

Undulatory Swimming: A Topological and Computational Model

Gwinnett School of Math, Science, and Technology

Lawrenceville, Georgia

Team Leader

Dobromir Iliev

Team Members

Anish Goyal & Ricardo Guardado

12 April 2024

Table of Contents

I	Brainstorming	3
II	Research	5
III	Reflections	32
IV	Required Forms	54
V	Research Plan	54
VI	Risk Assessment Form	58
VII	Student Checklist	60
VIII	Research Question/Goal	62
IX	Hypothesis	63
X	Data	64
XI	Graphs	65

XII	Statistical Analysis	66
XIII	Photo Documentation	67

Part I

Brainstorming

- Enhanced Biomimicry: investigate other organisms with unique undulatory swimming patterns and incorporate those patterns into our design.
- Explore aspects of fish locomotion, such as lateral line sensing, and integrate them for improved efficiency.
- Advanced Fluid Dynamics Modeling: Refine the computational fluid dynamics (CFD) model by incorporating real-time environmental data, allowing for dynamic adjustments in response to changing conditions.
- Turbulence and water currents in the CFD model to simulate more realistic aquatic environments.
- Machine Learning Integration: Implement machine learning algorithms to optimize the undulatory swimming patterns based on real-time feedback from the environment.
- Train the system to adapt to different terrains, depths, or obstacles for enhanced autonomy.
- Sustainable Materials: exploring sustainable materials research for 3D printing the model, aligning with ecological considerations for underwater exploration.
- Multi-Agent Systems, investigating the feasibility of deploying multiple submersibles working collaboratively in a swarm, sharing information and optimizing propulsion collectively.
- Sensory integration, by integrating additional sensors, such as temperature or pressure sensors, to expand the submersible's data collection capabilities and environmental awareness.
- Energy harvesting which is the implementation of energy harvesting technologies, such as solar or hydrodynamic energy, to supplement the power source and extend the operational duration.

-
- Adaptive control strategies: Develop adaptive control strategies that allow the submersible to dynamically adjust its undulatory motion based on real-time sensory input, optimizing energy efficiency.
 - Underwater communication: enhance communication capabilities for remote operation, possibly through acoustic or optical communication methods.
 - Underwater Navigation and Mapping: Incorporate navigation algorithms to enable the submersible to autonomously navigate and map underwater environments, contributing to ocean exploration.
 - Integration with Environmental Monitoring: Explore possibilities for integrating environmental sensors that contribute to scientific data collection, such as monitoring water quality, temperature, or marine life.
 - Humanitarian and Environmental Applications: Explore applications in environmental conservation, such as monitoring coral reefs or underwater ecosystems, contributing to biodiversity preservation.
 - Collaboration with Marine Biologists: Collaborate with marine biologists to gain insights into specific fish behaviors and optimize the submersible's design accordingly.
 - Educational Outreach: Develop educational materials and outreach programs to engage students and the public in understanding the importance of biomimicry and robotics in ocean exploration.
 - Miniaturization and Micro-Robotics: Investigate the feasibility of miniaturizing the design for applications in confined underwater spaces or for studying microorganisms.

Part II

Research

Source #1

Student Researcher Ricardo Guardado

Type of Source Journal Article

APA Citation

Oliveira, R. (2013). *Mind the Fish: Zebrafish as a Model in Cognitive Social Neuroscience*.

Frontiers <https://www.frontiersin.org/articles/10.3389/fncir.2013.00131/full>

Notes

- Social competence in animals is key for their survival - there is a great need for behavioral flexibility in social interactions to navigate changes in the social environment.
- Social behavior, often times motivated from neurological activity, possesses impact on biological mechanisms
- A cognitive level of Analysis can be used in the interface between the brain and behavior to develop theories and hypotheses about social behavior. This allows for the testing of predictions.
- Teleost fish as a model could be effective because of its closely related species and the availability of genetic tools for studying neural networks.
- There is a need for a general appraisal mechanism that assesses the valence and salience of social stimuli across different sensory modalities and functional domains.
- Optogenetic and transgenic techniques enable the monitoring of neural activity and experimental manipulations in Zebrafish larvae

-
- The integration of brain activity imaging with behavioral tasks in Zebrafish provides an opportunity to explore cognitive processes and their neural correlates.

Questions this source helped me answer

- Why are Zebrafish used to study social neurocognition and behavior?
- What element in science can be used to study the linkage between neuroscience behavior and social behavior in environments.
- What mechanisms have been used to help humans draw connections between neural and social activity

Questions I have for the author

- Is it possible to make computation models using neurological data alone?
- What models can be used for fish that have irregular movement patterns? For example, sharks can only swim in the background.
- Social Neuroscience—field that investigates the neural mechanisms underlying social processes and behavior
- Cognitive level of Analysis—focuses on understanding mental processes such as perception, memory, language, problem-solving, and decision-making.
- Magnetic resonance imaging—uses powerful magnets, radio waves, and a computer to generate detailed images of the internal structures of the body.

Summary

The proposed conceptual framework involves recognizing key components of social competence, including social value and preferences. Zebrafish are a valuable model for comparative social neuroscience due to the diversity of social systems among closely related species and the availability of genetic tools.

Implications for my project/thinking

To make a working computational model in an environment like the ocean, we can use neurological data collected from the locomotion of Zebrafish to make this model because of the benefits that using this model would provide.

Source #2

Student Researcher Ricardo Guardado

Type of Source Research Article

APA Citation

Beem K. (2020). *Rolling in the Deep: Climate Change and Deep Sea Ecosystems*. Columbia University. <https://climatesociety.ei.columbia.edu/news/rolling-deep-climate-change-and-deep-sea-ecosystems>.

Notes

- Deep sea ecosystems play a crucial role in oceanic-planetary regulatory systems. Through deep ocean mixing, they help drive ocean currents and absorb heat and atmosphere CO₂, acting a buffer against climate change.
- Anthropogenic climate change poses specific threats to deep sea ecosystems, including ocean acidification, deep sea warming, and decreased availability of food. These impacts could affect the deep sea's ability to serve as a carbon and heat sink.
- The health of deep sea ecosystems is closely connected to the health of surface water ecosystems.
- Methane vent communities on the ocean floor house unique species of bacteria and worms adapted to feed on methane, preventing it from reaching the surface. This methane oxidation process captures a significant amount of methane annually, mitigating its impact on the atmosphere.
- Ocean warming, induced by climate change, could lead to a climate-induced shift in warm currents, potentially releasing large amounts of frozen methane from the seafloor.
- Despite the importance of the deep sea, there is a lack of policy and scientific information about this ecosystem, which constitutes 90% of Earth's livable area.

-
- Anthropogenic—refers to processes, activities, or conditions that originate from or are caused by human actions, particularly those that have an impact on the environment or natural resources.
 - Methane vent communities—ecosystems that exist around deep-sea hydrothermal vents where methane is released from the ocean floor.
 - Induced climate change—Changes in the Earth’s climate that are primarily caused or triggered by human activities.

Questions this source helped me answer

- What are some effects of climate change on the ocean floor and other deep parts of the ocean?
- How are ecosystems tied with ocean conditions?
- Can ecosystems be used as a means of analyzing how human activity is affecting the oceanic environment?

Questions I have for the author

- How feasible do you think it is to construct a moving robot that can reach deep parts of the ocean and take images and record other data such as temperature over long periods of time to see the effects?
- What type of data would give us valuable insights on the fact that the ocean floor and its environment is changing over time based on human activity?

