attached to the motor shaft that protrudes from the robot head to hold flagella, and a head embedded with Fig. 3.4((**d**)) eight steel bearing balls (G25 Chrome Steel-AISI 52100) with an approximate diameter of 7.74 mm and mass of 2.06 grams, Fig. 3.4((**e**)) two 3.7V 200mAh rechargeable 502025 LiPo batteries (from Du litter energy battery), and Fig. 3.4((**f**)) one DC geared motor (from uxcell) with a nominal voltage of 3V, a power output of 0.35W, and a stall current of 0.55A. Inspired by natural creatures such as scorpions, snakes, and sand lizards, which have heads that are not spherical but more triangular, the head of our robot is designed as a cuboid (see Fig. 2 in Du *et al.*) or nose cone (see Fig. 3.4(**b**)). According to Euler-Bernoulli beam theory-based analyses, illustrated in Section 3.4.1.2, a smaller ω_h is advantageous for the realization of design 2. As a result, compared to the previous cuboid head design, the head in Fig. 3.4(**b**)) has not only a different shape but also higher inertia and a more frictional surface. Metal balls in Fig. 3.4(**(d)** are added to increase the inertia. Both head shapes contribute to the fluidization of the GM in front of the robot. To fabricate the elastic flagella, we applied the molding and casting techniques developed by Lazarus *et al.*[135] and Miller *et al.*[108]. The silicone-based rubber (vinylpolysiloxane from Elite Zhermack) injected into a PVC tube (e.g., from VWR International) mold and the inner and outer diameters of the PVC tube are the same as in our previous work[125], 3.175 and 6.35 mm. The radius of elastic flagella can be varied by using PVC tubes with different inner and outer diameters, allowing our robot platform to be effortlessly scaled up or down. The relationship between n and robot speed v is investigated in detail and will be explained in Section 3.6.1.

The spinning soft flagella propel the robot at a rotational speed of ω_t. Since the torque is system-balanced, the head is then actuated at a rotational speed of $-\omega_h$. The control parameter is the rotational speed of the flagella relative to the head, $\omega_T = \omega_t + \omega_h$. To vary ω_T, we replaced the geared motors while maintaining the other components the same instead of adding an encoder, which would significantly increase the robot size. The size and weight of all the motors are almost the same, (15-17) × 12 × 10 mm (L×W×H) and 13-15g even if they supply different ω_h. When necessary, we added electrical tapes around the motor to account for the small differences in size and weight for different motors. Moreover, the value of ω_t is proportional to the battery voltage and will decrease when the voltage drops. Subsequently, to maintain a constant ω_t, we fully charged the batteries before each experiment and recharged them after each approximately ten-minute experimental trial.

3.5.2 Granular medium

A granular medium has to be chosen to complement the untethered flagellated robot design. Too much internal friction between granules prevents the robot head from fluidizing the granules in front and propelling forward, while too little friction results in insufficient flagellar propulsion. Internal friction during locomotion is correlated with the phase transition of GM from solid to fluid, which is controlled by temperature and volume fraction (VF)[129]. The less effort required to alter the VF of GM, the more beneficial it is for studying the locomotion in GM. Apart from the reason mentioned above, we chose gel soil water crystal beads (from EBOOT) as the GM for their transparency. Robotic performance, e.g., the robot position and the rotational speeds of head and flagella, could be viewed outside the medium using a traditional digital camera (Nikon D3400). The beads had a diameter of 2.5 mm when dried and $d_f = 9.4 \pm 0.4$ mm when completely saturated with water. The size of beads is controllable by adjusting the time they are submerged in water and is reversible after dehydration. During experiments, the robot for design 1 ran in the GM with $d_f = 9.4 \pm 0.4$ mm whereas the one for design 2 ran in the GM with $d_f = 5.2 \pm 0.4$ mm (see Fig. 3.4(b)). The corresponding VF values were approximately 0.52 and 0.54. In the former case, v decreases as n decreases (design 1), while in the latter case, the converse (design 2) is true. Throughout all experiments in designs 1 or 2, the temperature variation of GM is kept to a maximum of 0.5 degrees, and the granule configuration, including size, density, and homogeneity, is maintained.

3.5.3 Locomotion in granular medium

We used a transparent acrylic cylindrical tube (from FixtureDisplays, Amazon) with an inner radius of 53 mm and an axial length of 1220 mm as the reservoir for the GM to conduct the locomotion experiments. The tube was filled with GM and placed horizontally, perpendicular to the direction of gravity. Before each experiment trial, we placed the robot at one end at the center of the cross-section of the tube. Since the force applied by the GM on the robot in the cylindrical tube is axially symmetric, the confinement effect of the relatively close bounding wall is canceled out. As the robot was positioned at the center and thus surrounded by compact granules, the drag-induced lift[6] was suppressed. Hence, the robot actuated by the rotating flagella would move in GM along a roughly straight line. The video camera captured this movement at a frame rate of 29.98 fps. In addition, both the head and flagella were marked with markers of a different color than the corresponding robot components. For instance, as shown in Fig. 3.4(**b**), a black marker was attached to the yellow-colored head, and a red marker was attached to one of the dark green colored elastic flagella. Together with the GM's transparency, this operation ensured the accuracy of counting the rotational speeds of the robot head (ω_h) and flagella (ω_t) and locating the position of the robot, s in recorded experimental videos.

3.6 Results and discussion

Recall that there are two control parameters in our study, the total rotational speed of the motor embedded in the head, ω_T, and the number of flagella, n. As shown in Fig. 3.1, when the motor is powered on, the actuation from the motor, ω_T, is split into a constant rotational speed of the head, ω_h, and flagella, ω_t, and the relationship $\omega_T = \omega_h + \omega_t$ is always satisfied ($\omega_T, \omega_h,$ and ω_t are all non-negative values). Because the system is torque-balanced, the head and flagella rotate in opposite directions. The net propulsive force is the residual of the propulsion generated by flagellar deformation (with the deflection w) and the drag on the head and flagella from the GM, propelling the entire robot forward (along with $-y$ direction) at speed v as illustrated in Fig. 3.2. In this section, we display the result comparison among the experiments and beam theory-based and DER-based numerical simulations. Since there are two designs, the sections below cover two sets of fitting parameters, experimental data, and numerical results.

3.6.1 Speed of the robot

The speed of the robot is the rate of change of its position with respect to the corresponding time interval, i.e. $v = \frac{\Delta s}{\Delta t}$. Except in some cases where sometimes the position of the robot remains unchanged (and "jamming" happens) when the number of flagella is large (e.g., $n = 4$ and $n = 5$), the position of the robot increases proportionally with time. Hence, v is a constant throughout each experiment trial. Since we apply RFT to model the hydrodynamic forces onto the robot from the GM, periodic "jamming" phenomenon cannot be captured by our simulator. We focus only on robots with $n = 2$ and $n = 3$ flagella that maintain a constant v over time. However, "jamming" (i.e. the viscosity of the GM in front of the head increases) can be effortlessly modeled as a function of the robot configuration and integrated into the DER-based simulator. We exhibit experimental results in Figs. 3.5(**a**)(**b**) (design 1) and (**c**)(**d**)(**e**)(design 2). To obtain each data point (($\bar{\omega}_T$, $\bar{v}$) in Fig. 3.5(**a**)(**e**) and ($\bar{\omega}_T$, $\bar{\omega}_h$) in Fig. 3.5(**b**)(**d**) where $\bar{v} = v\eta_p L^4/(EI)$ and $\bar{\omega}_h = \omega_h \eta_p L^3/(EI)$), we randomly selected three separate one-minute sequence from every ten-minute experimental trial, repeated the operation for three different experimental trials, processed the data for every one-minute sequence, and calculated the average and variance (of nine values). Notice that there are no data points between $\bar{\omega}_T = 0$ and $\bar{\omega}_T \approx 200$ in Figs. 3.5(**a**)(**b**) and $\bar{\omega}_T = 0$ and $\bar{\omega}_T \approx 18$ in Figs. 3.5(**c**)(**d**). This is because we observed from experiments that if ω_T is below a threshold (ω_T^s), the robot would stay stationary ($v = 0$) but started to move continuously otherwise. The threshold changed depending on the robot design, such as the shape, surface friction and inertia of the robot head and flagellar length. For example, $\omega_T^s \approx 50$ rpm ($\bar{\omega}_T \approx 200$) for the robot used in design 1[125] while $\omega_T^s \approx 33$ rpm ($\bar{\omega}_T \approx 18$) for the robot in design 2.

Note that the robots for designs 1 and 2 have distinct characteristics, such as the length of flagella L, and all data in Fig. 3.5 normalized and hence dimensionless. Normalization involves design parameters, L for example. As a result, though the ranges of $\bar{\omega}_T$ in Fig. 3.5(**a**)(**b**) and (**c**)(**d**) are seemingly different, they really overlap quite a bit ($\omega_T = 0 - 250$ rpm in Fig. 3.5(**a**)(**b**) and $\omega_T = 0 - 100$ rpm (Fig. 3.5**c**)(**d**)). To keep the same number of data points, and make ω_T low enough for Euler-Bernoulli beam theory to apply, we did not include any additional data in Fig. 3.5(**c**)(**d**).

3.6.2 Parameter fitting for simulations

Next, we will show the DER-based and beam-based numerical simulation results for locomotion modeling. Recall that C_1 (in Eqs. 3.25 and 3.23) and C_2 (in Eqs. 3.27 and 3.24) are two fitting parameters to account for the shape and surface roughness. Additionally, μ, the constant used to quantify the body-granule friction coefficient, is the third fitting parameter. As detailed next, C_1, C_2, and μ in both designs 1 and 2 were generated from experimental data of 2-flagellar locomotion, and the same values were used in simulations to predict 3-flagellar locomotion. The predicted outcomes were compared to the corresponding experimental data and validated the fitting process.

The normalized total rotational speed of motor, $\bar{\omega}_T$, compares the period of angular rotation to the elasto-viscous relaxation time, $\bar{T} = \eta_p L^4/(EI)$, where EI is the bending modulus of the flagella and η_p is the drag coefficient along the

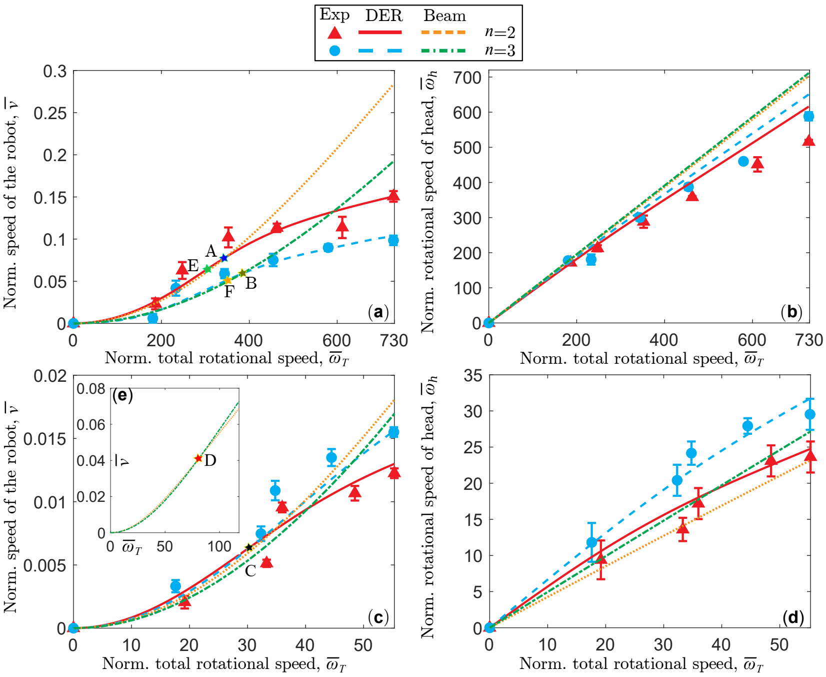

Figure 3.5: Comparison among the experimental data, DER-based simulation results and nonlinear Euler-Bernoulli beam(*NLB*)-based framework prediction. **(a)** **(c)** The plot of normalized robot movement speed $\bar{v}$ versus normalized total rotational speed of the robot $\bar{\omega}_T$; **(a)** Comparison of experimental data, DER-based simulation results and beam-based calculation in design 1, **(c)** comparison of experimental data and DER-based simulation outcomes of design 2, **(e)** beam-based framework prediction of the appearance of design 2 after point D; **(b)(d)** The plot of normalized rotational speed of the robot head $\bar{\omega}_h$ versus normalized total rotational speed of the robot $\bar{\omega}_T$; **(b)** Comparison of experimental data, DER-based simulation results and beam-based calculation in design 1, **(d)** comparison of experimental data, DER-based simulation outcomes and beam-based framework prediction result of design 2.

perpendicular direction [131], i.e. $\bar{\omega}_T = \omega_T \eta_p L^4/(EI)$. Similarly, the rotational speed of the head is normalized as $\bar{\omega}_h = \omega_h \eta_p L^4/(EI)$ and the velocity of the robot is normalized as $\bar{v} = v\eta_p L^3/(EI)$. In Figs. 3.5**(a)(b)** (for design 1), we plot the normalized speed of the robot, $\bar{v}$, versus the normalized total rotational speed of the motor, $\bar{\omega}_T$ and normalized rotational speed of the head $\bar{\omega}_h$ versus $\bar{\omega_T}$, respectively. All the experimental data, DER-based and beam-based simulation results for $n = 2$ and $n = 3$ are shown in the figures. The experimental data ($\bar{v}$ vs. $\bar{\omega}_T$ and $\bar{\omega}_h$ vs. $\bar{\omega}_T$) for $n = 2$ were adopted to determine the best fit values of the fitting parameters: $C_1 = 2.420, C_2 = 0.039$ and $\mu = 6.828$ (for design 1). These parameter values were then applied into the numerical simulators to simulate the locomotion performance of $n = 3$. In design 2, the same technique was applied in Figs. 3.5(**c**)(**d**) to find the fitting parameters that minimize the fitting error between experimental and DER-based simulation results : $C_1 = 28.750, C_2 = 0.938$ and $\mu = 2.125$ (for design 2). Figs 3.5(**a**)(**c**) demonstrate a high degree of agreement between experiments and DER-based simulations. The robot designs for designs 1 and 2 are distinct, and the relevant physical parameters were detailed in Section 3.4.3.

The following modeling assumptions can partially cause the slight disparity between the experimental and simulation results:

- In this fluid (i.e., the granular medium) structure (i.e., the robot) interaction modeling of robot locomotion, we assume that RFT can characterize the drag from the GM (the fluid).

- The structure model (DER) assumes that the flagella are infinitesimally thin elastic rods.

- It is assumed that the drag force acting on the head is considered to be linearly proportional to the velocity, and torque on the head is linearly proportional to its angular speed.

In addition, inevitably, there are experimental errors, structural defects introduced during fabrication, for instance. Nevertheless, the reasonable consistency between experiments and simulations suggests the validity of RFT in this context.

3.6.3 Speed vs. robot geometry

To quantify the effect of geometrical design, e.g., the number n and length L of flagella, we perform a parameter sweep along the angular velocity to systematically study the mechanical response of robots in both designs.

3.6.3.1 Speed vs. number of flagella

After performing the first round of experiments and data processing, we observed a counter intuitive phenomenon from Fig. 3.5(**a**) that the robot with two flagella ($n = 2$) moved faster than the one with three flagella ($n = 3$), at the same value of $\bar{\omega}_T$. Moreover, the relationship between $\bar{v}$ versus $\bar{\omega}_T$ is nonlinear. We then used the optimal set of fitting parameters described in Section 3.6.2 to perform the beam-based framework calculation; the results (($\bar{v}$ vs. $\bar{\omega}_T$ and $\bar{\omega}_h$ vs. $\bar{\omega}_T$)) are plotted in Figs. 3.5(**a**)(**b**). We denote points A and B in Fig. 3.5(**a**) as the intersection of beam-based and DER-based simulation results for $n = 2$ and $n = 3$ in design 1, respectively. As can be observed, the beam-based simulation results are close to DER-based simulation results when $\bar{\omega}_T$ is less than the value at point A ($n = 2$) and point B ($n = 3$), but deviate in other cases. All these observations indicate the significant flagellar deformation and the tight coupling between the head and flagella.

Nonetheless, the beam-based mechanics analysis framework in Section 3.4.1.1 tells us that if there is no head or the head is fixed along the x-axis (in Fig. 3.2), $\mathbf{F}_p$ will increase proportionally to the increase in n. The propulsive speed of the robot, as illustrated in Fig. 2, is the result of a complex battle between the projection of $\mathbf{v}_t$ along $-x$ (propulsion) and the sum of $\mathbf{F}_p$ in Eq. 3.23 and the projection of $\mathbf{v}_p$ along $-x$ (friction force). Section 3.4.1.2, on the other hand, elucidates that the effect induced by the head is one of the reasons for the difference between designs 1 and 2. Notice that $\bar{\omega}_h$ occupies more than 80% of $\bar{\omega}_T$ in Fig. 3.5(**b**), indicating the majority of the increment in $\bar{\omega}_T$ is spent growing $\bar{\omega}_h$. As a result, to achieve design 2, which is predicted by the beam-based framework, we have to slow down the rotational speed of the head, ω_h. Referring to Eq. 3.24, this suggests that while the torque keeps the same, the product of

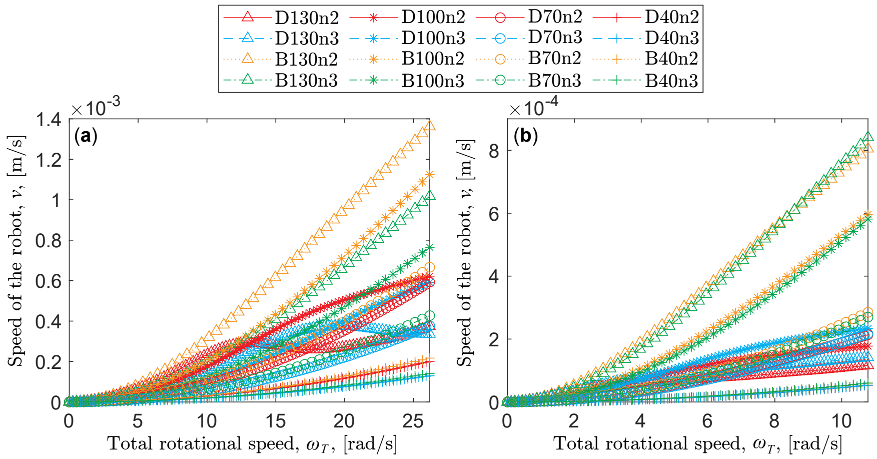

Figure 3.6: Comparison between DER-based simulation results and Euler-Bernoulli beam-based framework prediction in terms of flagellar number and length. (**a**)(**b**) The plot of robot movement speed $\bar{v}$ versus total rotational speed of the robot $\bar{\omega}_T$; **(a)** corresponds to design 1, while (**b**) corresponds to design 2. In the abbreviated legend, “D" stands for DER-related results and “B" represents results from the nonlinear beam theory-based framework; the number following “D"/“B" represents the length of each flagellum in millimeters(mm), “n2" denotes two flagella and “n3" denotes three flagella. For example, “D130n2" stands for the results of a DER-based simulation of a robot with two 130mm flagella.

C_2 and μ should be larger. As shown in Fig. 3.5(**d**), the ratio of $\bar{\omega}_h$ to $\bar{\omega}_T$ in design 2 drops to around 55%. Meanwhile, the product of fitting parameters C_2 and μ for design 2 is greater than the product for design 1 as mentioned in Section 3.4.3, which makes physical sense and highlights the predictive capability of our beam-based framework to help with design and motion prediction. Section 3.5.1 details how this was accomplished by updating the robot design in design 2 relative to the one utilized in design 1. We performed experiments with the updated robot design and the results are shown in Figs. 3.5(**c**)(**d**). Note that all of the results that pertain to beam theory in Fig. 3.5 are *NLB w/o head*, as illustrated in Section 3.4.1.2. As mentioned in the last session, when n equals two, the parameter fitting technique was leveraged to match the experiments and simulations. Point C in Fig. 3.5(**c**) stands for the conjunction of DER-based simulation results of $n = 2$ and $n = 3$. Experimental data point out that a two-flagellar robot moves slower than a three-flagellar robot while all other parameters remain constant. There is no evident difference in $\bar{v}$ between $n = 2$ and $n = 3$ simulation results for values of $\bar{\omega}_T$ smaller than the value at point C, which was captured by experiments. However, beam-based prediction does not show design 2 in Fig. 3.5(**c**) but it demonstrates design 2 when we increase the range of $\bar{\omega}_T$ as shown in Fig. 3.5(**e**). Furthermore, our goal of reducing the proportion of $\bar{\omega}_h$ taken from $\bar{\omega}_T$ is verified in Fig. 3.5(**d**). In Fig. 3.5(**e**), we display the beam-based framework predictions with the fitting parameters $C_1 = 28.750, C_2 = 0.938$ and $\mu = 2.125$ plugged in. Here, D represents the watershed point, beyond which the robot with $n = 3$ moves faster than the robot with $n = 2$ and vice versa. The beam-based prediction follows the same trend with the DER-based simulation in Fig. 3.5(**c**) even though the watershed points are drastically different. Also, the magnitudes of $\bar{v}$ and $\bar{\omega}_h$ predicted by the beam-based framework are a lot larger than those obtained from experiments and DER-based simulation. This further supports our assertion that robot locomotion involves large flagellar deformation and close interaction between the head and flagella. In summary, the simple beam-based analytical framework qualitatively captures the relationship between the speed of the robot and the number of flagella, though not quantitatively.

One more thing that needs our attention is the nonlinear increment of slope in Fig. 3.5(**a**). Both slopes for $n = 2$ and $n = 3$ initially increase in magnitude until they reach a point after which they begin to decrease in magnitude. We refer to the point for $n = 2$ as point E and the one for $n = 3$ as point F. The points E and F are conspicuous because, at these points, the unit magnitude increase in $\bar{\omega}_T$ turns into the steepest increase in $\bar{v}$, indicating the maximum efficiency of the robot locomotion is reached. The same phenomenon happens in design 2 (see Fig. 3.5(**c**)). Last but not least, point E is near point A and F is in the neighborhood of point B. This could mean that an increase in large flagellar deformation implies a decline in locomotion efficiency.

3.6.3.2 Speed vs. flagellar length

As previously stated, the data in Fig. 3.5 has been normalized. To help readers visualize the physical scenarios, we plot the simulation results from DER and *NLB* in Fig. 3.6, which depicts the relationship between the speed of the robot v in terms of each flagellar length, L, and the total rotational speed of the motor, ω_T. As illustrated in Fig. 3.6(a)(design 1), DER demonstrates that when the number of flagella, i.e. n is fixed, the distinction between DER and nonlinear beam theory (*NLB*) becomes more discernible as flagella become longer. When ω_T is small, DER results indicate that v grows as the value of L increases, but this trend breaks down when ω_T is big. In comparison, *NLB* behaves more consistently and linearly, i.e. when n is constant, v increases with the growth in L. These observations make sense in light of the fact that we previously acknowledged that DER is capable of capturing the nonlinearity of flagellar deformation, whereas *NLB* is not. The DER results displayed in Fig. 3.6(b)(design 2) show that as L increases, the difference between v of $n = 3$ and $n = 2$ does not always increase, but in the range of ω_T presented, v of $n = 3$ is greater than v of $n = 2$. In comparison to Fig. 3.5(c), *NLB* predicts the appearance of design 2 after a threshold is reached. However, this threshold becomes smaller when L is larger. In reality, we tested designs 1 and 2 on robots with $n = 4$ and $n = 5$ flagella. Nonetheless, “jamming" happened from time to time randomly. Hence, only simulation data are presented here.

3.6.4 Deflection of beam end

In the last session, we state that the simulation results from the beam-based framework (*NLB w/o head* design in Section 3.4.1.2) are qualitatively compatible with both experimental and DER-based simulation results. We also emphasize the importance of substantial flagellar deformation in relation to the performance of robot locomotion. Consequently, in Fig. 3.7, we compare beam-based deflection in three regimes, *LB*, *NLB*, and *NLB w/o head* as described in Section 3.4.1.2. The simulation parameters are identical to those for design 1 in Section 3.4.3 and to the fitting parameters for design 1 in Section 3.6.2. As seen in Eq. 3.18, the effect of the head in the flagellar deflection is ignored in regime *LB*, so the rotational speed of the flagella, ω_t is chosen as the independent variable. To see more noticeable difference in flagellar deflection among three regimes, we picked the scenario with larger flagellar deformation, design 1, and the corresponding fitting parameters to compute ω_h in *NLB* and *NLB w/o head* regimes. As seen in Fig. 3.7, when ω_t is relatively small ($\omega_t \lesssim 5$ rpm and $\bar{\omega}_T \lesssim 20$), the flagellar deformation drops to the *LB* regime introduced in Section 3.4.1.2 and the difference in deflection of the end of the beam (flagella) among the three regimes is subtle ($\lesssim 5\%$). Moreover, as can be observed in Fig. 3.7, the deflection of the end of the beam in regime *NLB w/o head* is almost always equal to that in *NLB* throughout the range of $\bar{\omega}_T$. This corroborates our statement that the head velocity-induced effect could be ignored in Section 3.4.1.2.

