Gait Optimization for a Multilink Anguilliform Swimmer

By

Alexander Joshua Wiens Department of Mechanical Engineering McGill University, Montreal

August 2012

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master of Engineering (Thesis).

©Alexander Joshua Wiens 2012

Dedication

This document is dedicated to my father, Ron Wiens, for his guidance, support, and encouragement in pursuing my goals.

Acknowledgements

I would like to thank my advisor, Professor Meyer Nahon, for his prudent guidance and editorial advice throughout my master's project.

I would also like to thank NSERC for their financial support through the CGS-M scholarship.

Abstract

Hyper-redundant mechanisms (HRMs), also known as snake-like robots, have been the target of a small but focused research push over the past four decades. Consisting of a simple kinematic chain with a large number of redundant degrees of freedom (DoF), they can act manipulators approximating the form and function of an elephant's trunk, or undulatory locomotors mimicking the motions of snakes, worms, and other creatures. Although past research on locomotion has largely restricted itself to land-based studies, hyper-redundant mechanisms are inherently well suited to aquatic propulsion. Their structural form allows them to directly mimic the swimming motions of anguilliform fish. Biological anguilliform swimmers are both maneuverable and extremely efficient, however, these properties hinge upon finely tuned body deformations. The current understanding of undulatory swimming does not provide a clear method to optimally define these deformations for a highly articulated system. The present study solves this issue by developing a scheme capable of producing optimal gaits for a hyper-redundant swimmer. The optimization process consists of a self-propelled swimming model and a custom particle swarm algorithm. The proposed scheme is used to produce optimal gaits for efficient swimming over a range of different velocities and for high acceleration. Although the development of the gait generation process is an end in itself, the properties of the optimal swimming kinematics also provide insight on HRMs and undulatory swimming in a more general sense. Simple control strategies, key issues for design, and potential topics for future work are extracted from the results.

Abrégé

Les recherches précédentes on montrés que des mécanismes hyper-redondants (MHR) sont fortement adaptables en bougeant sur la terre. Cependant, leurs capacités pourraient aussi être étendues aux environnements aquatiques par la propulsion semblable à celle de l'anguille. Les nageurs anguilliforme naturels sont autant manuvrables qu'extrêmement efficaces. Cependant, ces propriétés dépendent de déformations très spécifiques du corps. La compréhension actuelle de la nage ondulatoire ne fournit pas de méthode claire ou de façon optimale afin de définir ces déformations pour un système fortement articulé. L'étude présentée ici résout cette question en développant un processus informatique capable de produire des démarches optimales pour un robot hyper-redondant nageant. Le processus est composé d'un modèle nageant et d'un algorithme d'essaim de particules faits sur mesure. Cette solution d'optimisation est utilisée pour produire des démarches efficaces pour la natation sur une gamme de vitesses différentes et pour la haute accélération. Bien que le développement du processus d'optimisation soit une fin en soi, les propriétés de la cinématique de la nage optimale fournis aussi un aperu sur les MHRs et sur la natation ondulatoire dans un sens plus général. Des stratégies de contrôle simples, des problèmes-clés pour le design, et des sujets potentiels pour le travail à venir sont extraits des résultats.

Contents

| De | edica | tion | | ii | | | |
|-------------------------|---------------------|---------|---|----|--|--|--|
| A | Acknowledgements ii | | | | | | |
| A | Abstract | | | | | | |
| A | brégé | | | iv | | | |
| Li | st of | Figure | 28 | ix | | | |
| Li | st of | Tables | | xi | | | |
| 1 | Intr | oducti | on | 1 | | | |
| | 1 | Hyper- | Redundant Mechanisms | 2 | | | |
| | | 1.1 | Land-Based Locomotion | 2 | | | |
| | | 1.2 | Aquatic Locomotion | 6 | | | |
| 2 Swimming Optimization | | 9 | | | | | |
| | | 2.1 | Modelling the Fluid-Swimmer Interaction | 10 | | | |
| | | 2.2 | Past Work on Swimming Gait Optimization | 12 | | | |
| | 3 | Thesis | Objectives and Layout | 13 | | | |
| 2 | Mod | lelling | | 15 | | | |
| | 1 | Geome | try and Kinematics | 15 | | | |
| | | 1.1 | Mechanism Geometry | 15 | | | |
| | | 1.2 | Swimming Kinematics | 16 | | | |

| | 2 | Dynamics Model | | | |
|-----------------------------------|-----|-------------------------------------|---|----|--|
| | 3 | Fluids Model | | | |
| | | 3.1 | Lateral Fluid Forces | 22 | |
| | | 3.2 | Axial Fluid Forces | 24 | |
| | | 3.3 | Fluid Moment | 25 | |
| | 4 | Demonstrating the Model | | | |
| | 5 | Energ | getics | 27 | |
| 3 | The | e Opti | mization Algorithm | 31 | |
| | 1 | The E | Efficient Swimming Optimization Problem | 32 | |
| | 2 | The C | Genetic Algorithm | 33 | |
| | | 2.1 | Outline of Genetic Algorithm Structure | 33 | |
| | | 2.2 | Modifications to the Basic Genetic Operators | 34 | |
| | | 2.3 | Testing the Modified Algorithm | 39 | |
| | 3 | Particle Swarm Optimization | | | |
| | | 3.1 | Outline of Particle Swarm Algorithm Structure | 40 | |
| | | 3.2 | Modifications to the Basic Particle Swarm Algorithm | 42 | |
| | | 3.3 | Testing the Algorithm | 43 | |
| | 4 | Comparing the Algorithms | | | |
| 4 Efficient Swimming Optimization | | | | 45 | |
| | 1 | Revis | iting the Gait Parameterization | 46 | |
| | 2 | Optin | nization Results | 50 | |
| | 3 | Discu | ssion | 55 | |
| | | 3.1 | Stochastic Gait Optimization and Efficient Swimming | 55 | |
| | | 3.2 | Control and Design for Efficient Swimming | 56 | |
| | | 3.3 | Negative Work and Energy Recovery | 58 | |
| 5 | Acc | celerat | ion Optimization and Swimming Control | 61 | |
| | 1 | Accel | eration Maneuvers in Fish | 62 | |
| | 2 | Setting Up the Optimization Problem | | | |

| | | 2.1 | Parameterizing the Maneuver | 64 | |
|---|-----|--|--|----|--|
| | | 2.2 | The Objective Function and Power Constraints | 65 | |
| | 3 | Results and Discussion | | | |
| | | 3.1 | Results | 67 | |
| | | 3.2 | Discussion | 70 | |
| | 4 | Combining the Optimal Behaviours for Control | | | |
| | | 4.1 | Velocity Tracking Control | 74 | |
| | | 4.2 | Velocity Tracking with Waypoint Navigation | 76 | |
| 6 | Con | clusio | ns and Recommendations | 81 | |
| | 1 | The Swimming Model | | | |
| | 2 | The Optimization Algorithm | | | |
| | 3 | The O | ptimization Results | 84 | |

List of Figures

| 1.1 | Photograph of the ACM-III mechanism. | 3 |
|-----|--|----|
| 1.2 | Photograph of ACM-R3 mechanism. | 4 |
| 1.3 | HRM designs actuated through active wheels or tracks | 5 |
| 1.4 | Examples various HRM behaviours | 6 |
| 1.5 | HELIX and ACM-R5 swimming robots | 7 |
| 1.6 | Photograph of Amphibot amphibious robot. | 8 |
| 1.7 | Six link segment of Robot Anguille | 9 |
| 2.1 | Visual depiction of ten link swimming mechanism | 16 |
| 2.2 | Diagram of variables used in description of Serpenoid Curve | 17 |
| 2.3 | Comparison of the four body-caudal fin swimming modes | 19 |
| 2.4 | Force and vector quantities used in dynamics model | 20 |
| 2.5 | Velocity components and link coordinate for torque calculation | 26 |
| 2.6 | Velocity plot for swimmer accelerating from rest | 27 |
| 2.7 | Power balance for swimmer accelerating from rest | 29 |
| 2.8 | Plot of joint power for actuators near tail of mechanism | 30 |
| 3.1 | Visual depiction of roulette wheel selection process | 35 |
| 3.2 | Diagram of gait encoding applied within the genetic algorithm | 37 |
| 3.3 | Visual depiction of crossover process. | 38 |
| 3.4 | Visual depiction of the modified crossover process | 38 |
| 3.5 | Test runs for genetic algorithm. | 39 |
| 3.6 | Plots depicting convergence of the PSO swarm. | 41 |

| 3.7 | Diagram of damped reflection process. | 42 |
|------|---|----|
| 3.8 | Test runs for Particle Swarm algorithm | 43 |
| 3.9 | Comparison of optimization test runs | 44 |
| 4.1 | Crude propagating wave gait. | 48 |
| 4.2 | Crude flapping gait | 48 |
| 4.3 | Optimal transport economies from various parameterizations | 49 |
| 4.4 | Relative comparison of optimal transport economies | 49 |
| 4.5 | Optimal frequencies for efficient swimming | 50 |
| 4.6 | Optimal wavelength for efficient swimming | 51 |
| 4.7 | Optimal transport economy during efficient swimming | 51 |
| 4.8 | Optimal Strouhal Number for efficient swimming | 52 |
| 4.9 | Optimal amplitude for efficient swimming (abs. work) | 53 |
| 4.10 | Optimal amplitude for efficient swimming (net work) | 53 |
| 4.11 | Comparison of gait envelopes. | 54 |
| 4.12 | Power distribution during efficient swimming. | 59 |
| 4.13 | Thrust distribution during efficient swimming | 60 |
| 5.1 | Traces of live fish executing fast-starts | 63 |
| 5.2 | Ramp time and wavelength for optimal acceleration | 67 |
| 5.3 | Amplitude for optimal acceleration. | 68 |
| 5.4 | Optimal acceleration kinematics. | 69 |
| 5.5 | Power distribution during acceleration | 71 |
| 5.6 | Comparison of optimal efficiency and acceleration amplitude profiles. | 73 |
| 5.7 | Acceleration of high efficiency gaits | 74 |
| 5.8 | Diagram of control loop for speed tracking | 75 |
| 5.9 | Speed error during acceleration from rest. | 76 |
| 5.10 | Diagram of desired heading calculation | 77 |
| 5.11 | Snapshot of mechanism during turn. | 78 |
| 5.12 | Diagram of complete control scheme. | 79 |
| 5.13 | Path and speed error during navigation task | 79 |

List of Tables

| 5.1 | Comparison | of mechanism | and pike durin | g acceleration | 70 |
|-----|------------|--------------|----------------|----------------|--------|
| | | | | | |

Chapter 1

Introduction

Hyper-redundant mechanisms (HRMs), also known as snake-like robots, have been the target of a small but focused research push over the past four decades. Consisting of a simple kinematic chain with a large number of redundant degrees of freedom (DoF), they can act manipulators approximating the form and function of an elephant's trunk, or undulatory locomotors mimicking the motions of snakes, worms, and other creatures.

In terms of locomotion they have proven themselves to be highly adaptable. Through the work of number of groups, it has been shown that they are capable of negotiating a wide range of obstacles and terrain types [1, 2, 3]. Although past research has largely restricted itself to land-based studies, hyper-redundant mechanisms are inherently well suited to aquatic propulsion. With the ability to operate underwater, HRM's could potentially fulfill a wide range of robotic applications which call for high maneuverability in complex aquatic environments. Examples include inspection of mechanical systems, search and rescue in flooded areas, and demining operations [2, 3]. These aquatic capabilities could likely be implemented while maintaining the ability to operate on land, producing a highly adaptable amphibious system.

On both land and water, HRM locomotion is achieved through periodic deformations which generate a net displacement of the robot's body [1]. Due to the complexity of the system, defining these motions in an optimal or even functional manner is a nontrivial issue. Faced with this problem, engineers have turned to biological systems for guidance. Although there is a considerable body of literature on the theoretical basis of natural undulatory locomotion, our knowledge of the topic is incomplete. As such, reverse engineering these biological systems to truly replicate their performance has been a daunting task. Considerable time and resources have been invested in solving this problem for mechanisms on land, but relatively little attention has been given to gaits for aquatic HRM locomotion. The work presented here aims to address this issue by developing an algorithmic solution capable of generating optimal HRM gaits for a variety of swimming behaviours.

1 Hyper-Redundant Mechanisms

The following section gives an overview of previous studies on HRM locomotion. Although our work is targeted at hyper-redundant swimming, this review begins with a survey of land-based HRMs to establish the history of the field. While numerous researchers have contributed to the knowledge of HRM locomotion, focus is given to three primary groups which represent of the bulk of previous development. Following this, a discussion of previous work on the aquatic capabilities of HRMs is provided.

1.1 Land-Based Locomotion

The first HRMs were developed by Hirose in the the early 1970's. At this point the primary motivation of his group was to understand and replicate the undulatory movement of snakes. Considering previous studies of snake locomotion to be incomplete, Hirose began his efforts by first developing an improved understanding of the subject through observing live animals. His resulting theory of serpentine kinematics and locomotion methods is laid out in [4]. Hirose applied this theory to develop the the ACM-III (Active Cord Mechanism), the first robot to achieve true snake-like locomotion. Pictured in figure 1.1 the device uses passive wheels along the base to provide the directional friction properties required for undulatory locomotion [2]. In terms of dimensions the ACM-III robot was actually quite large, each of its 20 links weighs 1.4kg resulting in a mass of 28kg for the entire mechanism. The total length of the system is 2m.



Figure 1.1: Photograph of the ACM-III mechanism. Locomotion is achieved using an undulatory gait and passive wheels. [2].

In [2], Hirose highlights the next major HRM development of his group as the ACM-R3 robot. Designed and built in 2001, the device is composed of 20 discrete links. It has a total mass of 12 kg and a length of 1.8m. Once again passive wheels are used for directional friction. As shown in figure 1.2, it is capable of three-dimensional deformation. This is achieved by offsetting every second joint axis by 90 degrees. Three-dimensional body movements allow the device to replicate the more complex behaviours of natural undulatory locomotors. Examples include sinus lifting, a gait in which body segments aligned with the direction of travel are lifted to eliminate axial friction, and sidewinding, a gait applied by snakes on low shear surfaces.

In addition to these two landmark projects, Hirose's lab has developed a several additional HRM-like mechanisms to explore different actuation methods and geometries.



Figure 1.2: Photograph of ACM-R3 mechanism, an improved snake-like robot capable of three dimensional actuation. [2].