Summary

The health of deep-sea ecosystems is closely connected to surface water ecosystems, and their ability to capture and store carbon is essential for mitigating climate change. The article highlights methane vent communities on the ocean floor, where unique species help prevent methane from reaching the atmosphere.

Implications for my project/thinking

We can use this as an application to our computational model. If we were to make a model that is energy efficient and is able to use enhanced locomotion for movement, we could use our model to discover deep parts of the oceans that can read how they are being affected by climate change that is induced by human activity over long periods of time.

Source #3

Student Researcher Ricardo Guardado

Type of Source Journal Article

APA Citation

Cregg, J., Kiehn O., Leiras, R. (2022). *Brainstem Circuits for Locomotion*. Annual Reviews.

<https://www.annualreviews.org/doi/full/10.1146/annurev-neuro-082321-025137>

Notes

- Midbrain nuclei, specifically the cuneiform nucleus play important roles in mediating the initiation and speed control of locomotion as well as selection of different gates
- Activation of CnF-Vglut2 neurons initiates locomotion with a full range of speeds and gaits, while PPN-Vglut2 activation specifically induces slow alternating gaits.
- Discrepancies exist in the outcomes of studies on PPN-Vglut2 neurons, indicating potential variations in targeting approaches and stimulus paradigms.
- Targeting PPN-Vglut2 neurons broadly can lead to variable effects, emphasizing the need for precise stimulation in different subpopulations for specific outcomes.
- The study of brainstem circuits for locomotor control has progressed from classical methods to a combination of electrophysiological, molecular genetics, and behavioral approaches
- Brainstem pathways involved in stopping and steering locomotion have been identified alongside those related to initiation and speed control.

Questions this source helped me answer

- What role do brain stem cells play in the locomotion of organisms?
- How complicated can computational neuroscience with brainstem signals become when it comes to studying very complex organisms?

-
- Is it practical to make a computational neuroscience model with organisms that have very extensive and complex neurosystems?

Questions I have for the author

- A lot of the experimentation was done on species that are not very complex neurologically. However, for more complex organisms like humans, what differences could be discovered in the finishing discussed in the article?
- Because you discovered brainstem pathways that were related in function, how easy is it to trace locomotion of activity when both pathways are at work? What are some things that distinguish one from the other?

Summary

Downstream circuits transmitting MLR signals to the spinal cord have been identified, with parallel pathways for stopping and steering locomotion also recognized. Although the precise way in which the locomotors behave is unknown, there is optimism about establishing a functional connectome from the brainstem to the spinal cord. This connectome would clarify how the locomotor system is implemented and how traits are controlled by command signals.

Implications for my project/thinking

- Trying to develop our own neurological data that record the synaptic activity of a certain species would not be practical as it requires extensive experimental setup and is difficult to track the activity of every neuron in opening and steering locomotion in the species.
- It would be best for us to obtain our synaptic data from an online source that is publicly released so that we do not get into the issue of trying to obtain our own data
- If we were to experiment, it would be best to use a species that has a reputation for being feasible to work with in terms of the complexity of their neurological system. For example, an easy species to work with are drosophila flies.

Source #4

Student Researcher Anish Goyal

Type of Source Journal Article

APA Citation

Roussel, Y., Gaudreau, S. F., Kacer, E. R., Sengupta, M., & Bui, T. V. (2021). *Modeling spinal locomotor circuits for movements in developing Zebrafish*. eLife, 10, e67453.
<https://doi.org/10.7554/eLife.67453>

Notes

- The study focuses on understanding the operation of spinal circuits dedicated to locomotor control in developing Zebrafish.
- Iterative models were built, coupling developing Zebrafish spinal circuits with simplified musculoskeletal models to replicate coiling and swimming movements.
- Neurons in the models were based on morphologically or genetically identified populations in the Zebrafish spinal cord.
- Simulations involved intact spinal circuits, circuits with silenced neurons, and circuits with altered synaptic transmission to explore the role of specific spinal neurons.
- Analysis of firing patterns and phase relationships aimed to uncover mechanisms behind locomotor movements in developing Zebrafish.
- Simulations demonstrated the transition between coiling and swimming and highlighted the importance of contralateral excitation to multiple tail beats.
- The models provided insights into the sensitivity of spinal locomotor networks to various factors, including motor command amplitude, synaptic weights, axon length, and firing behavior.

Questions this source helped me answer

- What specific morphologically or genetically identified populations of neurons in the Zebrafish spinal cord were used in the models?
- How did the simulations demonstrate the transition between coiling and swimming, and what mechanisms were identified during this transition?

Questions I have for the author

- What were the key findings regarding the sensitivity of spinal locomotor networks to factors such as motor command amplitude, synaptic weights, axon length, and firing behavior?

Vocabulary

- Spinal Circuits: Neural circuits or networks within the spinal cord responsible for specific functions, such as locomotor control.
- Zebrafish: A small tropical fish commonly used in scientific research, particularly in the study of vertebrate development and genetics.
- Musculoskeletal Models: Simplified models that represent the interaction between muscles and bones in the context of movement and locomotion.
- Contralateral Excitation: Excitation of neurons on the opposite side of the body, common in neural circuits involved in coordinating movements on both sides.
- Axon Length: The length of the axon, a part of a neuron that conducts electrical impulses away from the cell body.
- Firing Behavior: The pattern and frequency of action potentials (nerve impulses) produced by neurons.
- Coiling: A movement pattern characterized by the bending or winding of the body.

-
- **Synaptic Transmission:** The process by which nerve impulses pass across the synapse (junction between neurons) to transmit signals. **Rhythm Generation:** The generation of rhythmic patterns, often associated with repetitive movements such as locomotion.

Summary

Utilizing iterative models, the researchers coupled Zebrafish spinal circuits with simplified musculoskeletal models to replicate coiling and swimming movements. The models were constructed based on morphologically or genetically identified neuron populations in the Zebrafish spinal cord. Simulations included intact spinal circuits as well as circuits with silenced neurons or altered synaptic transmission.

Implications for my project/thinking

The created open-source models from this research paper help to advance humanity's knowledge in spinal locomotion in fish. Such models can be used to form a baseline for analysis on energy efficiency in a project.

Source #5

Student Researcher Anish Goyal

Type of Source Website

APA Citation

Oliver K. Ernst, Ph. D. (2020, July 9). *Headless setup of Raspberry Pi—once and for all*. Medium. <https://medium.com/practical-coding/headless-setup-of-raspberry-pi-once-and-for-all-de5a2c4f715b>

Notes

- Use balenaEtcher to flash the SD card with the chosen image (a good alternative is RPi flasher)
- Set up Wi-Fi by enabling the NetworkManager daemon and editing `/etc/wpa_supplicant.conf`
- Can enable SPI, I2C, UART, LCD screen, NFC, and Docker through the RPi configuration tool
- Devices that connect to the Raspberry Pi via SSH must share the same network as the Pi

Questions this source helped me answer

- How can I set up a Raspberry Pi headlessly (without monitor/keyboard)?
- What are the essential steps for configuring Wi-Fi and enabling SSH on a headless Raspberry Pi?
- How can I enable optional features like SPI, I2C, UART, and LCD via fbtft on a Raspberry Pi?

Questions I have for the author

- Can the guide be adapted for older Raspberry Pi models? What are the specific advantages of using a headless Raspberry Pi setup?

-
- Why did the author choose to use Raspberry Pi OS Lite (32-bit) in making the guide?