3.7 Conclusion

In this study, we designed a low-cost experimental setup, an untethered robot actuated by multiple soft flagella moves in granules. Meanwhile, we developed two numerical simulators, one based on Euler-Bernoulli beams and the other on discrete differential geometry (DDG) framework, to simulate the performance of the articulated locomotion in granular media (GM). Both numerical tools use resistive force theory (RFT) to model the drag force exerted by the GM on the robot and Stokes' law to model the external force/moment applied to the robot head by the GM. Initially, experiments unveiled a counterintuitive phenomenon: the robot's speed decreases as the number of flagella increases

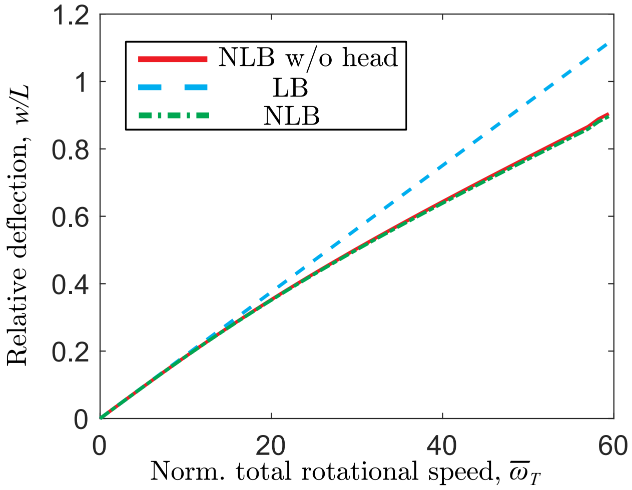

Figure 3.7: Comparison of the flagellar relative deflection w/L versus the normalized total rotational speed of the motor, $\bar{\omega}_T$, in regimes *NLB w/o head*, *LB*, and *NLB*. Physical parameters are identical to those in Section 3.4.3 for design 1 and to the fitting parameters in Section 3.6.2 for design 1.

(design 1). However, the beam-based simulator predicts the existence of the converse case (design 2) on the condition that the rotational speed of the head is suppressed to be zero. The robot design was then modified to slow down the rotational speed of the robot head.

The DDG-based simulator models the robot into a composition of Kirchhoff elastic rod and discretizes it into a series of mass-spring systems. As a consequence, the elastic energy of the robot structure is the linear sum of the discrete elastic energies associated with each spring, which include the stretching and coupled bending and twisting energies. At each time step, the equations of motion formulated can be summarized as follows: at each degree of freedom (DOF), the sum of elastic forces (i.e., the negative gradient of elastic energies) and external forces equals the lumped mass multiplied by the acceleration of that DOF. Especially, the actuation of the robot (i.e., the rotational speed of the motor) is modeled by a time-varying natural strain (twist) at the node standing for the head. This method enables us to simulate the robot locomotion fully implicitly.

Both beam-based and DDG-based frameworks capture designs 1 and 2 successfully. Although the beam-based simulator can quantify only the trend in both designs while the DDG-based method accurately reproduces experiments, it is still exhilarating because the simple beam theory-based analysis depicts the complicated locomotion system. Simultaneously, the discrepancy between beam-based prediction and experiments indicates the large flagellar deformation and nontrivial coupling between the head and flagella. Additionally, both simulators, validated experimentally, shed light on the highly nonlinear functional relationship between the locomotion performance, such as speed and efficiency, and its physical parameters, such as the number of flagella. The nonlinear dependency of the speed of the robot concerning the rotational speed of the motor necessitates the development of a design tool for optimal control of this class of robots. Last but not least, thanks to the simplicity of the beam-based analysis framework and the computational efficiency of the DDG-based simulator, they can be exploited to perform parametric studies and identify the optimal design and control this class of articulated robots as long as they locomote through the GM. We chose a smooth and soft GM with a large particle diameter.

3.8 Summary and Outlook

We have introduced a simple analytical framework based on Euler-Bernoulli beam theory to study the dynamics of rather complicated untethered robot with multiple soft flagella. In the meantime, we presented simulation results from a numerical framework that is based on DDG, given in Ch. 2. The qualitative system performance was successfully captured by the simple analytical framework, which serves as the ground truth for validating the results acquired from DDG-based simulations. One thing to emphasize is that such a simple analytical framework is able to anticipate the existence of two trends in the performance of the robot: alternative robot head designs can sometimes boost robot speed with an increase in the number of flagella and reduce robot speed in other situations. The simplicity of this analytical framework can be adaptable to investigate the locomotion in viscous fluids as well. Moving forward, it would be interesting to apply this method to predict limbed locomotion of soft robots in 3D.

Also, scaling power requirements for locomotions moving in harsher GM-like sand will be an interesting challenge.

CHAPTER 4

Simple flagellated soft robot framework for locomotion comprehension near air-liquid interface

A wide range of microorganisms, e.g. bacteria, propel themselves by rotation of soft helical tails, also known as flagella. Due to the small size of these organisms, viscous forces overwhelm inertial effects and the flow is at low Reynolds number. In this fluid-structure problem, a competition between elastic forces and hydrodynamic (viscous) forces leads to a net propulsive force forward. A thorough understanding of this highly coupled fluid-structure interaction problem can not only help us better understand biological propulsion but also help us design bio-inspired functional robots with applications in oil spill cleanup, water quality monitoring, and infrastructure inspection. Here, we introduce arguably the simplest soft robot with a single binary control signal, which is capable of moving along an arbitrary 2D trajectory near air-fluid interface and at the interface between two fluids. The robot exploits the variation in viscosity to move along the prescribed trajectory. Our analysis of this newly introduced soft robot consists of three main components. First, we fabricate this simple robot and use it as an experimental testbed. Second, a discrete differential geometry-based modeling framework is used for simulation of the robot. Upon validation of the simulation tool, the third part of this study employs the simulations to develop a control scheme with a single binary input to make the robot follow any prescribed path.

We introduce the motivation in § 4.1, followed by relevant literatures in

§ 4.2. We provide details on experiments and simulations in § 4.3. In § 4.4, we list the relevant physical parameters that affect the motion of the robot. Next, a simple control scheme that needs a single binary input for the robot to pursue the desired motion path is given in § 4.5. Eventually, § 4.6 concludes the paper and points out the potential future work. The content of this chapter has appeared in Ref. [66].

4.1 Motivation

Inspired by the inherent structural compliance of living creatures, soft swimming robots are designed to be lifelike and better emulate the movement of creatures in nature. Such soft robots often exploit structural deformation for functionality. Propulsion of bacteria by rotation of flexible tail-like *flagella* [16] is a source of inspiration for soft robot design. Flagella-propelled bacteria have been cited to be the "most efficient machines in the universe" [136] as they can swim at speeds up to tens of body lengths per second. Interestingly, large deformation and buckling in flagella can be used to control the swimming direction of bacteria [1]. Bacteria were found to swim in circles near close boundaries [127], walls for example. The underlying cause was found to be the different drag coefficents on the sides closer to and further from the boundary. Bacteria were then discovered to show circular motion as well near the air-fluid interface [137]. We aim to develop functional robot with soft flagella. While running the robot in viscous fluids and it is challenging to make the robot neutrally buoyant, we found that the robot always trying to turn rather than moving straight. Finally, it turned out that it was due to the uneven distribution of the viscosity of viscous fluids. This discovery benefited from our modeling and its integration into fast-running soft robot simulator, which will be illustrated in details in this chapter.

In this chapter, we adopt this paradigm of using deformation for functionality in soft structures. The number of flagella on our flagellated robot platforms is variable effortlessly. Not only can these platforms demonstrate exceptional and adaptable mobility for applications in biologically-inspired field robotics, but they can also serve as a testbed for elucidating the locomotion of soft biological species. Due to the existing limits in simulating the dynamics of soft material systems, the design and operation of soft robots are frequently based on arduous trial and error, and it can be challenging to reconcile qualitative findings to underlying concepts of kinematics, mechanics, and tribology. Advancement of the research in this field will considerably benefit from this computational framework for generic soft robot modeling that can aid in design, control, and experimental analysis. The speedy simulator as a tool for users to design practical cheap robots to function in fluid or granular medium.

The remainder of this chapter is organized as follows. We give relevant released work in §4.2. After that, in §4.3, we provide details on experiments and simulations in §4.3. In §4.4, we list the relevant physical parameters that affect the motion of the robot. Next, a simple control scheme that needs a single binary input for the robot to pursue the desired motion path is given in §4.5. Eventually, §4.6 concludes this chapter.

4.2 Literature review

The typical fluid flow around a swimming bacterium is of low Reynolds number, around 10^{-4}, where the viscous force dominates the inertial counterpart owing to the small size of bacterial cells. Scallop theorem [62] establishes that a motion invariant under time reversal cannot achieve net propulsion in this regime. Flagellar propulsion is a mechanism that overcomes this barrier. A flagellar bacterium consists of a cell body and one or more flagella; a rotary motor generates relative rotation between the cell body and the flagella. This rotation creates a net propulsive force forward. Typically, bacterial flagella are helical and their propulsion has been extensively studied [94]. In these studies, the viscosity and density of the fluid medium are usually assumed to be constant. However, in practice, such ideal fluid medium is not feasible. Near the boundary between fluid and air (or at the interface between two immiscible fluids), viscosity and density both vary spatially. We will show that this variation in viscosity can be exploited to build a very simple robot (composed of naturally `straight` flagella) that is capable of following any prescribed trajectory near the boundary. Fig. 4.1 shows snapshots of the robot moving along a triangular trajectory. This simple low-cost robot with a single binary control input can have applications in ocean oil spill cleanup, water quality monitoring, and pipe inspection. Interestingly, it has been reported that the motion of flagellated bacteria near air-liquid interface is circular [137]. If the angular velocity of the motor is constant in the robot introduced in this study, its trajectory is also circular.

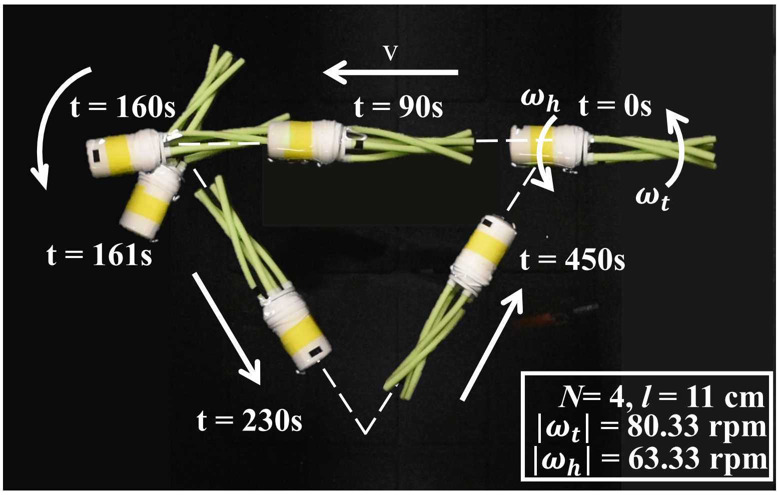

Figure 4.1: Snapshots of the robot (top view) moving along a triangular trajectory. Number of tails $N = 4$; tail length $l = 11$ cm; angular velocity of tail and head is 80.33 and 63.33rpm, respectively. The sign of the angular velocity is flipped at specific timepoints to achieve the triangular trajectory. Trajectory design is discussed in Section 4.5.

Due to the simplicity of the robot, it is amenable to miniaturization. A variety of robots have been developed in microscale for propulsion in marine environments. Microscale mobile robot fabrication, such as artificial bacterial flagella [138, 139], is restricted by the key bottleneck: miniaturization of power source and onboard actuation. The corresponding control strategies are often dependent on external magnetic field. While our prototype robot is centimeter-sized, we use a viscous fluid medium (glycerin) to maintain low Reynolds number. The findings are mostly presented in non-dimensional form and do not depend on the size of the system (as long as the Reynolds number is low). In the future, the simplicity of the proposed robot design can be exploited to develop untethered autonomous micro-robots.

In this work, we develop an economical centimeter-scale, simple-to-assemble, and self-contained robot comprised of a cylindrical head and a rotating disk containing two or more soft polymeric tails, actuated by the motor within the head. The motor generates a relative rotation between the head and the tails; therefore, the robot head and tails rotate in opposite directions. The magnitude of the angular velocities are determined by the torque balance of the system. The rotation leads to hydrodynamic (viscous) forces on the soft tails leading to elastic deformation; this deformation generates a net propulsive force that is used by the robot to translate in fluid. If the robot is in an infinite fluid bath, the direction of motion is parallel to the axis of the cylindrical head. However, in practice, such fluid bath with uniform viscosity and density is not practical. We exploit this variation and the robot (under constant angular velocity) moves along a line that is slanted with the axis of the head. Depending on the sign of the angular velocity, the robot moves clockwise or anti-clockwise along a circle. By periodically switching the sign of the angular velocity, the robot achieves a net translation along a straight line. We show that the robot can move along a straight line simply by switching the angular velocity; a constant angular velocity lets the robot make a turn. A simple control law is designed where the robot approximates a prescribed trajectory by a piece-wise linear function. To understand the physical principles, a simulation tool is developed where the structure is modelled using the Discrete Elastic Rods (DER) algorithm [78, 83] and the fluid forces are implemented using Resistive Force Theory (RFT) [92]. We show that elementary physics can be used to explain the propulsion mechanism of this robot.

Our contributions are as follows. We introduce a simple untethered soft robot that exploits variation in viscosity and elastic deformation in its tails to follow a pre-planned trajectory. A complete framework comprising of experiments, simulations, and controls is described to study the flagellated robot. The simulation tool is faster than real-time on a contemporary computer and can be used to generate data to formulate a control strategy. The physics behind the locomotion is elaborated. The simplicity of the robot and the small number of moving parts can eventually lead to miniaturization of this robot.

4.3 Methods

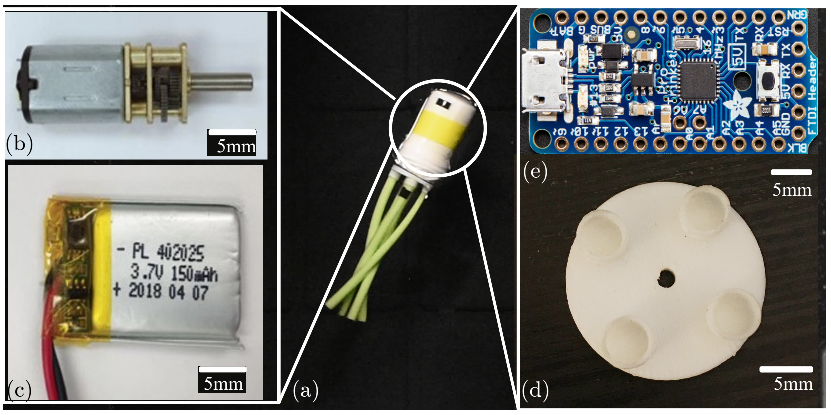

Figure 4.2: Compositive view of the experimental setup. (a) The robot with $n = 4$ tails in glycerin (top view). The head is comprised of (b) a DC geared motor, (c) a battery, (d) a 3D printed circular disc connecting the tails to the rotating motor shaft, and (e) a microcontroller to control the rotational speed of the motor.

In this section, we first introduce the framework of our low-cost simple robot design, consisting of a rigid head and multiple soft polymeric tails, which are called flagella in reference to the bacterial structure. The motor, which is em-

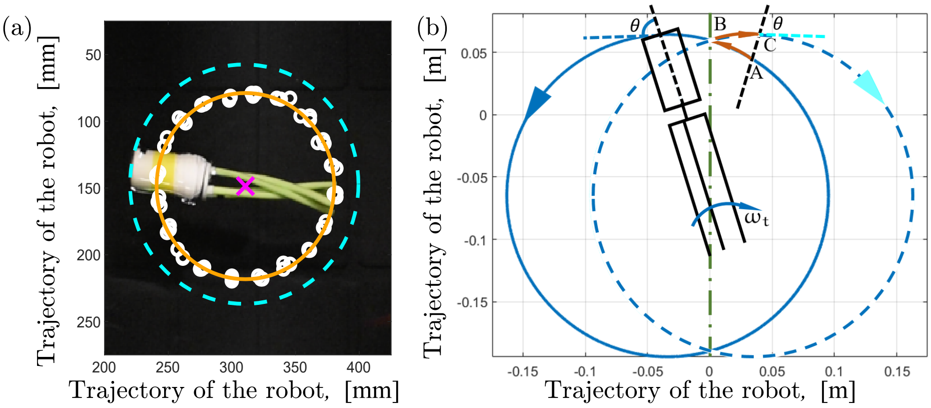

Figure 4.3: (a) Circular movement of the robot in glycerin. (b) Trajectories of the robot head (solid and dashed circles) when the robot starts from the same place but with different signs of the angular velocity of the motor.

bedded in the head with its shaft protruding out, actuates the tails to rotate along the long axis of the robot body. A series of experiments using this robot as a testbed were conducted with the number of tails, length of tails, rotation direction, and speed of head and tails as variables. Our DER-based simulation has successfully simulated the dynamics of a single rotating flexible flagellum [74]. We therefore aim to leverage the simulation to deepen our understanding of the hydrodynamics when the robot is near the air-fluid boundary and the robot's movement. Through parameter fitting, we match the experimental and simulation results obtained from our DER-based simulator in conjunction with RFT. It turns out that they match almost perfectly quantitatively.

4.3.1 Robot design and experimental setup

Motion of flagellar bacterial propulsion near the air-liquid interface is circular [137]. Our experiments are primarily designed to investigate the hydrodynamics and the movement of soft flagellated robot near the air-fluid boundary with the tail number, tail length, rotation orientation, head velocity, and tail velocity all changing. The design of the robot presented in this paper is a prototype of an untethered system that integrates a full set of functional elements: a palm-sized self-contained robot, low Reynolds number fluid (i.e. glycerin), and a rectangular tank. It is not yet entirely designed for any specific realistic application. Instead, it serves as the starting point for the development of a family of untethered robots with elastic tails in viscous fluid.

Glycerin with a density of 1.26 g/mL and viscosity $\mu_0 = 1.49$ Pa-s at $25°$C is selected as the fluid medium. The density of our lightweight and compact robot is slightly less than that of glycerin and it remains submerged near the air-fluid boundary. The robot in Fig. 4.2 is comprised of a head, multiple elastic tails, and a 3D-printed plate attached to the motor shaft to hold these tails. The robot head is a cylinder with a radius of 1.6cm and height of 6cm, which contains inside (b) one DC geared motor (uxcell) with 3V nominal voltage, 0.35W nominal power and 0.55A stall current (c) one 3.7V 200mAh rechargeable 502025 LiPo batteries, and (e) a 5V, 16MHz adafruit pro trinket. The motor is embedded inside the head with its shaft protruding out, and its rotation direction and speed are controlled by changing the PWM value in the program running in the trinket. The radius of the cylindrical head is $R = 1.6$cm. Some copper wires are attached to the outer surface of the robot head to make it balanced horizontally. During all experiments, the robot's tails are fully submerged in glycerin while 30% of the head is exposed to the air. In order to count the rotation speed of robot's head and tails clearly and conveniently, we stick a colored marker on one side of the robot's head and one of its tails. A digital camera (Nikon D3400) is used to record the robot's movement from the bird's eye view with its lens facing right down. The tails are made from Vinyl Polysiloxane using well established molding and casting techniques [72]. The Young's modulus is $E = 1.2$ MPa [72] and cross-sectional radius is $r_0 = 3.2$ mm. Since the material is near incompressible (Poisson ratio $\nu \approx 0.5$), the shear modulus is $G = E/3$. In order to generate enough of experimental data for parameter fitting in simulations, we vary the number of tails, $N = 2, 3, 4, 5$, and the length of tails, $l = 5, 7, 9, 11, 13, 15$ cm, with a DC geared motor mentioned above actuating the tails with a rated angular velocity of 150 rpm. Note that the actual angular velocity of the motor varies depending on the number of tails and is not necessarily 150 rpm, which ensures a Reynolds number $< 10^{-1}$.

4.3.2 Experiment trials

Images are extracted from the recorded experimental videos for data processing. Fig. 4.3(a) shows the trajectories of the tip of the robot head and the tip of a tail for a constant value of angular velocity of the motor ($\omega = 143.66$ rpm). The rotation directions of the robot head and tails around the long axis (i.e. axis of the cylindrical head) are opposite, as the system is untethered and torque-balanced. If the magnitude of the angular velocities of the head and the tail are ω_h and ω_t, respectively, and the angular velocity of the motor is ω, then $|\omega_h| + |\omega_t| = |\omega|$. The torque on the robot's head is balanced by the torque on the tails. As illustrated in Fig. 4.3(a), we also find that the whole robot circles around the vertical axis that is perpendicular to the air-fluid interface (y-axis in Fig. 4.5(a)) when its motor rotates unidirectionally, clockwise or counterclockwise. The open circle are the trajectory of the tip of robot tails; these points are fitted to the solid circle with the cross sign as the center. Similarly, the dashed circle is the circle fit to the trajectory of the tip of the robot head with the cross sign as the center.

Next, when we flip the sign of the angular velocity of the motor from the same initial orientation, the robot turns to circle around the vertical axis in the opposite direction. Specifically, as shown in Fig. 4.3(b), the solid and dashed circles are the trajectories of the robot head when the whole robot circles clockwise and counterclockwise (about the vertical y-axis), respectively. In both the cases, the initial orientation is along the dash-dot line. These two circles have the same radius but do not coincide with each other. To understand this, note that the angle, θ, between the long axis of the robot (dashed line in Fig. 4.3(b)) and the tangential direction of the circular trajectory is not 90°. As a result, if the tip of the robot's head starts to rotate from point A and rotates counter-clockwise along curve AB first and then rotates along curve BC after flipping the rotation direction of the motor, the robot will move forward and generate a translational movement. The net translation is the line segment AC. In summary, periodically switching the angular velocity ω of the robot between positive and negative values (keeping the same magnitude) results in a net straight-line trajectory. If the angular velocity of the robot about y-axis is ω_{yr}, the robot will make a turn by an angle α if the motor's angular velocity is maintained at ω for a period of α/ω_{yr}. Note that ω_{yr} is a function of various geometric, material, and fluid parameters (See Section 4.4). This is where a comprehensive simulation tool and a physics-based understanding, to be discussed in the next section, can guide us to develop a control law.

4.3.3 Numerical Simulation

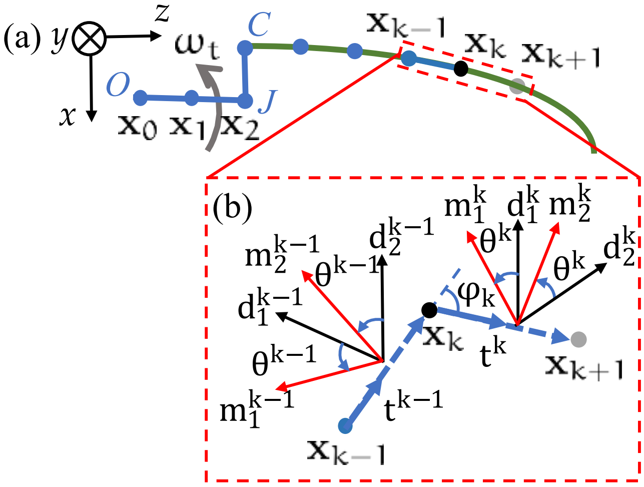

Figure 4.4: (a) Discrete representation of the soft robot. (b) Three nodes, two edges, and the associated reference and material frames.

We develop a numerical simulation based on the Discrete Elastic Rods (DER)

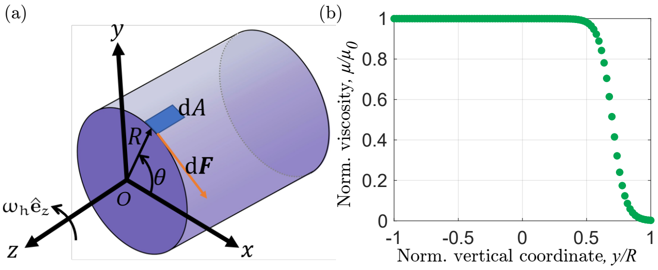

Figure 4.5: (a) Schematic showing drag $\mathrm{d}\mathbf{F}$ on the cylindrical head when the head is rotating along its long axis (z-axis). (b) Normalized viscosity as a function of normalized vertical coordinate $\frac{y}{R}$.

method; a tutorial exposition to DER can be found in Ref. [83]. Since the simulation framework we developed in this chapter is very similar to the one in Ch. 2, please refer to § 2.4. The key difference and thus contribution in numerical methods of the present chapter and previous work (§ 2.4) is to model the effect of open boundary onto the locomotion. In DER, the robot is discretized into n nodes, as shown in Fig. 4.4(a). There are three nodes ($\mathbf{x}_0, \mathbf{x}_1$, and $\mathbf{x}_2$) on the head and equal number of nodes on each tail (for illustration purposes, only one leg is shown in Fig. 4.4(a)). It is necessary to have three nodes on the head to model actuation using a natural twist that varies with time (more on this later in this section). Two adjacent nodes, $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, are connected by an *edge*, $\mathbf{e}^k = \mathbf{x}_{k+1} - \mathbf{x}_k$. Three adjacent nodes and the intermediate two edges form an elastic rod. Partial discretization is shown for illustration purpose only. O is the center of the robot head while OJ stands for the protruded motor shaft, JC denotes half of the 3D printed circular disc, and J is the center of the 3D printed disc.

Fig.4.4(b) is a closeup showing how DER works for the elastic rod comprised of three nodes (blue, black, and grey dots in the dashed rectangle), $\mathbf{x}_{k-1}$, $\mathbf{x}_k$, and $\mathbf{x}_{k+1}$, and two edges $\mathbf{e}_{k-1}$ and $\mathbf{e}_k$ where $\mathbf{e}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ $(1 \leq k \leq N-1)$. Each edge is associated with an orthonormal reference frame, $\{\mathbf{t}^k, \mathbf{d}_1^k, \mathbf{d}_2^k\}$, and an orthonormal material frame, $\{\mathbf{t}^k, \mathbf{m}_1^k, \mathbf{m}_2^k\}$. Both of these frames are *adapted*, i.e. the first director $\mathbf{t}^k$ is the unit vector along the edge $\mathbf{e}^k$. The simulation moves forward with time taking small steps of Δt. During the simulation loop, the reference frame is updated through parallel transport in time. We omit the details of time parallel transport; Ref. [83] includes a pedagogical introduction to this method. Since the material frame shares a common director $\mathbf{t}^k$ with the reference frame, only a scalar angle θ^k (see Fig.4.4(b)) is necessary to describe the material frame. The degrees of freedom (DOF) vector of the robot is then $\mathbf{q} = [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \theta^0, \theta^1, \ldots, \theta^{m-1}]$, where n is the number of nodes and m is the number of edges. The total number of DOF is `ndof` $= 3n + m$.