The Slim Slime robot, is a novel pneumatically actuated snake-like robot[2]. Through the unique properties of its actuators the mechanism is able to contract and expand longitudinally while also bending its body in a continuous fashion. This allows the device employ worm-like creeping gaits for locomotion in spaces too small for the undulatory motions of snakes. A number projects undertaken by Hirose's group have also explored the potential of combining active wheels or tracks with an HRM body [2]. These mechanisms exploit the inherent adaptability of the elongated snake-like form, while employing simplistic locomotion methods which are much simpler to control. Two examples of such devices are pictured in figure 1.3.

Hirose's pioneering work introduced the topic hyper-redundant locomotion to the engineering research community. His early studies of undulatory locomotion led to the design of the first biomimetic snake-like mechanisms and over the years his group has consistently been at the forefront of HRM hardware development. In terms of applications, Hirose envisions numerous possibilities for HRMs, examples include exploration of complex environments, inspection or repair of pipes and other mechanical systems, and active versions of elongated tools such as endoscopes, ropes, or hoses. However, while Hirose's group has pushed to develop the necessary hardware for these applications, each task also represents a complex control problem. Even simple motions,



Figure 1.3: Examples of HRM designs actuated through active wheels or tracks [2]. The mechanism on the left is the ACM-R4 and the device on the right is the Genbu-III.

such as raising the mechanism's head, require that numerous joints are actuated in a coordinated fashion. The hyper-redundant nature of the system makes it virtually impossible for a human operator to directly accomplish this. Instead, control must be executed through carefully designed predefined maneuvers.

The work of Burdick and his students (Chirikjian, Ostrowski, and Choset) at Cal-Tech during the 1990's, and the continued work of Choset at Carnegie Mellon University over the past decade represent two of largest contributions to the HRM control problem. During the early 90's Chirikijan and Burdick developed a computationally efficient method to fit the discrete body of an HRM to a continuous backbone curve [5]. This allowed them to develop a series of HRM behaviours using continuous functions to define the system's posture. Within their work, they explore the kinematics required to mimic a variety of undulatory locomotion methods, while investigating additional issues such as adapting to variations in terrain, obstacle avoidance, and grasping operations [5, 6, 7]. At approximately the same time as Chirikjian's work, Ostrowski and Burdick developed an analytical model of Hirose's ACM-III mechanism based on a Lagrangian approach [8, 9]. Using geometric mechanics, the model is separated into two distinct components: internal changes in body configuration and the resulting displacement in the inertial frame. This approach allows Ostrowski to comment on the controllability of the system and suggest periodic locomotive gaits based on the mathematical connection between the model components [8].

Choset's work with Burdick explores sensor based motion planning with HRMs as a potential application [10], however, his most significant contributions occured after he founded the BioRobotics Lab at Carnegie Mellon University. Over the past 10-15 years Choset and his students have worked on a number problems related to state estimation [11], mechanism design [12], and motion planning/gait generation[13, 14] for HRMs. The sum of these efforts has a produced a series of hyper-redundant robots capable of a wide range of behaviours. Examples include undulatory locomotion on grass, mud, or confined spaces, and climbing of ramps, stairs, or even poles. A number of these behaviours are pictured in figure 1.4.



Figure 1.4: Examples various HRM behaviours developed by the BioRobotics group at Carnegie Mellon University [15].

1.2 Aquatic Locomotion

In recent years, HRM research has expanded significantly. Numerous groups have constructed prototype devices and are currently using them to explore various methods of locomotion. HRM swimming in particular has received the attention of several researchers within the past decade. Hirose's group at the Tokyo Institute of Technology, demonstrated the feasibility of aquatic HRMs through the HELIX and ACM-R5 robots. HELIX [2], pictured in figure 1.5, is an amphibious HRM capable of locomotion on land and in water. While using standard undulatory locomotion on land; in water it mimics the spiral swimming kinematics of spirochete bacteria. The design of the device was later refined to produce the ACM-R5 which uses simple undulatory motions in water rather than a complex helical gait [16, 2]. Choset's group has also shown that their HRMs are capable of surface swimming, however no work has been published on specific swimming kinematics or aquatic mechanism design.



Figure 1.5: Right: The HELIX, a swimming snake-like robot which uses a helical gait [17]. Left: The ACM-R5, an improved amphibious HRM [2].

Ostrowski continued his research on undulatory locomotion, as a Professor at the GRASP lab of the University of Pennsylvania. Applying the geometric methods of his previous work, Ostrowski and McIsaac, develop a mathematical model of a swimming snake-like robot. To reduce the physics of the system to a tractable problem they quantify fluid forces as a basic linear drag term [18]. In [19] a perturbation analysis is applied to the model to identify gaits for turning, forward, and even transverse swimming. In [20], these methods are extended to produce a closed loop control scheme for an undulatory swimmer. The resulting algorithm is tested with a simple 5-link servo actuated mechanism. Through these experiments, it is demonstrated that their control solution allows the mechanism to track simple paths while rejecting external disturbances.

Crespi and Ijspeert, researchers of the BioRobotics Laboratory at EPFL, are currently pursuing a project to develop a biomimetic amphibious robot. In its present state, the robot consists of a kinematic chain of discrete links connected by angular position servos. As shown in figure 1.6, legs can added or removed to replicate the locomotion methods of various animals such as eels, salamanders, or centipedes. The project aims to develop a functional amphibious robot which can be used as a test-bed to explore novel control solutions based on the nervous systems of real animals. Multiple prototypes of the device have already been produced [21] and in [22] it is shown that these system can be controlled via central pattern generators roughly tuned using online optimization techniques.



Figure 1.6: Amphibot, designed by the BioRobotics group at EPFL. Shown here with and without legs [23].

The final major HRM research initiative discussed here, is the the Robot Anguille project of the ROBEA program in France. Motivated by the astounding performance of biological swimmers, several French researchers are collaborating to produce an eel-like autonomous underwater vehicle. The project consists of several parallel research streams to develop hardware [24], low-cost simulations [25, 26, 27], and three dimensional control [28] for an elongated, highly articulated swimmer. In their most recent publications, the robot remains incomplete [24], however solutions have been developed for actuated three DoF joints, compliant waterproof skin, and onboard control and power. In its final form the envisioned system contains 12 links and has total length over 2m. Based on current actuator choices it has a maximum power output of 100w and can swim continuously for 20 minutes. A six body segment portion of the mechanism is shown during testing in figure 1.7, along with a single segment with the outer skin removed.



Figure 1.7: Left: A six link system during underwater testing. Right: Photograph of a single mechanism link with outer skin removed [24].

2 Swimming Optimization

Despite the fact that HRM swimming has been a topic of interest for over a decade, the problem of generating truly optimal swimming gaits has been given relatively little attention. In most cases, a simple sinusoidal motion pattern is applied to achieve forward propulsion. Although biological swimmers exhibit impressive performance, it has been found that this performance hinges upon specific swimming kinematics. In their optimization work [21], Crespi and Ijspeert find that the velocity of their swimmer can change significantly for even small variations of the gait parameters. In their well-known work on bioimimetic swimming, Barrett and Triantafyllou find that efficiency drops off rapidly if optimal kinematics are altered [29]. Considering this, locomotive gaits which appear similar in a qualitative sense may provide very different end results. Consequently, a crude of replication of fish-like motions is likely inadequate to reproduce the capabilities of natural swimmers. Achieving truly optimal performance will require a solution for producing well tuned swimming gaits.

Gaits for land-based locomotion in HRMs have typically been defined through a manual tuning process. This is effective in producing functional HRM behaviours, however it is poorly suited to generating highly optimized kinematics. Due to the complexity of the encompassing optimization problem, the sheer size of the of required parameter space would necessitate a massive time investment to adequately explore it using online tuning methods. Simulation based optimization has been explored by a number of groups looking at other types of swimming mechanisms, however it has never been applied for HRM-like swimmers. Simulation based optimization requires two key components: a self-propelled swimming model, and an effective optimization scheme. The model is used to predict the mechanism's performance as an emergent property of its gait and the resulting interactions with the surrounding fluid. The optimization process then applies this model to search for gait kinematics which maximize specific performance parameters such as efficiency or speed. The following thesis aims to address the HRM swimming problem by developing a simulation based gait optimization solution which is tailored to meet the needs of HRM geometry and actuation characteristics. In preparation for this, the following section will first provide an overview of previous work related to the modelling problem. Once this has been established, a review is given on past studies related to simulation based swimming optimization.

2.1 Modelling the Fluid-Swimmer Interaction

The modelling component of the optimization scheme presents a major challenge; the fluid mechanical basis of undulatory swimming has been a topic of interest for almost a century [30]. Among early works on undulatory swimming, two of the most notable analyses are the resistive approach proposed by Taylor [31] and Lighthill's reactive model [32, 33, 34, 35]. The reader may refer to [36] for a summary of their models. Both Lighthill's and Taylor's work contributed significantly to our understanding of undulatory swimming, however their analytical methods require a number of major simplifying assumptions to produce tractable mathematical solutions. The validity of these assumptions varies significantly based on the flow regime and geometry of the swimmer in question. For small swimmers (< 10cm) in low Reynolds Number flow, viscous forces dominate and Taylor's drag based approach is most applicable.

In developing his reactive analysis, Lighthill acknowledges this fact, hypothesizing that a solely resistive model may be applicable up to Re as high as 10³ for elongated cylindrical animals [34]. However, Lighthill argues that in larger swimmers, evolved to maximize added-mass effects, reactive forces are the primary source of thrust. Unfortunately, large anguilliform swimmers and HRMs typically lie somewhere between these two extremes. Consequently, resistive and reactive forces both play a critical role in their propulsion [37].

Considering this, a representative model of HRM swimming must account for both drag and added mass effects. There are a number significant analytical works in addition to the models of Taylor and Lighthill, noteworthy examples include [38, 39, 40], nevertheless an analytical solution which encompasses both resistive and reactive forces has yet to be realized. The advent of low-cost high-power computing technology has enabled researchers to instead approach the problem through CFD based investigations. However, devising effective and efficient CFD methods remains difficult as the swimming problem involves transient flow structures around a deforming body coupled to the fluid which surrounds it. A number of research groups have developed CFD solutions capable of dealing with these issues and have applied them to great effect [41, 42, 43, 44, 45, 46]. However, the computation time required by such methods remains prohibitive in the context of gait optimization. As a result, the CFD approach is generally restricted to studies which investigate a narrow scope of swimming conditions.

When low computational cost is required, researchers often turn to low-order semiempirical models. Within these models, the analytical methods for both added mass and drag based propulsion are combined to produce a swimmer which is propelled by both effects. This approach is frequently employed in biological studies for the calculation of internal powers and torques during recorded fish maneuvers [47]. In this role, the model is limited to inverse dynamics, however, a number of groups have extended it to study swimming performance in a forward sense, where the motion of the swimmer is determined as function of body deformation and fluid-swimmer interaction. Although this method cannot produce the elegant mathematical results of its analytical constituents nor the fidelity of CFD based solutions, it has been shown to provide reasonably accurate, low cost simulations of intermediate Reynolds number swimming [37, 48, 26]. Based on this, it is taken as the method of choice for the optimization work presented here.

2.2 Past Work on Swimming Gait Optimization

The second component of the optimization scheme is the optimization algorithm itself. Due to the complexity of the optimization space, gradient-free approaches are well-suited to the swimming problem. These methods generally provide improved performance in large optimization spaces where gradient based methods tend to get stuck in local optima. Over the past decade, a number of research groups have combined gradient-free optimization with swimming simulation to study undulatory propulsion. Morgan and Smart apply a drag-based swimming model with both a genetic algorithm and simulated annealing to generate optimally efficient gaits for a four link swimmer [49]. Kuo and Grierson model a three segment carangiform swimmer and employ a genetic algorithm to explore gaits for both efficient and fast-start swimming. They apply a simplified fluids model based on the Kutta-Jukowski theorem to quantify the thrust generated by the foil-like tail of the swimmer [50]. The most rigorous optimization study to date is likely the work of Kern and Koumoutsakos [44]. They use a custom evolutionary algorithm and a three dimensional numerical solution to the Navier-Stokes equations to produce gaits for both efficient and high velocity swimming with an anguilliform swimmer. However, due to the computational cost of their model only two optimization runs are executed, one for each fitness goal.

It should be noted that a small number of groups have conducted optimization studies using physical robots to experimentally assess gait fitness. In the work of Crespi and Ijspeert, mentioned earlier, Powell's method is used to generate roughly optimized gaits for a variety of behaviours. In [51], Barrett et al. employ a genetic algorithm to maximize the efficiency of the well-known swimming robot, RoboTuna. Interestingly, the time required to evaluate a gait with the physical robot is actually significantly shorter than full CFD simulation (minutes versus hours), however the process requires constant supervision and resetting of the robotic system.

3 Thesis Objectives and Layout

The work presented here aims to develop an algorithmic solution capable of generating optimal HRM gaits for a variety of swimming maneuvers. To accomplish this, a semi-empirical swimming model is applied within a custom gradient-free optimization algorithm. The resulting optimization scheme is then used to generate gait definitions for two swimming behaviours: efficient locomotion and rapid acceleration. For each optimization goal the trends and properties of the resulting swimming kinematics are used to suggest design modifications for future HRMs while highlighting simple but effective methods for mechanism control

This optimization process is developed and then applied within the proceeding document as follows. First, the swimming model is presented in Chapter 2. This includes discussions of the kinematics, dynamics, and energetics of the simulated swimming mechanism. In Chapter 3, the optimization algorithm is laid out. As highlighted in the preceding review, genetic algorithms are the method most commonly applied in previous work. A custom GA was initially developed to serve as the optimization component of the gait design process, however, its performance was found to be somewhat lacklustre for our particular problem. Considering this, an alternative gradient-free method known as Particle Swarm Optimization is investigated. Relative to GA, it is found to produce consistently superior results while requiring considerably less development time. Based on this, it is chosen as the final algorithm for the optimization study. The structures of the both the GA and PSO algorithms are presented, and test runs are then compared to demonstrate the superiority of the PSO approach. In Chapter 4 the swimming model and PSO algorithm are applied to generate optimally efficient swimming gaits. The process involves the development of an appropriate gait parameterization, presentation of the optimization results, and a discussion of the resulting kinematics as they relate HRM design and control. This process is repeated to produce high acceleration gaits in Chapter 5. In the last section of Chapter 5, a simple control solution is proposed to demonstrate that the optimal efficiency and acceleration kinematics can be concisely but effectively combined for swimming control.

Chapter 2

Modelling

The first component of the gait optimization algorithm is the swimming simulation. The development of the self-propelled model is broken into three sections. The discussion begins with the kinematics of the simulated mechanism. This includes a description of the swimmer's geometry and an overview of biological swimming modes. Once this has been done, the dynamics equations of the system are presented. With the dynamics established, the fluid-swimmer interaction model is developed. Finally, having fully elucidated the three major components of the model, the performance of the completed simulation is demonstrated and the energetics of the system are investigated in preparation for the analysis of efficient swimming.