Vocabulary

- Headless Setup: Configuring a device without the need for a monitor or keyboard.
- SPI (Serial Peripheral Interface): A synchronous serial communication protocol used to transfer data between a master device and one or more peripheral devices.
- I2C (Inter-Integrated Circuit): A communication protocol used to connect low-speed devices in a master-slave configuration.
- UART (Universal Asynchronous Receiver-Transmitter): A hardware communication protocol used for serial communication between devices.
- LCD (Liquid Crystal Display): A type of flat-panel display technology commonly used in devices like monitors and screens.

Summary

This source is a detailed guide for a headless setup of a Raspberry Pi Zero WiFi using Raspberry Pi OS Lite. It covers flashing the SD card, configuring WiFi, enabling SSH, and optional features such as SPI, I2C, UART, and LCD via fbtft.

Implications for my project/thinking

Learned how to set up a Raspberry Pi headlessly, which is valuable for this project and doesn't require extensive hardware setup. I also learned how to configure Wi-Fi and SSH integration between end devices.

Source #6

Student Researcher Anish Goyal

Type of Source Journal Article

APA Citation

Dukes, X., Littleton, J., Neville, T., Rollerson, C., & Quinney, A. (n.d.). *Object detection on Raspberry Pi*. American Society for Engineering Education. <https://peer.asee.org/object-detection-on-raspberry-pi.pdf>

Notes

- The source discusses the implementation of object detection on Raspberry Pi using machine learning models
- The project involves using a web camera, Raspberry Pi Kits (Model B+), and a Google USB accelerator to enhance detection speed.
- They implemented object detection using SSD-MobileNet for general objects and extend it to recognize weapons through transfer learning.
- Preliminary validation results indicate effective detection of general objects and certain weapons, but the detection speed for weapons needs improvement.

Questions this source helped me answer

- How is object detection implemented on Raspberry Pi's using machine learning models with minimal resource cost?
- What are the key components involved in such a project, including hardware and software?
- What challenges are addressed in implementing an object detection toolchain?

Questions I have for the author

- What optimizations were considered in implementing the SSD-MobileNet model?
- Were there any bottlenecks in using a Raspberry Pi, such as poor memory/CPU performance?

Vocabulary

- Transfer Learning: When a model trained on one task is adapted to a second related task.
- Edge Devices: Devices located close to the source of data that process data locally rather than relying on cloud computing.

Summary

This source is a senior design project implementing object detection on Raspberry Pi using SSD-MobileNet to perform weapon detection through transfer learning. The project uses a Google USB accelerator to enhance detection speed and seamless software communication via the SSH (Secure Shell) protocol.

Implications for my project/thinking

The use of transfer learning can be applied to create a model for undulatory swimming by fine tuning certain parameters. A Raspberry Pi headless display is best-suited to meet this end.

Source #7

Student Researcher Dobromir Iliev Iliev

Type of Source Research Article

APA Citation

Thomas, A., Bates, K., Elchesen, A., Hartsock, I., Lu, H., & Bubenik, P. (2021b, May 12). *Topological data analysis of C. elegans locomotion and behavior*. Frontiers. <https://www.frontiersin.org/articles/10.3389/frai>

Questions I have for the author

- Could the author provide more details on the specific experimental setups and conditions used in the study?
- How does the synthesis of skeleton data through persistent homology contribute to a better understanding of stereotypical behaviors in *C. elegans*?
- Are there any limitations or challenges encountered in applying persistent homology to behavior analysis, and how were they addressed?
- What were the key findings regarding the sensitivity of spinal locomotor networks to factors such as motor command amplitude, synaptic weights, axon length, and firing behavior?

Vocabulary

- Persistent Homology: The study of topological features in a dataset that persist across multiple scales.
- Sliding Window Embeddings: Transforming time series data into point cloud data while preserving temporal information.
- Microfluidic Devices: Devices designed to manipulate small amounts of fluid, used in confining *C. elegans* for experiments.
- Viscosity: A measure of a fluid's resistance to flow.

Summary

The study applies persistent homology to analyze *C. elegans* behavior in different experimental conditions, distinguishing and classifying locomotion patterns. The method outperforms simpler approaches, providing insights into stereotypical behaviors and environmental impacts. The use of sliding window embeddings facilitates the detection of characteristic cycles in time series data, contributing to a quantitative summary of complex behaviors.

Implications for my project/thinking

This source enhances my understanding of how persistent homology can be applied to behavior analysis, offering a potential method for quantifying and classifying patterns in my own research. The emphasis on distinguishing behaviors and the synthesis of skeleton data may inspire new approaches or considerations in my project. Additionally, it prompts me to explore the limitations and challenges associated with applying persistent homology to behavior studies.

Source #8

Student Researcher Dobromir Iliev Iliev

Type of Source Research Article

APA Citation

Saputra, A. A., Botzheim, J., & Kubota, N. (2023, June 3). *Neuro-cognitive locomotion with dynamic attention on topological structure*. MDPI. <https://doi.org/10.3390/machines11060619>

Notes

- The research proposes a novel concept in locomotion generation that integrates a cognitive model from an ecological psychology viewpoint.
- The goal is to decrease the gap between the cognitive model and the locomotion model, moving from a hierarchical to a parallel system structure.
- The attention controller is highlighted as a crucial starting process for handling exteroceptive sensory information required by further cognitive processes.
- It processes 3D point-cloud data as exteroceptive sensory information and controls the density of topological structures in localized areas for detailed object perception.
- A dynamic density topological map construction process, called DD-GNG, is introduced based on the Growing Neural Gas (GNG) algorithm.
- Compared to other topological reconstruction methods, DD-GNG is claimed to reduce processing time significantly (up to 70
- Affordance detection is emphasized as a critical element in integrating the relationship between attention and locomotion.
- The proposed affordance detector provides semantic function in locomotion behavior, offering object information in the context of the robot's current capabilities.

-
- Experimental trials in both simulation and real robot performance demonstrate the effectiveness of the proposed system in short-term adaptation to obstacles.
 - The robot showcases the ability to change limb swinging patterns and foothold targets in response to sudden obstacles.
 - The contributions of the research include the dynamic density topological map construction process, real integration between attention and affordance, and the development of a locomotion system that integrates external sensory information in short-term adaptation.
 - The paper mentions the potential for future development in implementing the dynamic attention model, especially in mobile robot applications.
 - The concept of neuro-cognitive locomotion is seen as having high prospects for achieving dynamic locomotion that integrates cognition with the locomotion generator.

Questions this source helped me answer

- How does the proposed locomotion generation concept integrate cognitive models from an ecological psychology viewpoint?
- What role does the attention controller play in handling exteroceptive sensory information?
- What is the dynamic density topological map construction process, and how does it differ from other methods?
- How does the affordance detector contribute to the integration between attention and locomotion?
- What are the key findings and results from the experimental trials in simulation and real robot performance?

Questions I have for the author

- Can you provide more details on the specific parameters and settings used in the dynamic density topological map construction process (DD-GNG)?

-
- How do you envision further developing the locomotion generator part to consider more sensory feedback?
 - Are there specific limitations or challenges encountered during the experiments that you plan to address in future research?

Vocabulary

- Exteroceptive: Relating to stimuli that arise outside an organism.
- Ecological Psychology: A field of psychology that studies the relationship between individuals and their environments.
- Affordance: The potential actions or interactions that an individual perceives in their environment.
- GNG (Growing Neural Gas): A neural network algorithm used for clustering and topological mapping.