The core of the simulation is a solver (integrator) of following equations of motion.

$$m_i \ddot{q}_i = F_i^e + F_i^h, \tag{4.1}$$

where m_i is the lumped mass at the i-th DOF, q_i is the i-th element of the DOF vector, F_i^e is the i-th element of the `ndof`-sized elastic force vector $\mathbf{F}^e$, and F_i^h is the i-th element of the `ndof`-sized external (hydrodynamic) force vector $\mathbf{F}^h$. Hereafter, dot $(\dot{\ })$ represents derivative with respect to time.

First, we describe the elastic forces. The elastic energy is composed of three modes: stretching, bending, and twisting. Each component is given by

$$\begin{aligned} E_k^s &= \frac{1}{2} EA \left(\frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{\bar{\mathbf{e}}^k} - 1 \right)^2 |\bar{\mathbf{e}}^k| \\ E_k^b &= \frac{1}{2} EI (|\kappa_k - \kappa_k^0|)^2 \frac{1}{l_k} \\ E_k^t &= \frac{1}{2} GJ (|\tau_k - \tau_k^0|)^2 \frac{1}{l_k} \end{aligned} \tag{4.2}$$

where $EA = E\pi r_0^2$, $EI = \pi E r_0^4/4$, $GJ = \pi G r_0^4/2$, $|\bar{\mathrm{e}}_k|$ is the length of edge $\mathbf{e}_k$ in the undeformed state, κ_k is the curvature vector at node $\mathbf{x}_k$ (related to the *turning angle* ϕ_k in Fig. 4.4(b)) while κ_k^0 is the undeformed curvature for the same node, τ_k is the integrated twist (related to $\theta^{k+1} - \theta^k$ in Fig. 4.4(b)) while τ_k^0

represents the natural twist at node $\mathbf{x}_k$, and $l_k = (|\bar{\mathbf{e}}_{k-1}| + |\bar{\mathbf{e}}_k|)/2$ is the Voronoi length of the node in undeformed state. Moreover, integrated twist at time t_i can be expressed as $\tau_k^i = \delta\theta_k + \underline{\tau}_k(t_i)$ where $\delta\theta_k = \theta_k(t_i) - \theta_k(t_{i-1})$ and $\underline{\tau}_k$ is the reference twist [78]. The total elastic energy is $E^e = \sum_k E_k^s + \sum_k E_k^b + \sum E_k^t$. The elastic force vector is simply $\mathbf{F}^e = -\frac{\partial}{\partial \mathbf{q}} E^e$.

It is important to note that the elastic stiffness parameters are not the same throughout the rod. These parameters for the soft tails are described in the previous section. However, as the head and disc are rigid (OJC portion in Fig. 4.4(a)), we set the values of EA, EI, GJ on this segment to be very large so that no deformation takes place.

In order to mimic actuation by the motor rotating at an angular velocity $\omega(t)$, we set the natural twist of the second node (τ_1^0) at the beginning of each iteration to be

$$\tau_1^0(t) = \omega(t). \tag{4.3}$$

Next, we describe the formulation of the hydrodynamic force (i.e. viscous drag) vector $\mathbf{F}^h$.

Hydrodynamic force on robot head: The cylindrical head with radius R is translating with a velocity $\dot{\mathbf{x}}_1$ and rotating about its axis with an angular velocity of $\omega_h \equiv \dot{\theta}^0$. The hydrodynamic drag on a cylinder (external force on $\mathbf{x}_1$ in DER) can be decomposed into two parts:

$$\mathbf{F} = \mathbf{F}_v(\dot{\mathbf{x}}_1) + \mathbf{F}_\omega(\omega_h), \tag{4.4}$$

where $\mathbf{F}_v(\dot{\mathbf{x}}_1)$ and $\mathbf{F}_\omega(\omega_h)$ are the drag forces due to translation and rotation, respectively. The former quantity is a function of the translational velocity, $\dot{\mathbf{x}}_1$, of the head while the latter is a function of the angular velocity, ω_h.

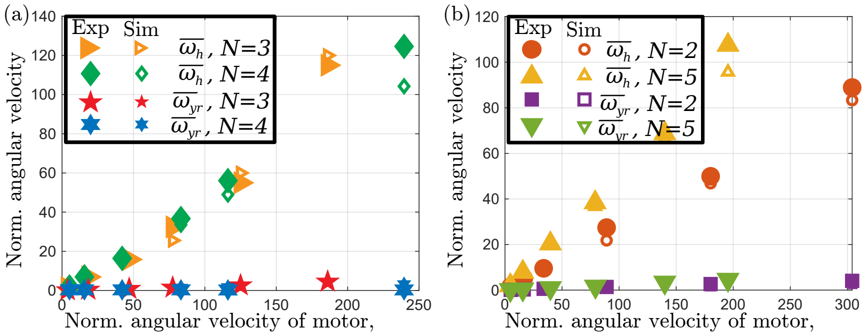

Figure 4.6: (a) Experimental and simulation data on $\bar{\omega}_h$ and $\bar{\omega}_{yr}$ as functions of the normalized angular velocity of the motor, $\bar{\omega}$, at two different values of the number of tails ($N = 3$ and 4). This data are used to estimate C_t, C_r, C_{yr}. (b) Same data from experiments and simulations but with $N = 2$ and 5. In simulations, the estimated values of C_t, C_r, C_{yr} from (a) were used.

Drag due to the translation on a sphere is given by Stokes' law as

$$\mathbf{F}_v = -6\pi\mu_0 R\dot{\mathbf{x}}_1, \tag{4.5}$$

where R is the radius of the spherical object and $\dot{\mathbf{x}}_1$ is the velocity of the object relative to the fluid. Since the robot head is cylindrical and there is no closed form expression for drag on a cylinder, we use a numerical coefficient C_t (to be evaluated through data fitting) to express the drag as

$$\mathbf{F}_v = -C_t 6\pi\mu_0 R\,\dot{\mathbf{x}}_1. \tag{4.6}$$

For the robot studied in this paper, the viscosity varies along the vertical direction. Fig. 4.5(a) shows a schematic of the head and $x - y - z$ is the body fixed frame. The vertical direction y is perpendicular to the air-fluid interface. This interface where the viscosity changes rapidly from μ_0 (fluid) to 0 (air) is at $y \sim R$. The fitting parameter C_t in Eq. 4.6 also depends on the functional relationship between viscosity μ and vertical position y.

Critical to the propulsion of this soft robot is the drag force $\mathbf{F}_\omega$ originating from this variation in viscosity.

The viscosity μ is a function of the y-coordinate, i.e. $\mu = \hat{\mu}(y)$. The specific functional form of μ does not matter as we will be using fitting parameters. We pick the following expression for viscosity,

$$\mu = \mu_0 \frac{1}{1 + \exp\left(k\left(\frac{y-h}{R}\right)\right)}, \tag{4.7}$$

where h is the location (close to the inter-medium boundary) where glycerin starts to mix with air and k is the “sharpness" of the transition from $\mu = \mu_0$ to $\mu = 0$. In Fig. 4.5, we used $h = 0.7R$ and $k = 20$. Note that Eq. 4.7 is an analytical approximation to the Heaviside function.

Referring to Fig. 4.5, a small area element $\mathrm{d}A = R\mathrm{d}\theta\mathrm{d}z$ on the surface of the cylinder rotating at an angular velocity of ω_h (along the z-axis) is picked. The magnitude of the force on this infinitesimal element is

$$dF \sim \mu\omega_h \; R\mathrm{d}\theta\mathrm{d}z, \tag{4.8}$$

with its direction along negative $\hat{\mathbf{e}_\theta}$, which is the unit vector along the tangential direction. The force along the x axis is

$$dF_x \sim dF \sin\theta = \mu\omega_h \sin\theta \; R\mathrm{d}\theta\mathrm{d}z, \tag{4.9}$$

and the force along the y axis is

$$dF_y \sim -dF \cos\theta = -\mu\omega_h \cos\theta \; R\mathrm{d}\theta\mathrm{d}z. \tag{4.10}$$

The horizontal component (x axis) of the total force on the cylinder with length L is obtained by integrating dF_x;

$$F_x \sim \int_{z=0}^{L} \int_{\theta=0}^{2\pi} \mu\omega_h \sin\theta \; R\mathrm{d}\theta\mathrm{d}z, \tag{4.11}$$

$$\implies F_x = -1.403\mu_0\omega_h RL. \tag{4.12}$$

Since we do not know the exact form of $\mu = \hat{\mu}(y)$, a fitting parameter C_{yr} is used and Eq. 4.12 can be reformulated as

$$F_x = -C_{yr}\omega_h\mu_0 RL. \tag{4.13}$$

The vertical component (y axis) of the total force is

$$F_y \sim -\int_{z=0}^{L} \int_{\theta=0}^{2\pi} \mu\omega_h \cos\theta \; R\mathrm{d}\theta\mathrm{d}z = 0, \tag{4.14}$$

i.e. there is no vertical hydrodynamic force.

In summary, the hydrodynamic drag on the head (applied on the center of mass of the head) due to rotation (ω_h) is

$$\mathbf{F}_\omega(\omega_h) = -C_{yr}\omega_h\mu_0 RL\hat{\mathbf{e}_x}. \tag{4.15}$$

The hydrodynamic moment on the head (applied on the first edge θ^0 in DER) is

$$F_\omega = -C_r 8\pi\omega_h\mu_0 R^3, \tag{4.16}$$

where C_r is a numerical prefactor (fitting parameter in our study). Note that if the head was spherical, we would have $F_\omega = -8\pi\omega_h\mu_0 R^3$.

Hydrodynamic force on tails: The hydrodynamic force on the nodes belonging to the soft tails is formulated using RFT [92, 100]. The force on node

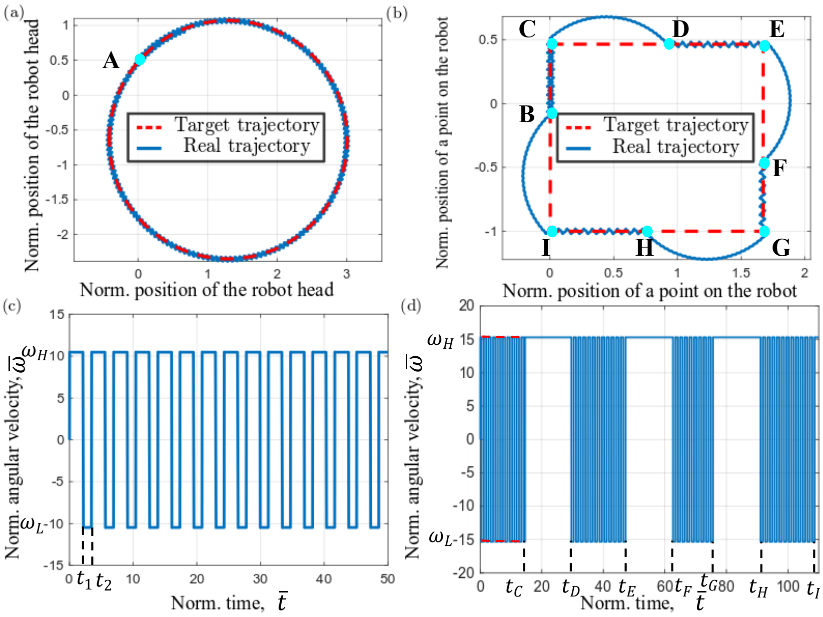

Figure 4.7: (a) Prescribed circular trajectory (dashed line) and real path (solid line) viewed from top. (b) Prescribed rectangular path (dashed line) and the real path of the robot (solid line). In (a) and (b), position has been normalized by tail length, l. Corresponding control signals (angular velocity) with time for (a) circular and (b) square trajectories.

$\mathbf{x}_k$ (moving with velocity $\dot{\mathbf{x}}_k$) is

$$\mathbf{F}_{\mathrm{RFT}} = -\mu_{||} (\mathbf{t} \cdot \dot{\mathbf{x}}_k)\mathbf{t} l_k - \mu_{\perp}[\dot{\mathbf{x}}_k - (\mathbf{t} \cdot \dot{\mathbf{x}}_k)\mathbf{t}] l_k, \tag{4.17}$$

where $\mathbf{t}$ is the tangent vector on node $\mathbf{x}_k$, l_k is the Voronoi length (described earlier), and $\mu_{||} = 2\pi\mu_0/[\log(l/r_0) - \frac{1}{2}]$ and $\mu_{\perp} = 4\pi\mu_0/[\log(l/r_0) + \frac{1}{2}]$ are the RFT drag coefficients along the tangential and perpendicular directions.

The expressions of the forces in Eqs. 4.4, 4.16, and 4.17 are used to populate the external force vector $\mathbf{F}^h$ of size `ndof`.

Parameter fitting: As mentioned in Section 4.3.2, the tail length at each tail number varies from $5 - 15$ cm in experiments and we have 6 data-points for a specific tail number. Now that the hydrodynamic forces on the robot have been analyzed, there are three numerical prefactors (C_t, C_r, and C_{yr}) that need to be obtained from data fitting. Our fitting strategy is to take the experimental data for $N = 3$ and $N = 4$ tails and find the set of parameters (C_t, C_r, and C_{yr}) that result in the best match between experiments and simulations. To evaluate the fitness of a given set of parameters, we use the following metrics: (i) angular velocity of the head, ω_h and (ii) angular velocity of the robot around the vertical axis, ω_{yr}. In the experiments, we used the same motor with a full PWM value signal sent by the microcontroller. As we vary the length of the tails, the actual angular velocity of the motor, ω, changes. Fig. 4.6(a) presents ω_h and ω_{yr} as functions of ω. All other parameters (except l) are kept fixed.

The best fitting parameter set that realizes the smallest error, 14.8%, between experimental and simulation results in Fig. 4.6(a) is $C_t = 4.0 \pm 0.33, C_r = 2.06 \pm 0.156, C_{yr} = 6.0 \pm 0.5$. After the application of this fitting parameter set, the predicted simulation results for 2 and 5 tails turn out to match well with experiments with a 10% error as shown in Fig. 4.6(b). This agreement indicates that the physics of this robot has been captured using the hydrodynamic model presented earlier in this section.

We then apply the attained fitting parameters to predict values of ω_h and ω_{yr} when the number of robot tails is 2 and 5, as in Fig. 4.6(b). During the experiments, even though we use the same motor with a full PWM value signal sent by the microcontroller, once the tail number and tail length change, the output torque of the motor will vary as well, causing a change in the actual angular velocity. This means that our simulator can only generate the same number of datapoints as experiments to fit both results because there are two variables changing in the simulation input at the same time, l and ω. As a result, we only represent the simulation outcome by discrete datapoints in Fig. 4.6.

4.4 Parameter space

In this section, we list the relevant physical parameters that affect the motion of the robot. Note that there is an intrinsic time-scale [131] in this problem $\mu l^4/EI$. We use this time-scale to normalize various quantities (overbar represents normalization), e.g. $\bar{\omega} = \omega \mu l^4/EI$ is normalized angular velocity of the motor and $\bar{t} = tEI/[\mu l^4]$ is normalized time. The set of physical parameters that describe the system is $\{C_t, C_r, C_{yr}, l/R, L/R, l/r_0, \bar{\omega}, N\}$; these are the inputs to our simulation tool. The angular velocity is a function of time. The simulation outputs the trajectory of the robot with time. In the next section, we will address the inverse problem where $\bar{\omega}$ has to be computed, given a prescribed trajectory.

The output of the simulation (i.e. trajectory of the robot) when $\bar{\omega}$ is constant with time can be encapsulated with two parameters: $\bar{\omega}_{yr}$ and R_{yr}/l, where R_{yr} is the radius of the circle in Fig. 4.3(b). If the sign of the angular velocity is flipped every T seconds, the output can be captured by θ (Fig. 4.3(b)) and effective speed v (distance traveled along a straight line per unit time).

4.5 Control for path planning

In this section, we present two examples of the inverse problem where the trajectory (circle and square) is prescribed and the angular velocity of the motor has to be computed. All the physical parameters are the same as those in Section 4.3: $\{C_t, C_r, C_{yr}, l/R, l/r_0\} = 3.0, 2.8, 2.0, 6.875, 34.375$. The intrinsic time-scale is $\mu_0 l^4/(EI)$ = 2.207 seconds. Number of tails is $N = 2$.

In the first example in Fig. 4.7(a), the robot starts from point A and needs to follow a circular path (the radius of this circle is not equal to R_{yr}). Here, we introduce one of the simplest possible control schemes (Fig. 4.7(c)) where the angular velocity of the motor is either ω_H or ω_L ($\omega_H = -\omega_L$). We rather arbitrarily choose $\omega_H = 10$ (and $\omega_L = -10$). The remaining task is to compute the timepoints ($t_1, t_2, \ldots$ in Fig. 4.7(c)) at which the angular velocity has to be switched. To make the robot swim along a circle, the motor first rotates counterclockwise for normalized duration t_1, causing the robot to traverse a clockwise arc of angle θ. Then, the motor rotates clockwise for a marginally shorter duration $t_2 - t_1$, causing the robot to move through a slightly smaller arc of angle $\theta - \Delta\theta$. This input, alternating between a short counterclockwise rotation and a longer clockwise rotation, is repeated to form a zig-zag circular path in Fig. 4.7 (c).

In the second example in Fig. 4.7(b), the robot has to follow a rectangular trajectory. It is obvious that the robot will follow a straight line if $\bar{\omega}$ switches between ω_H and ω_L every T seconds. In Fig. 4.7(d), this is the case when the robot has to follow a straight line ($0 \leq \bar{t} \leq t_C$, $t_D \leq \bar{t} \leq t_E$, $t_F \leq \bar{t} \leq t_G$, $t_H \leq \bar{t} \leq t_I$). Once the robot arrives at one corner of the rectangular path, C as displayed in Fig. 4.7(b), the motor keeps rotating in one direction (time from t_C to t_D) until the robot finishes turning 90° and it reaches point D. The same protocol of turning is applied at points E, G, and I.

4.6 Conclusions and future work

In summary, we built a framework comprised of a simple untethered soft robot, a numerical simulator, and a simple control scheme that enables the robot to follow any prescribed trajectory. Our low-cost, easy-to-assemble, untethered soft flagellated robot offers a convenient and practical platform for users to study hydrodynamics near the air-liquid interface in viscous fluid. The robot is able to follow any prescribed 2D trajectory through a simple control method with a single binary input. In addition to the low cost, this simplicity points to possible miniaturization of the robot. As the size of the robot gets smaller, viscous effects start to dominate and the flow approaches low Reynolds number. The propulsion mechanism of the proposed robot relies on low Reynolds assumption and provides a blueprint for micro-robots.

A future direction of research is to exploit the efficiency of the simulator to train a neural network that models the input - output relationship of this problem. In reality, we have publications to be pushed out already. That neural network then can serve as an inverse design to formulate the control signal, given the prescribed trajectory. For example, if the user requires a robot capable of swimming near the air-fluid interface at a speed of approximately 1 cm/s, our network is able to output the corresponding physical and material parameters, such as the number of flagella required, their cross-sectional radius, lengths, and Young's modulus. Further, our framework may be used to investigate if imperfection can be utilized for functionality, i.e. whether the fact that certain flagella are shorter than others can help the robot steer or not.

CHAPTER 5

Like bacteria: untethered underwater robots exploiting flagellar instability for steering

Uni-flagellar bacteria are discovered to steer by buckling their hook, the flagellar base. This mechanism has never been validated by robotic experiments due to the coupled difficulty of making underwater robots compact, neutrally buoyant and functional with a compliant flagellum. We are the first to verify it: when a helical flagellum rotates along its axis, the robot moves forward straight and steers when the flagellum buckles in a viscous fluid. Moreover, we use our untethered soft robot to investigate the relationship between the flagellar buckling threshold and its geometrical and material properties. Simultaneously, we develop a discrete differential geometry (structure) - Regularized Stokeslet Segments method (flagellar hydrodynamics) - Stokes' law (head hydrodynamics) simulator to model the locomotion. Simulations are validated against experiments. We also release a gimbal-based design (tethered) and a simple but practical approach to characterize the mapping between buckling thresholds of tethered and untethered robots to make flagellar robot design adaptable. This approach alleviates the pain of finding a motor whose rotational speed covers the buckling threshold with enough torque by saving roughly $\frac{2}{3}$ of experimental time. Furthermore, the robot is too underactuated [140] to be controlled to follow a 3D trajectory robustly by flagellar buckling alone; hence, we propose a “mass-transformer" mechanism to aid in out-of-plane movement. Overall, our comprehensive framework including physics-based modeling, motion planning, and control is a recipe for untethered flagellar robots that can reach any 3D destination.

Our model system is comprised of a spherical rigid head and a helical elastic flagellum.

The locomotion is an intricate interplay between the elasticity of the flagellum, the hydrodynamic loading, and the flow generated by the moving head. This flagellated system follows a straight path if the angular velocity of the flagellum is below a critical threshold. Buckling ensues in the flagellum beyond this threshold angular velocity and the system takes a nonlinear trajectory. We consider the angular velocity as the control parameter and solve the inverse problem of computing the angular velocity, that varies with time, given a desired nonlinear trajectory. Our results indicate that bacteria can exploit buckling in flagellum to precisely control their swimming direction. We describe the underlying motivation and relevant literature in § 5.1. The numerical simulation procedure is detailed in § 5.4.2. The buckling instability of helical filament and a data-driven approach for the trajectory design are in § 5.4. We conclude a summary and point out potential directions for future research in § 5.7. The content of this chapter has appeared in Ref. [75].

5.1 Motivation

Bacteria achieve propulsion in Stokes regime fluid flow (Reynolds number $\ll$ 1) [141] by rotating one or more helical slender filaments, i.e., flagella. This propulsion is the result of a rather complicated fluid-structure interaction (FSI) between the flagellar structural flexibility and the viscous forces imposed by the surrounding flow. This FSI problem may induce geometrically nonlinear deformation, which can be exploited for functionalities, such as turning [1], bundling [142, 143], tumbling [144], and polymorphic transformations [145]. The bundling and unbundling between multiple flagella contribute to bacterial reorientation, which is essential for multi-flagellar bacteria to climb chemical gradients in search of nutrition or escape toxins [146].

5.2 Literature review

Although flagellar propulsion has been extensively studied, it was not until recently that the instability of the hook was discovered to be utilized to steer by uni-flagellar bacteria such as *Vibrio alginolyticus* [1]. We name this mechanism as “buckle-to-turn”. Previous studies investigating simulations [26, 72] demonstrate that the hydrodynamic force of the flow can cause the flagellum or hook to buckle; however, an untethered robot with a flexible flagellum that can replicate bacterial “buckle-to-turn" has not yet been described, let alone such kind of robots that can achieve a preplanned 3D trajectory. A scaled-up millimeter or centimeter robot that could do so seamlessly would be a great experimental tool for researchers investigating bacterial mechanics and would guide the design of micro-flagellated robots that have enormous potential in targeted drug delivery [2]. Macroscale flagellar robots can also be utilized as cost-effective oil spill skimmers, pipeline inspection, and fishing baits [147].

The propulsion of uni-flagellum-propelled systems such as bacteria and our bacteria-inspired robot, consists of three factors: (1)elasticity of the flagellum, (2)flagellum-induced flow, and (3)the hydrodynamics of the head. Flagellar propulsion has been well theoretically and computationally investigated. Firstly, the flagellum is modeled as a Kirchhoff elastic rod [81] while the flagellum-generated hydrodynamics in a viscous fluid are modeled by Resistive Force Theory (RFT) [92, 5]. This framework demonstrated that the flagellum can experience buckling instability when its rotational frequency surpasses a threshold [26]. Nonetheless, RFT only matches qualitatively with experiments regarding the phenomenon because it is a local hydrodynamic force model, ignoring the interaction between the flow induced by distant flagellar portions [5, 100, 148]. More recently, the instability of a flexible helical rod rotating in a viscous fluid was studied [72] by combining Discrete Elastic Rods (DER) [73, 83] - a fast algorithm introduced by the computer graphics community to simulate the dynamics of slender structures in movies, such as fur and hair, and Lighthill's slender body theory (LSBT) - a long-range hydrodynamic force model. This DER-LSBT framework was validated quantitatively against experiments using a scaled-up clamped flagellum (not a robot). The same framework was then used to study the propulsion and instability of a rotating helical rod subjected to an axial flow [149] and the effect of a close rigid boundary on flagellar propulsion [74]. Under the postulation that flagellar buckling is critical for changing the swimming direction of uni-flagellar propulsion [1], the framework was further enhanced to model the trajectory of bacteria-inspired uni-flagellar robots [75]. The simulator also predicted that robots with one soft flagellum can follow any prescribed trajectory by changing a single scalar input - the flagellar angular velocity. Nonetheless, experiments with untethered robots to verify this prediction are missing because of the challenge of developing underwater robots that are compact, neutrally buoyant, and functional with a soft flagellum. To retain mobility while avoiding the edge effect in a glycerin tank with a low Reynolds number, the robot must be untethered, neutrally buoyant, and small. Furthermore, if the flagellum is too soft, it does not provide sufficient propulsion and does not buckle otherwise. Moreover, the simulation framework above uses LSBT, which has twofold weaknesses: (i)Because of the discontinuity between the local and nonlocal hydrodynamic terms in the formulation of LSBT, numerical issues appear when simulating the interaction among segments on the rods, i.e., when the rod buckles and some of its segments approach others. (ii)The spatial discretization in the LSBT-DER architecture [72] is determined by the ratio between the flagellar arclength, and the cross-sectional radius. The distance between two consecutive nodes on the rod must be sufficiently small, about less than twice the cross-sectional radius. This causes the simulation's computing speed to be slow.