1 Geometry and Kinematics

1.1 Mechanism Geometry

While snakes and eels can have hundreds of vertebrae, hyper-redundant robots typically have significantly fewer discrete links. The mechanism simulated for the optimization study is represented as a kinematic chain made up of ten segments connected by single degree of freedom pin joints as shown in figure 2.1. To approximate the geometry of a typical swimming HRM, each segment is represented as an elliptical cylinder with its major diameter parallel to the joint axis. The head and tail segments are assumed to have streamlined hemispherical caps to reduce drag. The motion of the system is restricted to a two dimensional plane to ensure that the problem remains tractable for gait optimization. Biological swimmers on the scale of the mechanism typically employ planar gaits which suggests that this is a reasonable simplification. Swimming HRMs presented in previous works range from 50 cm up to 2 m [16, 25] in length. To roughly approximate this range, the length of the simulated mechanism is chosen as 1m. The cylindrical links are specified to have a major diameter of 0.1m and a minor diameter of 0.05m. Since the mechanism is made of up ten segments, each cylinder is 0.1m in length. The mechanism is assumed to have same density as water to eliminate buoyancy effects.



Figure 2.1: Visual depiction of ten link swimming mechanism.

1.2 Swimming Kinematics

The Serpenoid Curve

The basic gait observed in undulatory locomotors consists of a body wave which propagates from head to tail. Initially, it was believed that this wave could be best represented as sinusoidal variation of lateral body displacement, however in 1993 Hirose observed that this definition required an unnatural pattern of muscle activations



Figure 2.2: Diagram of variables used in description of Serpenoid Curve.

along the animal's body. Based on this, he proposed an alternative gait definition known as the Serpenoid Curve. The Serpenoid Curve describes the body undulations as a sinusoidal wave of curvature rather than lateral position. Since body curvature is directly controlled by the antagonistic muscle pairs along the animal's spine this definition ensures a smooth sinusoidal activation pattern. In terms of Cartesian coordinates, the Serpenoid Curve is mathematically defined as follows,

$$x(s) = \int_0^s \cos(a\cos(b\sigma) + c\sigma) d\sigma, \qquad (2.1)$$

$$y(s) = \int_0^s \sin(a\cos(b\sigma) + c\sigma) d\sigma, \qquad (2.2)$$

where s is the body-coordinate along the specified curve as depicted in figure 2.2. x(s) and y(s) are the cartesian coordinates of the point delineated by a given value of s and σ is a dummy variable for integration. It can be shown that this curve has the following curvature expression,

$$\kappa(s) = ab\sin(bs) - c. \tag{2.3}$$

Variables a, b, and c determine the properties of the curvature wave. While the definition of the Serpenoid curve is considerably more complex than a simple Cartesian sinusoid, it reduces to a relatively compact form when discretized to a function of joint angles for a segmented mechanism [52]. This makes it ideal for defining the motions of the simulated HRM. The resulting joint angle expression is given as follows,

$$\phi_i = \alpha \sin(i\beta + \beta/2) + \gamma, \qquad (2.4)$$

where ϕ_i is the angle between consecutive segments and,

$$\alpha = a |\sin(\beta/2)|, \ \beta = b/2, \ \gamma = -c/n.$$

$$(2.5)$$

The coefficient, α , determines the amplitude of the serpenoid curve, γ introduces constant curvature to the body, and the β terms inside define the phase difference between consecutive segments. This expression, derived in [52], is simplified to define the gait of the simulated mechanism during straight line motion. The γ term is dropped to ensure a symmetrical gait for straight swimming and the $\beta/2$ term is eliminated, as the gait of the swimmer depends only on the relative phase of the joint angles. A time term is added inside the sine function to generate undulations as the simulation advances. These changes result in the following gait definition,

$$\phi_i = \alpha \sin[2\pi (iL/\lambda - t/T)]. \tag{2.6}$$

Undulatory Swimming Modes

In most swimmers, the amplitude of the curvature wave grows as it travels towards the tail. Applying this knowledge, the joint angles of the simulated mechanism can be defined using the discrete approximation of Hirose's Serpenoid curve coupled with an exponential amplitude function. The resulting expression is given as follows,

$$\phi_i = A e^{\epsilon i L} \sin[2\pi (i L/\lambda - t/T)]. \tag{2.7}$$

This expression is used as the gait parametrization during the process of developing the swimming model and the optimization algorithm. It will be revisited in Chapter 4, prior to the formal optimization study.



Figure 2.3: Comparison of the four body-caudal fin swimming modes a) anguilliform b) subcarangiform c) carangiform d) thunniform (Figure taken from [54]).

Biological swimmers which propel themselves through body and caudal fin undulations are often classified based on the amplitude function $(Ae^{\epsilon iL})$ and wavelength λ which best describe their gait [45]. They are typically divided into four primary categories: anguilliform, subcarangiform, carangiform, and thunniform [53]. Fish classified as anguilliform swim with a wavelength of around 60% to 70% of body length and an amplitude profile with relatively gentle slope which engages most of the body. In the subcarangiform mode, the wavelength increases and anterior undulations are attenuated. These trends continue in the transition to the carangiform mode. Carangiform swimmers such as sunfish or trout, employ a wavelength slightly larger than total body length and body undulations are typically limited to the last third of the fish [30]. The final class, thunniform, applies to fish which are propel themselves almost entirely through oscillations of a foil like tail, examples include tuna and swordfish. Figure 2.3 provides a visual comparison of the four swimming classes. The reader is referred to [53] for further information on the various swimming modes.

2 Dynamics Model

With the geometry and kinematics of the mechanism established, it is now possible to lay out the dynamics model of the system. The particular dynamics problem addressed here is that of forward dynamics: given a joint motion time history, find the ensuing system motion and required joint torques. Previous works on the modelling of a discretized swimmer employ a variety of approaches to accomplish this. A Lagrangian method is applied by McIsaac and Ostrowski [55] for the purposes of a perturbation study. By contrast, the Robot Anguille researchers approach the problem from a robotics background and thus apply a recursive Newton-Euler algorithm [25]. For the investigation presented here, we elect to apply a simple force balance method similar to that of Ekeberg [56]. The simplicity of this approach makes it relatively easy to implement and transparent to the reader.

The balance method consists of formulating the moment and x-y force equations and kinematic constraints for each segment of the simulated swimmer, this generates six equations per link. Since the robot moves freely in the inertial frame, the position constraints are dropped for the head link. This results in a total of 6n-3 equations to completely define the system, where n is the total number of body links. To illustrate the process of creating the model, the equations for a three link eel are developed here. Figure 2.4 provides a diagram of the necessary vector quantities.



Figure 2.4: Force and vector quantities used in dynamics model.

Known Variables:

- + $\mathbf{F}_{i_{fluid}},\,T_{i_{fluid}}$ Force and moment on link i due to fluid-structure interaction

- m_i, I_i Mass and angular inertia of link i

Unknown Variables:

- \mathbf{F}_i Internal force at joint i
- \mathbf{M}_i Internal moment at joint i
- \mathbf{a}_i Acceleration of i^{th} link's centre of mass
- $\dot{\theta}_i, \ddot{\theta}_i$ Angular velocity and acceleration of i^{th} link

$$Link 1 \quad m_1 \mathbf{a}_1 = \mathbf{F}_1 + \mathbf{F}_{1_{fluid}} \tag{2.8}$$

$$\mathbf{I}_{1}\ddot{\theta}_{1} = \mathbf{T}_{1} + \mathbf{T}_{1_{fluid}} + \mathbf{r}_{1,1} \times \mathbf{F}_{1}$$

$$\tag{2.9}$$

Link 2
$$m_2 \mathbf{a}_2 = -\mathbf{F}_{2-1} + \mathbf{F}_2 + \mathbf{F}_{2_{fluid}}$$
 (2.10)

$$\mathbf{I}_{2}\ddot{\theta}_{2} = -\mathbf{T}_{1} + \mathbf{T}_{2} + \mathbf{T}_{2_{fluid}} - \mathbf{r}_{2,1} \times \mathbf{F}_{1} + \mathbf{r}_{2,2} \times \mathbf{F}_{2}$$
(2.11)

$$\ddot{\theta}_2 = \ddot{\theta}_1 + \ddot{\phi}_1 \tag{2.12}$$

$$\mathbf{a}_{2} = \mathbf{a}_{1} - \dot{\theta}_{1}^{2} \mathbf{r}_{1,1} + \ddot{\theta}_{1} \times \mathbf{r}_{1,1} + \dot{\theta}_{2}^{2} \mathbf{r}_{2,1} - \ddot{\theta}_{2} \times \mathbf{r}_{2,1}$$
(2.13)

$$\mathbf{Link 3} \quad m_3 \mathbf{a}_3 = -\mathbf{F}_2 + \mathbf{F}_{3_{fluid}} \tag{2.14}$$

$$\mathbf{I}_{3}\ddot{\boldsymbol{\theta}}_{3} = -\mathbf{T}_{2} + \mathbf{T}_{3_{fluid}} - \mathbf{r}_{3,2} \times \mathbf{F}_{2}$$

$$(2.15)$$

$$\ddot{\theta}_3 = \ddot{\theta}_2 + \ddot{\phi}_2 \tag{2.16}$$

$$\mathbf{a}_{3} = \mathbf{a}_{2} - \dot{\theta}_{2}^{2} \mathbf{r}_{2,2} + \ddot{\theta}_{2} \times \mathbf{r}_{2,2} + \dot{\theta}_{3}^{2} \mathbf{r}_{3,2} - \ddot{\theta}_{3} \times \mathbf{r}_{3,2}.$$
(2.17)

The force and moment balance equations consist of constant terms defined by the properties of the mechanism, fluid forces calculated based on the velocity and acceleration of the system, and unknown internal forces. The constraint equations consist of linear and angular expressions based on the joint angles (ϕ_i) defined by the prescribed gait. In total there are 6n - 3 unknown variables: $\mathbf{T_i}$ and $\mathbf{F_i}$ for n - 1 joints and, $\mathbf{a_i}$ and θ_i for n links. At each time step, the system of equations is solved simultaneously for all unknowns. The accelerations are then numerically integrated using a MATLAB variable step solver (ODE45) to advance the simulation.

3 Fluids Model

The force and moment balance equations require that we define the fluid terms $\mathbf{F}_{i_{fluid}}$ and $\mathbf{T}_{i_{fluid}}$ for each link of the mechanism. As highlighted in Chapter 1, this is not a simple task. The proposed mechanism swims at a Reynolds number of approximately 10^4 to 10^5 . Taylor and Lighthill's models can provide reasonable approximations of fluid forces at low and high Re respectively, however intermediate Reynolds numbers remain somewhat intractable. Under these conditions both added mass and drag play an important role. A detailed representation of these fluid effects can only be achieved through a full numerical solution to the Navier-Stokes equations. Unfortunately, the prohibitive computational costs of such methods are not suitable within the intended optimization approach. A solution to this issue is proposed by Jordan in [37]. A modification of his method is applied here to produce a computationally efficient selfpropelled swimming model suitable for an investigation of efficient swimming.

3.1 Lateral Fluid Forces

The primary fluid effect within the system are the lateral forces on the segments of the swimmer's body as they oscillate transversely in the fluid. Jordan's method for quantifying these forces is essentially a hybrid of the resistive and reactive approaches proposed by Taylor and Lighthill. The lateral forces are quantified as a linear combination of pressure drag and added mass effects. Viscous forces are also present in the lateral direction, however, their magnitude is negligible so they are not included in the model. Each link of the mechanism is considered as an isolated segment and end effects are ignored. The proposed expression for the lateral fluid force on each link is given as follows,

$$\mathbf{F}_{i_{fluid \perp}} = \mathbf{F}_{i_{D\perp}} + \mathbf{F}_{i_{A\perp}}$$

$$\mathbf{F}_{i_{D\perp}} = -\frac{1}{2} C_D dL \rho \mathbf{v}_{\perp} |\mathbf{v}_{\perp}|$$

$$\mathbf{F}_{i_{A\perp}} = -C_A m_{added} \mathbf{a}_{\perp}.$$
(2.18)

Subscript \perp denotes the perpendicular component of a vector relative the longitudinal link axis, while $\mathbf{F}_{i_{D\perp}}$ and $\mathbf{F}_{i_{A\perp}}$ represent the drag and added mass components of the perpendicular fluid force. The rest of the variables are defined as follows, C_D is the drag coefficient, C_A is the added mass coefficient, d is major diameter of the body, L is link length, ρ is density, m_{added} , is the theoretical added mass of the link in the lateral direction, and \mathbf{v} and \mathbf{a} are velocity and acceleration. The Reynolds number based on section diameter and average lateral velocity is approximately 10⁴ to 10⁵. For the 2:1 elliptical cylinders which make up the swimmer, the steady state values of C_D and C_A are 2 and 1 respectively. However, the drag and added mass coefficients of the system should ideally be determined experimentally as they may vary due to unsteady effects. For a submerged cylinder in a crosswise oscillatory flow, the values of the C_D and C_A are primarily dependent on two dimensionless parameters, Re and the Keulegan-Carpenter (K_C) number [57]. Based on the geometry in question these parameters are defined as follows,

$$Re = \frac{\rho U_m d}{\mu}, \quad K_C = \frac{U_m T}{d}, \quad (2.19)$$

where U_m is the peak velocity during oscillation, μ is dynamic viscosity, and T is the oscillation period. For studies involving sinusoidal motion the K_C number can be reduced to a dimensionless oscillation amplitude, where a is the oscillation amplitude, tude,

$$\delta = \frac{a}{d}.\tag{2.20}$$

There are a number of experimental studies which provide C_D and C_A over a range of Re and δ values, however the majority are intended for large structures at sea and do not cover values applicable to the swimmer. Fortunately, a single study, [58], was found which investigates the Re and δ ranges of the simulated mechanism during efficient locomotion (Re: $10^4 - 10^5$, δ : 0.5 - 2). Within the study, coefficient values are provided for elliptical cylinders of several axis ratios. For an axis ratio of 2:1, C_D is found to vary from approximately 2 to 3 and C_A remains relatively constant, varying from 0.93 to 1.

In reality, the flow around the simulated mechanism differs some what from the oscillating cylinder experiments. In addition to transverse motions, the segments are also subject to sustained axial velocities and oscillatory rotations. Considering this, we elect to simply apply the steady-state coefficients in our optimization study. Nonetheless, the empirical results presented in [58] suggest that during efficient swimming the fluid force coefficients may not deviate significantly from steady-state values.