Summary

The research introduces a novel concept in locomotion generation, integrating ecological psychology principles. The attention controller processes 3D point-cloud data, and a dynamic density topological map construction process (DD-GNG) efficiently clarifies details of objects. Affordance detection contributes to integrating attention and locomotion. Experimental trials demonstrate the system's effectiveness in short-term adaptation to obstacles.

Implications for my project/thinking

- integrating principles from ecological psychology in the design of cognitive models for robot locomotion.
- Dynamic density topological maps for detailed object perception in the robot's surroundings.
- Affordance detection can enhance the adaptability of a robot to unforeseen obstacles.

Source #9

Student Researcher Dobromir Iliev Iliev

Type of Source Research Article

APA Citation

Mulase, M., & Penkava, M. (2012, March). *Topological recursion for the Poincaré polynomial of the combinatorial moduli space of curves*. Redirecting. <https://doi.org/10.1016/j.aim.2012.03.027>

Notes

- Euler characteristic of moduli space of smooth algebraic curves with distinct marked points.
- Harer-Zagier's closed formula, Penner's relation to quantum field theory.
- Ribbon graphs in Penner model, isomorphism of topological orbifolds.
- Penner model as the generating function, matrix integral
- Introduction to Eynard-Orantin topological recursion theory.
- Definition of Poincaré polynomial and its uniqueness (Theorem 1.1).
- Symmetric differential and its relation to Eynard-Orantin theory.
- Topological recursion's inductive structure and its role in various areas.
- Virasoro Constraint and Differential Equation
- Derivation of differential recursion equation for Poincaré polynomial (Theorem 5.1).
- Initial values calculation (Section 6) and consequences of the differential equation.
- Proof that Poincaré polynomial is a Laurent polynomial (Theorem 7.2).
- Invariance under the coordinate change (Proposition 7.4).

Questions this source helped me answer

- How is the Poincaré polynomial related to the Euler characteristic of the moduli space of smooth algebraic curves with marked points?

-
- What is the significance of ribbon graphs in the context of the Penner model and quantum field theory?
 - How does topological recursion theory contribute to the understanding of the Poincaré polynomial?
 - What is the differential equation satisfied by the Poincaré polynomial, and how does it connect to the Virasoro constraint condition?
 - In what way is the Poincaré polynomial related to the intersection numbers of the Deligne-Mumford stack?

Questions I have for the author

- Can you elaborate on the motivation behind using Eynard-Orantin topological recursion theory in this context?
- How does the Laurent polynomial property of the Poincaré polynomial impact its applications and interpretations?
- Could you provide more insights into the coordination change invariance of the Poincaré polynomial?

Vocabulary

- Moduli space: A mathematical space that represents the collection of all geometric objects with a certain property.
- Ribbon graph: A graph used in mathematical studies, particularly in the context of algebraic curves.
- Topological recursion theory: A mathematical theory used to compute certain topological invariants.
- Virasoro constraint condition: A condition related to the Virasoro algebra, often encountered in string theory and mathematical physics.

-
- Laurent polynomial: A polynomial with both positive and negative degree terms, allowing for negative powers of the variable.

Summary

The paper explores the Poincaré polynomial's connection to the Euler characteristic of moduli spaces, specifically focusing on smooth algebraic curves with marked points. It introduces the use of ribbon graphs in the Penner model and establishes the Poincaré polynomial through topological recursion theory. The paper provides a differential equation for the Poincaré polynomial, connecting it to the Virasoro constraint condition and showcasing its Laurent polynomial properties.

Implications for my project/thinking

This source deepens my understanding of the mathematical intricacies involved in studying algebraic curves and moduli spaces. The use of topological recursion theory and the connection to the Virasoro constraint condition could inspire new perspectives or methodologies in the topological model. Additionally, the Laurent polynomial properties suggest potential applications or considerations for polynomial structures in related contexts.

Source #10

Student Researcher Dobromir Iliev Iliev

Type of Source Research Paper

APA Citation

Mitin, I., Korotaev, R., Ermolaev, A., Mironov, V., Lobov, S. A., & Kazantsev, V. B. (2022, Novem-

ber 28). *Bioinspired propulsion system for a thunniform robotic fish*. MDPI. <https://doi.org/10.3390/biomimetics704>

Notes

- The robot's maximum speed was approximately 0.4 body lengths per second (BL/s), achieved at the maximal tail beat frequency of 3.4 Hz and a maximal tail deflection amplitude of 105 mm.
- The speed of the robot was found to increase with the frequency of tail fin oscillations. Different frequencies showed qualitatively similar profiles, shifting to higher speeds as the tail beat frequency increased.
- The dependencies of the robot's speed on the amplitude of tail beats exhibited non-monotonic shapes with seemingly oscillatory patterns. Two linear trends with different slopes were observed for small and larger amplitude values, intersecting at an amplitude of about 50 mm.
- The oscillatory character of the speed-amplitude dependencies may result from a resonant interaction of the robot's oscillating body and hydrodynamic waves reflected from the pool walls.
- The cost of transport (COT), a measure of energy efficiency, showed a minimum corresponding to the change in the angle of the regression lines in the speed-amplitude dependencies. COT values were higher compared to live tuna but varied in an interval of 30-70 J kg⁻¹m⁻¹ for different values of frequency and swimming speed.

-
- Increasing the tail oscillation amplitude above a threshold of about 50 mm led to a decrease in swimming efficiency, indicating that there is an energetically preferable range of traveling speeds up to a threshold speed.
 - Generally, increasing the swimming speed is preferable by increasing the oscillation frequency of the caudal fin rather than the amplitude.

Questions this source helped me answer

- How does the thunniform swimming robot's speed vary with tail beat frequency and amplitude?
- What is the influence of dynamic parameters on the efficiency of the robotic fish's propulsion system?
- What is the cost of transport (COT) for the thunniform swimming robot, and how does it compare to live tuna?
- What are the advantages and limitations of the simplified design of the robotic fish's tail section?
- How do the experimental results contribute to the understanding of biomimetic robotic fish design?

Questions I have for the author

- Can you elaborate on the reasons behind the non-monotonic shapes and seemingly oscillatory patterns observed in the dependencies of the robot's speed on tail beat frequency and amplitude?
- How might the resonant interaction of the robot's oscillating body and hydrodynamic waves affect its performance, and are there potential ways to mitigate this effect?
- Could you provide insights into potential applications of the thunniform swimming robot and areas for future research in this field?

Vocabulary

- Thunniform: Relating to a type of locomotion characterized by undulation limited to the rear part of the body, as observed in certain fishes like tunas.
- Flexor/Extensor: Muscles responsible for bending (flexor) and straightening (extensor) a joint or body part.
- Hydrodynamics: The study of fluid motion and the forces acting on solid bodies immersed in fluids.
- Resonant Interaction: The phenomenon where an oscillating system is influenced by external forces or frequencies that match its natural frequency, resulting in increased amplitude or efficiency.
- Cost of Transport (COT): A measure of the energy required to move a unit mass over a unit distance at a given speed.

Summary

The article discusses experiments with a thunniform swimming robot designed to emulate the movement of tuna fish. The robot's propulsion system involves an elastic cord with flexor/extensor mechanisms simulating muscular contractions. The experiments revealed that the robot's speed increased with tail beat frequency and exhibited non-monotonic patterns with amplitude. The cost of transport (energy efficiency) showed variations, and the simplified design of the robotic fish's tail section influenced its speed and efficiency. The study emphasizes the need for dynamic feedback systems to compensate for physical fluctuations and suggests future research directions.