5.3 Definition of the problem

Experiments [1, 143] and simulations [72, 26] have demonstrated that the hydrodynamic force can cause buckling of the flagellum or the hook [150, 151, 152]. Refs. [72, 75] demonstrated that flagellar buckling could be triggered if the rotational speed of the motor actuating the flagellum exceeds a threshold. In this article, we assume that the hook and flagellum are made of the same material to simplify the fabrication of untethered soft flagellated robots, the same idea as Ref. [72]. Bacteria-inspired flagellated robots are typically controlled by external magnetic field [153, 154], electric field [155], acoustic excitation waves [43], and chemically powered propulsion [156, 157, 158]. Buckling, on the other hand, can be induced in any elastic material simply by changing a single scalar variable – the rotational speed of the flagellum [72].

Thawani et al. [86] incorporated the effect of the head on the locomotion of a model uniflagellar bacterium; however, the flagellum was assumed to be rigid. We seek to bridge this gap and incorporate all the three aforementioned components to demonstrate that the buckling instability can be used by the uniflagellar system to follow a prescribed 3D trajectory.

Here, experimentally, we create an untethered robot with a soft flagellum, called BacteriaBot, that replicates bacterial "buckle-to-turn" mechanism and can achieve any 3D trajectory via flagellar buckling with the aid of an internal dynamic mass redistribution system. The flagellar buckling is simply triggered by the rotational speed change of a motor, a scalar. BaceriaBot is arguably the world's simplest autonomous soft robot in contrast to other released bacteria-inspired flagellated robots that are controlled by an external electric [159] or magnetic field [138], acoustic excitation waves [160, 161], or chemically driven propulsion [61]. Thanks to our modular design and mature fabrication techniques in our precision model experiments, the error between theoretical calculations and experimental implementations is within 0.1 grams.

Computationally, we develop a numerical framework that incorporates (i) discrete differential geometry (DDG) to model the robot structure, (ii) state-of-the-art long-range hydrodynamic force model - Regularized Stokeslet Segments (RSS) method [28] on the flagellum, and (iii) Stokes' law for the viscous forces on the robot head. In contrast to aforementioned LSBT, RSS has two main strengths: (i) a continuous flow field with no numerical issues [5] when two rod segments get closer and even contact with each other. (ii) Unlike the DER-LSBT framework [75], in which the spatial discretization length is determined by the ratio of the flagellar arclength to the cross-sectional radius, the spatial discretization in our DDG-RSS framework can be much coarser. Hence, the computation of our simulator is more than an order of magnitude faster than DER-LSBT. We quantify the relationship between flagellar geometrical properties and its buckling threshold (ω_t^b), which is further employed to validate the numerics against experiments. Due to the computational efficiency of our simulator, it can be used as an efficient tool for designing robots, e.g., finding the smallest cheapest motor by predicting the motor torque needed to actuate a flagellum with specific physical parameters and the motor rotational speed needed to cover ω_t^b.

Moreover, we propose an ingenious tethered gimbal configuration and map the tethered and untethered ω_t^b. This saves around $\frac{2}{3}$ of the trial-and-error exploratory voyage of a motor with a maximum rotational speed of ω_t^b, substantially easing the arduous nature of the process. To make the robot system more robust, the mass center and buoyancy center should be further away, yet they should almost intersect if the robot needs to flip. This irreconcilable contradiction makes design a robot capable of autonomously and robustly reaching a 3D destination impossible. This is ascribed to the innate nature of flagellar robots - highly underactuated and uncontrollable of the direction of flagellar buckling. As a result, inside the robot head, we develop a mass redistribution system based on a screw mechanism that is activated by sensing signals from an embedded Inertial Measurement Unit (IMU). We refer to this mechanism as a “mass-transformer". This mechanism is vital when the robot needs to move out of plane, i.e., when ascending or diving. We then implement it in our DDG-RSS simulator by enforcing a force-coupled torque and quantify the map between the speed the mass transforms and the angular velocity that the robot head tilts. This map tells us how fast and how long the mass needs to be moved if the robot is desired to steer towards a specific direction and thus enables online control of the robot to reach a 3D destination. Overall, our study develops an underwater BaceriaBot, arguably the world’s simplest autonomous soft robot to mimic bacterial “buckle-to-turn" mechanism and introduces a comprehensive framework for physics-informed modeling, motion planning, and control of a fully untethered soft robot. The framework enables offline co-design of the geometrical and material parameters of the robot, as well as closed-loop motion planning and control for generic soft flagellar robots. The quantitative match between experimental and simulation results demonstrate our computational framework is effective not only in low Reynolds number fluid Re<0.1 but also can be a useful tool for robot design and control in Re 0.2. The gimbal-pinned flagellum is used instead of clamped flagellum because of its higher similarity to the untethered soft flagellar robots, the flagellum is affected by the disturbance by the part above the hook.

The effect of gravity is a common issue encountered in almost all underwater robot design because it is undoable for perfect match between the gravity and buoyancy. Our simulator can also take care of it by an “added-mass" approach that will be described in §5.4.2.1. Numerical tools should be able to take care of this, especially for soft robots such our soft flagellated robots. This is because recent research has showed significant difference of the performance of the robot, such as its velocity, between the scenarios where the robot head is ahead of flagellum and the robot tail (flagellum).

5.4 Results

In this section, we first present two sets of platforms in §5.4.1: a gimbal setup and an untethered robot consisting of a rigid cylindrical head and a soft helical flagellum. The gimbal setup is for collecting data of the performance of a pinned rotating flexible flagellum to tune the regularization parameter of RSS method in our numerical framework against experimental results. Subsequently, we exhibit the untethered robot we designed and developed that can utilize buckling to steer. Meanwhile, we present how only one tuning parameter in RSS is tuned to match the buckling threshold in simulations and experiments. After that, a head is added to the tethered robot. Next, we briefly describe our simulation framework and offer details about how it quantitatively captures the flagellar buckling of our fully untethered soft-flagellar robot. Third, we discuss how the simulator helps predict the buckling threshold of the untethered robot, assisting us to find the motor whose rotational speed covers the buckling threshold. Next, we propose “mass-transformer" mechanism to redistribute the mass inside the robot head and enable the untetehred robot to follow any 3D preplanned trajectory together with flagellar buckling and embodied IMU perception. The total time for each trial of buckling threshold test is 500 seconds. We realize that the simulated speed of the robot stabilize after 30 seconds so to speed up the data fitting process, we set up the total simulation time as 41 seconds.

5.4.1 Desktop-scale experimental system design

The robot is comprised of a rigid head and a soft flagellum that rotate in the opposite directions while achieving the translational movement. Experimentally, the design and development of untethered robots is a time-consuming back-and-forth process of trial-and-error. The pain could be relieved if the modeling of this type of system is simple and can work as a predictive tool to guide experiments. However, the fact contradicts desires. As mentioned in § 1 of this Thesis, the complexity of modeling of untethered soft flagellated robots resides in the intricate coupling between nonlinear geometry, which includes structural instability, and hydrodynamics. In comparison, investigations using tethered robots (without the head) are considerably simpler. After three years of intermittent work on the design and development of untethered soft flagellated robots, we have discovered the practical framework displayed in Fig. 5.1. The rationale is to first develop tethered experiments and explore the underlying mechanics, and then add the robot head experimentally and investigate its effect from the mechanics side. The first step is to develop a simple tethered experimental setup and get the buckling threshold of a flagellum with specific geometrical and physical parameters. Second, we develop simulations to match the above experiments. Next, we add a robot head for effective untethered locomotion but without flagellar buckling. After that, we add the effect of the head into simulations to match experiments. Finally, as the physics of untethered locomotion are fully comprehended, the validated simulators can be utilized to predict what system modifications are necessary to achieve “buckling-to-turn".

Different from the experimental setup used in [72], in which a flagellum is clamped to a motor and spun at a constant rotational speed, we designed a gimbal with degrees of freedom most similar to a robot with a free head, as shown in Fig. 5.4. The experimental system setup for desktop-scale experiments

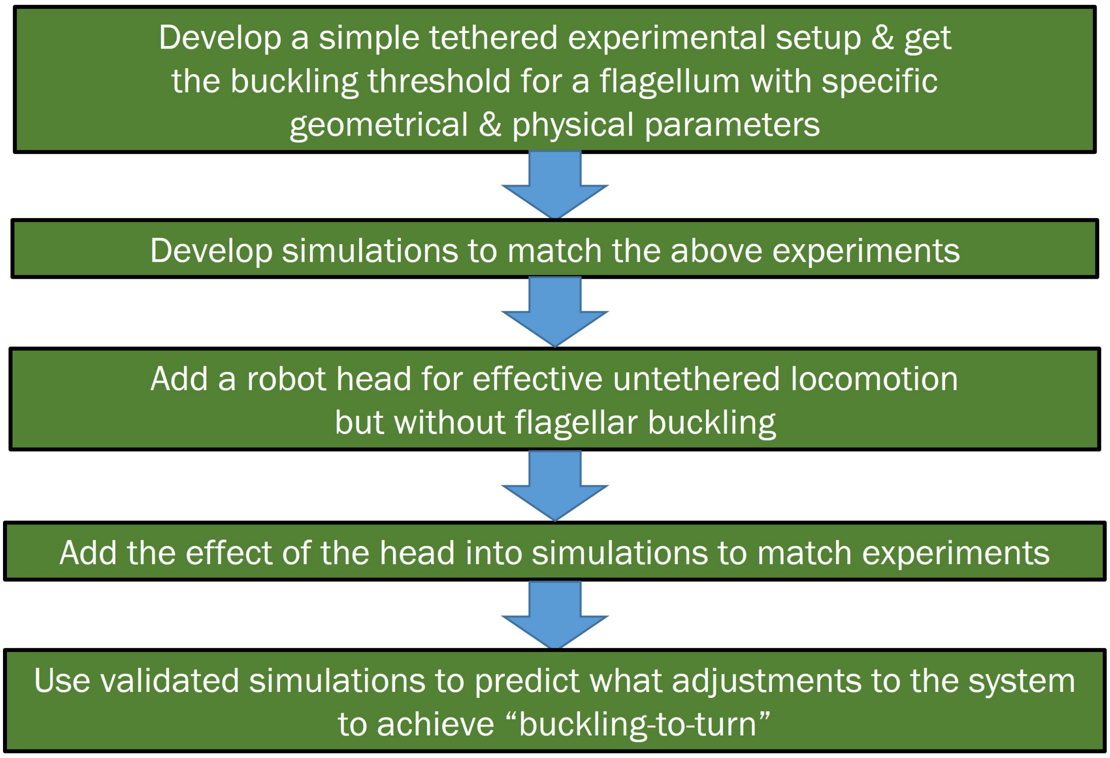
Develop a simple tethered experimental setup & get the buckling threshold for a flagellum with specific geometrical & physical parameters
Develop simulations to match the above experiments
Add a robot head for effective untethered locomotion but without flagellar buckling
Add the effect of the head into simulations to match experiments
Use validated simulations to predict what adjustments to the system to achieve “buckling-to-turn”

Figure 5.1: The flow chart to be followed to design from tethered robots and untethered soft-flagellated robots to achieve flagellar buckling.

is classified into tethered (§5.4.1.1) and untethered setups (§5.4.1.2).

5.4.1.1 Tethered setup

Reynolds number of the flagellated bacterial locomotion [62] is defined as follows

$$Re = \frac{\rho_m r_0 \omega_t R_t}{\mu}, \tag{5.1}$$

where ρ_m is the density of the glycerin, r_0 is cross-sectional radius of the flagellum, ω_t is the rotational speed of the flagellum, R_t is the radius of the helical flagellum, and μ represents the viscosity of glycerin that heavily relies on its temperature [72]. It can be seen that the parameters with a larger controllable space are r_0 and R_t. As the design process of the robot is troublesome with trial and error. Moreover, our goal is not only a robot that is capable of buckling to turn, but also capable of locomotion. Therefore, when R_t increases, r_0 should typically also be larger because the buckling threshold of the flagellum is too low if r_0 is too small and thus it is less feasible for the robot to propel forward. Eq. 3.23 will show that the drag force acting on the robot head is proportional to its radius, R_H. As a result, R_H is smaller, the drag on the robot head is reduced and thus the swimming efficiency of the robot is increased. More importantly, in order to analogize the mechanics of bacteria, the Reynolds number must be kept low, i.e., $Re < 10^{-1}$. As a result, R and r_0 should be as small as possible.

Fig. 5.4(a) depicts our experimental setup: a tethered(clamped at a gimbal)/untethered robot swimming in viscous glycerin. The robot consists of a rigid 3D-printed (PLA) head with an inner structure illustrated in Figs. 5.6- 5.8 and a soft helical flagellum. As a model for flagella, we fabricate a series of elastomeric rods with vinylpolysiloxane (VPS) [108] by independently varying each of the geometric parameters (axial length, l, helix radius, R, pitch, λ, and cross-sectional radius, r_0, or area moment of inertia, $I = \pi r_0^4/4$) and material properties (the Young' modulus, E). By updating the 3D-printed mold whose

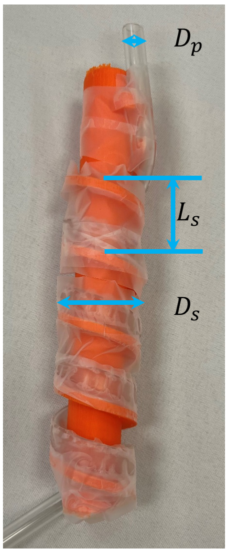

Figure 5.2: Picture of a PVC tube mold wrapping around a 3D-printed mold. D_p represents the diameter of the PVC tube, the distance between the two consecutive crests is L_s, which equals the pitch of the helical flagellum.

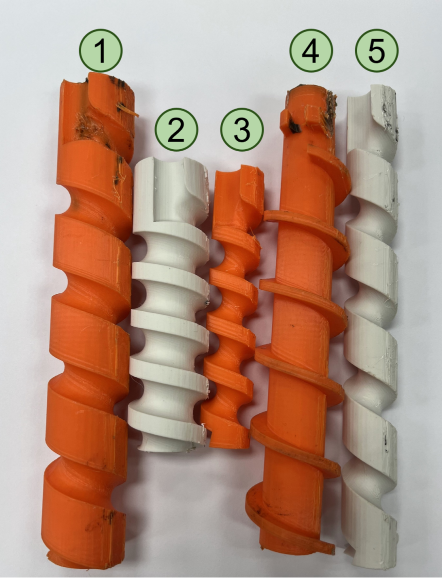

Figure 5.3: 3D-printed indented (1, 2, 3, 5) and outdented supports (4) with different axial length, l, helix radius, R and pitch, λ to hold PVC molds for polymeric flagellar fabrication.

parameters are shown in Fig. 5.2: the diameter of the PVC tube is D_p, and the parameters of the support are: $L_s = \lambda \ll D_p$ and its diameter D_s equals the diameter of the robot head D_h, i.e., $D_s = D_h$.

We apply the method introduced in Ref. [162] and analyze the shape of a suspended annulus [162] of the flagellum to determine the value of E corresponding to various polymers. The flagellum was assumed to be incompressible (Poisson' ratio, $\nu \approx 0.5$). During fabrication [108], a polyvinyl chloride (PVC) tube was wrapped about a 3D-printed indented (①, ②, ③, ⑤) and outdented supports (④) supports as shown in Fig. 5.3. It is experimentally verified that if the ratio of λ to R, i.e., $\frac{\lambda}{R}$, is large to a certain extent, the deviation of flagellar fabrication is smaller if indented supports are used instead of outdented supports assuming the wall thickness of the PVC tubes is fixed. The density of the flagellum was adjusted to match the density of glycerin by adding fine iron fillings (SHINCO) of radius ≈0.04 mm to the VPS polymer, prior to casting. Once cured and demolded, the flagellum was either pinned at one end to the gimbal (tethered) or clamped at the 3D-printed plate, immersed in a bath of glycerin.

Referring to Fig. 5.4, it can be seen that ① a helix whose deformation is captured by ② two perpendicular cameras (top and side). It is clamped to a DC motor that is equipped with two encoders and embedded into a ③ gimbal (whose CAD is shown in (b) with its rotational axes marked in dashed lines). The helix is fully submerged in viscous glycerin that is contained in ④ a tank (122cm×45cm×51.5cm). a single DC motor (uxcell)(in Fig. 5.4(d)) with an encoder is embedded within the gimbal (③ in Fig. 5.4(a)). The motor is actuated by an external⑤ DC power supply, whose speed is measured by the encoder, calculated and read by the board with ⑥ an Arduino Uno, and then transferred to ⑦ a laptop with a monitor capable of displaying the readings. With the motor speed shown on the screen, we are able to control the rotational speed of the motor precisely (≈ 0.01 rpm). In this way, the buckling threshold can be measured accurately.

Three rulers are used to measure the coordinates of each point on the robot (⑧ the ruler used to measure the displacement of the robot in the x direction, ⑨ the vertical ruler for measurement of the displacement in the z direction, and one placed at the bottom of the tank measuring displacements in the y direction but not shown in this figure). Note that ruler ⑨ is fixed as part of the 80-20 frame, which is composed of two primary parts, ⑩, which is clamped onto the wooden station and serves as a guide for the other (movable) part. The camera ② providing a top view and the stick to which ③ the gimbal is mounted are attached to the movable part. We closely regulate the viscosity of the glycerin in the tank by monitoring its temperature with ⑪ a temperature probe and by placing ice bags in the tank.

5.4.1.2 Untethered setup

Making underwater untethered robots neutrally buoyant so that the effect of gravity on their movement direction is insignificant is the first obstacle to over-

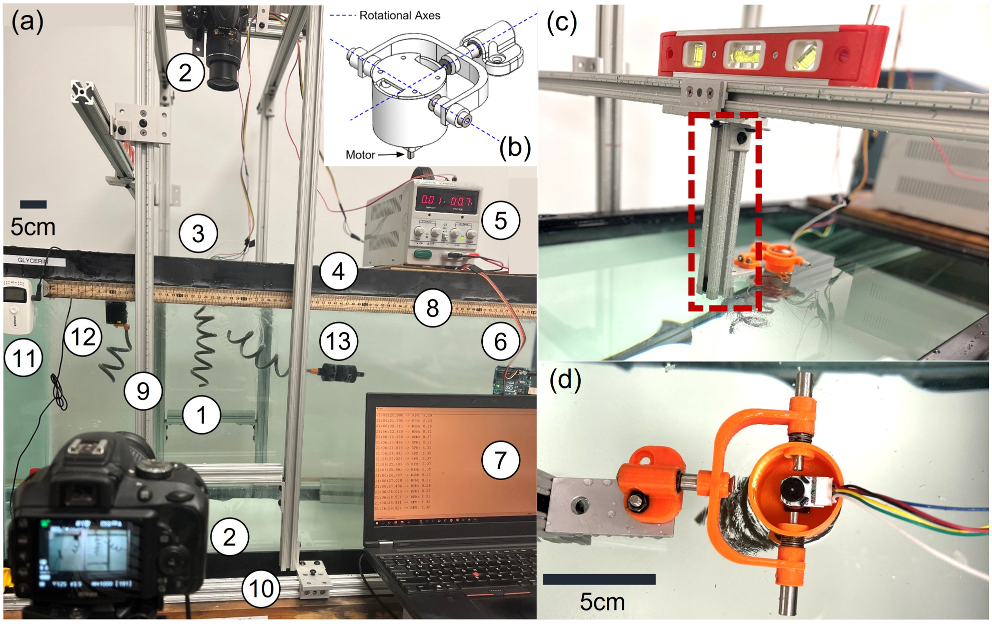

Figure 5.4: Experimental setup of tethered flagllated robots in glycerin for buckling threshold exploration and untethered flagellated robots for “buckling-to–turn" exploration. (a)Photograph of the experimental setup: ① a helix whose deformation is captured by ②two perpendicular cameras (top and side). It is clamped to a DC motor that is equipped with two encoders and embedded into a ③ gimbal (whose CAD is shown in (b) with its rotational axes marked in dashed lines). The helix is fully submerged in viscous glycerin that is contained in ④ a tank (122cm×45cm×51.5cm). The motor in the gimbal is powered by ⑤ a DC power supply whose voltage is adjustable to alter the rotational speed of the motor actuating the helical flagellum, during which the flagellar buckling threshold can be found. The rotational speed of the motor is read by ⑥ an Arduino Uno board connected to ⑦ a laptop with the serial monitor displaying the real-time rotational speed of the dc motor. Based on this information, the experimenter can immediately read the flagellar velocity when buckling happens. Three rulers are used to measure the coordinates of each point on the robot (⑧ the ruler used to measure the displacement of the robot in the x direction, ⑨ the vertical ruler for measurement of the displacement in the z direction, and one placed at the bottom of the tank measuring displacements in the y direction but not shown in this figure). Note that ruler ⑨ is fixed as part of the 80-20 frame, which is composed of two primary parts, ⑩, which is clamped onto the wooden station and serves as a guide for the other (movable) part. The camera ② providing a top view and the stick to which ③ the gimbal is mounted are attached to the movable part. We closely regulate the viscosity of the glycerin in the tank by monitoring its temperature with ⑪ a temperature probe and by placing ice bags in the tank. Two types of untethered robots ⑫ and ⑬ are displayed in this figure. Untetethered robot ⑬ is designed first with electronic components and counterweights compactly arranged inside the robot head to achieve the “mass-transformer" mechanism. Then, robot ⑫ that has no slider in its head but has the same radius as robot ⑬ is developed. Robot ⑫ simply goes up, but robot ⑬ plans its motion by buckling its flagellum and employing the“mass-transformer" when a sharp turn (e.g., a flip) is necessary; (c) the side view showing how the gimbal is mounted to a stick (marked in the red rectangle) that is attached to the movable part of the 80-20 frame; (d) the top view illustrating how the dc motor with encoders is embedded in the gimbal. See text for the properties of rod and fluid.

come (challenge I). The experimental setup for untethered soft flagellated robots (⑫ and ⑬) is illustrated in Fig. 5.4. Note that the gimbal-based arrangement is not required for research with untethered robots. In untethered experiments, two types of untethered robots ⑫ and ⑬ are displayed in this figure. For untethered research, we only require a robot, two cameras, three rulers, a temperature probe, and a tank. Untetethered robot ⑬ is designed first with electronic components and counterweights compactly arranged inside the robot head to achieve the “mass-transformer" mechanism. Then, robot ⑫ that has no slider in its head but has the same radius as robot ⑬ is developed. Robot ⑫ simply goes up, but robot ⑬ plans its motion by buckling its flagellum and employing the“mass-transformer" when a sharp turn (e.g., a flip) is necessary; Eq. 4.6 demonstrates that the drag on the robot head is a linear function of the radius of the robot head, R_c. Hence, structure within the head of an untethered robot must be compact, which is the second challenge of underwater untethered soft flagellated robots (challenge II). As our objective is not only to establish the simplest testbeds for researchers to study flagellar propulsion in viscous fluids, but also to push the robot towards full autonomy for 3D motion planning and controllable locomotion. There is no optimum orientation for the flagellum due to the fact that its density is matched to the density of glycerin during fabrication. The neural buoyancy of the robot is therefore dependent on the neutral buoyancy of the robot head. Additionally, although the robot gets propelled by its flagellum, the head decides the orientation of the robot when the robot is under the “push" mode [163] because the robot is in the “head-tail" (i.e., head moving in front of the tail) moving mode [164]. Bacterial head was proved to be prolate [165, 166, 167, 168]. Experiments have demonstrated that the greatest space utilization reaches its maximum when the robot head is cylindrically shaped as displayed in Fig. 5.5 with its exploded view depicted in Fig. 5.7, given that most components inside it are rigid. It is well acknowledged that the relative position of the center of mass (COM) and center of volume/buoyancy (COV) determines its robustness. Furthermore, the placement of COM beneath COV is required for the robot to move vertically, which makes it difficult for the robot to travel horizontally with stability. In order to enable the robot to plan its motion to reach any point in the 3D space, we have to design the COM of the robot head beneath its COV. Moreover, we have to evaluate the relationship between the distance between them and how much flagellar buckling can change the orientation of the robot head. Their quantitative relationship is explored and offered in the coming publication.