3.2 Axial Fluid Forces

Forces in the axial direction arise through various fluid effects: viscous shear on the mechanism's surface, form drag on the body, and axial added mass. In a number of previous studies [55, 56], axial forces are completely neglected as they are deemed to be negligible with respect to the much larger transverse forces. This approach is reasonable when the mechanism is swimming with a non-optimal gait, however, when the gait is tuned for efficiency, transverse forces generate relatively small amounts drag in the direction of travel. Consequently, axial effects become a primary source of drag to balance out thrust during steady locomotion.
Due to the elongated streamlined shape of the mechanism, pressure and added mass effects are relatively small in comparison to frictional drag. Based on this, they are neglected in the formulation of the model. The frictional forces are quantified using the coefficient based approach proposed by Taylor in [31]. The friction drag expression is given as follows, where subscript || denotes the perpendicular component of a vector relative the longitudinal link axis.

$$\mathbf{F}_{i_{D\parallel}} = -\frac{1}{2} \ \rho dL \ C_f \mathbf{v}_{\parallel} |\mathbf{v}|. \tag{2.21}$$

The value of the friction coefficient (C_f) is selected from a review of published experimental data presented by Paidoussis in [36]. The Reynolds number based on the swimmer's length and forward velocity is roughly 10^5 , considering this, the axial friction coefficient is chosen as $C_f=0.03$.

3.3 Fluid Moment

Torsional drag and added mass effects $(T_{i_{fluid}})$ are also incorporated into the fluids model. Many previous studies of multi-link swimmers neglect fluid moments as they have little effect on the overall motion of the system [55, 56]. This is confirmed through our own investigations, however the fluids moments are found to contribute significantly to the required actuation torques at the mechanism's joints. Since these torques directly determine the power consumed by the system, it is critical that they be accurately represented for the efficiency optimization. Considering this, the fluid moments are incorporated into the dynamics equations using a finite segment approach. The transverse drag and added mass expressions are formulated as functions of link coordinate "x" and then integrated over each body section to determine the resulting torque. The drag moment (T_{i_D}) integral is evaluated numerically, while the added mass moment (T_{i_A}) reduces to a simple analytical form. The required vector quantities and link coordinate are defined in figure 2.5. The resulting expressions are given as follows,



Figure 2.5: Velocity components and link coordinate for torque calculation.

$$\mathbf{T}_{i_{fluid}} = \mathbf{T}_{i_{A}} + \mathbf{T}_{i_{D}}$$

$$\mathbf{T}_{i_{A}} = -C_{M} \ m_{added} \ \frac{L^{2}}{12} \ \ddot{\theta}_{i}$$

$$\mathbf{T}_{i_{D}} = -\frac{C_{D}\rho d}{2} \int_{-L/2}^{L/2} x \mathbf{u}^{i} \times |\mathbf{v}_{x\perp}^{i}| \mathbf{v}_{x\perp}^{i} dx.$$
(2.22)

4 Demonstrating the Model

With the fluids terms defined, the swimming model is now complete. An example simulation demonstrating the model's performance is provided in figure 2.6. The gait of the mechanism is prescribed using the expression given in (2.7), with T = 1s, A = 0.001 rad, $\epsilon = 7$, and $\lambda = 1m$. The swimmer accelerates from rest and then gradually approaches an equilibrium state in which drag forces balance out the thrust produced by its gait. At the beginning of the simulation the swimmer is completely straight with all joint angles set to zero. A ramp function is appended to the expression given in (2.7) to transition the mechanism to the steady state gait without introducing discontinuities. This ramp consists of a fifth order polynomial constrained to provide continuity of joint accelerations at the beginning and end of the ramp period. In the simulation shown here the ramp is set to 5 seconds, however it is lowered to 1 second for optimization trials to reduce simulation time. During optimization the fitness of



Figure 2.6: Velocity plot for swimmer accelerating from rest to a constant speed. Gait is defined with (2.7) where T = 1s, A = 0.001 rad, $\epsilon = 7$, and $\lambda = 1m$.

the swimmer's gait is evaluated once it reaches steady state. For our purposes, this is considered to occur once the velocity change from one gait cycle to the next is less than 1% of the current speed.

5 Energetics

In preparation for the efficient swimming optimization study, the energetics of the simulated mechanism are now dicussed. Power is input to the system through torques generated at the pin joints between each pair of body segments (P_{joints}). This energy is then either stored as kinetic energy ($P_{inertia}$) or dissipated to the surrounding fluid (P_{fluid}). Perfect joints are assumed so there are no losses due to internal friction or other non-ideal effects. The evolution of the various power quantities during acceleration from rest is shown in figure 2.7. Each of the three powers can be mathematically defined as follows,

$$P_{joints} = \sum_{i=1}^{n-1} -T_i \dot{\phi}_i \tag{2.23}$$

$$P_{fluid} = \sum_{i=1}^{n} (\mathbf{F}_{i_{fluid}} \cdot \mathbf{v}_i + \mathbf{T}_{i_{fluid}} \cdot \theta_i)$$
(2.24)

$$P_{inertia} = \sum_{i=1}^{n} (I\ddot{\theta}_{i}\dot{\theta}_{i} + m\mathbf{v}_{i}\cdot\dot{\mathbf{v}}_{i}).$$
(2.25)

For the optimization problem it will be necessary to quantify the efficiency of the mechanism. The efficiency of conventional aquatic vehicles is often evaluated using Froude efficiency as given by [32],

$$\eta = \frac{Thrust * Velocity}{Power Input}$$
(2.26)

However, the ultimate goal of efficiency optimization study is to generate gaits which minimize the energy required to propel the eel over a given distance. This can be directly represented as transport economy: swimming velocity over power input. Through early optimization trials it was found that Froude efficiency actually favoured gaits with power requirements significantly higher than those produced using the transport economy metric. Froude efficiency maximizes the ratio between thrust and power input rather than minimizing power consumption. In conventional aquatic vehicles, drag is independent of the propulsion system. As a result, Froude efficiency and transport economy are theoretically equivalent for optimization purposes. However, in undulatory swimmers both drag and thrust are determined by the system's gait, consequently the two efficiency metrics converge to different swimming regimes. Taking this into account, transport economy is applied as the efficiency metric within the formal optimization work.

To calculate transport economy the power consumed by the system must be determined. The theoretical power input to the eel can be simply evaluated using the joint power expression given by (2.23). A plot of the individual joint powers at the tail of the mechanism during a typical gait is provided in figure 2.8. The resulting curves reveal that multiple joints are subject to significant periods of negative work. This finding is consistent with the results of a number of previous works [59, 60]. Using (2.23) to define power consumption implies that this negative work offsets the cost of positive work. Through the elasticity of their muscles and tendons, animals and humans are often able to capture negative work and exploit it to reduce the overall cost of motion. However, attempts to mimic this process in man-made systems have, for the most part, been unsuccessful. Gains are usually negated by the additional



Figure 2.7: Power balance for swimmer accelerating from rest to a constant speed. Gait is defined with (2.7) where T = 1s, A = 0.001 rad, $\epsilon = 7$, and $\lambda = 1m$.

weight and complexity of regenerative systems [61].

Conventional hyper-redundant robots make no attempt at energy recovery; they are typically actuated by a series of angular position servos. Based on this, these periods of negative work would actually result in a significant energy cost as the mechanism must draw power to produce resistive torques. The power consumption due to negative work is largely dependent on specific actuator properties. A servo model could be employed to approximate the resulting power draw, however it is likely that future robotic swimmers will employ a different means of actuation. Thus, to quantify the cost of negative work in a more general sense, we calculate power consumption as the absolute value of the theoretical joint power. This approach has been applied in previous studies of HRM locomotion [61]. The absolute joint power is integrated over a complete swimming cycle to determine average power consumption. This results in the following expression,

$$P_{Avg} = \frac{1}{T} \int_{\tau}^{\tau+T} \sum_{i=1}^{n-1} |-T_i(t) \dot{\phi}_i(t)| dt.$$
 (2.27)

As highlighted in the introduction, the differing negative work responses of biological and artificial systems may significantly alter the performance and optimal kinematics of the simulated mechanism relative to a live eel. To investigate this issue, the op-



Figure 2.8: Plot of joint power for actuators near tail of mechanism during undulatory gait. Gait is defined with (2.7) where T = 1s, A = 0.001 rad, $\epsilon = 7$, and $\lambda = 1m$.

timization study will first be conducted using the absolute joint power expression in (2.27) and then repeated with net joint power to allow for recovery of negative work. This will provide a comparison of the optimal gaits for an HRM versus a swimmer capable of regenerative actuation.

Chapter 3

The Optimization Algorithm

Fluid systems are known to have complex optimization spaces containing numerous minima and maxima [44]. Consequently, the development of a customized approach was necessary to generate satisfactory results for the swimming optimization study. This chapter documents the development of this approach and demonstrates the efficacy of the final algorithm. The discussion begins with a formal definition of the optimization problem. Once this has been established, the development and testing of the algorithm is examined in sections that follow.

Two different optimization approaches were explored during the development process, both consisted of stochastic schemes which use the swimming model to evaluate the performance of the system at various operating points. The first method we tested, the genetic algorithm, is the standard solution for stochastic gait optimization. However, through a preliminary investigation, it was found that an alternative method known as Particle Swarm Optimization (PSO) might be better suited to our problem. Considering this, a PSO algorithm was developed and tested in addition to the GA. The following section explains the development process of both algorithms. A brief review of each algorithm's basic structure is first provided. This is then followed by an outline of the modifications required to tailor the algorithm to the HRM swimming problem. The modified algorithm is then tested through an extensive series of optimization runs. Finally, with both algorithms fully developed and tested, the relative performance of the two approaches is discussed. It is found that PSO provides the best overall performance: it consistently produces gaits of higher fitness than those of the GA, while being much simpler to implement.

1 The Efficient Swimming Optimization Problem

When formulating an optimization problem, the objective function and constraints must be properly defined. Although the final optimization study explores both efficient and fast-start maneuvers, only the efficient swimming problem is used during development. The ultimate goal of the efficiency optimization is to minimize the power required to swim at a specified velocity. This is accomplished by maximizing the following objective function which consists of the transport economy metric [61] and a penalty term to enforce the desired swimming speed,

$$F = V_f / P_{Avg} + C_v, \tag{3.1}$$

$$C_v = 1 * 10^{10} \ [Heaviside(V_d - V_f) * (V_d - V_f)^2], \tag{3.2}$$

where F is the fitness of a given gait and the C_v is the penalty term. From early trials it was observed that, when maximizing transport economy without the constraint, the optimization algorithm naturally converges to the slowest possible gait due to the quadratic nature of the drag model. C_v is used to penalize gaits with a forward velocity, V_f , below the desired swimming velocity, V_d . This causes the optimization algorithm to converge to the most efficient gait at the specified velocity limit.

For the purposes of testing the algorithm the gait parametrization given in Chapter 2, (2.7), is employed. Limits are applied to the parameters β , A, and λ to restrict the swimmer to kinematically feasible gaits. The gait period (T) is constrained to match the capabilities of current swimming HRMS. The imposed ranges are defined as follows,

$$0 \le \beta \le 10$$

$$0 \le A \le 0.5 rad$$

$$0.2 \le \lambda \le 3m$$

$$1 \le T \le 5s.$$

(3.3)

In summary, the efficiency optimization problem applied during algorithm development consists of maximizing (3.1) with respect to β , A, λ , and T subject to the constraints defined in (3.3).

2 The Genetic Algorithm

The first optimization approach tested is the genetic algorithm. First developed in the 1960's, genetic algorithms mimic the process of biological evolution to generate optimal solutions for engineering problems. To accomplish this, the process of evolution is simplified to a relatively compact algorithm consisting of a few basic steps. The basic genetic algorithm, as outlined by [62], consists of three standard components: mating pair selection, cross-over, and mutation.

2.1 Outline of Genetic Algorithm Structure

The progression of the basic GA is given as follows. First, a population of candidate solutions is randomly generated. For the problem posed here, each solution consists of a four parameter swimming gait. The solutions are then encoded as binary strings, as will be detailed in the following sections. These strings serve as "chromosomes" to represent the genetic material of each population member. With the initial population initialized and encoded, the fitness of each candidate solution is evaluated. Fitness is defined using (3.1), where the simulation is used to determine power input

 (P_{input}) and swimming velocity $(V_{forward})$. This represents the bulk of the computational cost of the algorithm as the swimming simulation must be run once for each gait. Once this is done, a mating selection process is applied to choose solution pairs. Each pair of solutions then exchanges genetic material through the crossover process to generate two child solutions. Finally, a mutation operator is applied to the new generation which then replaces the parents. This cycle of mating and mutation continues until a predetermined convergence condition is satisfied. If the algorithm is effective, the population will gradually evolve over numerous generations to produce solutions with optimal fitness. The key steps of the standard GA are outlined in the following pseudocode. A detailed description of how each step is implemented within the final genetic algorithm is provided in the following subsections.

Basic Genetic Algorithm Structure

1. Randomly generate and encode a population of N swimming gaits.

Begin Loop

- 2. Evaluate fitness of each population member
- 3. Selection of mating pairs
- 4. Mating by crossover
- 5. Mutation of children End Loop

2.2 Modifications to the Basic Genetic Operators

Selecting Mating Pairs

Once the fitness values have been determined, the first step of the optimization cycle is to select mating pairs from the population. The classic selection process is the roulette wheel method [62]. A hypothetical roulette wheel is divided into segments representing each population member. The size of a given segment is proportional to the fitness of the corresponding gait. To select a mating pair the wheel is simply spun twice. This is done until N/2 couples have been generated, where N is the total number of individuals in the population. This process favours the reproduction of the best gaits while still allowing the algorithm to explore the parameters within lower fitness solutions. The hypothetical roulette wheel for a five member population is depicted in figure 3.1 to demonstrate this process.



Figure 3.1: Visual depiction of roulette wheel selection process [63].

The standard roulette wheel selection method is poorly suited to the optimization problem posed in section 2. The constraint term, C_v , results in a large fitness discrepancy between gaits above and below the desired velocity. The fitness of solutions that do not meet the required velocity condition is several orders of magnitude lower than those that do. As a result, the few gaits which satisfy the velocity constraint during early generations constitute over 99% of the selection wheel. This imbalance results in premature convergence and poor final results. To address this issue, a ranking method [64] is applied in place of the classical approach. The solutions are sorted according to fitness and the size of each solution's roulette wheel segment is proportional to its sorted ranking rather than the actual fitness value. This approach ensures that all solutions are reasonably represented during the selection process; discouraging the domination of a small number of solutions during early generations.