Implications for my project/thinking

Understanding the impact of tail beat frequency and amplitude on the robotic fish's performance can inform the design of biomimetic propulsion systems in aquatic robots. The insights into energy efficiency and the trade-off between speed and amplitude provide valuable considerations

for optimizing the performance of robotic fish. The acknowledgment of variability in experimental results and the need for dynamic adjustments align with the challenges in developing responsive and adaptive control systems for aquatic robots in real-world environments.

Part III

Reflections

Reflection #1

Date 11/19/2023

Initials Ricardo Guardado

During my internship at Georgia Gwinnett College and working with neuroscience data, I learned how to identify peaks in neurological activity in MATLAB using the threshold function to determine the minimum level of synaptic activity for a peak to be considered a peak by MATLAB. I figured it was a good idea to use this same function with our future neuroscience data that we were going to use in order to build the computational model. This would allow us to determine the activity level in a certain amount of time and use this information to our advantage in order to make the computational model for testing.

Reflection #2

Date 11/21/2023

Initials Ricardo Guardado

After taking notes on a research article that discussed the use of computational thinking with Zebrafish, we were able to start considering the use of Zebrafish as our source of the necessary neuroscience data in order to make a computational model that was feasible and easy to use. We discovered that the use of Zebrafish for testing offers many advantages, one being the proximity with other species. Their rapid development and accessibility to genetic analysis make the Zebrafish an excellent model system for molecular and mechanistic studies of neurodevelopment. With this information, we plan on considering Zebrafish as the source of our neuroscience data for the creation of the computation model.

Reflection #3

Date 11/23/2023

Initials Ricardo Guardado

Today, I took more notes on a research article that talked about how oceans are affected by climate change in areas that are not easily accessible or seen by humans. This can be seen as an application to our future computational model as we plan on making a model that is energy efficient that can access any environment based on the neurological activity from the surrounding environment. With this, we can take our model to the next level by identifying which materials are the most durable for sea exploration. This could allow us to take images of parts of the ocean that are not easily accessible to humans.

Reflection #4

Date 11/25/2023

Initials Ricardo Guardado

Today, I continued work at my internship. In MATLAB, I learned how to normalize some of the data so that small variances in the data do not interfere with the identification of peaks in the synaptic activity. This allowed us to identify peaks that were not “small noise peaks”, therefore allowing us to identify peaks in the data that were significant to notice. We can use this function to our advantage with our future synaptic data so that we can route out unnecessary noise and be able to identify peaks to locomotion more efficiently.

Reflection #5

Date 11/27/2023

Initials Ricardo Guardado

Today I provided my group members with some of the work and research articles I had been using during my internship that could be considered for assistance in constructing our neuroscience computational model. Because I was waiting for confirmation regarding the specific neuroscience data we were going to use, I did not conduct any specific research on specific animals that could be used to represent our computational model.

Reflection #6

Date 11/29/2023

Initials Ricardo Guardado

Today, we realized that if our group wanted to make a computational model that was energy efficient, we would also need to make a topological model that would eliminate confounding variables during our experimentation. Because of this, we began considering the construction of a topological model for our computational model to work with. We began doing research with this.

Reflection #7

Date 11/31/2023

Initials Dobromir Iliev Iliev

To create a topological model we first needed to create a velocity field model. We needed to do this because our topological model utilized the dot product function to minimize the force of impact onto the object itself; as a result to calculate the surface lines we needed a velocity field model. I started to research different types of velocity field models, finding Evragov and Noaa's HYCOM model. These models took a wide range of variables including the pressure field, viscosity, time step between the waves, and other variables that all culminated in a net velocity vector into their model. I took note and added my observations in the logbook.

Reflection #8

Date 12/01/2023

Initials Dobromir Iliev Iliev

Once I completed my research on other velocity fields models I then worked on implementing those features on our models; I also noted the data structures that these models used. The HYCOM model used a stack data structure with a PyTorch framework for each of the variables while Evargov's model used a queue based data structure. While the HYCOM might appear to be significantly slower, it used parallel processing resulting in faster initialization and graphing. I then worked on a data structure that incorporated queue based data structure with parallel computing however I ran into issues since some of the processes would run faster than others resulting in a memory leak. To fix these issues I implemented a priority queue system which calculated the computational complexity and dequeued computational complex tasks from overloaded threads and instead queued those tasks into another thread. As a result our model initialized the velocity fields at a time complexity of $O(n \log n)$, as $O(n \log n)$ is the time complexity necessary to input the object into Numpy's linspace for 3D graphing.

Reflection #9

Date 12/01/2023

Initials Dobromir Iliev Iliev

After setting up the circuit, I went back to the model and noticed the time complexity for graphing took significantly longer than the time complexity for initializing the function. I looked over the code and noticed that the time complexity for graphing was in fact $O(n^3)$ as it used a tuple to graph the 3D grid from the data provided. To fix the issue with graphing I then implemented a heap based data structure and an octree which componentized the index of the 3D grid into 8 quadrants into 8 dictionaries respectively. By implementing these data structures and algorithms the time complexity was significantly reduced in terms of graphing. From there I took the velocity field model as a VRML file format into MATLAB's simulink. I then created a dot product math function to find the surface lines to create a 3D model. Lastly I 3D printed that model using a sls file format from MATLAB's simulink in onshape using a 3D printer.

```

1  % Step 1: Call Python Script
2  system('python /Users/dobromirilieiev/Documents/GitHub/Undulatory-Swimming-A-Topological-and-C_
   ↪  omputational-Model/TopologicalModel.py');
3
4  % Step 2: Read Velocity Fields
5  load('velocity_fields.mat'); % Load velocity fields data
6
7  % Step 3: Calculate Dot Product
8  dot_product_result = dot(velocity_field_1, velocity_field_2);
9
10 % Step 4: Generate Surface Line
11 % Assuming you have a function find_surface_line() implemented
12 surface_line = find_surface_line(dot_product_result);
13
14 % Step 5: Iterate Across 3 Dimensions
15 % Assuming you have a function for iteration, iterate across the dimensions
16 % For example:
17 % [X, Y, Z] = ndgrid(1:size(surface_line, 1), 1:size(surface_line, 2), 1:size(surface_line,
   ↪  3));
18 % Iterate through X, Y, Z and modify the surface line accordingly
19
20 % Step 6: Export to VRML
21 vrmllwrite('output_model.vrml', surface_line);

```

Figure 1: Generating the topological model

Reflection #10

Date 12/05/2023

Initials Dobromir Iliev Iliev

I began working on the circuit, specifically by focusing on the INA219 power circuit and connecting it with SSH powershell. I had a small issue connecting through SSH through the raspberry because windows has connection port issues. It turns out that Windows 11 did not like it. Therefore, I fixed the port issues by installing another program. Furthermore, our current model appears to be too small to handle the stepper motors and isn't buoyant—therefore I designed a larger model so our object can float.