Eventually, it turns out that a mass-redistribution mechanism is needed within the robot head is required for the robot to follow any 3D prescribed trajectory. For this purpose, we conceived mechanism similar to electromagnetic catapults to move balls based on the orientation of the robot head measured by an IMU. Yet, the mechanism is too complicated to be fit into a compact robot head and controlling the movement of the ball is difficult. Finally, we designed a lead screw-nut-based mechanism, called “mass-transformer", to redistribute the mass on a slider (whose details are illustrated in Fig. 5.6 encapsulated as ⑦ in Fig. 5.7) within the head. To enable the ① IMU to detect the orientation of the head in real time as the self-embodied feedback and avoid tangled wire connections between the movable parts on the slider and the fixed parts inside the caps, the components that are held onto the slider (③ a 3D-printed housing) are: ① an IMU (ADXL345), ② a microcontroller (Adafruit Pro Trinket 5V or customized soft PCB board), ④ a DC motor (uxcell, 3V), ⑤ Tungsten weights working as counterweights, and ⑥ two Vintrons 3.7V 140mAh LiPo batteries. ⑤ are the most dense and cost-effective off-the-shelf weights. Integrating the components on the housing ensured experiment repeatability, which is vital considering the sensitivity of underwater robot experiments to neutral buoyancy. There are two motors inside the head of the untethered robot, one (embedded in the slider ⑦ in Fig. 5.7) for the “mass-transformer" mechanism (with a threaded shaft), and the other for actuating the flagellum (⑪ in Fig. 5.7). Both motors are connected to ② and controlled by PWM signals. Note that ② is chosen over boards with comparable capacity, such as Arduino Nano, Adafruit ItsyBitsy M0 Express, and Adafruit Trinket M0, due to its greater aspect ratio, which reduces R and thus the drag on the head. Consequently, r_0 is decreased, ensuring the locomotion in a low Reynolds number regime according to Eq. 5.1. The most important reason of choosing this board is that it has two sets of interrupt pins that are needed to independently control the rotational directions and speeds of two motors within the head. ⑥ Two LiPo batteries are linked in parallel to match the voltage range of motors and store enough power for the robot to operate several hours before needing to get recharged. Fig. 5.7 represents the exploded view of the robot body SOLIDWORKS model. The robot body consists of three outer parts, ⑧ a PLA head cap, ⑨ a PLA shell, and ⑩ a PLA tail cap. At the center of a ⑧, there is a nut embedded and fixed. Together with ⑦ the assembled slider, they contribute to the “mass transformer". The components of “mass transformer" are arranged in such a way to maximize the axisymmetry of the robot head. “Mass transformer" enables precise and robust control of the moving distance of the housing. The actual development of the robot is demonstrated in Fig. 5.8. Untethered soft flagellated robot design and fabrication is delicate, and thus the following tips and tricks (highlights) are suggested during the robot design and fabrication:

- Two “noses" (as denoted in Fig. 5.5) are extended out to give ample travel to the assembled slider (⑦ in 5.7) for mass redistribution. This ingenious design decreases R_c substantially. In addition, counterweights such as solder can be wrapped around the noses for delicate density matching;
- Two “L"-shape locks are designed on ⑧ and ⑩ (in Fig. 5.7) to prevent

the relative rotation between ⑧, ⑩ and ⑨. Therefore, it helps waterproof the connection between caps and the body;

- “Mass-transformer" mechanism is designed to redistribute the mass inside the robot head to follow any 3D preplanned trajectory;
- The length of each wire is fixed for all robot fabrications. For reproducible trials, we did our best to maintain the location of wires with tiny tolerances, and to maintain the robot’s neutral buoyancy;
- Two slots are carefully crafted inside the tiny robot head to prevent mass rotation and reduce friction between the moving assembled slider and the body wall;
- Batteries and microcontroller (two sets of interrupt pins) should be compatible;
- Electrical pins (marked in red rectangles in Fig. 5.8) are exposed out of ⑩ the tail cap to conveniently test the robot statically and dynamically without repeated disassembly and reassembly.
- During the robot design and fabrication, components must be positioned within the robot head to make it centrosymmetric;
- While designing the head, COM and COV in SOLIDWORKS should be matched around the head center;
- The outer shell (⑨ in Fig. 5.7) is coated with Krylon K05160107 ColorMaster and with wax (which is fluidic when applied) sealing the connection between caps and the body (for waterproofing).

It is observable that the design and fabrication of the robot head is meticulous and sort of pulling one hair affecting the whole body. Counterweight holder

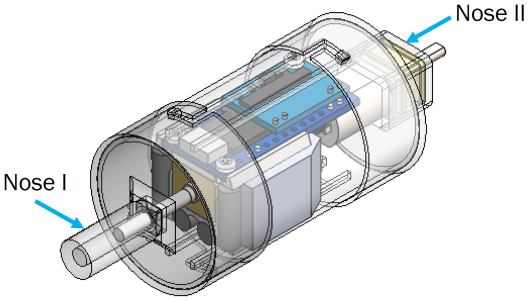

Figure 5.5: SOLIDWORKS model of an untethered soft flagellated robot.

should be exposed to conveniently match the density of the robot and glycerin (designs 1 and 2). Using digital imaging, we reconstructed the deformed configurations of the flagellum and quantified its dynamics. To ensure constant and reproducible values for the viscosity, we placed one thermometer to inspect the temperature of the glycerin bath and to use ice bags to accurately control the temperature within ±0.5°C. Due to the difficulty of cooling down all the glycerin evenly inside the tank for a higher viscosity, the viscosity of glycerin is only controlled to be μ [Pa · s] = 1.03(± 0.07 Pa · s). The density of glycerin is $\rho_m = 1.24$ g/cm^3, and our rods had a roughly the same but slightly higher density ($\lesssim$ 3%) than glycerin.

Due to the internal torque balancing, the robot's head and flagellum rotate in the opposite direction, just like bacteria. The component of the drag along the robot's moving direction on a helical flagellum that rotates along its long axis propels is the propulsive force, generates a translational movement (the direction of which is denoted by the red arrow).

Neutral buoyancy is closely related to waterproofing, which has historically been one of the most significant obstacles for underwater robotics. We conquered this by meticulously modularizing components in the robot head and theoretically balancing force and torque during the design and fabrication stages. Even every wire has its location to stay so that experiments are repeatable after

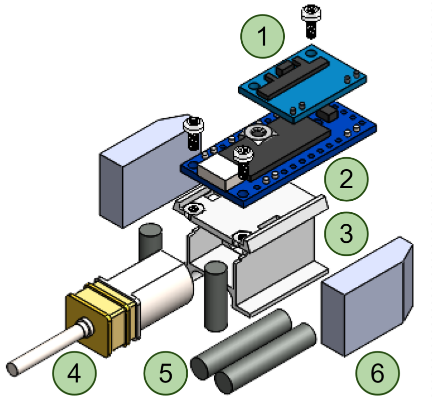

Figure 5.6: Exploded view of SOLIDWORKS model of the slider within the robot head. The slider consists of ① an IMU that detects the orientation of the robot head, and ② a microcontroller mounted onto ③ a 3D-printed housing, attached, on which are also attached ④ a DC motor (uxcell) with a threaded shaft, ⑤ Tungsten weights, and ⑥ two Vintrons 3.7V 140mAh LiPo batteries.

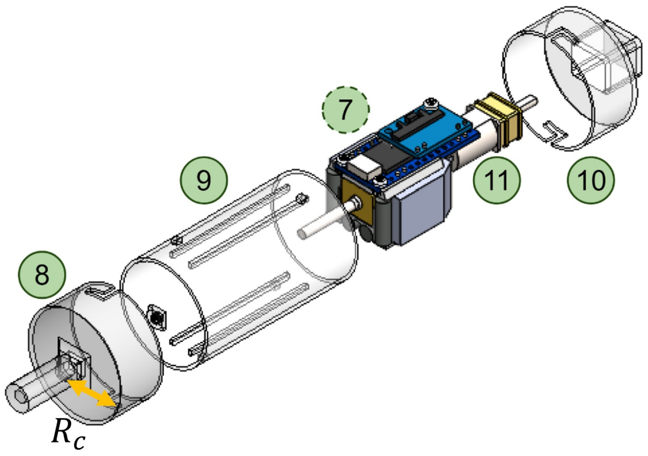

Figure 5.7: Exploded view of the robot body SOLIDWORKS model. The robot head is comprised of ⑦ an assembled slider located inside the head, ⑧ a PLA head cap, ⑨ a PLA shell, and ⑩ a PLA tail cap. ⑪ a DC motor actuating the tail – flagellum is embedded inside ⑩.

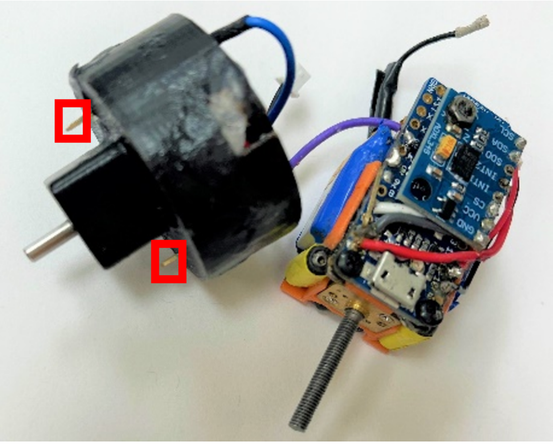

Figure 5.8: Picture of the actual development of the robot head (the head cap is not included). The pins marked in red rectangles are electrical pins exposed out of (⑩ in Fig. 5.7) the tail cap to conveniently test the robot statically and dynamically without repeated disassembly and reassembly.

disassembly and reassembly. Technically, we sprayed the PLA head with thin waterproof paints (Krylon K05160107 ColorMaster) and then applied heated fluidic wax to seal the connection between the body of head and the two caps.

Before each experiment is conducted, the robot is slowly placed in the desired initial position and alignment (robot perpendicular to the bottom of the tank). We used, as far as possible from the bottom of the tank (rigid boundary condition) and the glycerin-air interface (open boundary condition) such that the motion is not significantly disturbed by boundaries. The motion is recorded with a video camera (920 x 1080 pixels, Nikon D3400, 60 frames per second). The speed of the robot is obtained by measuring its displacement in time. The location of the robot is denoted by the location of the head, determined using software MATLAB; the speed is deduced from the position of the head using a central difference scheme.

5.4.2 Fully implicit DDG-RSS-Stokes' numerical framework

We develop a fully-implicit DDG-RSS framework to simulate the dynamics of the robot. Our numerical framework combines three components: (i) Discrete

Differential Geometry (DDG) approach [73, 83, 72, 67] to describe the structure and geometrically nonlinear deformation of flexible filaments, flagella [66, 67]; (ii) Regularized Stokeslet Segments (RSS) method for the nonlocal hydrodynamic force produced by slender structures in viscous fluids, i.e., a flagellum in glycerin [28]; and (iii) Stokes' law for the viscous drag exerted on the cylindrical robot head. Our software implementation will be released as a GitHub repository. The details of this DDG-based tool was introduced in §1.3. Our DDG-RSS-Stokes' numerical framework starts with a discrete representation of the robot. We provide a schematic diagram of the robot in experiments and simulations in Fig. 5.9 to explain both the experimental (Fig. 5.9(a)) and numerical setup (Fig. 5.9(b)). The configuration of the robot is fully described by the locations of N nodes (with lumped masses): $\mathbf{x}_j = [x_j, y_j, z_j]$ $(0 \leq j < N$ where N is the number of nodes), and the twist angles, θ^j $(0 \leq j < N_e$ where N_e is the number of edges and $N_e = N - 1$ for the robot studies in this paper). The N nodes correspond to $N-1$ edges, $\mathbf{e}^0, \ldots, \mathbf{e}^{N-2}$ such that $\mathbf{e}^j = \mathbf{x}_{j+1} - \mathbf{x}_j$. From here, we use subscripts to denote node-based quantities and superscripts for edge-based quantities. Every two adjacent nodes contribute to a stretching spring. Every three neighboring nodes form a torsional spring. Consequently, each node is associated with bending, stretching, and twisting energies. The DOF vector for the robot is $\mathbf{q} = \left[\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N-1}, \theta^0, \theta^1, \ldots, \theta^{N_e-1}\right]^T$, where the superscript ^{T} denotes transpose. Note that two nodes, nodes 0 and 1, are used to discretize the head and node 2 is the conjunction point of the motor shaft and 3D-printed plate. The length of edges $\mathbf{e}^0$ and $\mathbf{e}^1$ are identical and equals the radius of the head, R_H in Fig. 5.9(b). Note that the R_H is decided by Nodes 2 and 3, and edge $\mathbf{e}^2$ represent the 3D-printed plate that holds the soft flagellum, so the length of $\mathbf{e}^2$ equals the radius of the helix, R_x. Edge $\mathbf{e}^3$ stands for the flagellar base, hook [1], so its length, L_3, is changeable. Because of the fact that the robot head and 3D-printed plate are rigid, edge $\mathbf{e}^2$ is enough to

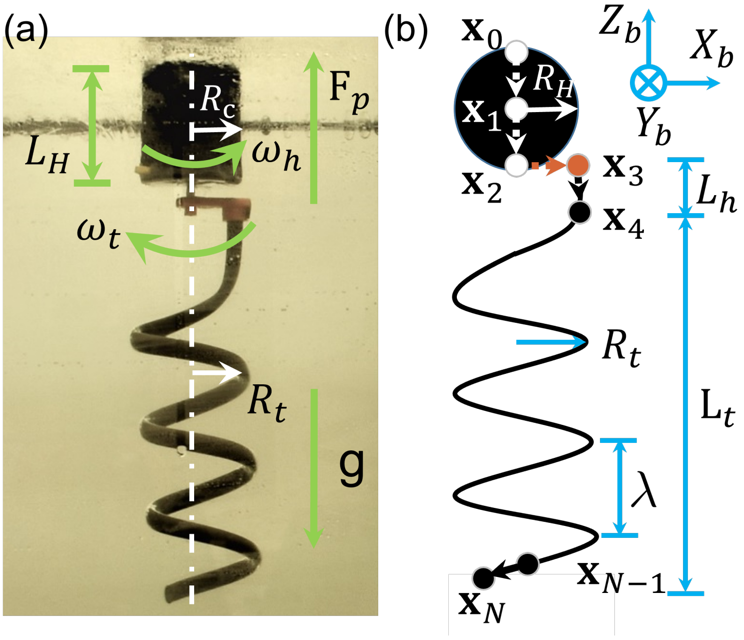

Figure 5.9: Geometry of the soft flagellated robot with a rigid cylindrical head. (a) In experiments, the robot is neutrally buoyant in glycerin. The propulsive force generated by the flagellum is F_p. The total rotational speed of the motor actuating the flagellum is the sum of the rotational speed of the robot head, ω_h, and the flagellum, ω_t, i.e., $\omega_T = \omega_h + \omega_t$. The direction of vector g denotes the direction of gravity. (b)The geometry and the geometrical discretization of the robot by DDG. The robot head is represented by nodes $\mathbf{x}_0$ and $\mathbf{x}_1$ and the bending, twisting, and stretching springs comprised of edges $\mathbf{e}_0$ and $\mathbf{e}_1$(not shown). The two dashed white lines (against the black circle that stands for a virtually equivalent head with the same volume as the original cylindrical head) denote the discretization of the robot head. The length and radius of the rigid cylindrical head is L_H and R_c, respectively. The length of the flexible hook (a part of the flagellum) is $L_h = ||\mathbf{x}_4 - \mathbf{x}_3||$. The radius and pitch of the helical flagellum is R_t and λ, respectively. The axis length of the flagellum is L_t. Note that the coordinate axis system $X_b - Y_b - Z_b$ is the one associated with the world frame and it apply to both (a) experiments and (b) simulations.

represent the 3D-printed plate and the plate need not be finely discretized. All remaining nodes with locations $\mathbf{x}_4$ to $\mathbf{x}_{N-1}$ represent the soft helical flagellum and fall on a helical shape in their undeformed state at time $t = 0$ and the length of each edge on the flagellum is the same.

Each edge, $\mathbf{e}^j$, has an orthonormal adapted reference frame $\left\{\mathbf{d}_1^j, \mathbf{d}_2^j, \mathbf{t}^j\right\}$, and material frame $\left\{\mathbf{m}_1^j, \mathbf{m}_2^j, \mathbf{t}^j\right\}$, whose details can be found in our previous work [66]. Given that the head has a greater mass, the masses $m_j(0 \leq j < N)$ of all nodes other than nodes 0 and 1 on the head are identical while experimental measurements tell us that $m_0 = m_1 = 10m_x$.

Similarly, the head is rigid and thus the axial stiffness EA of the stretching springs S_0, S_1, as well as bending stiffness $EI = \frac{\pi}{4}Er_0^4$ and twisting stiffness $GJ = \frac{\pi}{2}Gr_0^2$, of torsional springs T_0, T_1, T_2 are 1,000 times those of other springs. In above equations, E is the Young's modulus, G is the shear modulus, $A = \pi r_0^2$ is the cross-sectional area, and r_0 is the cross-sectional radius of the flagellum.

The elastic energy associated with springs in the robot can be computed from the strains, the details of which are illustrated in Section IV-B of our prior work [67]. The total elastic energy of the robot is the linear sum of (1) stretching energies $E_s = \sum_{js=0}^{js=N-2} E_s^{js}$, (2) bending energies $E_b = \sum_{jb=1}^{jb=N-2} E_b^{jb}$, and (3) twisting energies $E_t = \sum_{jt=1}^{jt=N-2} E_s^{jt}$, the formulations of which are Eqs. 6-8 in [67]. Note that our robot's geometry here are significantly different with our previous released work [72, 75, 66, 67] because of the existence of robot head, flagellar hook, and uni-flagllum (no "joint" node as introduced in [67]). The elastic stretching, bending, and twisting forces exerted on each node can be calculated as the negative gradient of corresponding elastic energies. We use a natural twist associated with edge $\mathbf{e}^1$ that varies with time to model the actuation - a rotating motor. This operation is ingenious because it is a vivid description of actuators and does not add artificial energies, as boundary conditions do. On nodes 3 to $N-1$, external forces are the nonlinear hydrodynamic force from the viscous fluid that is modeled by the RSS method, while on nodes 0 and 1, external forces are the drag forces and torques from the viscous fluid that is modeled by Stokes's law. These forces are used in equations of motion, which are fully implicitly solved for the next time step. See Section 5.6 for a deep discussion of the simulator solving dynamics. The above framework applies to the untethered soft flagellated robot. For the gimbal setup, we could have remove node 0 and related degrees of freedom, i.e., its position vector and twist angle. However, to make the numerical framework boundary conditions are applied to fix the x, y, z coordinates the first node

The values of parameters inside the simulator is listed in the table below. The denotations below are consistent with those in Fig. 5.9.

λ (m)	0.036
R_t (m)	0.018
R_H (m)	0.018
L_h(m)	0.0188
C_1	1.8
C_2	1.37
r_0(m)	varying
E	varying
ν	0.5
deltaTime (s)	1e-4
totalTime (s)	500
tolerance	0.01
maxIter	100
ρ_m (kg/m^3)	1240
gVector	0.0 0.0 0.0
μ	1.03
ϵ	1.03r_0
omegaHigh	5.0
omegaLow	0.0
timeHigh	1.0
timeLow	0.0
axisLengthInput	0.144
deltaLengthInput	5e-3
render	1
gimbal	1
saveData	1

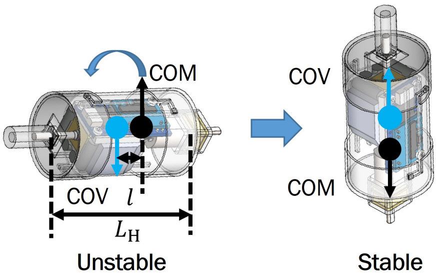

Figure 5.10: Orientation of the robot head versus the relative position between its center of mass (COM) and center of volume (COV).

5.4.2.1 “Added mass" method for simulating uneven mass distribution

Uneven mass distribution is the most common issue encountered during fabrication, and it has a significant impact when it occurs along the long axis of the cylindrical robot head. This is because it will directly cause the mismatch between the center of mass (COM) and center of volume/buoyancy (COV). As illustrated in Fig. 5.10, COM and COV must coincide for the robot to be able to swim horizontally with stability. Consequently, during the design phase in SOLIDWORKS, we match them by carefully arranging the components (and thus masses). However, it was experimentally confirmed that even if there is only 0.01mm between them, the robot head will transition from an unstable (horizontal) orientation to a stable equilibrium (vertically). As the head is the robot’s navigator, it controls the robot’s direction. This procedure is expensive to reverse because flagellar buckling is insufficient to flip the robot from vertical to horizontal position. We develop a “mass-transformer" mechanism so that the robot can be controllable to follow a prescribed 3D trajectory. The mass redistribution is achieved by controlling the rotational speed and time of the motor ④ in Fig. 5.6. Correspondingly, “mass-transformer" can be simulated through a time-varying “added mass" method.

In the simulator, the robot is discretized and represented in Fig. 5.9(b). The torque generated by the uneven distribution of the head is simulated by adding an extra mass onto nodes $\mathbf{x}_0$ and $\mathbf{x}_2$ to form a force couple/moment. Our robot head is assumed to be radially symmetric, which is empirically confirmed by the fact that the robot head and flagellum always rotate at a constant speed. The COM of the robot head can be measured by placing the cylinder-shaped head on a thin blade that functions as a pivot, and then two halves separated by the pivot will be balanced by the lever principle. The COV is determined as the midpoint along the length of the cylinder. As exhibited in Fig. 5.10, if the distance between COM and COV is l, extra mass added onto nodes $\mathbf{x}_0$ and $\mathbf{x}_2$ are calculated as follows:

$$M_a = \frac{\rho_m \pi L_H R_c^2 l/2}{L_H/2} = \rho_m \pi R_c^2 l, \tag{5.2}$$

where ρ_m has the same meaning as Eq. 5.1. The mass unevenness of the flagellum can also be simulated in a similar manner, albeit it can be avoided by the use of accurate fabrication techniques.

Compared to the previous work [73, 83, 72, 67], our simulation framework has several highlights:

1. The actuation, e.g., rotation of the motor, is readily embedded in the DDG approach by updating the natural twist with time, which is different from typical DDG-based simulators where undeformed configuration of a structure is fixed and assumed to be invariant through the simulation. Hence, our simulator is fully implicit and can thus converge at a larger time step, requiring less computation time;

2. We are the first to develop a DDG-RSS-Stokes' simulation framework for simulating untethered locomotion in viscous fluids that incorporates state-of-the-art RSS method, which is continuous and has no requirements for discrete edge length, to model the nonlinear hydrodynamic force by taking

into account the interaction among (even distant) nodes on soft filaments. Furthermore, we demonstrate its precision by systematic desktop-scale experiments. We further investigate and validate the applicability of this framework in Reynolds number ≈ 0.2, which cannot be considered a low Reynolds number regime;

3. We develop an “added-mass" method in the aforementioned DDG-RSS-Stokes’ framework to consider the most common problem in experiments – uneven mass redistribution and adapt it to simulate the controllable mass-redistribution mechanism – “mass-transformer".

5.4.3 Characterization of tethered flagellar buckling threshold

Since untethered locomotion involving flagellar buckling is a complicated system to model, we must first validate the simulation framework introduced above in a tethered setup before applying it as a predictive tool for finding a motor that can buckle the flagellum. For this, we developed a frictionless gimbal with degrees of freedom (DOF) as shown in Fig. 5.4(b). Compared to prior studies [72], the gimbal-based system will allow the gimbal-pinned “head" to have as many DOF as an untethered robot. This reduces the difference between the tethered configuration learned by the simulation and the untethered locomotion for which the simulation is used to estimate performance.

r_0 (mm)	ω_t (rpm)	Re
0.792	1.1034	0.0020
1.188	1.4551	0.0039
1.584	3.0375	0.0109
1.980	7.3607	0.0330
2.376	12.6588	0.0683
2.772	25.2744	0.1590
3.168	36.9887	0.2659

5.4.3.1 Effect of geometrical parameters

We varied the radius and number of pitches of the flagellum to change its geometrical parameters and investigated their effect on the buckling threshold. The results are displayed in Figs. 5.12 and 5.11. The similarity between experimental data (in filled triangles) and simulation data (in hollow triangles) in Figs. 5.12 and 5.11 indicates that DDG-RSS successfully captures the hydrodynamics from the surrounding fluid on the flagellum. Also, the result makes physical sense: in general, the flagellar buckling threshold, ω_b, increases with flagellar radius whereas decreases with flagellar pitch number, N. One thing deserves our notice is that there exists a falgellar radius turning point before which ω_b reduces with the increase of r_0 and after which ω_b grows with the growth of r_0.

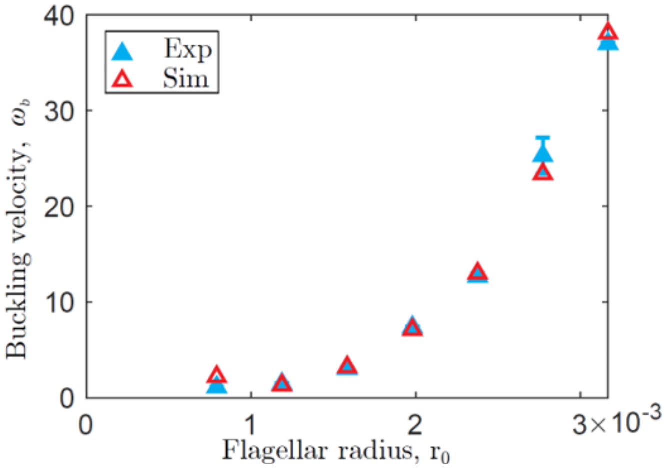

Figure 5.11: The flagellar buckling velocity VS. flagellar radius

5.4.3.2 Effect of Young's modulus

In addition to modifying flagellar geometrical characteristics, we modified its physical property, Young's modulus and plotted the results in Fig. 5.13. Again, experiments and simulations match nicely. ω_b becomes larger as the flagellum stiffens.

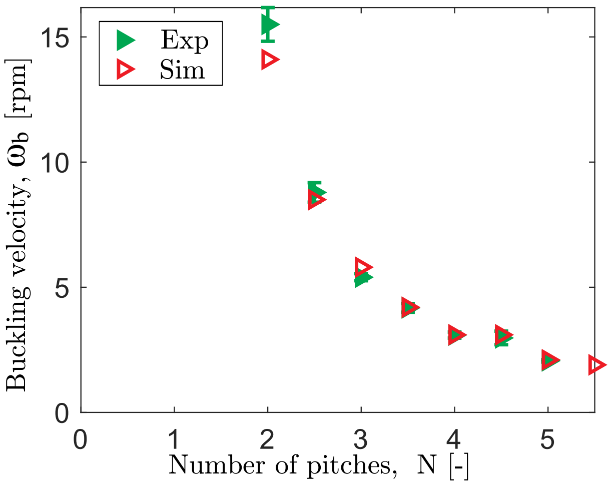

Figure 5.12: The flagellar buckling velocity VS. number of flagellar pitches.