In addition to the ranking method, a second modification was applied to the selection process. The basic GA outlined in [62] specifies that the selection process should be done with replacement. This means that each parent is chosen at random from the full population pool and can selected multiple times over the crossover process. This allows a single population member to serve as both parents within a given couple. When this occurs, two children identical to the parent chromosome are produced. The odds of pairing with two identical parents increases with each additional copy. As a result, this effect can build upon itself to rapidly generate a nearly uniform population. This was observed to be a major issue during trial optimization runs. Fortunately, this issue is easily resolved by enforcing unique couples. This is essentially a simplified representation of the incest taboo observed by most animal species. A number of researchers have extended this concept to create comprehensive incest-prevention schemes which track solution families and disallow pairings of closely related population members [62]. While certainly interesting, an approach of this complexity was avoided in the interest of reducing development time.

Gait Encoding and Crossover

The second step of the genetic optimization cycle is the mating/crossover process. The gait parametrization used during algorithm development consists of four parameters: amplitude (A), wavelength (λ), exponent (β), and period (T). To represent each gait as a chromosome, the four parameters are encoded as a binary string through standard decimal to binary conversion. The number of bits representing each gait

parameter was chosen to provide a reasonable balance between precision and chromosome length. Amplitude, wavelength, and period are represented using 6 bits, while the exponent parameter is represented using 7 bits as it varies over a relatively large range. This results in a total of 25 bits to represent each gait, as shown in figure 3.2.



Figure 3.2: Diagram of gait encoding applied within the genetic algorithm.

Once the mating pairs have been selected, the crossover operation is applied. Crossover consists of cutting the binary string of each parent at a random location and then swapping the resulting string segments to create two children. This operation is depicted in figure 3.3. During basic crossover only the gait parameter at the crossover point is altered through a new combination of genetic material, the rest are simply transferred directly from the parent chromosomes. Through preliminary optimization runs it was found that this encourages the survival of suboptimal parameter values from early generations. To encourage the creation of new genetic material the single chromosome string representing each gait was separated into four distinct chromosomes, one for each gait parameter. The new encoding scheme is represented in Figure 3.4. During the mating process a crossover point is randomly selected within each chromosome. This results in children which contain new combinations of the existing genetic material for each parameter rather than large sections of genetic code directly transmitted from parent to child. The modified crossover operation was observed to increase the diversity of genetic material within the population and provided better end results.

The crossover operation of the GA algorithm was also modified to include elitism.



Figure 3.3: Visual depiction of crossover process.



Figure 3.4: Visual depiction of the modified crossover process.

First proposed in [65], elitism consists of directly transferring fittest members of the population from one generation to next. The children with the lowest fitness are dropped to maintain a constant population size. The optima within the swimming optimization space are relatively peaked. Consequently, they can easily be lost after they are found due to the stochastic nature of the algorithm. The elitism operator protects the best solutions to ensure that their genetic material survives within the population over multiple generations. Within our algorithm the top 5% of the population is maintained from one generation to the next

Mutation

The final step of the optimization cycle is mutation. At this stage a small number of mutations are introduced to the child population. This allows the algorithm to generate new genetic material even after the majority of the population has converged to a given region of the optimization space, making the it robust to local optima. The mutation process is the only portion of the algorithm implemented in standard form: after the crossover process there is small chance for any bit within the child population to flip to the opposite value. Through preliminary runs, a mutation rate of 3% was found to provide good performance.

2.3 Testing the Modified Algorithm

A large battery of test runs was conducted to evaluate the effectiveness of the modified genetic algorithm in its final form. A total of 20 runs were executed. Each run had a population size of 100 individuals which was evaluated for 300 generations. The desired swimming velocity was set to 0.3m/s for the first 10 runs and 0.7m/s for the second 10. The resulting fitness values of the best overall individual in each final generation are presented in figure 3.5. These results are discussed alongside the PSO results in section 4.



Figure 3.5: Test runs for genetic algorithm. Efficiency is optimized at swimming speeds of 0.3 and 0.7 m/s.

3 Particle Swarm Optimization

3.1 Outline of Particle Swarm Algorithm Structure

Based on the findings of a preliminary investigation, PSO was explored as an alternative to GAs for our optimization problem. PSO was first proposed by Kennedy and Ebehart in 1995 [66]. The algorithm aims to mimic a group of animals cooperatively searching for a goal such as food or shelter. Within the algorithm these animals are represented as a swarm of particles which explore the parameter space of the optimization problem. For the efficient swimming study, the location of each particle within the search space is represented as vector of the gait parameters $(P_j = [\beta A \lambda T_{gait}]^T)$. The particles cooperatively explore the parameter space searching for regions of optimal fitness. The standard algorithm structure consists of the following steps,

Basic PSO Structure

1. Randomly initialize coordinates (P_j) and velocity (V_j) for N particles

Begin Loop

- 2. Find best point in history of each particle ${\cal P}_j^{best}$
- 3. Find the overall best point in history of all particles G^{best}
- Update Velocities and Positions see update equations
 End Loop

Velocity Update :
$$V_j^{New} = w * V_j + C_1 * rand() * (P_j^{best} - P_j)$$
 (3.4)
+ $C_2 * rand() * (G^{best} - P_j)$

(3.5)

 $\textbf{Position Update}: \quad P_j^{New} = P_j + V_j^{New}$

Initially, a swarm of particles is distributed randomly throughout the optimization space. The fitness of each particle's position is then determined by running the swimming simulation with the gait parameters which correspond to its coordinate vector. After the fitness of the entire swarm has been evaluated, each particle then adjusts its velocity and position based on its own fitness history and the fitness information of the rest of the swarm. The coefficients C_1 and C_2 determine the relative weighting of global and personal fitness during this process. Typically, they are both simply set to one. By repeating the fitness evaluation and position/velocity update over and over, the particles travel through the space searching for high fitness regions. If the algorithm functions properly, the swarm gradually collapses over time, eventually converging to a small optimal region. The convergence process is depicted in figure 3.6 for a swarm of 40 particles over 100 velocity/position updates.



Figure 3.6: Plots depicting the convergence process of the PSO swarm. The randomly distributed point cloud collapses to relatively small region of the optimization space.

When using the basic PSO algorithm the swarm particles sometimes get stuck in unstable oscillations. This causes them travel large distances back and forth within the optimization space during each position update. Two standard measures are usually applied to discourage this behaviour. First, the maximum allowable velocity is restricted to control the total displacement of the particles within each position update. In our case, the imposed velocity limit along each dimensions of the optimization space is set to be half of the total range of the corresponding parameter. Second, an inertia term, w, is added to the velocity update equation [67]. When w is less than one, it introduces a damping effect which dissipates energy from the system and ensures that the swarm collapses over time.

3.2 Modifications to the Basic Particle Swarm Algorithm

As with the genetic algorithm, each component of the PSO structure can be modified as needed to suit the the optimization task at hand. Fortunately, the basic PSO algorithm required relatively few changes to produce good results for the efficient swimming problem. In the end, only a single customization was needed. The standard PSO algorithm allows particles to travel freely along all dimensions of the optimization space, however, the gait parameters must be constrained as defined in (3.3). To implement this, limits are enforced on swarm particles using the damped reflection method suggested in [68]. This essentially encloses the swarm within a solid box. When a particle reaches a boundary, it effectively bounces off the box wall while losing a fraction of its initial velocity as shown in figure 3.7. This approach discourages the swarm from settling on local optima at the boundaries of the parameter space.



Figure 3.7: Diagram of damped reflection process. Particle effectively bounces off the y limit, while losing a portion of its initial velocity.

3.3 Testing the Algorithm

As in the case of the GA, 20 optimization runs were executed to evaluate the effectiveness of the PSO algorithm. Each run had a swarm size of 100 particles which was evaluated for 300 iterations. The desired velocity was set to 0.3m/s for the first 20 runs and 0.7m/s for the second 10. The inertia parameter is set to 0.9, as preliminary trials indicated that this value provided reasonable convergence while still allowing the swarm to fully explore the optimization space. Based on standard practice for PSO algorithms, the coefficients C_1 and C_2 in the velocity update equation are both set to 1. The highest fitness value out of the entire swarm during each run is presented in figure 3.8.



Figure 3.8: Test runs for Particle Swarm algorithm. Efficiency is optimized at swimming speeds of 0.3 and 0.7 m/s.

4 Comparing the Algorithms

The results produced during the testing of the GA and PSO algorithms are compared in figure 3.9. The amalgamated fitness values indicate that the PSO algorithm is superior to the GA. At high velocity the gaits produced using PSO have transport economies tightly clustered around 0.52m/J. At low velocity three PSO runs appear to have converged on a local maximum at 3.6m/J, however the rest are scattered fairly close to 4m/J. In contrast to this, the GA results exhibit a relatively high degree of variance. While the best GA gaits are roughly equivalent to the maximum fitnesses produced by the PSO algorithm, overall, the GA is significantly less consistent.



Figure 3.9: Comparison of optimization test runs of GA and PSO algorithms.

With further development it is likely the GA could eventually match the performance of the PSO algorithm. Within the genetic approach, the implementation of an appropriate encoding scheme and mating process is critical to achieving good results [62]. However, designing these elements requires an understanding of the problem which can only acquired through a time consuming iterative development process [62]. Within the intended optimization work, it will be necessary to investigate a variety of different gait parameterizations and fitness goals. Considering this, it is essential that our optimization process is robust and relatively quick to implement. Based on the development process presented here, the PSO algorithm appears to be far superior in this respect. It produces gaits with consistent fitness values, higher than those of the GA, while requiring significantly less customization. Taking this into account, it chosen for application in the optimization studies of Chapters 4 and 5. The results of these investigations support the findings suggested here. The proposed PSO algorithm produces consistent results for both the efficient and fast-start swimming optimization problems in their final form.

Chapter 4

Efficient Swimming Optimization

With the model and optimization routine fully developed, they can now be applied together to generate optimal swimming behaviours. The first behaviour explored is efficient swimming as defined in the optimization problem used for algorithm development. The full solution to the proposed optimization task is presented in the following chapter. The study begins with a discussion of the gait parametrization. The gait definition provided in Chapter 2, (2.7), represents a relatively narrow variety of body undulations. A more general representation is developed in its place to investigate a wider range of locomotory modes. Once the final parametrization is established, the formal optimization study begins. Optimization runs are conducted over a range of speeds. Each run is executed twice, first with and then without energy recovery. The resulting gaits are discussed in relation to the efficacy of the proposed optimization scheme, and the control and design of swimming HRMs. A simple but optimal strategy for velocity control and key problems for future work are developed based on the trends within the optimal gaits. In addition to this, energy recovery is found to increase efficiency by a factor of ten. This increase is observed to arise through a distinct change in swimming kinematics.

1 Revisiting the Gait Parameterization

The gait parametrization given in (2.7) is based on observations of live undulatory swimmers, however the internal dynamics and actuator properties of a robotic system are very different from the muscles of biological fish. Damping, stiffness, other losses, and negative work response differ significantly between the two systems. Considering this, it may not be correct to assume that the optimal gaits for the mechanical system can be described within the constraints of a biological gait function. Taking this into account, a more general gait definition is implemented in its place.

The generalized representation must balance complexity and compactness [61]; it should be compact enough to facilitate convergence while still allowing enough freedom to explore all relevant gait geometries. Finding this balance is an iterative process. The most general method for gait representation is to simply use the joint angles at each simulation time step as optimization variables. However, the sheer size of this parametrization scheme would severely hinder convergence and likely produce poor results. Based on this, we elect to begin with a Fourier series approach. A completely generalized Fourier parametrization would theoretically require an infinite number of series terms. However, actuator limits of both biological and man-made swimmers typically limit body movement to a relatively low frequency region. Based on this, higher order terms can be neglected. In fact, most swimming studies employ only a single sinusoid per joint [49, 50, 69]. Each joint trajectory is then fully described using only four variables: phase (α_i) , amplitude (A_i) , period (T_i) , and offset (ψ_i) . Period is assumed to be the same for all joints to prevent asynchronous body movements. This results in a total of 3n + 1 parameters to define a complete swimming gait. Constraints are imposed based on the limitations of current HRMs [16, 70]. The proposed parametrization is expressed mathematically as follows,

$$\phi_i(t) = A_i \sin(\frac{2\pi t}{T} + \alpha_i) + \psi_i \quad i = 1, 2, 3...(n-1),$$
(4.1)

$$0 \le A_i \le \pi/2 \ rad \qquad 0 \ \le \psi_i \le 2\pi$$

$$-\pi/2 \le \psi_i \le \pi/2 \ rad \qquad 1 \ \le T \le 5s.$$

$$(4.2)$$

Several optimization trials were conducted to test the Fourier parametrization. Two emergent swimming behaviours were observed in the resulting gaits: a propagating wave mode similar to anguilliform or carangiform swimmers and a flapping mode somewhat similar to the legs of a frog or a jellyfish. Figures 4.1 and 4.2 illustrate the kinematics of the two gait types produced by the algorithm. While the system successfully propels itself through the water, the gait definitions are clearly not optimal. The motions of the swimmer are highly asymmetrical in both cases. To enforce symmetry in the flapping and wave gaits, two modifications of the basic Fourier parametrization are proposed. They are defined as follows,

Propagating Wave :
$$\phi_i(t) = A_i \sin(2\pi t/T + \alpha_i)$$
 $i = 1, 2...(n-1),$ (4.3)

Flapping:
$$\phi_i(t) = A_i \sin(2\pi t/T + \alpha_i) + \psi_i \quad i = 1, 2...5.$$
 (4.4)

In the propagating wave parametrization the offset variable (ψ_i) is removed to enforce symmetry about the axial direction. For the flapping mode, joints one to five are freely defined. The motions of one to four are then mirrored onto joints seven to ten to enforce a symmetry about the center joint. A series of optimization trials is conducted using the two modified Fourier parametrizations, the resulting fitness values are presented in figure 4.3.The comparison shows that the wave gait is significantly more efficient than the flapping gait. Based on this, the flapping parametrization is dropped and sole focus is given to the propagating wave representation.

The amplitude and phase curves produced by the Fourier Wave parametrization indicate that the system is converging on simple, smooth variable profiles. Consequently, it is unnecessary to explicitly define the amplitude and phase at each joint. Based on this, five spline nodes are used in place of the ten joint parameters to define both the amplitude and phase of the gait. This reduces the dimension of the problem to



Figure 4.1: Crude propagating wave gait produced using the general Fourier representation. Displacement is exaggerated to clearly show kinematics.