Reflection #11

Date 12/07/2023

Initials Dobromir Iliev Iliev

The project has great potential integration with LIDAR technology. Scalability is demonstrated through our differential equation models. The flexibility of the project extends not only to different environments but also to diverse model sizes, highlighting its potential across a wide range of scales. Our modular circuitry design emerges as a key contributor to scalability. The ability to integrate various components without extensive redesign offers versatility in motor selection. LIDAR's detailed sensory outputs, as demonstrated in Queralta et al.'s work, can be seamlessly integrated into our project through ROS drivers. This opens avenues for real-time environmental adaptation, enabling the model to dynamically adjust its speeds based on the surroundings. The potential synergy between our closed-loop feedback control system and LIDAR technology holds promise for optimizing motion planning and enhancing overall efficiency. I then worked on analyzing our topological model for 1 million velocity vectors compared to the Evragov and simplified HYCOM model. I found that the time our model took to initialize and graph was significantly less than the other models, however, the memory allocation was about the same. I also created shell script in Bash to ensure that the same environment was measuring time and energy usage.

```
1      #!/bin/bash
2
3      # Path to the memory analysis tool
4      MEMORY_TOOL=/Users/dobromiriliev/Documents/GitHub/Undulatory-Swimming-A-Topological-
↵      and-Computational-Model/TopologicalModel.py
5
6      # Path to the program
7      PROGRAM=/Users/dobromiriliev/Documents/GitHub/Undulatory-Swimming-A-Topological-and-
↵      Computational-Model/TopologicalModel.py
8
9      # Loop 50 times
10     for ((i=1; i<=50; i++))
11     do
12         echo "Running analysis $i"
13         memray run /usr/bin/python3 /Users/dobromiriliev/Documents/GitHub/Undulatory-Swi
↵         ming-A-Topological-and-Computational-Model/TopologicalModel.py
14         echo "-----"
15     done
```

Figure 2: Memory analysis of our model using Memray

```

1  # Path to the program
2  PROGRAM=/Users/dobromiriliev/Documents/GitHub/Undulatory-Swimming-A-Topological-and-Computational-Model/TopologicalModel.py
3
4  # Number of times to run the program
5  NUM_RUNS=50
6
7  # Array to store execution times
8  declare -a execution_times
9
10 # Run the program NUM_RUNS times
11 for ((i=1; i<=$NUM_RUNS; i++))
12 do
13     # Capture the output of the time command
14     output=$(time -p $PROGRAM 2>&1)
15
16     # Extract the real time (execution time) from the output
17     execution_time=$(echo "$output" | grep -oP "(?<=real\s)\d+(\.\d+)?" | tail -n 1)
18
19     # Store execution time in the array
20     execution_times+=($execution_time)
21
22     echo "Run $i: Execution time = $execution_time seconds"
23 done

```

Figure 3: Time analysis performed on the Topological Model

Reflection #12

Date 12/09/2023

Initials Dobromir Iliev Iliev

I continued to perform analysis on the circuit and I noticed that the average power consumption of the closed-loop system and the expected values were close. Specifically it measured 0.0545 kWh which was close to the expected of 0.05 kWh; normally the variance ranges from 15-20% power loss, so this was surprising.

While the results are surprising, a note of caution is sounded, since long-term trials could shed light on whether power consumption remains constant or exhibits variations. To enhance our understanding, considerations for different power sensors and a larger testing environment are proposed, paving the way for future experiments.

Additionally, the variance introduced by the SLA printing technology in our model's design needs to be considered. Despite calibration efforts, the ridges along the surface may contribute to some level of variance. To further minimize these variations we could use Selective Laser Sintering (SLS), different motor drivers, as well as use thermoplastic urethane (TPU) for our model.

Reflection #13

Date 12/10/2023

Initials Anish Goyal

I started my work on creating a usable Raspberry Pi for the project. First, I installed the official Raspberry Pi Imager with the command `yay -S rpi-imager-bin` on my Arch Linux installation. Originally, I was going to install Aspertis onto the disk, which is a systems-level OS for embedded IoT devices, but I knew it would be really complicated for the others to work with. So, I decided to install Arch Linux ARM for Raspberry Pi 4 instead using the flasher tool. I also purchased an extra USB-C cable with a minimum amperage of 3A to avoid any power issues.

Reflection #14

Date 12/12/2023

Initials Anish Goyal

Using an SD card reader, I connected the Raspberry Pi filesystem to my laptop. First, I created a directory in the `/mnt` partition:

```
sudo mkdir -p /mnt/raspbian
```

I then installed some additional dependencies (`qemu-user-static`, `binfmt-support`) to get support for user mode emulation binaries, allowing the group to launch processes compiled for the ARM architecture on their host CPU. `binfmt-support` also allows the Raspberry Pi to run foreign ELF binaries directly. I then unmounted the SD card reader boot medias to the current working directory:

```
1 sudo umount /run/media/anish/firmware
2 sudo umount /run/media/anish/boot
3 sudo umount /run/media/anish/general_storage
```

And mounted the actual file systems to my Arch installation:

```
1 sudo mount /dev/sda3 /mnt/pi
2 sudo mount -t proc /proc /mnt/pi/proc
3 sudo mount -t sysfs /sys /mnt/pi/sys
4 sudo mount -o bind /dev /mnt/pi/dev
```

The next step is to copy the QEMU ARM emulation binary to the `/usr/bin` folder on the RPI:

```
sudo cp /usr/bin/qemu-arm-static /mnt/pi/usr/bin/
```

And now I can chroot (“change root”) into the SD Card, modifying it as though it were my actual system!

Reflection #15

Date 12/14/2023

Initials Anish Goyal

It's time to chroot into the system:

```
1 sudo chroot /mnt/pi
2 sudo chroot /mnt/pi /bin/bash
3 sudo chroot /mnt/pi /usr/bin/bash
4 sudo arch-chroot
```

At this point, I expected to be taken into the Arch Linux installation within the SD card, but I was gravely mistaken. I still wasn't able to chroot into the system because it wasn't detecting some core linked library files in `/usr/share/lib` and `/var/lib` to emulate the `LD_PRELOAD` environment variable. After 1 hour of debugging, I was able to figure out a solution. The explanation for this is quite complicated, but I can best explain it as follows:

On UNIX devices, whenever an application needs to use a system call (typically written in the C programming language or one of its derivatives), it runs the function associated with that call in one of the many linked libraries in the location `/lib`, typically using `glibc` or `libc.so.6`. However, ARM architectures of Arch Linux don't have support for linked libraries, so they implemented something "hacky:" overriding the `LD_PRELOAD` variable, which is responsible for asking the dynamic linker to load a particular shared object as a dependency before invoking any system calls. Essentially, if we can get `LD_PRELOAD` to point to our normal system outside the chroot, it should work. And I did that like so:

```
sudo chroot /mnt/pi /bin/sh
```

This "links" all of the binaries currently installed on my system with the Raspberry Pi disk, allowing me to chroot successfully.