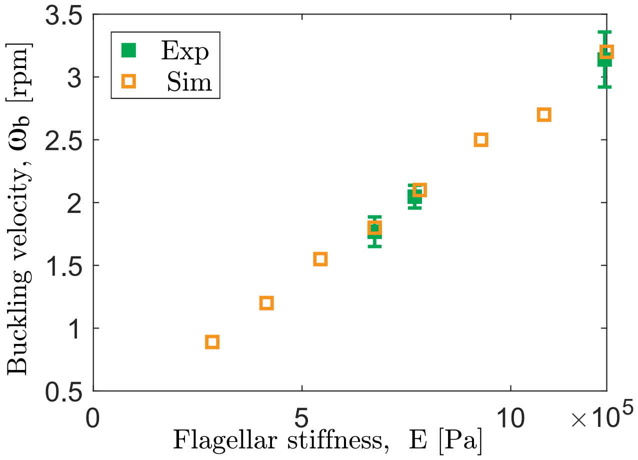

Figure 5.13: The flagellar buckling velocity VS. Young' modulus of the flagellum.

5.4.4 Characterization of untethered flagellar propulsion velocity versus the rotational motor speed

Since the robot's head shape is not perfectly spherical and there might be slippage between the robot head and the fluid, fitting parameters are required to calibrate Stokes' law, which will be explained in greater details in §5.6.3. Fitting parameters are determined from matching some experimental and simulation results. The same set of parameters is then incorporated into the DDG-RSS-

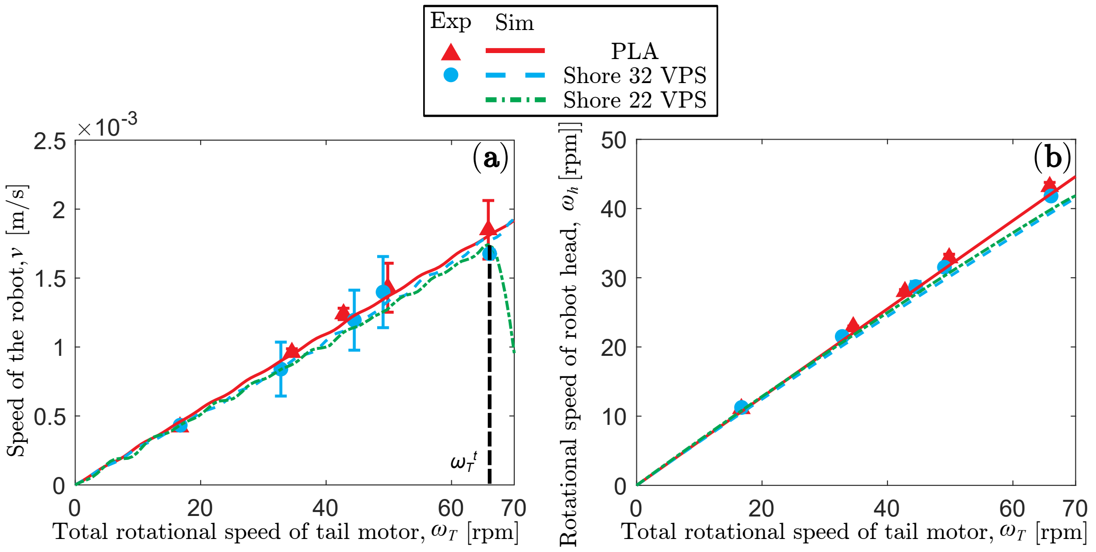

Figure 5.14: Parameter fitting to match the experimental data (triangles and circles) and DDG-RSS-Stokes' simulation results (solid, dashed, and dash-dotted lines) of untethered robots with a flagellum made of PLA and Shore 32 VPS with (a)the speed of the robot, v, versus the total rotational speed of the tail motor, ω_T, and (b)the rotational speed of the robot head versus ω_T. Using a PLA flagellum (solid line) and Shore 32 VPS (dashed line), the untethered robot has a pitch of 3.5. The Young's modulus of PLA is 4.107GPa and Shore 32 (experimentally measured) is 1.225MPa. Fitting parameters $\mathscr{C}$: $C_t = 1.80 \pm 0.02, C_r = 1.37 \pm 0.025$ can realize the smallest fitting error between experimental and simulation results. Next, dash-dotted lines plot the predicted results of an untethered robot with a 3.5-pitch flagellum made of Shore 22 VPS (with a Young's modulus of 0.77MPa). The vertical black dashed line marks ω_T (≈ 67 rpm), the point at which the flagellum buckles, denoted by ω_T^t.

Stokes' framework to predict the performance of systems with parameters that the simulator has never seen. The theoretical framework is validated if the predicted performance matches experiments accurately. We fitted the experimental and simulation data for untethered robots with a 3.5-pitch flagellum made of PLA and Shore 32 VPS and identified the following fitting parameters: $C_t = 1.80 \pm 0.02, C_r = 1.37 \pm 0.025$. The Young's modulus of PLA and Shore 32 is experimentally measured as 4.107GPa and 1.225MPa, respectively. The least fitting error between experimental and simulation results can be achieved. Experimental data for an untethered robot with a PLA flagellum are plotted as triangles in Fig. 5.14 while the corresponding simulation results are plotted as continuous lines. Fig. 5.14(a)plots the speed of the robot, v [m/s], versus the total rotational speed of the tail motor, ω_T; Fig. 5.14(b) plots the rotational speed of the robot head, ω_h [rpm], versus ω_T [rpm]. Experimental data for an untethered robot with a Shore 32 flagellum are plotted as circles in Fig. 5.14 while the corresponding simulation results are plotted as dashed lines. Next, dash-dotted lines plot the predicted results of an untethered robot with a 3.5-pitch flagellum made of Shore 22 VPS (with a Young's modulus of 0.77MPa). The vertical black dashed line marks ω_T (≈ 67 rpm), the point at which the flagellum buckles, denoted by ω_T^t. We did not measure the 3.5-pitch flagellum made of Shore 32 VPS. Rather, we conducted experiments on an untethered robot with a Shore 22 VPS flagellum with 3 pitches this is because we want to see the prediction accuracy of the simulator for an untethered locomotion with a different geometry (e.g., number of pitches) and stiffness (Young's modulus). Amazingly, Fig. 5.14 demonstrates the accuracy of DDG-RSS-Stokes' framework. The vertical black line in Fig. 5.14(a) also denotes the flagellar buckling threshold, ω_T^t. Comparing Fig. 5.15(a) with Fig. 5.14(a), we can observe that ω_T^t is larger for a 3-pitch Shore 22 VPS flagellum than for a 3.5-pitch one.

One more thing to deserve our attention is that before flagellar buckling

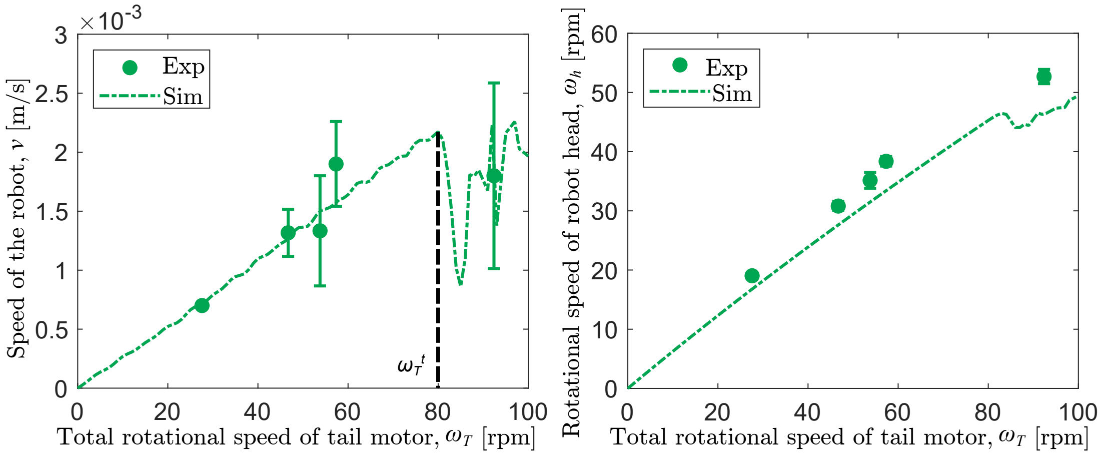

Figure 5.15: Predicted (with fitting parameters $\mathscr{C}$) performance of untethered robots with a 3-pitch Shore 22 VPS flagellum versus experimental results: (a)the speed of the robot, v, versus the total rotational speed of the tail motor, ω_T, and (b)the rotational speed of the robot head versus ω_T.

occurs, the propulsion efficiency of a robot with a flexible flagellum (e.g., Shore 32 VPS) is the same to that of a robot with a rigid flagellum (e.g., PLA).

5.4.5 Map from tethered to untethered: locomotion performance

Users can first apply the gimbal-based setup to find the flagellar buckling threshold (calibrating the DDG-RSS framework) and then add a robot head but do not involve flagellar buckling, after which the DDG-RSS-Stokes's framework is verified. The results above validated our DDG-RSS-Stokes' simulation framework. Therefore, the simulator can work as a predictor to predict the performance of untethered locomotion. Consequently, our simulator can then forecast the performance of robots with a flagellum with different geometrical and physical parameters. Further, it is capable of determining a motor with a rotational speed exceeding the flagellar buckling threshold with the same robot head.

5.4.6 “Mass-transformer" for autonomous motion planning

Although we prove that flagellar buckling enables the robot to turn and reconfigure, the movement (direction and orientation) of the robot is not completely controllable due to underactuation. The rotational direction and speed of the motor are controllable. Plus, the initial shaft position where the motor starts to rotate is also controllable. If the flagellum is isotropic and the aforementioned controllable parameters are set to identical values, the robot should follow the same trajectory. However, because of fabrication and experimental errors such as the flagellar fabrication flaw, the trajectory of the robot is not identical even if the controllable parameters are set the same. As a consequence, to advance the robot’s autonomy, we designed and developed “mass-transformer" mechanism explained in § 5.4.1.2.

5.5 Discussion

Our results demonstrate that the structural instability, i.e., flagellar buckling enables the robot to steer but it has a limited capability to fully control the robot’s direction. This kind of robots has applications in fishing baits, oil spill cleanup, water quality monitoring, and infrastructure inspection. Our research intends to develop an untethered robot with a soft flagellum that can serve as a simple yet functional testbed for replicating bacterial behavior and comprehending the fundamental mechanics of bacterial locomotion. The movement of the robot should drop in the low Reynolds number regime. Because of this, the robot was designed to be as simple and compact as feasible. To be employed in practice, this type of robots must be enlarged, which is feasible due to the fact that our robot is amenable to dimensional scaling, customization to specific needs, and fabrication via accessible additive manufacturing techniques. In the near future, we will investigate the robot’s scalability.

We integrated “mass-transformer" inside the robot head to enable the robot to reach any 3D destination. This is merely the starting effort showing the potential of our robot’s 3D motion planning capability. Similar to the concept in [1], our robot’s simplicity makes it an ideal platform for rapidly generating a huge quantity of data in a short time, which can be further learned by machine learning algorithms for 3D motion planning [152]. This is made possible by a trait of our robot, which is controllable by updating a sequence of motor rotational speeds.

5.6 Materials and Methods

Our study includes the fabrication and simulation of the robot; therefore, this section is primarily divided into two parts, experiments and simulations. The robot has two major structural components: a hard robot head and a soft polymeric flagellum. Hence, we have § 5.6.1 and § 5.6.2 to describe important details of specialties of the development of our robot. Following these two sections are the numerical implementations of simulating the robot in §5.6.3, §5.6.4, and §5.6.5. Each time step in the simulator is essentially to solve a linear system, which can be under ill-conditioned. To solve the ill-conditioned cases, we introduce the idea of how to use a preconditioning method in §5.6.6. In the end, we report how to plan the motion of the robot in §5.6.7.

5.6.1 Flexible circuit design and fabrication

As mentioned in Section 5.4.1, the compactness of the robot head is critical for efficient propulsion. Aside from the IMU and the ATmega328P chip, which cannot be produced as soft, the remainder of the circuit is soft-made to reduce the head radius. We customized the flexible printed circuit board (PCB) that can actuate and control two motors, as well as read readings from IMU while with pins for extension. Furthermore, we fabricated our soft, custom-designed PCBs

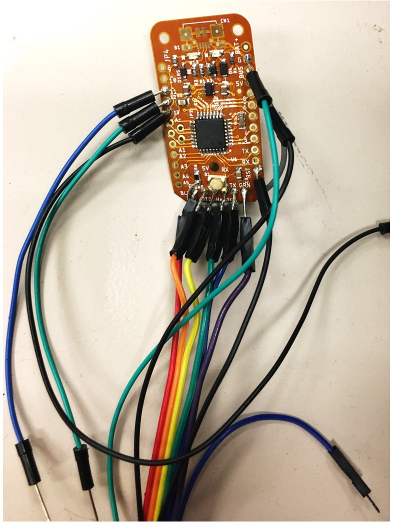

Figure 5.16: Our customized PCB board design and final board fabrication. All operations, from board design in EAGLE to chemical etching and final soldering, are performed in-house (done in the lab). The PCB now functions flawlessly and can be wrapped around the robot's head.

by following the steps below in order. On a sheet of paper, we first adhered a small piece of high-temperature Kapton tape. Then, using solid ink, a Xerox Phaser 8860/8860 printer was used to print the circuit onto the previously prepared paper. The circuit was then etched with a solution consisting of a 1:2 volume ratio of hydrochloric acid and hydrogen peroxide. Other parts will be eliminated with the exception of the circuit covered by solid ink. Lastly, electric components, such as Schottky diodes, resistors, and LEDs, are soldered onto the circuit wires with a hot air gun airflow (Rework Station 500°C Soldering Station Adjustable). The circuit design and the ultimate outcome is depicted in Fig. 5.16. In summary, the circuit that functions flawlessly is completely customized by us.

5.6.2 Rapid prototyping of soft polymeric flagella

The silicone-based rubber VPS (Elite Double 22 and 32; Zhermack) flagellum used in experiments were cast using polyvinyl chloride (PVC) tubes as molds. To fabricate flagella that are neutrally buoyant with glycerin, we added iron fillings while mixing the base and catalyst of VPS (Elite Double 22 and 32; Zhermack) in a 1:1 mass ratio. The normal curing time of VPS is around 20 minutes. Before injecting the mixture into a PVC mold, air bubbles are extracted from the mixture using a dryer vent. Iron fillers were added to the VPS (base) prior to the catalyst to prolong the operational duration by lengthening the curing period. In a mixture of iron and VPS, iron fillings account for 10.3% by mass. We combined fine iron fillers (Eisco Labs) with the VPS base before adding the catalyst. We then injected this final mixture into PVC tubes that had been wrapped around a 3D-printed support. After the mixture has cured, we carefully cut and demolded the PVC tubes.

5.6.3 Fully-implicit DDG-based simulation of system dynamics

We give details of how the robot geometry is discretely represented in § 5.4.2. The force and torque on the cylindrical head are described by Stoke's law. The drag on a rigid sphere moving through a viscous fluid is

$$\mathbf{F}_v = -6\pi\mu R\dot{\mathbf{x}}_1, \tag{5.3}$$

where R is the radius of the sphere, μ is the dynamic viscosity of the glycerin, and $\dot{\mathbf{x}}_1$ is the velocity of the object relative to the fluid. Due to the irregular shape of the robot head and the lack of a closed form expression for the drag, we use a numerical coefficient C_t (to be evaluated through data fitting) to model the drag as

$$\mathbf{F}_v = -C_t 6\pi\mu_0 R\,\dot{\mathbf{x}}_1. \tag{5.4}$$

As shown in Fig. 5.5, the shape of robot head is finalized irregular after numerous trials as designs such as spherical or perfectly cylindrical head requires a larger radius and thus cause a larger drag based on Eq. 5.4.

5.6.4 Regularized Stokeslet Segments method-based hydrodynamics modeling

In this section, we present the relation between the velocity of each node on the flagellum and the corresponding hydrodynamic force applied. Note that RSS is only applied onto $N-3$ nodes (nodes with DOF of $[\mathbf{x}_3, \ldots, \mathbf{x}_{N-1}]$) on the flagellum (in Fig. 5.9).

RSS is a regularization [169] of fundamental singular solution of Stokes flow, which describes the flow induced by a singular point force [170]. In RSS, the velocity $\mathbf{u}(\tilde{\mathbf{x}})$ at the position being evaluated $\mathbf{x}$ because of a regularized force $\mathbf{f}(\mathbf{x})$ applied at $\mathbf{x}$ is as follows

$$8\pi\mu\mathbf{u}(\tilde{\mathbf{x}}) = \left(\frac{1}{R} + \frac{\epsilon^2}{R^3}\right)\mathbf{f}(\mathbf{x}) + \frac{(\mathbf{f}(\mathbf{x})\cdot\mathbf{r})\mathbf{r}}{R^3}, \tag{5.5}$$

where μ is the fluid viscosity, $\mathbf{r} = \tilde{\mathbf{x}} - \mathbf{x}$, $R^2 = |\mathbf{r}|^2 + \epsilon^2$, and ϵ is the regularized parameter, which usually takes the value of ... Consider an edge, for instance, the one connecting nodes $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ with an edge length of Δl. A point on this edge should be located at $\mathbf{x}_\alpha = \mathbf{x}_i - \alpha\mathbf{v}$ (with $\mathbf{v} = \mathbf{x}_i - \mathbf{x}_{i+1}$ and $|\mathbf{v}| = \Delta l$).

As displayed in Fig. 5.17, we assume a linear force density, $\mathbf{f}_\alpha = \mathbf{f}_{r\_i} - \alpha(f_{r\_i+1} - f_{r\_i})$, along the centerline of the segment, such that the velocity at location $\tilde{\mathbf{x}}$ due to the linear force density is

$$8\pi\mu\mathbf{u}(\tilde{\mathbf{x}}) = \Delta l\int_0^1 \left[\frac{1}{R_\alpha} + \frac{\epsilon^2}{R_\alpha^3}\mathbf{f}_\alpha + \frac{(\mathbf{f}_\alpha\cdot\mathbf{r}_\alpha)\mathbf{r}_\alpha}{R_\alpha^3}\right]d\alpha, \tag{5.6}$$

where $\mathbf{r}_\alpha = \tilde{\mathbf{x}} - \mathbf{x}_\alpha$ and $R_\alpha^2 = |\mathbf{r}_\alpha|^2 + \epsilon^2$. Assuming that $\mathbf{f}_\alpha$ is a polynomial of α, the velocity $\mathbf{u}(\tilde{\mathbf{x}})$ in Eq. 5.6 can be formulated as [28]

$$(8\pi\mu/\Delta l)\mathbf{u}(\tilde{\mathbf{x}}) = \mathbf{f}_{r\_i}(T_{0,-1} + \epsilon^2(T_{0,-3}) + \mathbf{f}_{r\_i+1}(T_{1,-1} + \epsilon^2 T_{1,-3}) + \sum_{n=0}^{3}\mathbf{f}_n T_{n,-3}, \tag{5.7}$$

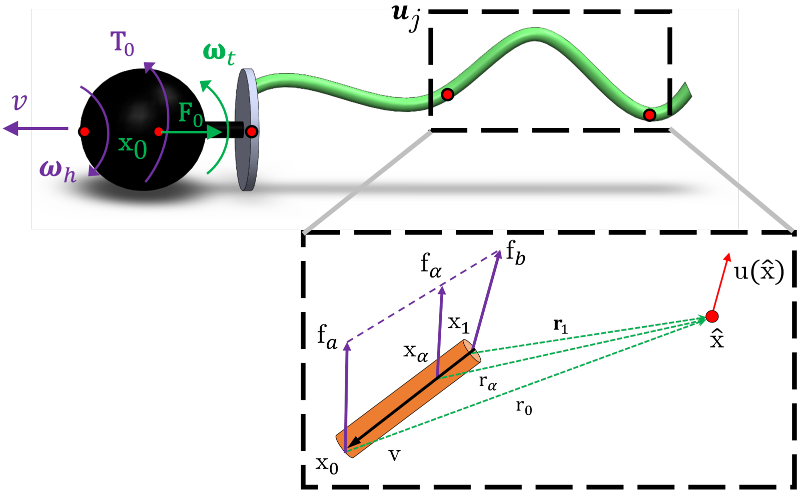

Figure 5.17: Notations associated with the flow $\mathbf{u}(\hat{\mathbf{x}})$ at point $\hat{\mathbf{x}}$ generated by a line segment from $\mathbf{x}_0$ to $\mathbf{x}_1$. Note $\mathbf{r}_\alpha = \hat{\mathbf{x}} - \mathbf{x}_\alpha$ and $\mathbf{v} = \mathbf{x}_0 - \mathbf{x}_1$.

where coefficients $\mathbf{f}_n$ are obtained as follows

$$\mathbf{f}_0 = (\mathbf{f}_i \cdot \mathbf{r}_i)\mathbf{r}_i, \tag{5.8a}$$

$$\mathbf{f}_1 = (\mathbf{f}_i \cdot \mathbf{v})\mathbf{r}_i + (\mathbf{f}_i \cdot \mathbf{r}_i)\mathbf{v} + (\mathbf{f}_{i+1} \cdot \mathbf{r}_i)\mathbf{r}_i, \tag{5.8b}$$

$$\mathbf{f}_2 = (\mathbf{f}_i \cdot \mathbf{v})\mathbf{v} + (\mathbf{f}_{i+1} \cdot \mathbf{r}_i)\mathbf{v} + (\mathbf{f}_{i+1} \cdot \mathbf{v})\mathbf{r}_i, \tag{5.8c}$$

$$\mathbf{f}_3 = (\mathbf{f}_{i+1} \cdot \mathbf{v})\mathbf{v}. \tag{5.8d}$$

The $T_{k,l}$ terms in Stokeslet Segments can then be computed by integrating over

α:

$$T_{0,-1} = \frac{1}{\Delta l}\log[R_\alpha \Delta l + (\mathbf{r}_\alpha \cdot \mathbf{v})]|_0^1, \tag{5.9a}$$

$$T_{0,-3} = -\frac{1}{R_\alpha[R_\alpha \Delta l + (\mathbf{r}_\alpha \cdot \mathbf{v})]}\Bigg|_0^1, \tag{5.9b}$$

$$T_{1,-1} = \frac{R_\alpha}{(\Delta l)^2}\Bigg|_0^1 - \frac{(\mathbf{r}_i \cdot \mathbf{v})}{(\Delta l)^2}T_{0,-1}, \tag{5.9c}$$

$$T_{1,-3} = -\frac{1}{R_\alpha(\Delta l)^2}\Bigg|_0^1 - \frac{(\mathbf{r}_i \cdot \mathbf{v})}{(\Delta l)^2}T_{0,-3}, \tag{5.9d}$$

$$T_{2,-3} = -\frac{\alpha}{R_\alpha(\Delta l)^2}\Bigg|_0^1 + \frac{1}{(\Delta l)^2}T_{0,-1} - \frac{(\mathbf{r}_i \cdot \mathbf{v})}{(\Delta l)^2}T_{1,-3}, \tag{5.9e}$$

$$T_{3,-3} = -\frac{\alpha^2}{R_\alpha(\Delta l)^2}\Bigg|_0^1 + \frac{2}{(\Delta l)^2}T_{1,-1} - \frac{(\mathbf{r}_i \cdot \mathbf{v})}{(\Delta l)^2}T_{2,-3}. \tag{5.9f}$$

Before applying RSS directly to a discrete slender rod-like structure, let us provide the expression of the velocity of a point x̃ along a curve with an arclength s (and arclength of L) on a continuous rod in a force field:

$$8\pi\mu\mathbf{u}(\tilde{\mathbf{x}}) = \int_0^L \left[\frac{1}{R} + \frac{\epsilon^2}{R^3}\mathbf{f} + \frac{(\mathbf{f}\cdot\mathbf{r})\mathbf{r}}{R^3}\right] ds. \tag{5.10}$$

Now, we move on to the case of our discrete robot with $N-3$ nodes and $N-4$ edges on the flagellum. The discretized version of Eq. 5.10 is the following

$$8\pi\mu\mathbf{u}(\tilde{\mathbf{x}}) = \sum_{j=3}^{N-2}(\mathbb{A}_1^j\mathbf{f}_j + \mathbb{A}_2^j\mathbf{f}_{j+1}), \tag{5.11}$$

where $\mathbb{A}_1^j$ and $\mathbb{A}_2^j$ are 3x3 matrices shown below:

$$\mathbb{A}_2^j = |\mathbf{v}_j|[(T_{1,-1}^{j,j+1} + \epsilon^2 T_{1,-3}^{j,j+1}) + T_{1,-3}^{j,j+1}(\mathbf{r}_j\mathbf{r}_j^T) + T_{2,-3}^{j,j+1}(\mathbf{r}_j\mathbf{v}_j^T + \mathbf{v}_j\mathbf{r}_j^T) + T_{3,-3}^{j,j+1}(\mathbf{v}_j\mathbf{v}_j^T)], \tag{5.12a}$$

$$\mathbb{A}_1^j = |\mathbf{v}_j|[(T_{0,-1}^{j,j+1} + \epsilon^2 T_{0,-3}^{j,j+1}) + T_{0,-3}^{j,j+1}(\mathbf{r}_j\mathbf{r}_j^T) + T_{1,-3}^{j,j+1}(\mathbf{r}_j\mathbf{v}_j^T + \mathbf{v}_j\mathbf{r}_j^T) + T_{2,-3}^{j,j+1}(\mathbf{v}_j\mathbf{v}_j^T)] - A_2^j. \tag{5.12b}$$

Eq. 5.12 is then used to build a linear system

$$\mathbf{U} = \mathbb{A}\mathbf{F} \tag{5.13}$$

that expresses the relation between the velocity along the flagellum and the force density applied on it. Here, $\mathbf{U} = [\dot{\mathbf{x}}_3, \dot{\mathbf{x}}_1, \ldots, \dot{\mathbf{x}_{N-1}}]^T$ is the velocity vector of nodes on the flagellum. Note that each node on the flagellum satisfies the no-slip boundary condition, i.e., the velocity at each node equals the viscous fluid velocity there. $\mathbf{F} = [\mathbf{f}_3, \ldots, \mathbf{f}_{N-1}]^T$ is the force density vector. The hydrodynamic force at j-th node is

$$\mathbf{F}_j^t = \mathbf{f}_j \Delta l_j, \tag{5.14}$$

with Δl_j being the Voronoi length (in Fig. 3.1 in [83]).