Figure 4.2: Crude flapping gait produced using the general Fourier representation. Displacement is exaggerated to clearly show kinematics.

a 11 variables (5 amplitude nodes + 5 phase nodes + 1 period). Constraints remain unchanged from (4.3), the spline parametrization is thus given as follows,

$$\phi_i = A(s)\sin(\frac{2\pi t}{T} + \alpha(s)), \quad s = i \cdot l, \quad i = 1\dots(n-1).$$
 (4.5)

Test results for the new spline parametrization are provided in figure 4.3 and 4.4. Due to improved convergence, the resulting gaits generally exhibit an increase in efficiency relative to those produced by the Fourier approach. Through experimentation with the model it was found that the spline phase profiles produced by the optimization process can be replaced by a linear approximation with minimal effect on the performance of the swimmer. Based on this, the 5 phase spline nodes are replaced with a single wavelength variable which defines a constant phase shift along the body. This reduces the generalized representation to a total of only 7 variables. The new wavelength variable is restricted to 0.2m to 3m as preliminary runs indicated that the optimal value falls within this range, all other constraints remain unchanged. The modified spline parametrization is expressed as follows,



Figure 4.3: Comparison of optimal transport economies produced using the various parameterizations. Spline - AP refers to parametrization with 5 spline nodes for both amplitude and phase, Spline - A refers to parametrization with 5 spline nodes for amplitude and a single wavelength for phase.



Figure 4.4: Relative comparison of optimal transport economies produced using the various parameterizations. The Fourier - Wave results are used as a baseline.

$$\phi_i = A(s) \sin[2\pi(\frac{s}{\lambda} - \frac{t}{T})], \quad s = i \cdot l, \quad i = 1 \dots (n-1).$$

$$(4.6)$$

As shown in figure 4.4, this final iteration of the gait parametrization provides the best overall performance. The low speed efficiency is similar to the previous parameterizations, while the efficiency at higher speeds increases considerably. Considering this, this representation is applied in the formal optimization study that follows.

2 Optimization Results

Using gait parametrization given in (4.6) and the PSO algorithm, the optimization scheme is run at 10 different swimming velocities evenly spaced from 0.1 to 1m/s. Two runs are executed at each velocity: one using absolute work as power consumption and another using net work to allow energy recovery. For each run, a population of 200 particles is evaluated over 600 iterations. This requires a run time of 10 to 15 hours on a Core i7 desktop CPU. The resulting optimal gait parameters are presented in figures 4.5 to 4.10.



Figure 4.5: Optimal frequency at each swimming velocity for optimization runs with and without energy recovery.

From these figures it is apparent that the trends in the optimal gait parameters are highly consistent despite being the product of a stochastic algorithm. This suggests that the optimization scheme is able to effectively identify curves of optimal performance within the parameter space. In figure 4.5 it can be seen that the frequency of the swimming mechanism increases linearly for both the net and absolute work cases up to the maximum frequency constraint. The absolute work swimmer maintains the maximum frequency from 0.4m/s onwards while the net work swimmer does so after 0.6m/s. The velocities at which each swimmer hits the frequency limit should be noted as they mark clear transitions in the other optimal gait parameters. The



Figure 4.6: Optimal wavelength at each swimming velocity for optimization runs with and without energy recovery. Vertical lines highlight where the swimmer hits the 1Hz frequency limit.



Figure 4.7: Optimal transport economy at each swimming velocity for optimization runs with and without energy recovery.

absolute work swimmer maintains a wavelength of approximately 1.3m to 1.4m across all swimming velocities. The net work swimmer exhibits a very different wavelength of around 0.55m to 0.6m until it hits the 1Hz frequency limit at 0.6m/s. After this, it converges towards the wavelength of the absolute work swimmer.

Consistent behaviour is also apparent in the amplitude curves of the optimal gaits. Despite having to optimize 5 independent spline nodes the PSO algorithm generates a series of curves which change in a smooth predictable fashion as velocity increases



Figure 4.8: Strouhal number of optimal gait at each swimming velocity for optimization runs with and without energy recovery. Vertical lines highlight where the swimmer hits the 1Hz frequency limit.

from 0.1 to 1.0 m/s. For the absolute work swimmer all curves below 0.4 m/s are extremely similar. At higher speeds the amplitude peak at the tail begins to shift towards the center of the mechanism and undulations appear at the head. The same trends are present in the amplitude curves of the net work swimmer, however, the changes begin at the net work frequency limit point of 0.6 m/s.

Having reviewed the optimization results it is apparent that both swimmers maintain a relatively constant amplitude curve and wavelength up to the point where they hit the 1Hz frequency limit. However, these values differ significantly in the absolute and net work cases. The absolute work swimmer employs a highly carangiform gait. Undulations are restricted to a small segment at the tail, while the forebody remains relatively straight. By contrast, the net work swimmer exhibits a clearly anguilliform gait; significant undulations are present along most of the swimmer's length. These kinematic differences are illustrated by the low speed gait envelope plots provided in figure 4.11. As highlighted earlier, the gait kinematics of both swimmers begin to change once they hit the frequency limit. As the two swimmers approach 1.0m/s, their amplitude and wavelength converge resulting in similar high speed gaits. This is demonstrated by the similarity of the high speed envelopes shown in figure 4.11.



Figure 4.9: Optimal amplitude curves WITHOUT energy recovery (absolute work). The slowest (0.1m/s) and fastest (1.0m/s) gaits are omitted to reduce clutter.



Figure 4.10: Optimal amplitude curves WITH energy recovery (net work). The slowest (0.1 m/s) and fastest (1.0 m/s) gaits are omitted to reduce clutter.



Figure 4.11: Comparison of gait envelopes for low (0.3m/s) and high (0.9m/s) speed gaits produced by absolute and net work optimization. The gait envelope is composed of snapshots of the mechanisms centerline throughout a full gait cycle.

It is also worth considering the Strouhal number at which the swimmer operates. The expression used to calculate Strouhal number is given as follows [29],

$$St = \frac{fA_{tail}}{V_{forward}},\tag{4.7}$$

where f is the undulation frequency, A_{tail} is the peak to peak amplitude of the tail beat, and $V_{forward}$ is the swimming velocity. In studies of biological eels and other fish it has been found that they typically swim at a constant St between 0.2 and 0.4 during efficient locomotion [71]. Figure 4.8 shows that, below the frequency limit, the absolute work swimmer undulates at a constant Strouhal number of St = 0.33. The net work swimmer also maintains a constant Strouhal number, but appears to swim best at approximately St = 0.4. In both cases the St rises to roughly St = 0.5 as the swimmers' speeds increase beyond the frequency limit.

The constant Strouhal number of the gaits produced by the optimization process suggests that the St related effects required for efficient swimming are somehow captured within the low-order approximation of fluid forces. This is somewhat surprising as St is typically linked to wake structure and vortex shedding; neither of which are accounted for here. In a recent study by Eloy [72], Lighthill's elongated body theory is used to predict optimal Strouhal numbers for a wide variety of swimming animals. The predicted optima are found to coincide well with biological data. Lighthill's swimming theory does not allow the wake to influence the dynamics of the swimmer. Its success in predicting biological behaviour in spite of this suggests that the characteristic wake structures often linked to optimal St values may simply be the signature of efficient swimming rather than the underlying cause. Eloy highlights that this idea has previously been proposed by Muller et al. in [73]. If true, this proposition may explain the link between St and efficiency within our swimming simulation.

3 Discussion

The efficient swimming gaits generated by the optimization study exhibit a number of clear trends. These trends will now be discussed in terms of the insight they provide for the control and design of an efficient swimming HRM. The discussion will consist of three three primary topics. First, we examine the efficacy of the proposed gait optimization process and explore and how it can be adapted to physical swimming mechanisms. Once this is done, the key trends of the optimization results are used to devise optimal strategies for mechanism control and design. Finally, a comparison of the absolute and net work gaits is used to examine the role of negative work recovery in efficient undulatory swimming.

3.1 Stochastic Gait Optimization and Efficient Swimming

From the results presented in this chapter, it is clear that the PSO algorithm generates a set of highly consistent swimming gaits. It serves as an effective tool for generating optimal gait definitions for the simulated mechanism. The gait parametrization is found to be a critical aspect of the optimization scheme. The performance discrepancy between the initial and final parametrization in this chapter demonstrates the importance of selecting an appropriate gait representation. The final parametrization was produced through an iterative process beginning with a relatively general gait representation and gradually paring it down to an ideal set of gait variables. The resulting definition is capable of representing the full range of amplitude curves required by the mechanism while at the same time limiting the swimmer's phase, offset, and frequency to an ideal segment of the full optimization space.

The final parametrization consists of a sinusoidal curvature wave defined by a single period and wavelength with five spline nodes for amplitude. This is actually quite similar to the biologically based parametrization given in 2.7, the only change is the amplitude function. However, the splined amplitude definition is important to the optimization process as it allows the mechanism to explore non-exponential curves. This is critical at higher velocities where the optimal amplitude curves exhibit local minima and maxima.

The next challenge in the development of the optimization process would be to experimentally test the gaits on a swimming mechanism. Due to the simplicity of the fluids model, the optimal gaits produced by the simulation may not transfer directly to the physical system. To adjust for differences between the simulation and reality the gaits presented here could be fine tuned on the actual mechanism. Our optimization approach provides a sound parametrization for this tuning process and a first approximation of the optimal gait solution.

3.2 Control and Design for Efficient Swimming

Although the specific gait parameters would likely require tuning to be truly optimal for a given robotic swimmer, there are overarching trends within the results which can be applied to a wide range of swimming mechanisms. Controlling a highly articulated swimmer in an optimal fashion is a challenging problem. The flexibility of the system presents many possible solutions for any given task. For instance, velocity can be modulated by varying any single parameter of the gait representation. In previous work on control for robotic swimmers, the chosen velocity tracking strategies are effective but somewhat arbitrary [74, 28, 75]. The trends exhibited by the optimal gaits allow us to derive an optimal strategy for modulating velocity while highlighting issues of interest for the design of future swimming HRMs.

From the optimization results it is apparent that the amplitude curves and wavelengths remain constant at low speeds while frequency and swimming velocity are linearly related. As a result, the numerator and denominator of the Strouhal function vary proportionally. This allows the system to maintain the optimal St value purely through frequency modulation. This velocity control strategy is actually observed in natural anguilliform swimmers [76]. Applying this knowledge to the control of a general HRM we arrive at the following conclusion. Velocity control through simple frequency modulation can provide optimal swimming efficiency. This means that only a single optimal amplitude curve must be determined for efficient low speed locomotion.

At speeds beyond the frequency limit the swimmer must resort to amplitude based velocity control. The amplitude changes are relatively similar in the absolute and net work gaits. As the desired swimming velocity rises, the amplitude peak at the tail creeps forward to engage most of the swimmer's body, while posterior amplitude is reduced to control tail tip oscillation. At the highest speeds, a significant amplitude peak appears near the head of the mechanism. These head rotations serve to ensure that the head remains aligned with the direction of travel thus minimizing drag. Due to the numerous complex changes in the amplitude curves, it is difficult to reduce them to simple trends for the control of a general swimming HRM. As a result, optimally efficient high speed locomotion can only be achieved through a direct application of the gait parameters. Based on this, a look up table approach would likely be required to implement the high speed results.

Comparing the optimal behaviour above and below the frequency limit, it is clear that the optimal solution for efficient swimming is considerably simpler below it. Unfortunately, the standard servos currently employed in most swimming HRMs are poorly suited to sustained oscillatory actuation. This results in restrictive frequency capabilities. Typical HRM's have a maximum frequency of 0.5 to 2Hz, however undulations frequencies of 5Hz or more are often observed in biological swimmers. An increase in frequency capability to match biological limits would not only simplify gait design and control, but also improve high speed efficiency, and increase maximum speed. Considering this, there is significant motivation to investigate high frequency actuation technologies for future work in biomimetic swimming and design.

3.3 Negative Work and Energy Recovery

The final issue addressed in this study is the role of energy recovery in undulatory swimming. From the gait envelopes presented in figure 4.11 it is clear that the absolute and net work optimizations produce distinct swimming methods. The absolute work swimmer favours a carangiform gait while the swimmer capable of energy recovery swims best in the anguilliform mode. Thus, the optimal kinematics are found to be highly dependent on the swimmer's handling of negative work.

Below the frequency limit the transport economy of the swimmer with energy recovery is roughly ten times that of the swimmer without. These results would not apply to a physical swimmer as it could not recover 100% of the negative work. However, it does suggest an opportunity for a significant increase in efficiency if the mechanism can recover at least a fraction of this energy. To understand how the absolute and net work gaits differ, a comparison of the power dissipated to the fluid along the mechanism in each case is provided in the left plot of figure 4.12. The theoretical power consumption at each joint is given in the right plot. The shape of the resulting joint power curves closely matches the bending power analysis of live anguilliform and carangiform swimmers presented by Hess in [59]. For the swimmer optimized without energy recovery, there is no benefit to negative work, consequently the system converges to a gait which simply minimizes all energy dissipated to the fluid. This results in negligible action by forebody actuators and power consumption is concentrated in the joints near the tail. For the gait optimized with regeneration, the anguilliform properties result in large regions of negative power dissipation along the swimmer's body where it actually extracts energy from the flow. This results in negative work at the joints which the swimmer exploits to reduce the overall cost of locomotion.



Figure 4.12: Left: Average power dissipated to fluid by each body segment, for absolute and net work swimmers at 0.3m/s. Right: Average power consumed at each joint, for absolute and net work swimmers at 0.3m/s

It can be logically concluded that the energy extraction arises entirely through added mass effects. By definition, drag forces will always oppose segment velocity. To optimize the negative work on a given segment, the PSO algorithm tunes the swimmer's gait to align a complex set of conditions. The relative phase of rotation and translation for specific body segments is set to optimize periods of energy extraction while minimizing power dissipation. While attempting to do this, the swimmer must also minimize losses due to link rotation and axial viscous drag. A deeper analysis is required to understand exactly how the optimal gait arises from a balance of these competing effects.

From a hydrodynamic perspective anguilliform swimming is thought to be inefficient. Lighthill's theory predicts that undulations of the anterior portion of the swimmer's body contribute very little to forward thrust [32]. The time-averaged thrust distribution of the optimal net and absolute work gaits is provided in figure 4.13. The resulting curves show that, as predicted, the anterior undulations generate a relatively small amount of net propulsive force. Despite this finding, a number of studies have shown that anguilliform swimmers are extremely efficient in terms of a cost of transport metric [77, 78], often outpacing their carangiform counterparts. The optimal gaits



Figure 4.13: Thrust distribution along absolute and net work swimmers for optimal gaits at 0.3m/s.

produced with and without recapture of negative work suggest that anguilliform kinematics may increase efficiency by allowing the swimmer to extract energy from the flow along the anterior portion of its body. In contrast to this, the carangiform mode simply minimizes the absolute power magnitude along the swimmer's entire length.
Chapter 5

Acceleration Optimization and Swimming Control

The efficient swimming results provide a basic framework for the velocity control of the mechanism, however many of the efficient gaits exhibit extremely poor acceleration. When starting from rest the robot can require up to thirty seconds to attain its steady state velocity. By contrast, biological swimmers are noted for their excellent acceleration performance. Considering this, an optimization study is conducted to develop high acceleration maneuvers which complement the efficient swimming results.