Reflection #16

Date 12/16/2023

Initials Anish Goyal

I installed `x11vnc-server` on the Raspberry Pi in case the team would ever need VNC capabilities (although I ultimately decided not to install a desktop environment) and `tigervnc-viewer` on the host machine. I used a free and open source software (FOSS) program called `linux-wifi-hotspot` to create a WLAN software access point (commonly referred to as a “hotspot”) for the Raspberry Pi:

```
sudo create_ap wlan0 eth0 wifi raspberry
```

Now, whenever I call the command `wihotspot`, the network “raspberry” will be created (I set the default password to “anish,” and I’ll eventually have to hard code this password when setting up wireless configuration for the RPI). The command didn’t work initially, but I realized I had to enable the `create_ap` service like so:

```
sudo systemctl enable create_ap
```

Reflection #17

Date 12/18/2023

Initials Anish Goyal

Now, it's time to finish installing Arch Linux ARM. Although the disk image has been installed to the Raspberry Pi, it's no more useful than just that—an image. While in **arch-chroot**, I ran the following commands to partition the SD card:

```
1  sudo fdisk /dev/sda # launches a TUI program for managing and formatting existing partitions
2  sudo mkfs.vfat /dev/sda1 # create the boot sector
3  mkdir boot
4  sudo mount /dev/sda1 boot # mount the boot sector to a folder named boot
5  sudo mkfs.ext4 /dev/sda2 # create an ext4 file system
6  mkdir root
7  sudo mount /dev/sda2 root # mount ext4 file system
```

I also installed the **zsh** shell, and configured the root user to use it by default instead of Bash (Bourne Again Shell):

```
chsh 0 /bin/zsh
```

It's finally time to extract the root file system from the official Arch Linux ARM source files:

```
1  wget http://os.archlinuxarm.org/os/ArchLinuxARM-rpi-armv7-latest.tar.gz
2  bsdtar -xpf ArchLinuxARM-rpi-armv7-latest.tar.gz -C root
3  sync
```

And finally, I unmounted the boot and root partitions:

```
umount boot root
```

Reflection #18

Date 12/20/2023

Initials Anish Goyal

Before handing over the Raspberry Pi for experimentation to the group, I needed to configure two more things: SSH and wireless networking. I inserted the SD card back into the reader and chrooted into the system. I previously forgot to mount the `/proc` filesystem, so I made sure to also do that this time:

```
1  sudo mount /dev/sda2 arm_mountpoint
2  cd arm_mountpoint
3  sudo chroot . bin/bash
4  sudo mount -t proc proc /proc
```

I’ve been able to install packages to the Raspberry Pi while chrooted because the host system passed my network configuration to the guest file system. However, if booting the Raspberry Pi natively, it will not have any connection. Therefore, I did the following to set up a permanent network connection:

```
1  sudo pacman -S networkmanager # helper package to configure network connections
2  sudo nmtui
```

After entering the Network Manager TUI selection screen, I choose “activate a network connection.” I made sure my laptop was emitting the personal hotspot with SSID raspberry and password anish, configured those details in nmtui, and pressed “Activate.” I also configured the network to automatically connect on device boot (or on network disconnect with a 5 second time interval).

Then, I installed ssh capabilities onto the Raspberry pi with

```
sudo pacman -S openssh-server
```

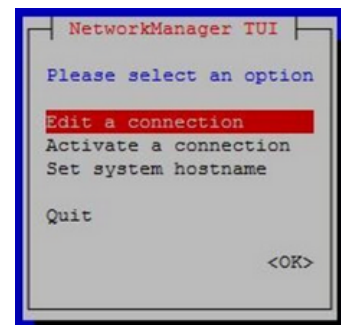


Figure 4: nmtui selection screen

And enabled the OpenSSH daemon with:

```
1 sudo systemctl enable sshd
2 sudo systemctl start sshd
```

To test whether the SSH server was indeed working and the RPi was connected to the proper network, I launched another terminal and tried connecting via password authentication and a static IP address (connecting via the host name didn't work unfortunately, but that's because DHCP typically doesn't work on wireless access points, BUT ONLY FOR THE FIRST CONNECT):

```
ssh alarm@192.168.12.44
```

And it was successful! I was able to remotely send commands via emulating the Raspberry Pi disk through chroot, even though it wasn't *technically* remotely connected. Before unmounting the disk, I removed the default user “alarm” on the Arch Linux ARM disk, and changed the hostname of the RPi (also named “alarm” by default) to archlinux-mini. I also copied the private and public keys of the Raspberry Pi onto my own system, so that I would be able to connect in the future without having to enter a password:

```
1 ssh-copy-id root@archlinux-mini
2 ssh root@archlinux-mini # shouldn't prompt a password
3 pwd # should be in the root user's home directory (/root)
```

Finally, I created a backup of the Arch Linux ARM filesystem by copying the entire disk to a flash drive. I unmounted all of the disks, took out the SD card from the reader, and put it into the Raspberry Pi. Once I connected the Raspberry Pi, it automatically connected to the network named “raspberry” and SSH was fully functional. The Raspberry Pi was now fully ready for our group's use case. I gave the RPi (along with the SD card reader if any of the files were to become corrupted) to Dobromir the next day.

Part IV

Required Forms

The required forms for this project are:

- GSEF Participation Agreement
- Official GSEF Abstract Form
- Checklist for Adult Sponsor [Form 1]
- Checklist for Adult Sponsor [Form 1a]
- Research Plan/Project Summary
- Approval [Form 1B]

They can be found on the following pages.

Part V

Research Plan

We expect that an optimized model with an optimized propulsion system will be significantly faster and more energy efficient because of previous research from the Institute of Georgia Technology and University of Chicago showing statistical significance when it comes to an optimized model or propulsion system respectively.

Expected Outcomes:

The model which is optimized for its design and its propulsion will be statistically significantly more energy efficient and faster than the other models.

Procedures:

1. Determine the baselines, requirements for the project, resources necessary and the parameters needed.
2. Create a CFD model of the environment using Navier Stokes, SIMPLE algorithms. Make assumptions and place values for the specifications of the project.
3. Use the 3D Vector space created by the CFD model to make a 3d model of the object.
Finish up the model and clear up any imperfections. Also split the model in half so we can attach the electronics to the model.
4. Make a machine learning model from the synaptic data.
5. Make a propulsion method which will take in the synaptic data and output the optimal propulsion necessary in the system.
6. 3D print the first half of the 3D Model and install electronics, specifically the breadboard and the raspberry pi to the model.
7. Configure the electronics on the model, install the methods, the propulsion method and ML models into the raspberry pi. Attach cables to the outside of the model and plug it to the power supply. If time permits, install the Jevois camera system.

Rationale:

There are two mechanisms behind undulatory swimming, propulsion and a computational fluid dynamic model (CFD). First by making a computational fluid dynamic model using Poincaré polynomials, polygonal linkages, and Navier Stokes equations with a Semi-Implicit Method for Pressure Linked Equations (SIMPLE), you can create a topological vector space of a model given a certain environment. By connecting the vectors you are able to create a 3D model that is optimized for the given environment. Next you are able to create a propulsion system by modeling the synaptic activity of a species which is already optimized to traverse through water, a fish. By modeling the synaptic activity you are able to create an energy-efficient and fast propulsion system. Combining both an optimized model and a propulsion system gives you a fast moving and energy efficient robot able to move through water. The project has implications for aquatic exploration, optimization for robots within liquid mediums, and energy efficiencies in pre-existing systems.

Research Questions:

Is the optimized robot with a propulsion system statistically significantly faster than a non-optimized robot? Is that robot statistically significantly faster than the optimized robot with no propulsion system? Is there a statistically significant difference in energy expenditure?

Hypothesis:

A model which is optimized for its liquid medium and optimized for its propulsion system will be more energy efficient and faster than a model which is not optimized for its propulsion and design.

Engineering Goal:

-
8. Glue the top half of the 3D printed model and begin setting up for the experiment: 2 containers, one with the optimized model and one without. Fill up the container with deionized water, attach the electronics to the power supply and exercise the safety precautions outlined.
 9. Record the data with a stopwatch and with the chip attached to the breadboard for the power consumption.
 10. Plug in the data to the data table we established and determine if there is a statistical significance in the time the model traveled and the power consumption.
 11. Safely dispose of the deionized water and clean up the project.