5.6.5 Numerical implementation of “mass-transformer"

The entire control logic of the system is as follows: the robot moves in a straight line when the flagellum remains straight, buckles its flagellum to turn a small angle (e.g., 15 degrees), and applies the “mass-transformer" to efficiently turn more than 90 degrees as this involves the switching between the two equilibrium states (horizontal and vertical). Back and forth time and energy-consuming experiments can help contribute to gain general feelings about how the robot should be controlled to realize a prescribed trajectory, including the control of “mass-transformer". To precisely control and forecast the robot’s trajectory, the “mass-transformer" must be accurately modelled in order for it to accurately follow any predetermined route. In reality, “mass-transformer" is simulated using the “added mass" method explained in §5.4.2.1.

5.6.6 Preconditioning for correcting the inaccurate simulator

In our simulator, we evaluate the fluid forces, f from the velocity from the velocities, u, at each of the N nodes on a rod. The force vs. velocity relation from Eqs. 3.42 and 2.14 is generated as follows

$$\mathbf{U} = \mathbf{AF}, \quad \text{where} \begin{cases} U_{3(p-1)+k} = u_k^p, \\ F_{3(m-1)+n} = f_n^m. \end{cases} \tag{5.15}$$

In the linear system, $1 \leq p, m \leq$ N (across the nodes) and $1 \leq k, n \leq$ 3 (spanning across three Cartsian dimensions, i.e., x, y, z). The matrix **A** has size $3N \times 3N$, and its expression is provided as follows

The logic is to solve the inverse problem of computing the forces represented by **F** knowing the matrix **A** and velocities denoted by **U**. If **A** is close to singular, numerical issues will appear. While solving this linear system, we follow the strategy below to avoid numerical issues.

We assume that the force varies smoothly with the arc-length, s, and f_n^m can be written as a polynomial of arc-length, s, such that

$$f_n^m = \sum_{r=1}^{P} p_n^r (\frac{s_m}{L})^{r-1}, \tag{5.16}$$

where p_n^r are the coefficients, $P-1$ is the degree of the polynomial. This assumes that the system may be adequately described by P degrees of freedom, similar to a singular value decomposition (SVD) procedure. The value of P is chosen as $\approx N/3$, and s_m is the arc-length parameter associated with node m. The force vector **F** can currently be expressed as

$$\mathbf{F} = \mathbf{CP}^*, \tag{5.17}$$

where $P_{3(r-1)+n} = p_n^r$ and **C** is a $3N \times 3P$ sized matrix. The entries of the matrix **C** can be calculated from Eq. 5.16. The force-velocity relation of Eq. 5.15 can now be written in a decomposed way as follows

$$\mathbf{U} = \mathbf{A}^*\mathbf{P}^*, \quad \text{where} \quad \mathbf{A}^* = \mathbf{AC}, \tag{5.18}$$

which is solved with SVD to obtain $\mathbf{P}^*$. The force vector, **F**, is then readily aquired from Eq. 5.17. The above method is called preconditioning. After applying it, we need to perform error evaluation. The error, E, associated with the above solution can is estimable from

$$E = \frac{||\mathbf{U} - \mathbf{AF}_s||}{||\mathbf{U}||}, \tag{5.19}$$

where $\mathbf{F}_s$ is our solution to Eq. 5.15, and $|| \cdot ||$ stands for the Frobenius norm. This error usually decreases with the number of reduced forces, N^*, increases; nonetheless, computational cost increases with increasing N^*. We observed that $N^* \approx N/3$ yields an error of $E \lessapprox 5\%$ at a reasonable computational cost. This is related to the uncertainty in our measurements of critical buckling velocity, ω_b, and the maximum propulsive force, was estimated to be the mean of this error, E, across all the time steps. Our solutions in this chapter did not need this preconditioning method because **A** is not that close to singular.

5.6.7 Closed-loop 3D motion planning

The overall robot is an underactuated system because only the direction, speed, and the starting orientation of the tail motor are controllable. Without the fabrication error of flagellum, the simulator predicts that the trajectory of the robot is fully controllable by controlling the above parameters. Controlling the tail motor to determine the direction of flagellar buckling and the motor for the realization of a "mass-transformer" is primarily involved. The results will be released in the coming publication.

5.7 Summary and Outlook

The objective of this chapter is to develop robots for scientific exploration, specifically flagellar locomotion in viscous fluids. Experimentally, we designed an untethered underwater robot with a rigid head and flexible flagellum that can swim in a straight path and steer by leveraging structural instability, such as flagellar buckling in a fluid with a low Reynolds number, the same life environment as bacteria. The flagellum is actuated by a motor rotating along the long axis of the flagellum, and flagellar buckling is triggered by a sudden rise in the rotational speed of the flagellum [72]. The development of an untethered underwater robot is intricate due to the numerous challenges involved: 1. making the robot waterproof and neutrally buoyant, 2. developing the robot as compact as possible to decrease its drag, 3. keeping the center of mass and buoyancy as close as possible at the original orientation, and 4. finding a flagellum with right geometrical and physical parameters. We demonstrate that although flagellar buckling can change the robot's movement direction to some amount, it is not powerful enough for the robot to turn over. As a result, theoretically speaking, a desired robot's trajectory is controllable simply using one control parameter - angular velocity. However, due to experimental errors such as the uneven distribution of mass on the flagellum and the robot head, the "mass-transformer" mechanism within the robot head is required for the robot to achieve any preplanned trajectory.

Computationally, we have introduced a DDG-RSS-Stokes' computational framework to simulate the geometrically nonlinear deformation of soft filament and bio-locomotion of uniflagellar soft robot moving in a low Reynolds fluid environment. The computational framework fully accounts for three components: elasticity of the flagellum, long-range hydrodynamic forces on the flagellum, and the flow due to the head. Due to the intricacy of such robots, the robot design process is an arduous procedure involving trials and errors. In contrast to untethered robots, tethered ones are far less challenging. In order to facilitate the understanding of flagellar locomotion and to aid in the search for a motor whose rotational speed exceeds the flagellar buckling threshold, a simple platform is established. Keeping as many degrees of freedom as possible, we designed a gimbal-based system to emulate the performance of untethered locomotion as closely as possible. We first demonstrated that DDG-RSS simulation is capable of modeling the flagellar buckling threshold at the tethered setup. Then, Stokes's law is augmented onto the DDG-RSS framework. The excellent agreement between experimental and simulated data of the robot' speed and the rotational speed of the head and flagellum validates the DDG-RSS-Stokes' architecture. Then, the effect of the head size on the motion of the entire system and the instability of the flagellum is further quantified. The size of the head appears to play a substantial impact in flagellar propulsion, according to our findings. We then studied how this instability may be used to control the swimming direction of uniflagellar bacteria and robots. Supported by the robustness and efficacy of the numerical tools, we performed systematic parameter sweeps to quantify the relationship between the turning angles and the the angular velocity varies with time. The results are used to solve the problem of making a prescribed turn by varying the angular velocity with time. A series of such turns can therefore be employed to follow any desired three-dimensional trajectory.

Such robots will only need a single control input, resulting in a simplified control strategy. Our research on the underlying mechanics and control rule could be applied to bacteria on a small scale. Moreover, we explore two distinct trajectory control schemes that are coupled with the flagellar physical and geometrical properties: 1.after each turn, we allow the flagellum to relax to its unbuckled helical state before considering another turn, and 2.the rotational speed of the motor actuating the flagellum is in the following order: $\omega_1 \ll \omega_t$, $\omega_2 \gg \omega_t$, and $\omega_3 < \omega_t$. In the future, the numerical tool can be combined with machine learning to create more generic model-based control strategies. Importantly, we employed a long-range hydrodynamic force model, and our framework may be extended to investigate multi-flagellated systems. In these more complicated systems, numerical simulation can serve as the data source for data-driven machine learning models. We believe that our numerical results will encourage further research on all of these fronts in order to promote a fundamental understanding of the biophysics of microbes and facilitate the building of superior functional soft robots.

CHAPTER 6

Conclusion

We have presented investigations into several representative projects that dynamically model untethered soft flagellated locomotion in viscous fluids and granular media. We developed a variety of robot designs, including multi-flagellated robots and uni-flagellated robots, for use in the granular media, near the air-fluid interface, and in viscous fluids.

In Ch. 2, we use articulated soft robots as an experimental platform to explore the locomotion in granular media (GM). Numerically, the same DDG-RFT-Stokes' framework is applied again to model the hydrodynamics of locomotion moving through GM. Numerical and experimental results match quantitatively with each other when the number of flagella is two or three, validating the applicability of RFT in GM. However, “stick-slip" or “jamming", i.e., the robot randomly gets stuck at the same position with time passing, happens when the number of flagella turns four or five. The simulator fails to capture this, which proves the limitation of RFT in GM. Moreover, our main finding is that increasing the number of flagella from two to three decreases the speed of the robot. This is kind of counter intuitive, proving the complexity of flexible flagellar locomotion, the competition between the drag and propulsion. This indicates that our simulator is potentially applicable for unknown physics exploration. We find that there is an optimal rotational speed at which maximum efficiency is achieved. This highlights that our validated simulator can be used as a design tool for soft robots.

In Ch. 3, we implemented an Euler-Bernoulli beam-based analytical framework that is both simple and capable of capturing the performance of the robot in GM. Fundamentally, a basic beam-based analytical framework may adequately represent the qualitative performance trend of such complex systems. In any case, this framework is not intended to quantitatively characterize the system; rather, it serves as a ground truth to ensure that the simulation results of the DDG-based simulator make physical sense.

In Ch. 4, we introduce arguably the simplest soft multi-flagellated robot with a single binary control signal, which can move along an arbitrary 2D trajectory near air-fluid interface and at the interface between two fluids. Our work explores the performance of flagellar propulsion near an open boundary instead of a closed boundary such as walls as the former is much less explored[74, 127] compared to the latter while the former has wide applications such as flagellated robots that are used as baits, and tools for oil spill cleanup, water quality monitoring, and infrastructure inspection. The robot is propelled by several motor-actuated soft flagella that are naturally straight but deformed once actuated. The propulsion comes from the structural compliance of polymeric flagella. Then, we develop a DDG-based simulator that can simulate actuators such as a rotating motor that actuates the flagellum. Experimentally, we find that the robot swims in circles near the air-fluid/fluid-fluid interface, the same as bacteria [127]. An important contribution of our simulation in this study is the observation that the actuation (e.g., rotation of motor) can be readily accounted for in a DDG-based framework by updating the undeformed configurations with time, which typically is fixed and assumed to be invariant through the simulation. The hydrodynamic force from the viscous fluid onto the flagella is modelled by RFT. Stokes' law is applied to model the force and torque on the cylindrical robot head. Theoretical analysis and DDG-RFT-Stokes' numerical simulations tell us that this behavior is caused by the uneven distribution of viscosity of the glycerin near the interface. Our simulations show quantitative agreement when compared against experiments. We investigate the performance of the robot versus the number of flagella. Our results demonstrate that though our robot looks nothing like a rod, our simulator is able to capture the dynamics of our robot relatively accurately. Additionally, our robot is capable of following any 2D prescribed trajectory through a binary control signal. Due to the geometrical simplicity, our computational framework is able to run faster than real time on a modern desktop, suggesting that our numerical approach moves a promising step toward a computational framework for soft robotic engineering. In the end, we briefly propose the idea of incorporating machine learning with our fast-running simulator as a handy inverse design tool of flagellated robots. The high computational efficiency of our simulator enables it as a tool of this class of robot.

In Ch. 5, we developed the first untethered underwater robot with a flexible polymeric flagellum that replicate bacterial utilizing structural instability for functionality, i.e., “buckling-to-turn" mechanism. Additionally, we show the effect of flagellar geometrical and physical properties on the performance of flagellar propulsion. To expedite the development of painless untethered robots, we implemented a gimbal-based setup and verified the logic to systematically investigate the underlying mechanics. We showed the effect of the head size on flagellar buckling. In addition, we demonstrate that while bacteria use buckling to steer, flagellar buckling is likely insufficient for a robust robotic system to follow any 3D prescribed route, especially when flipping itself over. As a result, we develop a “mass-transformer" mechanism to make the robot system robust and be able to reach a destination in the 3D space. Additionally, we are the first to demonstrate that the state-of-the-art continuous hydrodynamic model Regularized Stokeslet Segment (RSS) method can accurately model the hydrodynamic force on a rotating flagellum on an untethered robot (with a rigid head). We develop a numerical framework that incorporates (i) DDG to account for the elasticity of soft flagellum, (ii) RSS for the long term hydrodynamic flow by the rotating helical flagellum, and (iii) Stokes' law for the hydrodynamics induced by a spherical head. Our modular robot design enables researchers to use it as testbeds for studying generic flagellar propulsion. The “mass-transformer" mechanism together with simple flagellar buckling control scheme can be used for developing autonomous underwater robots for exploration and exploitation of new environments.

In conclusion, we developed various untethered soft multi- and uni-flagellated robots that can move or navigate through granular media or viscous fluids. They have significant application potential for oil spill cleanup, monitoring water quality, and infrastructure inspection. Researchers can use them as simple but practical testbeds to learn the mechanics of general flagellar locomotion in granular media and viscous fluids and to investigate how it is affected by the robot's physical and geometrical parameters and the environment, such as the density of granular media, the viscosity of the fluid, and the presence of boundaries. Simulators based on discrete differential geometry (DDG) were proposed concurrently. Due to the rapid execution and adaptability of our simulation frameworks, researchers are able to investigate flagellar locomotion, utilize our simulator to quickly generate a huge quantity of data, and combine machine learning algorithms to analyze the data for inverse design and control of robots.

[1] Kwangmin Son, Jeffrey S Guasto, and Roman Stocker. Bacteria can exploit a flagellar buckling instability to change direction. *Nature physics*, 9(8):494–498, 2013.

[2] Julio Bastos-Arrieta, Ainhoa Revilla-Guarinos, William E Uspal, and Juliane Simmchen. Bacterial biohybrid microswimmers. *Frontiers in Robotics and AI*, page 97, 2018.

[3] Bian Qian, Thomas R Powers, and Kenneth S Breuer. Shape transition and propulsive force of an elastic rod rotating in a viscous fluid. *Physical review letters*, 100(7):078101, 2008.

[4] Langquan Shui, Liangliang Zhu, Zhe Yang, Yilun Liu, and Xi Chen. Energy efficiency of mobile soft robots. *Soft Matter*, 13(44):8223–8233, 2017.

[5] James Lighthill. Flagellar hydrodynamics. *SIAM review*, 18(2):161–230, 1976.

[6] Ryan D Maladen, Yang Ding, Chen Li, and Daniel I Goldman. Undulatory swimming in sand: subsurface locomotion of the sandfish lizard. *science*, 325(5938):314–318, 2009.

[7] Andrew D Marchese, Robert K Katzschmann, and Daniela Rus. A recipe for soft fluidic elastomer robots. *Soft robotics*, 2(1):7–25, 2015.

[8] Ruxu Du, Zheng Li, Kamal Youcef-Toumi, and Pablo Valdivia y Alvarado. *Robot fish: Bio-inspired fishlike underwater robots*. Springer, 2015.

[9] Chenglei Wang, Hui Tang, and Xing Zhang. Fluid-structure interaction of bio-inspired flexible slender structures: a review of selected topics. *Bioinspiration & Biomimetics*, 2022.

[10] Julian Colorado, Antonio Barrientos, Claudio Rossi, and Kenny S Breuer. Biomechanics of smart wings in a bat robot: morphing wings using sma actuators. *Bioinspiration & biomimetics*, 7(3):036006, 2012.

[11] Matthew A Woodward and Metin Sitti. Multimo-bat: A biologically inspired integrated jumping–gliding robot. *The International Journal of Robotics Research*, 33(12):1511–1529, 2014.

[12] Jennifer Frame, Nick Lopez, Oscar Curet, and Erik D Engeberg. Thrust force characterization of free-swimming soft robotic jellyfish. *Bioinspiration & biomimetics*, 13(6):064001, 2018.

[13] Ziyu Ren, Wenqi Hu, Xiaoguang Dong, and Metin Sitti. Multi-functional soft-bodied jellyfish-like swimming. *Nature communications*, 10(1):1–12, 2019.

[14] Marcello Calisti, Michele Giorelli, Guy Levy, Barbara Mazzolai, B Hochner, Cecilia Laschi, and Paolo Dario. An octopus-bioinspired solution to movement and manipulation for soft robots. *Bioinspiration & biomimetics*, 6(3):036002, 2011.

[15] Caleb Christianson, Yi Cui, Michael Ishida, Xiaobo Bi, Qiang Zhu, Geno Pawlak, and Michael T Tolley. Cephalopod-inspired robot capable of cyclic jet propulsion through shape change. *Bioinspiration & Biomimetics*, 16(1):016014, 2020.

[16] Michael Silverman and Melvin I Simon. Bacterial flagella. *Annual review of microbiology*, 31(1):397–419, 1977.

[17] Centers for Disease Control and Prevention, howpublished = https://www.cdc.gov/.

[18] Lactobacillus acidophilus, howpublished = https://medlineplus.gov/druginfo/natural/790.html.

[19] Margaret Gemmell and W Hodgkiss. The physiological characters and flagellar arrangement of motile homofermentative lactobacilli. *Microbiology*, 35(3):519–526, 1964.

[20] Edward M Purcell. Life at low reynolds number. *American journal of physics*, 45(1):3–11, 1977.

[21] Eric Lauga. Life around the scallop theorem. *Soft Matter*, 7(7):3060–3065, 2011.

[22] James G Mitchell and Kazuhiro Kogure. Bacterial motility: links to the environment and a driving force for microbial physics. *FEMS Microbiol. Ecol.*, 55(1):3–16, January 2006.

[23] Antoni Van Leeuwenhoek. Observations, communicated to the publisher by mr. antony van leewenhoeck, in a dutch letter of the 9th octob. 1676. here english'd: concerning little animals by him observed in rain-well-sea- and snow water; as also in water wherein pepper had lain infused. *Philosophical Transactions of the Royal Society of London*, 12(133):821–831, 1677.

[24] Robert M Macnab and May Kihara Ornston. Normal-to-curly flagellar transitions and their role in bacterial tumbling. stabilization of an alternative quaternary structure by mechanical force. *Journal of Molecular Biology*, 112(1):1–30, 1977.

[25] Wanho Lee, Yongsam Kim, Boyce E Griffith, and Sookkyung Lim. Bacterial flagellar bundling and unbundling via polymorphic transformations. *Physical Review E*, 98(5):052405, 2018.

[26] Reinhard Vogel and Holger Stark. Motor-driven bacterial flagella and buckling instabilities. *The European Physical Journal E*, 35(2):1–15, 2012.

[27] Diego Lopez and Eric Lauga. Dynamics of swimming bacteria at complex interfaces. *Physics of Fluids*, 26(7):400–412, 2014.

[28] Ricardo Cortez. Regularized stokeslet segments. *Journal of Computational Physics*, 375:783–796, 2018.

[29] Carmel Majidi. Soft robotics: a perspective—current trends and prospects for the future. *Soft Robotics*, 1(1):5–11, 2014.

[30] Mohammad Khalid Jawed. *Geometrically Nonlinear Configurations in Rod-like Structures*. PhD thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 2016.

[31] Pedro M Reis. A perspective on the revival of structural (in) stability with novel opportunities for function: from buckliphobia to buckliphilia. *Journal of Applied Mechanics*, 82(11):111001, 2015.

[32] Robert K Katzschmann, Joseph DelPreto, Robert MacCurdy, and Daniela Rus. Exploration of underwater life with an acoustically controlled soft robotic fish. *Science Robotics*, 3(16):eaar3449, 2018.

[33] Xiaonan Huang, Kitty Kumar, Mohammad K Jawed, Amir Mohammadi Nasab, Zisheng Ye, Wanliang Shan, and Carmel Majidi. Highly dynamic shape memory alloy actuator for fast moving soft robots. *Advanced Materials Technologies*, 4(4):1800540, 2019.

[34] Jesse L Silverberg, Arthur A Evans, Lauren McLeod, Ryan C Hayward, Thomas Hull, Christian D Santangelo, and Itai Cohen. Using origami design principles to fold reprogrammable mechanical metamaterials. *science*, 345(6197):647–650, 2014.

[35] John A Rogers, Takao Someya, and Yonggang Huang. Materials and mechanics for stretchable electronics. *science*, 327(5973):1603–1607, 2010.

[36] Zhiyuan Yang, Dongsheng Chen, David J Levine, and Cynthia Sung. Origami-inspired robot that swims via jet propulsion. *IEEE Robotics and Automation Letters*, 6(4):7145–7152, 2021.

[37] Robert F Shepherd, Filip Ilievski, Wonjae Choi, Stephen A Morin, Adam A Stokes, Aaron D Mazzeo, Xin Chen, Michael Wang, and George M Whitesides. Multigait soft robot. *Proceedings of the national academy of sciences*, 108(51):20400–20403, 2011.

[38] Poojitha Vurtur Badarinath, Maria Chierichetti, and Fatemeh Davoudi Kakhki. A machine learning approach as a surrogate for a finite element analysis: Status of research and application to one dimensional systems. *Sensors*, 21(5):1654, 2021.

[39] Ouajdi Felfoul, Mahmood Mohammadi, Samira Taherkhani, Dominic De Lanauze, Yong Zhong Xu, Dumitru Loghin, Sherief Essa, Sylwia Jancik, Daniel Houle, Michel Lafleur, et al. Magneto-aerotactic bacteria deliver drug-containing nanoliposomes to tumour hypoxic regions. *Nature nanotechnology*, 11(11):941–947, 2016.

[40] Linda Turner, William S Ryu, and Howard C Berg. Real-time imaging of fluorescent flagellar filaments. *Journal of bacteriology*, 182(10):2793–2801, 2000.

[41] Yunus Alapan, Oncay Yasa, Oliver Schauer, Joshua Giltinan, Ahmet F Tabak, Victor Sourjik, and Metin Sitti. Soft erythrocyte-based bacterial microswimmers for cargo delivery. *Science robotics*, 3(17):eaar4423, 2018.

[42] Mariana Medina-Sánchez, Lukas Schwarz, Anne K Meyer, Franziska Hebenstreit, and Oliver G Schmidt. Cellular cargo delivery: Toward assisted fertilization by sperm-carrying micromotors. *Nano letters*, 16(1):555–561, 2016.

[43] Murat Kaynak, Adem Ozcelik, Amir Nourhani, Paul E Lammert, Vincent H Crespi, and Tony Jun Huang. Acoustic actuation of bioinspired microswimmers. *Lab on a Chip*, 17(3):395–400, 2017.

[44] Cornel Dillinger, Nitesh Nama, and Daniel Ahmed. Ultrasound-activated ciliary bands for microrobotic systems inspired by starfish. *Nature communications*, 12(1):1–11, 2021.

[45] Chaolei Huang, Jiu-an Lv, Xiaojun Tian, Yuechao Wang, Yanlei Yu, and Jie Liu. Miniaturized swimming soft robot with complex movement actuated and controlled by remote light signals. *Scientific reports*, 5(1):1–8, 2015.

[46] Ahmed Mourran, Hang Zhang, Rostislav Vinokur, and Martin Möller. Soft microrobots employing nonequilibrium actuation via plasmonic heating. *Advanced Materials*, 29(2):1604825, 2017.

[47] Baohu Dai, Jizhuang Wang, Ze Xiong, Xiaojun Zhan, Wei Dai, Chien-Cheng Li, Shien-Ping Feng, and Jinyao Tang. Programmable artificial phototactic microswimmer. *Nature nanotechnology*, 11(12):1087–1092, 2016.

[48] Stefano Palagi, Andrew G Mark, Shang Yik Reigh, Kai Melde, Tian Qiu, Hao Zeng, Camilla Parmeggiani, Daniele Martella, Alberto Sanchez-Castillo, Nadia Kapernaum, et al. Structured light enables biomimetic swimming and versatile locomotion of photoresponsive soft microrobots. *Nature materials*, 15(6):647–653, 2016.

[49] Debora Schamel, Andrew G Mark, John G Gibbs, Cornelia Miksch, Konstantin I Morozov, Alexander M Leshansky, and Peer Fischer. Nanopropellers and their actuation in complex viscoelastic media. *ACS nano*, 8(9):8794–8801, 2014.

[50] Li Zhang, Jake J. Abbott, Lixin Dong, Bradley E. Kratochvil, Dominik Bell, and Bradley J. Nelson. Artificial bacterial flagella: Fabrication and magnetic control. *Applied Physics Letters*, 94(6):064107, 2009.

[51] Soichiro Tottori, Li Zhang, Famin Qiu, Krzysztof K. Krawczyk, Alfredo Franco-Obregón, and Bradley J. Nelson. Magnetic helical micromachines: Fabrication, controlled swimming, and cargo transport. *Advanced Materials*, 24(6):811–816, 2012.