To accomplish this, the gait parametrization of the previous chapter is modified to represent the kinematics required for rapid acceleration. Once this has been established, an objective function suitable for generating high acceleration maneuvers is defined. Finally, the proposed optimization framework is applied within the PSO algorithm to produce the desired high acceleration gaits. The gaits produced by this process appear to be well optimized, however the resulting acceleration performance is well below that of natural fast-start swimmers. Design modifications to improve performance are suggested based on the physiological features of biological fish, and the limiting factors observed in the optimization results. Despite the moderate acceleration performance of the system, it is found that the kinematics for optimal acceleration can be easily merged with those of the efficiency study. Based on this, in the final section of this chapter the high acceleration gaits are combined with those of the efficiency optimization to produce a simple but effective velocity tracking scheme.

1 Acceleration Maneuvers in Fish

Maneuvers for acceleration from rest, known as fast-starts, have been extensively studied in natural fish. The most common fast-start strategies are typically broken down into two varieties: C-starts and S-starts. C-starts consist of three distinct kinematic stages: the preparatory stroke, the propulsive stroke, and continued swimming [79]. During the initial preparatory stroke, the muscles along one entire side of the fish's body rapidly contract. This causes the animal to take on a C-like shape. Once this stage of the maneuver is complete, the fish initiates the propulsive stroke by contracting the muscles on the other side of its body. This causes it to straighten out and whip its tail backwards. In the process, the tail generates a large amount of thrust causing the fish to accelerate forward. Backbone traces of a live fish executing a C-start are shown in figure 5.1. C-starts are generally associated with escape behaviour, the final trajectory of the fish to escape predators approaching from any direction. The entire maneuver typically lasts between 30 and 100 ms with final velocities of approximately 10 body lengths/s [79].

The second maneuver type, know as an S-start, generally serves a different role in fish fast-start behaviour. While C-starts are primarily used as an escape mechanism, S-starts are typically employed for predatory strikes [79]. The basic maneuver begins with a preparatory phase in which the fish contracts its body into an S shape as shown in Figure 5.1. The fish then transitions into series of periodic tail beats. The properties of this gait can vary based on the intended goal of the fish. In [80], Harper



Figure 5.1: Backbone traces of live fish executing C-Start and S-Start[79]. Oms indicates starting position, following frames are labeled according to time (also in ms).

and Blake observe that S-starts in pike typically consist of minimum of 1 complete gait cycle with additional half cycles added as need based on the intended velocity or displacement. S-starts usually have a slightly longer timespan than C-starts, ranging from 100-200 ms, and final velocity is similar but varies considerably based on the number of tail beats within the maneuver [79].

For the acceleration of the simulated mechanism, we elect to emulate the S-start behaviour. There are two primary reasons for this decision. First, the S-start produces straight, controlled, acceleration rather than the explosive turning of the C-start. This makes it far simpler to implement for the control of a physical mechanism. Second, it can be reasonably represented by a modification of the sinusoidal gait used in the efficiency optimization. This will prove invaluable in attempting to merge the optimal efficiency and acceleration kinematics within a single control solution.

2 Setting Up the Optimization Problem

2.1 Parameterizing the Maneuver

The first step in setting up the acceleration optimization problem is to develop an appropriate parameterization of the maneuver. To do this we adopt the basic sinusoid of the previous chapter and append a ramp function to transition the mechanism from its resting position to the full acceleration gait. This allows the optimization algorithm to generate S-start like maneuvers without a major departure from the methods established in the efficient swimming work. The resulting expression is given as follows,

$$\phi_i = R(t) \cdot A(s) \sin[2\pi(\frac{s}{\lambda} - t)], \quad s = i \cdot l, \quad i = 1 \dots (n-1).$$

$$(5.1)$$

The amplitude function, A(s), is once again defined using five splined nodes limited to the following range, $(0 \le A_i \le \pi/2rad)$. The wavelength constraint also remains unchanged, $(0.2 \le \lambda \le 3m)$. During preliminary optimization it was found that the optimization algorithm naturally converges to lower period limit to maximize acceleration. Based on this, the gait period is not included as a variable of the parameterization, instead it is fixed at the lower limit of the mechanism (1s).

In all previous chapters a simple ramp function, R(t), has been applied to start the swimmer from rest. It consists a fifth order polynomial which satisfies the following conditions,

$$R(t) = At^{5} + Bt^{4} + Ct^{3} + Dt^{2} + Et + F,$$

$$R(0) = 0, \quad R(t_{ramp}) = 1,$$

$$R'(0) = 0, \quad R'(t_{ramp}) = 0,$$

$$R''(0) = 0, \quad R''(t_{ramp}) = 0.$$
(5.2)

During the process of developing the acceleration parameterization, more complex ramps based on splined curves were considered. However, preliminary optimization runs indicated that they do not provide a significant increase in mechanism performance. Based on this, the simple ramp of previous work is also applied here. The ramp time, t_{ramp} , is included as an optimization variable and is limited to a range of $0 < t_{ramp} \leq 1s$. In summary, 7 parameters in total are required to define a given acceleration gait (5 amplitude nodes, 1 wavelength, 1 ramp time).

2.2 The Objective Function and Power Constraints

To evaluate the effectiveness of a given gait, the mechanism is allowed to accelerate for a fixed length of time and the fitness is then taken as the average acceleration over the complete maneuver. As stated earlier, S-starts in pike typically consist of at least one complete swimming cycle (two tail beats) with additional half cycles added as needed. During their experiments, Harper and Blake observed that average acceleration decreases with each appended tail beat [80]. Based on this, the fixed acceleration period for optimization was initially set to a single swimming cycle (1s). During preliminary optimization runs it was found that this produced gaits in which the swimmer curled up at the end of the maneuver to maximize acceleration. Although this results in high fitness values, it is not suitable for a transition to continuous swimming. To remedy this issue, the acceleration time was extended to 1.5 gait periods (1.5s). This ensures that system converges to a gait which can smoothly transition into additional swimming cycles.

In terms of constraints, the primary factor limiting the fast-start performance of the swimmer is power. In hyper-redundant mechanisms high-torque hobby servos are the standard actuation method. Servos in this category have a maximum output on the order of 10 Watts [81]. Although future HRMs will likely employ a different means actuation, this value is used as a rough approximation of the limitations at this scale. Power consumption is calculated as the absolute value of the system's theoretical power usage and the 10W constraint is applied to each joint individually. During

optimization the constraint is implemented as a penalty term, C_P . The resulting objective function and constraint expression are given as follows,

$$f = a_{avg} + C_P, \tag{5.3}$$

$$C_P = 1 \cdot 10^{10} \left[Heaviside(P_{Max} - P_{lim})(P_{Max} - P_{lim})^2 \right]$$
(5.4)

 a_{avg} is the average acceleration of the maneuver, P_{lim} is the 10W power limit, and P_{max} is the maximum individual joint power observed during a given acceleration maneuver.

3 Results and Discussion

Having established the acceleration optimization problem and parameterization, it is now possible to combine these components with the PSO algorithm to generate high acceleration maneuvers. The following section presents the results of this process and compares the optimal performance of the system to that of natural swimmers. It is found that while the results appear to be well optimized, the swimmer's acceleration is still far below that of natural fish. The primary causes of this performance discrepancy are investigated and design changes are suggested to improve acceleration for future mechanisms. In preparation for the control design task section 4, the acceleration results are also discussed in terms of how they relate to the efficiency results of the previous Chapter. It is found that optimal efficiency and acceleration kinematics exhibit a number of similarities which will facilitate the process of combining them within a single control scheme.

3.1 Results

To produce the desired optimal acceleration maneuvers the PSO algorithm is run ten times. This makes it possible to evaluate the the consistency of the results and verify that the algorithm is able to repeatedly converge on a single optimum set of parameters. As in the previous chapter, each run evaluates a swarm size of 200 particles over 600 iterations and all algorithm parameters (w, C_1, C_2) are unchanged from Chapter 3. The properties of the resulting gaits are presented in figures 5.2 and 5.3.



Figure 5.2: Left: Ramp times for optimal acceleration gaits. Right: Wavelengths for acceleration gaits.

A fitness plot is not included here as the final fitness was extremely consistent across all runs. Without exception, the best maneuver in each of the 10 optimization trials had an average acceleration between 0.75 and $0.77m/s^2$. From figure 5.2 it is shown that the wavelength and amplitude curves are also relatively consistent between runs. The optimal wavelength ranges from 1.3 to 1.4 with a mean value of 1.32m. Interestingly, this matches the range observed for efficient swimming with absolute work in Chapter 4. The amplitude curves for optimal acceleration demonstrate a similar correspondence. They are nearly identical to those for maximum efficiency at higher swimming speeds (0.9-1.0m/s), exhibiting the same major peaks at the mid-body and the head of the robot. These similarities are discussed further in Section 3.2.



Figure 5.3: Amplitude curves for optimal acceleration gaits. Runs 6 through 10 are omitted here to reduce clutter.

In contrast to the relatively consistent values for wavelength and amplitude, the PSO algorithm has identified several different values for ramp time, ranging from 0.77s to the upper limit of 1.0s. Despite this large spread, the final average acceleration was still constant across all ten runs. This suggests that the variation is not a major issue, it appears that the ramp period can change somewhat without significantly impacting final performance. Further investigation of the results did reveal a small difference between runs with shorter versus longer ramp periods. The longer ramp times tend to cause less deviation from the horizontal course. This difference is demonstrated through backbone traces of gaits with the highest and lowest ramp times (run 3: 1.0s and run 7: 0.77s) which are presented in figure 5.4. Considering that any sort of deviation will have to be corrected through control action which may impact the system's performance, this finding makes the 1.0s ramp time the preferable choice for implementation.

If the backbone traces of figure 5.4 are compared to those of natural fish (as shown in figure 5.1), from a qualitative perspective it is apparent that the acceleration kinematics of the robotic system are quite distinct from those used by biological



Figure 5.4: Backbone trace for acceleration gaits with the longest (Run 3 - 1.0s) and shortest (Run 7 - 0.77s) ramp times.

swimmers. The robot's undulations during acceleration are significantly higher in amplitude than those of the fish. This is primarily due to 1Hz limitation. The biological S-start in figure 5.1 consists of single full gait cycle which occurs over a period of 0.1s. Based on this, the swimmer is moving at a frequency of 10Hz. Since the mechanism is restricted to a maximum undulation rate of 1Hz, it is forced to use high amplitudes to maximize acceleration rather than a high swimming frequency.

A quantitative comparison of the mechanism's performance and that of natural fish is provided in table 5.1. The performance specifications of the acceleration maneuver from Run 3 are tabulated against the average performance values recorded for pike in [47] and [80]. The pike power numbers were calculated by Frith and Blake [47] using recorded kinematics and a fluids model similar to our own. From the resulting comparison it is clear that the optimal acceleration of the mechanism is well below that of the natural swimmer. The average and peak acceleration of the mechanism are roughly 29 times less than those of the pike. In terms of absolute value, the average and peak power consumption of the pike is more than twice that of mechanism. When mass is considered this difference increases tenfold; the pike has an average specific power of approximately 130W/kg. Interestingly, the ratio of the average specific powers of the mechanism is only 4.6W/kg. Interestingly, the ratio of the average specific powers of the mechanism and the pike, approximately 28, corresponds almost exactly to the ratio of their average acceleration.

| | Mechanism | Pike |
|-----------------------------|-----------|-----------|
| Avg. Acceleration (m/s^2) | 0.76 | 22* |
| Peak Acceleration (m/s^2) | 2.8 | 100** |
| Timespan (s) | 1.5 | 0.12* |
| Peak Velocity (m/s) | 1.15 | 2.6^{*} |
| Avg. Power (W) | 18 | 53* |
| Peak Power (W) | 42 | 90* |
| Mass (kg) | 3.9 | 0.4* |

Table 5.1: Acceleration performance for gait produced in run 3, compared to the performance of a pike. Power values represent the sum power throughout all 9 joints of the mechanism or the entire body of the fish.

*Pike fast-start performance is taken from [47]

**Peak acceleration during pike fast-starts is taken from [80]

3.2 Discussion

Designing to Improve Acceleration

Based on the preceding comparison it is quite clear that the primary factor limiting the mechanism's acceleration is power. With thirty times less specific power than the natural pike, the robot cannot possibly match its performance. This highlights current actuation and energy storage technology as one of the major barriers to achieving truly biomimetic swimming behaviour. Although these issues remain a limiting factor, there are number of biological design cues which could be applied to improve the performance of the mechanism within the constraints of conventional systems. During fast-starts in natural swimmers, power consumption is concentrated at the median segments of the body. Consequently, fish have adapted a muscle mass distribution which peaks at the mid-body and tapers off rapidly towards tail [59]. Plots of the average net and absolute power along the mechanism throughout the optimal acceleration gait of run 3 are provided in figure 5.5.



Figure 5.5: Mean net and absolute power along mechanism during the optimal acceleration gait of run 3.

From the absolute power curve it is clear that the mechanism's power consumption is also concentrated in a relatively small region around its mid section. Approximately 80% of the total power consumed throughout the maneuver is generated by only 4 of the 9 servos. Considering this, the mechanism's uniform actuator layout is poorly suited the acceleration task. Servos at the head and tail expend relatively little power, while the brunt of the load is left to the those at the centre body of the robot. Mimicking the natural design of biological swimmers, a redesign of the mechanism to concentrate actuation at the mid-sections of the body could significantly improve performance.

A second issue of concern is the negative work in the posterior joints of them mechanism. The presence of this work during the acceleration maneuver is demonstrated by the large discrepancy between the absolute and net power curves in figure 5.5. Fish have evolved high stiffness caudal muscles to passively resist bending near the tail at low energy cost. In the absence of this design feature, the simulated swimmer must expend significant quantities of power actively resist bending as the tail sweeps through the water. A solution to implement a similar elastic system within the mechanism would reduce power consumption and eliminate the need for large servos in the tail, thus facilitating the previously proposed redistribution of actuation power. Despite the potential benefits of these modifications it is not immediately clear as to whether or not they should be implemented. Configurations for a given task may compromise the mechanism's performance in another. Furthermore, considerable work would be required to implement any sort of change in an optimal fashion. Mass distribution, actuator properties, and maneuver kinematics are all directly coupled to one another, resulting in an extremely complex design problem.