Risks and Safety:

Since the project involves the usage of a model near a liquid there are safety precautions we will take. We will use deionized water to make sure a low chance of shock will occur. We will use duct tape and polyurethane around the model to make sure that no water enters the model. We will use a model that is only partially submerged in water with electrical wiring being above the water level. We will not touch the model until it has completed its course. We will have the supervision of an electrical engineer in our initial prototypes.

Data Analysis:

We will take the three data outcomes for speed: the time it took the unoptimized robot with no propulsion model, the time it took the optimized robot with no propulsion system, and the time it took the optimized robot with a propulsion system. We will compare the statistical significance of the optimized robot with a propulsion system to the other data types using a t-test to determine if the time is statistically significant, 5%. We will also analyze the energy expenditure (work/time) using a chip measuring power connected to the raspberry pi to

determine if the optimized robot also has a statistically significant amount of energy expenditure compared to the other two robots. The energy expenditure data will also use a t-test.

Bibliography:

Giraldo Londono, O. (2020, October 8). *Topology optimization of single- and multi-material structures: From single-physics to multi-physics designs*. GT Digital Repository.
<https://repository.gatech.edu/entities/publication/5ee026fc-d0a4-46d2-9ab7-185427ebff42>

Mitin, I., Korotaev, R., Ermolaev, A., Mironov, V., Lobov, S. A., & Kazantsev, V. B. (2022, November 28). *Bioinspired propulsion system for a thunniform robotic fish*. MDPI.
<https://www.mdpi.com/2313-7673/7/4/215>

Penkava, M. P., & Mulase, M. (2012, April 23). *Topological recursion for the Poincaré polynomial of the combinatorial moduli space of curves*. Advances in Mathematics.
<https://www.sciencedirect.com/science/article/pii/S0001870812001235?via%3Dihub>

Saputra, A. A., Botzheim, J., & Kubota, N. (2023, June 3). *Neuro-cognitive locomotion with dynamic attention on topological structure*. MDPI.
<https://www.mdpi.com/2075-1702/11/6/619>

Risk Assessment Form

Must be completed before experimentation; recommended for all projects. May be required for projects involving Human Participants, Hazardous Chemicals, Materials or Devices or Potentially Hazardous Biological Agents.

To be completed by the Student Researcher(s) in collaboration with Designated Supervisor/Qualified Scientist: (All questions must be answered; additional page(s) may be attached.)

- To be completed and signed by the Designated Supervisor (or Qualified Scientist, when applicable):**

I agree with the risk assessment and safety precautions and procedures described above. I certify that I have reviewed the Research Plan/Project Summary and the International Rules, including the science fair ethics statement and will provide direct supervision.

Designated Supervisor's Printed Name

Signature

Date of Review (mm/dd/yy)

Experience/Training as relates to the student's area of research

Position/Institution

Phone or email contact information

Part VII

Student Checklist

Student Checklist (1A) This form is required for ALL projects.	
1. a. Student/Team Leader: <u>Dobromir Iliev</u> Grade: <u>12</u>	
Email: <u>dobromiriliev@gmail.com</u> Phone: <u>4045109223</u>	
b. Team Member: <u>Ricardo Guardado</u> c. Team Member: <u>Anish Goyal</u>	
2. Title of Project: <u>Exploring the Fluid Dynamics of Undulatory Swimming: A Topological Analysis and Computational Modeling</u>	
3. School: <u>GSMST</u> School Phone: <u></u>	
School Address: <u>970 McElvaney Ln NW, Lawrenceville, GA 30044</u>	
4. Adult Sponsor: <u>Ilia Iliev</u> Phone/Email: <u>6783651509</u>	
5. Does this project need SRC/IRB/IACUC or other pre-approval? <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No Tentative start date: <u>October 20, 2023</u>	
6. Is this a continuation/progression from a previous year? <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No	
If Yes:	
a. Attach the previous year's <input type="checkbox"/> Abstract and <input type="checkbox"/> Research Plan/Project Summary	
b. Explain how this project is new and different from previous years on <input type="checkbox"/> Continuation/Research Progression Form (7)	
7. This year's experimentation/data collection:	
<u>11/15/23</u> <u>12/15/23</u>	
Actual Start Date: (mm/dd/yy) End Date: (mm/dd/yy)	
8. Where will you conduct your experimentation? (check all that apply)	
<input type="checkbox"/> Research Institution <input checked="" type="checkbox"/> School <input type="checkbox"/> Field <input checked="" type="checkbox"/> Home <input type="checkbox"/> Other: <u></u>	
9. Source of Data:	
<input checked="" type="checkbox"/> Collected self/mentor <input type="checkbox"/> Other Describe/url: <u></u>	
10. List the name and address of all non-home and non-school work site(s), whether you worked there virtually or on-site:	
Name <u></u> <u></u>	
Address: <u></u> <u></u>	
Phone/ email <u></u> <u></u>	
11. Complete a Research Plan/Project Summary following the Research Plan/Project Summary instructions and attach to this form.	
12. An abstract is required for all projects after experimentation.	

Part VIII

Research Question/Goal

What are you trying to accomplish?

The rationale for the project is that by incorporating models for synaptic activity and a model you can create a more energy efficient system. Zebrafish are meant to be in homeostasis, therefore biomimicry based upon their movement is likely to be more energy efficient. Additionally pre existing research showcases how the computational fluid dynamic model is efficient in solid state structures, therefore, there is precedent in the model working for a robotic model in aquatics. The unique challenges and conditions of the ocean can be represented with the model.

Part IX

Hypothesis

Include your independent and dependent variables AND justification for your hypothesis. Engineering projects also need a hypothesis because they must test their concepts. Include a separate null hypothesis.

Experimental hypothesis: The biomimetic propulsion system and topological model will exhibit statistically significant differences in energy expenditure.

Null Hypothesis: The biomimetic propulsion system and topological model does not differ in efficiency compared to the traditional propulsion system.

Part X

Data

Part XI

Graphs

Part XII

Statistical Analysis

Part XIII

Photo Documentation

References

- Evgrafov, A. (2004). Topology optimization of navier-stokes equations.
- Fillingham, J., Harrington, A., Deehan, M., Renta, I., Gugliotti, E., Matlock, G., Bayler, E., Davis, S., Ghirardelli, J., Grasso, M., Harmon, M., Liddel, M., Marshak, A., Arzayus, F., Newcomb, L., Wielgus, J., Schepel, K., Schattel, J., & Yencho, M. (2022). Noaa science report. <https://doi.org/10.25923/vsa1-gg59>.
- Qin, C., Yang, S., Chu, Y.-H., Zhang, H., Pang, X.-W., Chen, L., Zhou, L.-Q., Chen, M., Tian, D.-S., & Wang, W. (2022). Signaling pathways involved in ischemic stroke: Molecular mechanisms and therapeutic interventions. *Nature News*. <https://www.nature.com/articles/s41392-022-01064-1>.
- Queralta, J. P., Yuhong, F., Salomaa, L., Qingqing, L., Gia, T. N., Zou, Z., Tenunen, H., & Westerlund, T. (2019). Fpga-based architecture for a low-cost 3d lidar design and ... *IEEE Xplore*. <https://ieeexplore.ieee.org/abstract/document/8956928/>.
- Roussel, Y., Gaudreau, S. F., Kacer, E. R., Sengupta, M., & Bui, T. V. (2021). Modeling spinal locomotor circuits for movements in developing zebrafish. *eLife*, 10, e67453. <https://doi.org/10.7554/eLife.67453>.