[52] Shuaizhong Zhang, Zhiwei Cui, Ye Wang, and Jaap den Toonder. Metachronal -cilia for on-chip integrated pumps and climbing robots. *ACS Applied Materials & Interfaces*, 13(17):20845–20857, 2021.

[53] Christian Peters, Olgaç Ergeneman, Pedro D Wendel García, Michelle Müller, Salvador Pané, Bradley J Nelson, and Christofer Hierold. Superparamagnetic twist-type actuators with shape-independent magnetic properties and surface functionalization for advanced biomedical applications. *Advanced Functional Materials*, 24(33):5269–5276, 2014.

[54] Sylvain Martel, Charles C Tremblay, Serge Ngakeng, and Guillaume Langlois. Controlled manipulation and actuation of micro-objects with magnetotactic bacteria. *Applied Physics Letters*, 89(23):233904, 2006.

[55] Veronika Magdanz, Samuel Sanchez, and Oliver G Schmidt. Development of a sperm-flagella driven micro-bio-robot. *Advanced materials*, 25(45):6581–6588, 2013.

[56] Changyou Chen, Pingping Wang, Long-Fei Wu, and Tao Song. Biocompatibility of marine magnetotactic ovoid strain mo-1 for in vivo application. *Journal of Oceanology and Limnology*, 39(6):2107–2115, 2021.

[57] Sung Jun Park, Seung-Hwan Park, Sunghoon Cho, Deok-Mi Kim, Yeonkyung Lee, Seong Young Ko, Yeongjin Hong, Hyon E Choy, Jung-Joon Min, Jong-Oh Park, et al. New paradigm for tumor theranostic methodology using bacteria-based microrobot. *Scientific reports*, 3(1):1–8, 2013.

[58] Sunghoon Cho, Sung Jun Park, Seong Young Ko, Jong-Oh Park, and Sukho Park. Development of bacteria-based microrobot using biocompatible poly (ethylene glycol). *Biomedical microdevices*, 14(6):1019–1025, 2012.

[59] Daechul Park, Sung Jun Park, Sunghoon Cho, Yeonkyung Lee, Yu Kyung Lee, Jung-Joon Min, Bang Ju Park, Seong Young Ko, Jong-Oh Park, and Sukho Park. Motility analysis of bacteria-based microrobot (bacteriobot) using chemical gradient microchamber. *Biotechnology and Bioengineering*, 111(1):134–143, 2014.

[60] Douglas B Weibel, Piotr Garstecki, Declan Ryan, Willow R DiLuzio, Michael Mayer, Jennifer E Seto, and George M Whitesides. Microoxen: Microorganisms to move microscale loads. *Proceedings of the National Academy of Sciences*, 102(34):11963–11967, 2005.

[61] Mahama A Traoré, Ali Sahari, and Bahareh Behkam. Computational and experimental study of chemotaxis of an ensemble of bacteria attached to a microbead. *Physical Review E*, 84(6):061908, 2011.

[62] Fatma Zeynep Temel, Aydek Gokce Erman, and Serhat Yesilyurt. Characterization and modeling of biomimetic untethered robots swimming in viscous fluids inside circular channels. *IEEE/ASME Transactions on Mechatronics*, 19(5):1562–1573, 2013.

[63] Alperen Acemoglu and Serhat Yesilyurt. Effects of poiseuille flows on swimming of magnetic helical robots in circular channels. *Microfluidics and Nanofluidics*, 19(5):1109–1122, 2015.

[64] Zhou Ye, Stéphane Régnier, and Metin Sitti. Rotating magnetic miniature swimming robots with multiple flexible flagella. *IEEE Transactions on Robotics*, 30(1):3–13, 2013.

[65] Ali Oulmas, Nicolas Andreff, and Stéphane Régnier. Closed-loop 3d path following of scaled-up helical microswimmers. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1725–1730. IEEE, 2016.

[66] Yayun Du, Andrew Miller, and Mohammad Khalid Jawed. Simple flagellated soft robot for locomotion near air-fluid interface. In *2021 IEEE*

4th International Conference on Soft Robotics (RoboSoft), pages 472–478. IEEE, 2021.

[67] Yayun Du, Jacqueline Lam, Karunesh Sachanandani, and M Khalid Jawed. Modeling the locomotion of articulated soft robots in granular medium. *IEEE Robotics and Automation Letters*, 7(3):6495–6502, 2022.

[68] Sangmin Lim, Achyuta Yadunandan, and Mohammad Khalid Jawed. Bacteria inspired multi-flagella propelled soft robot at low reynolds number. *arXiv preprint arXiv:2111.12793*, 2021.

[69] Aydek Gokce Erman and Serhat Yesilyurt. Swimming of onboard-powered autonomous robots in viscous fluid filled channels. In *2011 IEEE International Conference on Mechatronics*, pages 348–353. IEEE, 2011.

[70] Saúl Gómez, Francisco A Godínez, Eric Lauga, and Roberto Zenit. Helical propulsion in shear-thinning fluids. *Journal of Fluid Mechanics*, 812, 2017.

[71] Tiantian Xu, Gilgueng Hwang, Nicolas Andreff, and Stéphane Régnier. Modeling and swimming property characterizations of scaled-up helical microswimmers. *IEEE/ASME Transactions on Mechatronics*, 19(3):1069–1079, 2013.

[72] Mohammad K Jawed, Noor K Khouri, F Da, E Grinspun, and Pedro M Reis. Propulsion and instability of a flexible helical rod rotating in a viscous fluid. *Physical review letters*, 115(16):168101, 2015.

[73] Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly, and Eitan Grinspun. Discrete elastic rods. *ACM Transactions on Graphics*, 27(3):63, 2008.

[74] MK Jawed and Pedro M Reis. Dynamics of a flexible helical filament rotating in a viscous fluid near a rigid boundary. *Physical Review Fluids*, 2(3):034101, 2017.

[75] Weicheng Huang and MK Jawed. Numerical exploration on buckling instability for directional control in flagellar propulsion. *Soft matter*, 16(3):604–613, 2020.

[76] Guoying Gu, Jiang Zou, Ruike Zhao, Xuanhe Zhao, and Xiangyang Zhu. Soft wall-climbing robots. *Science Robotics*, 3(25):eaat2874, 2018.

[77] Yichen Yan, Yusen Zhao, Yousif Alsaid, Bowen Yao, Yucheng Zhang, Shuwang Wu, and Ximin He. Artificial phototropic systems for enhanced light harvesting based on a liquid crystal elastomer. *Advanced Intelligent Systems*, 3(10):2000234, 2021.

[78] Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. Discrete viscous threads. *ACM Transactions on Graphics*, 29(4):1–10, 2010.

[79] Gustav Kirchhoff. Uber das gleichgewicht und die bewegung eines unendlich dunnen elastischen stabes. *J. reine angew. Math.*, 56:285–313, 1859.

[80] Leonhard Euler. *Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes sive solutio problematis isoperimetrici latissimo sensu accepti*, volume 1. Springer Science & Business Media, 1952.

[81] Gustav Kirchhoff. Ueber das gleichgewicht und die bewegung eines unendlich dünnen elastischen stabes. 1859.

[82] Eugene Cosserat. *Théorie des corps déformables*. Librairie Scientifique A. Hermann et Fils, 1909.

[83] M Khalid Jawed, Alyssa Novelia, and Oliver M O'Reilly. *A primer on the kinematics of discrete elastic rods*. Springer, 2018.

[84] Michael T Tolley, Robert F Shepherd, Bobak Mosadegh, Kevin C Galloway, Michael Wehner, Michael Karpelson, Robert J Wood, and George M Whitesides. A resilient, untethered soft robot. *Soft Robotics*, 1(3):213–223, 2014.

[85] Daniela Rus and Michael T Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475, 2015.

[86] Akanksha Thawani and Mahesh S Tirumkudulu. Trajectory of a model bacterium. *Journal of Fluid Mechanics*, 835:252–270, 2018.

[87] Federico Renda, Francesco Giorgio-Serchi, Frederic Boyer, Cecilia Laschi, Jorge Dias, and Lakmal Seneviratne. A unified multi-soft-body dynamic model for underwater soft robots. *The International Journal of Robotics Research*, 37(6):648–666, 2018.

[88] Andrew A Biewener. Biomechanics of mammalian terrestrial locomotion. *Science*, 250(4984):1097–1103, 1990.

[89] Michael H Dickinson, Claire T Farley, Robert J Full, MAR Koehl, Rodger Kram, and Steven Lehman. How animals move: an integrative view. *science*, 288(5463):100–106, 2000.

[90] Graham K Taylor, Robert L Nudds, and Adrian LR Thomas. Flying and swimming animals cruise at a strouhal number tuned for high power efficiency. *Nature*, 425(6959):707–711, 2003.

[91] Tingnan Zhang and Daniel I Goldman. The effectiveness of resistive force theory in granular locomotion. *Physics of Fluids*, 26(10):101308, 2014.

[92] James Gray and GJ Hancock. The propulsion of sea-urchin spermatozoa. *Journal of Experimental Biology*, 32(4):802–814, 1955.

[93] Baptiste Darbois Texier, Alejandro Ibarra, and Francisco Melo. Helical locomotion in a granular medium. *Physical review letters*, 119(6):068003, 2017.

[94] Eric Lauga and Thomas R Powers. The hydrodynamics of swimming microorganisms. *Reports on progress in physics*, 72(9):096601, 2009.

[95] Olivier Goury and Christian Duriez. Fast, generic, and reliable control and simulation of soft robots using model order reduction. *IEEE Transactions on Robotics*, 34(6):1565–1576, 2018.

[96] Jonathan Hiller and Hod Lipson. Dynamic simulation of soft multimaterial 3d-printed objects. *Soft robotics*, 1(1):88–101, 2014.

[97] Federico Renda, Vito Cacucciolo, Jorge Dias, and Lakmal Seneviratne. Discrete cosserat approach for soft robot dynamics: A new piece-wise constant strain model with torsion and shears. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5495–5502. IEEE, 2016.

[98] Michael W Hannan and Ian D Walker. Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots. *Journal of robotic systems*, 20(2):45–63, 2003.

[99] Bryan A Jones and Ian D Walker. Kinematics for multisection continuum robots. *IEEE Transactions on Robotics*, 22(1):43–55, 2006.

[100] Bruce Rodenborn, Chih-Hung Chen, Harry L Swinney, Bin Liu, and HP Zhang. Propulsion of microorganisms by a helical flagellum. *Proceedings of the National Academy of Sciences*, 110(5):E338–E347, 2013.

[101] David L Hu, Jasmine Nirody, Terri Scott, and Michael J Shelley. The mechanics of slithering locomotion. *Proceedings of the National Academy of Sciences*, 106(25):10081–10085, 2009.

[102] Ryan D Maladen, Yang Ding, Paul B Umbanhowar, Adam Kamor, and Daniel I Goldman. Mechanical models of sandfish locomotion reveal principles of high performance subsurface sand-swimming. *Journal of The Royal Society Interface*, 8(62):1332–1345, 2011.

[103] Mohammad K Jawed, Fang Da, Jungseock Joo, Eitan Grinspun, and Pedro M Reis. Coiling of elastic rods on rigid substrates. *Proceedings of the National Academy of Sciences*, 111(41):14663–14668, 2014.

[104] Changyeob Baek, Andrew O Sageman-Furnas, Mohammad K Jawed, and Pedro M Reis. Form finding in elastic gridshells. *Proceedings of the National Academy of Sciences*, 115(1):75–80, 2018.

[105] Jesús Pérez, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, José A Canabal, Robert Sumner, and Miguel A Otaduy. Design and fabrication of flexible rod meshes. *ACM Transactions on Graphics (TOG)*, 34(4):138, 2015.

[106] Seungmoon Song, Joohyung Kim, and Katsu Yamane. Development of a bipedal robot that walks like an animation character. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3596–3602. IEEE, 2015.

[107] Vittorio Megaro, Bernhard Thomaszewski, Maurizio Nitti, Otmar Hilliges, Markus Gross, and Stelian Coros. Interactive design of 3d-printable robotic creatures. *ACM Transactions on Graphics*, 34(6):216, 2015.

[108] Jay T Miller, Arnaud Lazarus, Basile Audoly, and Pedro M Reis. Shapes of a suspended curly hair. *Physical review letters*, 112(6):068103, 2014.

[109] Ryan D Maladen, Yang Ding, Paul B Umbanhowar, and Daniel I Goldman. Undulatory swimming in sand: experimental and simulation studies of a robotic sandfish. *The International Journal of Robotics Research*, 30(7):793–805, 2011.

[110] Andrew Choi, Dezhong Tong, Mohammad K Jawed, and Jungseock Joo. Implicit contact model for discrete elastic rods in knot tying. *Journal of Applied Mechanics*, 88(5), 2021.

[111] Yayun Du, Andrew Miller, and M Khalid Jawed. Mechanics-based analysis on flagellated robots. *arXiv preprint arXiv:2112.13927*, 2021.

[112] AE Hosoi and Daniel I Goldman. Beneath our feet: strategies for locomotion in granular media. *Annual review of fluid mechanics*, 47:431–453, 2015.

[113] George V Lauder, Erik J Anderson, James Tangorra, and Peter GA Madden. Fish biorobotics: kinematics and hydrodynamics of self-propulsion. *Journal of experimental biology*, 210(16):2767–2780, 2007.

[114] Kevin Y Ma, Pakpong Chirarattananon, Sawyer B Fuller, and Robert J Wood. Controlled flight of a biologically inspired, insect-scale robot. *Science*, 340(6132):603–607, 2013.

[115] Einar Leifson, BJ Cosenza, R Murchelano, and RC Cleverdon. Motile marine bacteria i: Techniques, ecology, and general characteristics. *Journal of Bacteriology*, 87(3):652–666, 1964.

[116] Geoffrey Ingram Taylor. Analysis of the swimming of microscopic organisms. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 209(1099):447–461, 1951.

[117] Seishi Kudo, Norio Imai, Megumi Nishitoba, Shigeru Sugiyama, and Yukio Magariyama. Asymmetric swimming pattern of vibrio alginolyticus cells with single polar flagella. *FEMS microbiology letters*, 242(2):221–225, 2005.

[118] Mikako Fujii, Satoshi Shibata, and Shin-Ichi Aizawa. Polar, peritrichous, and lateral flagella belong to three distinguishable flagellar families. *Journal of molecular biology*, 379(2):273–283, 2008.

[119] Giovanni Noselli, Alfred Beran, Marino Arroyo, and Antonio DeSimone. Swimming euglena respond to confinement with a behavioural change enabling effective crawling. *Nature physics*, 15(5):496–502, 2019.

[120] Ricardo Cortez. The method of regularized stokeslets. *SIAM Journal on Scientific Computing*, 23(4):1204–1225, 2001.

[121] MunJu Kim and Thomas R Powers. Deformation of a helical filament by flow and electric or magnetic fields. *Physical Review E*, 71(2):021914, 2005.

[122] Allison P Berke, Linda Turner, Howard C Berg, and Eric Lauga. Hydrodynamic attraction of swimming microorganisms by surfaces. *Physical Review Letters*, 101(3):038102, 2008.

[123] Saverio E Spagnolie and Eric Lauga. Hydrodynamics of self-propulsion near a boundary: predictions and accuracy of far-field approximations. *Journal of Fluid Mechanics*, 700:105–147, 2012.

[124] Costanza Armanini, Madiha Farman, Marcello Calisti, Francesco Giorgio-Serchi, Cesare Stefanini, and Federico Renda. Flagellate underwater robotics at macroscale: Design, modeling, and characterization. *IEEE Transactions on Robotics*, 2021.

[125] Yayun Du, Jacqueline Lam, Karunesh Sachanandani, and Mohammad Khalid Jawed. Modeling the locomotion of articulated soft robots in granular medium. *arXiv preprint arXiv:2103.03993*, 2021.

[126] Gustav Kirchhoff. *Über das Gleichgewicht und die Bewegung eines unendlich dünnen elastischen Stabes*. De Gruyter, 2020.

[127] Eric Lauga, Willow R DiLuzio, George M Whitesides, and Howard A Stone. Swimming in circles: motion of bacteria near solid boundaries. *Biophysical journal*, 90(2):400–412, 2006.

[128] Shashank Agarwal, Andras Karsai, Daniel I Goldman, and Ken Kamrin. Surprising simplicity in the modeling of dynamic granular intrusion. *Science Advances*, 7(17):eabe0631, 2021.

[129] Nick Gravish and Daniel I Goldman. Effect of volume fraction on granular avalanche dynamics. *Physical Review E*, 90(3):032202, 2014.

[130] Istvan Albert, JG Sample, AJ Morss, S Rajagopalan, A-L Barabási, and P Schiffer. Granular drag on a discrete object: Shape effects on jamming. *Physical review E*, 64(6):061303, 2001.

[131] Naïs Coq, Olivia Du Roure, Joel Marthelot, Denis Bartolo, and Marc Fermigier. Rotational dynamics of a soft filament: Wrapping transition and propulsive forces. *Physics of Fluids*, 20(5):051703, 2008.

[132] Basile Audoly and Yves Pomeau. Elasticity and geometry. In *Peyresq Lectures on Nonlinear Phenomena*, pages 1–35. World Scientific, 2000.

[133] Olaf Schenk and Klaus Gärtner. Solving unsymmetric sparse systems of linear equations with pardiso. In *International Conference on Computational Science*, pages 355–363. Springer, 2002.

[134] Julian Panetta, MINA Konaković-Luković, Florin Isvoranu, Etienne Bouleau, and Mark Pauly. X-shells: A new class of deployable beam structures. *ACM Transactions on Graphics (TOG)*, 38(4):1–15, 2019.

[135] Arnaud Lazarus, Jay T Miller, Matthew M Metlitz, and Pedro M Reis. Contorting a heavy and naturally curved elastic rod. *Soft Matter*, 9(34):8274–8281, 2013.

[136] William Dembski and Jonathan Wells. The design of life: Discovering signs of intelligence in living systems, 2008.

[137] Laurence Lemelle, Jean-François Palierne, Elodie Chatre, and Christophe Place. Counterclockwise circular motion of bacteria swimming at the air-liquid interface. *Journal of bacteriology*, 192(23):6307–6308, 2010.

[138] Li Zhang, Jake J Abbott, Lixin Dong, Bradley E Kratochvil, Dominik Bell, and Bradley J Nelson. Artificial bacterial flagella: Fabrication and magnetic control. *Applied Physics Letters*, 94(6):064107, 2009.

[139] Sangwon Kim, Seungmin Lee, Jeonghun Lee, Bradley J Nelson, Li Zhang, and Hongsoo Choi. Fabrication and manipulation of ciliary microrobots

with non-reciprocal magnetic actuation. *Scientific reports*, 6:30713, 2016.

[140] Russ Tedrake. Underactuated robotics: Learning, planning, and control for efficient and agile machines course notes for mit 6.832. *Working draft edition*, 3, 2009.

[141] MunJu Kim, James C Bird, Annemarie J Van Parys, Kenneth S Breuer, and Thomas R Powers. A macroscopic scale model of bacterial flagellar bundling. *Proceedings of the National Academy of Sciences*, 100(26):15481–15485, 2003.

[142] Mostyn T Brown, Bradley C Steel, Claudio Silvestrin, David A Wilkinson, Nicolas J Delalez, Craig N Lumb, Boguslaw Obara, Judith P Armitage, and Richard M Berry. Flagellar hook flexibility is essential for bundle formation in swimming escherichia coli cells. *Journal of bacteriology*, 194(13):3495–3501, 2012.

[143] Maira A Constantino, Mehdi Jabbarzadeh, Henry C Fu, Zeli Shen, James G Fox, Freddy Haesebrouck, Sara K Linden, and Rama Bansil. Bipolar lophotrichous helicobacter suis combine extended and wrapped flagella bundles to exhibit multiple modes of motility. *Scientific reports*, 8(1):1–15, 2018.

[144] Marina Sidortsov, Yakov Morgenstern, and Avraham Be'Er. Role of tumbling in bacterial swarming. *Physical Review E*, 96(2):022407, 2017.

[145] Nicholas C Darnton and Howard C Berg. Force-extension measurements on bacterial flagella: triggering polymorphic transformations. *Biophysical journal*, 92(6):2230–2236, 2007.

[146] Howard C Berg and Douglas A Brown. Chemotaxis in escherichia coli analysed by three-dimensional tracking. *Nature*, 239(5374):500–504, 1972.

[147] Yayun Du, Andrew Miller, and Mohammad Khalid Jawed. Simple flagellated soft robot for locomotion near air-fluid interface. In *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*, pages 472–478, 2021.

[148] Robert E Johnson. An improved slender-body theory for stokes flow. *Journal of Fluid Mechanics*, 99(2):411–431, 1980.

[149] Mohammad K Jawed and Pedro M Reis. Deformation of a soft helical filament in an axial flow at low reynolds number. *Soft matter*, 12(6):1898–1905, 2016.

[150] Emily E Riley, Debasish Das, and Eric Lauga. Swimming of peritrichous bacteria is enabled by an elastohydrodynamic instability. *Scientific reports*, 8:10728, 2018.

[151] Imke Spöring, Vincent A Martinez, Christian Hotz, Jana Schwarz-Linek, Keara L Grady, Josué M Nava-Sedeño, Teun Vissers, Hanna M Singer, Manfred Rohde, Carole Bourquin, et al. Hook length of the bacterial flagellum is optimized for maximal stability of the flagellar bundle. *PLoS biology*, 16(9):e2006989, 2018.

[152] Mojtaba Forghani, Weicheng Huang, and M Khalid Jawed. Control of uniflagellar soft robots at low reynolds number using buckling instability. *arXiv preprint arXiv:1810.03113*, 2018.

[153] Ambarish Ghosh and Peer Fischer. Controlled propulsion of artificial magnetic nanostructured propellers. *Nano Letters*, 9(6):2243–2245, 2009.

[154] Kathrin E Peyer, E Siringil, Li Zhang, and Bradley J Nelson. Magnetic polymer composite artificial bacterial flagella. *Bioinspiration & biomimetics*, 9(4):046014, 2014.

[155] Sébastien Fournier-Bidoz, André C Arsenault, Ian Manners, and Geoffrey A Ozin. Synthetic self-propelled nanorotors. *Chemical Communications*, (4):441–443, 2005.

[156] Veronika Magdanz, Mariana Medina-Sánchez, Lukas Schwarz, Haifeng Xu, Jens Elgeti, and Oliver G. Schmidt. Spermatozoa as functional components of robotic microswimmers. *Advanced Materials*, 29(24):1606301, 2017.

[157] Dengfeng Li, Yanting Liu, Yuanyuan Yang, and Yajing Shen. Chemically self-propelled 3d-printed microiots. In *2018 International Conference on Manipulation, Automation and Robotics at Small Scales (MARSS)*, pages 1–4, 2018.

[158] Alexander A. Solovev, Samuel Sanchez, Martin Pumera, Yong Feng Mei, and Oliver G. Schmidt. Magnetic control of tubular catalytic microbots for the transport, assembly, and delivery of micro-objects. *Advanced Functional Materials*, 20(15):2430–2435, 2010.

[159] Gabriel Loget and Alexander Kuhn. Electric field-induced chemical locomotion of conducting objects. *Nature communications*, 2(1):1–6, 2011.

[160] Daniel Ahmed, Mengqian Lu, Amir Nourhani, Paul E Lammert, Zak Stratton, Hari S Muddana, Vincent H Crespi, and Tony Jun Huang. Selectively manipulable acoustic-powered microswimmers. *Scientific reports*, 5(1):1–8, 2015.

[161] Nicolas Bertin, Tamsin A Spelman, Olivier Stephan, Laetitia Gredy, Michel Bouriau, Eric Lauga, and Philippe Marmottant. Propulsion of bubble-based acoustic microswimmers. *Physical Review Applied*, 4(6):064012, 2015.

[162] N Adami, A Delbos, B Roman, J Bico, and H Caps. Elasto-capillary collapse of floating structures-non-linear response of elastic structures under capillary forces. *arXiv preprint arXiv:1310.0329*, 2013.

[163] Klaas Bente, Sarah Mohammadinejad, Mohammad Avalin Charsooghi, Felix Bachmann, Agnese Codutti, Christopher T Lefèvre, Stefan Klumpp, and Damien Faivre. High-speed motility originates from cooperatively pushing and pulling flagella bundles in bilophotrichous bacteria. *Elife*, 9:e47551, 2020.

[164] Howard C Berg. Bacterial behaviour. *Nature*, 254(5499):389–392, 1975.

[165] Maira A Constantino, Mehdi Jabbarzadeh, Henry C Fu, and Rama Bansil. Helical and rod-shaped bacteria swim in helical trajectories with little additional propulsion from helical shape. *Science Advances*, 2(11):e1601661, 2016.

[166] Tapan Chandra Adhyapak and Holger Stark. Zipping and entanglement in flagellar bundle of e. coli: Role of motile cell body. *Physical Review E*, 92(5):052701, 2015.

[167] Victor Ravin, Liisa Raisanen, and Tapani Alatossava. A conserved c-terminal region in gp71 of the small isometric-head phage ll-h and orf474 of the prolate-head phage jcl1032 is implicated in specificity of adsorption of phage to its host, lactobacillus delbrueckii. *Journal of bacteriology*, 184(9):2455–2459, 2002.

[168] P Forsman. Characterization of a prolate-headed bacteriophage oflactobacillus delbrueckii subsp. lactis, and its dna homology with isometric-headed phages. *Archives of virology*, 132(3):321–330, 1993.

[169] Ricardo Cortez, Lisa Fauci, and Alexei Medovikov. The method of regularized stokeslets in three dimensions: analysis, validation, and application to helical swimming. *Physics of Fluids*, 17(3):031504, 2005.

[170] JR Blake and AT Chwang. Fundamental singularities of viscous flow. *Journal of Engineering Mathematics*, 8(1):23–29, 1974.