The optimization process developed here could serve as a powerful tool in exploring these issues. Despite the limitations of the simulated swimmer, the fast-start maneuvers produced by the algorithm are reasonably well-optimized. Across a large number of distinct algorithm runs, the gait design process has repeatedly converged on a very specific set of kinematic parameters. Considering this, the PSO algorithm appears to be reasonably robust, it is of capable generating highly efficient swimming gaits and maneuvers for rapid acceleration, with no modification to its basic structure. Based on this versatility, it is possible that additional maneuver types could be investigated, such as turning or braking. In addition to this, geometry and actuation related variables could be added the optimization problem to investigate various design modifications.

Comparing the Acceleration and Efficiency Results

The system parameters for optimal acceleration are strikingly similar to those of the high velocity gaits from the efficiency study. As stated earlier, for acceleration the optimal wavelength is approximately 1.32m, while the best wavelength for efficiency based on absolute power consumption is around 1.3-1.4m. The amplitude profiles of the two studies are compared in figure 5.6. From this plot, it is clear that the optimal acceleration profile once again exhibits the same characteristics as the high speed absolute power efficiency gaits.

During the efficiency study, optimization runs were conducted at velocity increments of 0.1m/s, the highest velocity achieved through this process was 1.0m/s. For compar-



Figure 5.6: Comparison of amplitude profiles from efficiency optimization and run 3 of the acceleration optimization.

ison, if the optimal acceleration gait is maintained over several cycles, the swimmer settles at steady-state speed of 1.03m/s. Considering this, it is likely that the steady speed of the optimal acceleration gait approaches the upper limit of the system. Consequently, if the optimal acceleration gait is appended to the efficient swimming results, a continuous spectrum of increasing velocity is produced. Within this spectrum, acceleration also increases in a continuous fashion. Velocity plots of the efficient swimming gaits are presented in figure 5.7. Acceleration performance rises monotonically as speed increases from 0.1 to 1.0m/s attaining its maximum at the 1.03m/sacceleration gait. This natural progression within the optimization results of the two studies serves as the basis of the velocity tracking scheme proposed in the following section.

4 Combining the Optimal Behaviours for Control

In this section, the high acceleration gaits are combined with those of the efficiency optimization to produce a simple but effective velocity tracking scheme. The resulting



Figure 5.7: Comparison of acceleration for optimally efficient gaits of varying velocities. Swimming speed is averaged over each undulation cycle to remove oscillation in velocity lines.

control strategy improves the poor acceleration performance of the efficient swimming gaits to produce an end result which is capable of rapid acceleration during velocity transitions and minimal energy consumption during steady-state locomotion. No rigorous testing of the scheme is presented beyond a few simple examples. Due to the complexity of system, in depth control work without physical experiments would be ill advised. The aim here, is to simply develop the basic concepts of the proposed tracking strategy. Through this process it is demonstrated that the optimization results can be implemented in a concise yet effective control scheme.

4.1 Velocity Tracking Control

The structure of the control scheme is based on the velocity tracking strategies observed in live eels. In, [82], Tytell investigates how the gait kinematics of the American eel relate to acceleration and velocity. He finds that swimming velocity is strongly correlated to body wave speed, while acceleration correlates to a deviation in tail tip velocity but is *independent* of wave speed. Assuming the gait the of eels approximates a sinusoidal wave with an exponential amplitude function, peak tail tip velocity and body wave speed are expressed in terms fundamental gait parameters as follows,

$$V_{tip} = A_{tail} \cdot 2\pi f \tag{5.5}$$

$$V_{wave} = f\lambda \tag{5.6}$$

To change V_{tip} without affecting V_{wave} , V_{tip} must either be modulated entirely through changes in amplitude (A_{tail}) , or wavelength (λ) must vary in conjunction with frequency (f). Unfortunately, noise in the amplitude and wavelength data prevents a firm conclusion as to which strategy the live eels actually use.

For the simulated swimmer, wavelength is relatively constant across all swimming speeds and optimization goals. This suggests that pure amplitude modulation is the simplest method to replicate the acceleration behaviour of natural eels, while frequency modulation should primarily serve to determine steady-state velocity. To accomplish this, the control structure presented in figure 5.8 is proposed.



Figure 5.8: Diagram of proposed control loop for speed tracking

The amplitude of the swimmer's gait is controlled though a basic feedback loop with a look-up table. The look-up table takes in a unitless set point value from the PID block and outputs a corresponding amplitude profile. The set point range is limited to between 0 and 1, where 0 corresponds to the constant amplitude profile of the low speed efficient gaits (0.1-0.4m/s) and 1 corresponds to the optimal acceleration gait. Intermediate set points are mapped to the amplitude profiles of the high speed efficient gaits (0.5-1.0m/s) to produce a continuous range. The gait frequency is determined through a feedforward path which converts the desired velocity to the appropriate frequency based on the optimization results. A rate limiter is included to prevent rapid changes in the frequency value. As a result, the feedforward path has little effect on the mechanism's short term response to changes in desired speed, instead, the amplitude controller dominates the initial response to tracking error. When a step change in desired speed occurs, the system transitions to the high acceleration amplitude profiles at the upper limit of the set point range and then gradually settles on the optimally efficient amplitude profile as frequency reaches a steady-state value. Thus, amplitude modulation is used for bursts of acceleration while frequency determines long term behaviour. In figure 5.9, the performance of the controller is demonstrated for acceleration from rest to three different final velocities (0.3, 0.6, and 0.9m/s). The PID gains are set to $K_P = 3sm^{-1}$, $K_I = 1.5m^{-1}$, and $K_D = 0.1s^2m^{-1}$. In all cases the system attains its desired velocity in roughly 1s, with rise time increasing slightly for the higher speeds.



Figure 5.9: Speed error during acceleration from rest using the proposed control loop. From left to right desired speeds are 0.3, 0.6, and 0.9m/s.

4.2 Velocity Tracking with Waypoint Navigation

In addition to starts from rest, the controller is also effective in transitioning between speeds and tracking the set velocity after high drag maneuvers such as turns. To demonstrate this, the control loop is embedded in a way-point navigation scheme taken from [83]. Within the navigation algorithm, the mechanism chases a virtual swimmer which is constrained to travel along a path laid out by a series of waypoints. The virtual swimmer's velocity is defined as follows,

$$V_{virtual} = V \frac{R^*}{R} \tag{5.7}$$

V is the actual speed of the simulated mechanism and $V_{virtual}$ is the speed of the virtual swimmer. R is the absolute distance between the mechanism and the virtual swimmer and R^* is a predefined distance value. By defining the virtual swimmer's speed in this fashion it tends to swim a distance R^* ahead of the actual mechanism. The mechanism tracks the virtual swimmer by selecting a heading to swim directly towards it at any given instant. This is accomplished using the following expression for desired heading ($\psi_{desired}$). As shown in figure 5.10, the terms R_x and R_y are the x and y components the displacement vector between the real and virtual mechanism.

$$\psi_{desired} = \arctan \frac{R_y}{R_x} \tag{5.8}$$



Figure 5.10: Diagram illustrating R_x and R_y used to calculate desired heading.

While the navigation algorithm provides a desired heading, a method to actually control the mechanism's heading has not yet been demonstrated. To accomplish this, a solution proposed by McIsaac in [20] is applied. The mechanism is given the ability to turn by adding an offset term (γ) to the sinusoidal gait expression as follows,

$$\phi_i = A(s)\sin[2\pi(\frac{s}{\lambda} - \frac{t}{T})] + \gamma, \quad s = i \cdot l, \quad i = 1\dots(n-1).$$
(5.9)

A nonzero value of (γ) causes the mechanism to assume a C-like shape while undulating, thus initiating the desired turn. The severity and direction of the turning motion is controlled by altering the magnitude and sign of the offset value. The heading controller used to define γ is given in the following equation and a snapshot of the turning procedure is given in figure 5.11.

$$\gamma = K_{\gamma}(\psi - \psi_{desired}) \tag{5.10}$$



Figure 5.11: Left: Mechanism during straight swimming. Right: Mechanism during offset turning maneuver. The mechanism is travelling in the negative x direction

Combining this turning process with the navigation scheme and velocity tracking loop produces the overall control structure illustrated in figure 5.12. In figure 5.13, this control structure is used to guide the mechanism through a series of waypoints while tracking a set velocity. The mechanism's path is shown on the left and the actual swimming velocity throughout the tracking task is given on the right. Step changes in desired speed occur at waypoints 1 and 2. The gains of the speed controller are unchanged and the heading gain, K_{γ} , is set to 0.15

The data presented in figure 5.13 demonstrates that the speed control algorithm is also effective for tracking a set velocity through turns and speed transitions. The mechanism begins by rapidly accelerating to the first velocity set point of 0.3m/s.

As it approaches the first waypoint it begins to turn, at the same time it must also respond to a step change up to a desired velocity of 0.8m/s. Using the speed control loop, it is able to do this relatively quickly. At the second waypoint the desired velocity drops to 0.6m/s. The drag of the sharp turn causes the actual velocity to dip below this value, however, once again by using the speed control loop the mechanism promptly returns to the specified setting.



Figure 5.12: Diagram of complete control scheme for velocity and path tracking.



Figure 5.13: Left: Path traced by mechanism while navigating through a series of waypoints. Right: Velocity tracking during the navigation task.

Although it has been shown that the proposed velocity tracking scheme is effective over a range of different situations, there are several issues which will have to be addressed in the development of a detailed control solution. For instance, the unsteady nature of the mechanism's velocity results in continuous small oscillations of the control signal. Further investigations are required to explore how this affects the performance of the gaits produced by the optimization process and how its impact can minimized or eliminated. In addition to this, the mechanism was observed to have relatively poor braking performance within the proposed scheme, it decelerates by assuming the lowest speed gait and gradually drifting down to the desired speed. Braking gaits or even featherable fins could be explored to resolve this problem. Finally, the implemented path following scheme is functional but in no way optimal. Further investigations could target alternative turning techniques and navigation methods. Regardless of these issues, the simple velocity tracking scheme demonstrates that the results of the optimal acceleration and efficiency studies can be combined within a single concise control solution. The resulting process provides high acceleration in response to velocity error and optimal efficiency during steady-state swimming.

Chapter 6

Conclusions and Recommendations

The work presented within this thesis has demonstrated an optimization scheme capable of producing effective swimming gaits for an n-link hyper-redundant mechanism. The scheme consists of a self-propelled swimming model applied within a custom PSO algorithm. The optimization process is used to generate gaits for two different objectives: efficiency and acceleration. The resulting gait definitions are then combined to propose a simple but effective control scheme for velocity tracking in a hyper-redundant swimmer. In each optimization case, the resulting gait kinematics and performance characteristics also provide insight on key issues related to mechanism design. For both efficiency and acceleration, negative work is found to be a significant factor in the mechanism's performance, while power and frequency limitations are identified as a major barrier to matching the behaviour of biological swimmers.

The following sections will review each major component of this optimization process: the swimming model, the optimization algorithm, and the optimization results. For each topic, key conclusions are highlighted and recommendations are then made for future work.

1 The Swimming Model

The modelling portion of the optimization process is adapted from a semi-empirical method initially proposed by Jordan in [37]. This approach is necessary to account for both drag and added mass within a single solution while maintaining relatively low computational cost. This computational efficiency makes it possible explore a broad optimization space while still producing reasonably refined swimming gaits. Jordan's method has been applied in the past for the simulation of both mechanical and biological swimmers [37, 48, 26]. However, the model has not been explicitly validated for a device of the exact geometry and dimensions simulated within the work presented here.

Considering this, experimental validation of the model is the obvious next step for further research. This would require the construction or acquirement of a physical HRM suitable for underwater operation. Although previous efforts to design HRMs have often required a considerable amount of design work and resources, these projects generally aim to produce robust final products capable of autonomous operation. To run validation experiments, a much simpler device would be adequate. For instance, the device employed by McIsaac and Ostrowski to validate their modelling work, was designed and constructed by an undergraduate researcher over the course of a summer internship [84]. In terms of the experiments, a reasonable starting point for the validation process would be constant velocity swimming. A variety of different gaits throughout the parameter space of the final gait definition (as given in Chapter 2) could be tested to evaluate the accuracy of the model in predicting steady-state behaviour.

Beyond simply reinforcing the results presented here, a firmly validated low-order model of intermediate Reynolds number swimming would represent a significant research contribution. The complexity of the encompassing fluids problem has made it impossible to effectively investigate undulatory swimmers within this regime using standard analytical or computational methods. This issue is further exacerbated by the fact that the parameter space of these systems is almost always too large for adequate study through experimental work. The development of a robust, accurate, low-order modelling solution would open this topic to the application of a variety of optimization and analytical techniques which could be used to highlight key regions interest for targeted experimental or computational study.

The issues discussed up to this point focus on the fluid-swimmer interaction, however the internal mechanics of the mechanism can also be targeted to improve fidelity. Servo models could be implemented to provide a better representation of energy consumption and system limitations, while elastic, damping, and frictional elements could added to simulate the effects of body dynamics. These features could potentially be critical to producing optimization results which are directly applicable to present HRMs, and will likely be necessary for many of the future research topics outlined the following sections.

2 The Optimization Algorithm

Based on the complexity of the swimming optimization task [44], Gradient-free methods were identified as the ideal option for the optimization component of the gait design scheme. In previous studies on similar problems, genetic algorithms (GA) are the gradient-free method of choice. Considering this, a custom GA was initially developed for the problem. To achieve decent performance several modifications were required, including elitism, rank-based selection, incest prevention, and multi-chromosome encoding. In its final form, the GA was able to produce reasonably optimized swimming gaits, however results remained somewhat inconsistent between runs. Based on this, a second gradient-free method, known as Particle Swarm Optimization (PSO), was investigated. Through testing, PSO was found to generate results superior in both absolute fitness and consistency, while requiring considerably less customization. Based on this, it is applied in the investigations of both Chapters 4 and 5. Within this work it is able to produce well-optimized gaits for both efficiency and acceleration. Although the scope of the present PSO algorithm is limited to gait kinematics, the results of optimization work suggest that the mechanism could benefit from investigations of several additional design variables (as will be discussed in section 3). This would require an extension of the optimization problem to include numerous variables related to geometry and actuator properties. This increase in design parameters will likely necessitate an improved optimization algorithm to maintain reasonable levels of convergence. In Chapter 4, the results produced with the initial 28 parameter gait representation show that the PSO algorithm clearly struggles when faced with large number of optimization variables. Fortunately there are numerous potential avenues for improving optimization performance. The PSO algorithm is a relatively simplistic optimization approach and as such it could undoubtedly be improved upon by replacing it or even combining it with more advanced techniques.

3 The Optimization Results

In the optimization study of Chapter 4, highly efficient swimming gaits are generated over a range of different velocities. To accomplish this, an appropriate gait parameterization is first developed and it is then applied with the swimming model and PSO algorithm. Through the optimization process, it is determined that the mechanism swims most efficiently by mimicking the gaits observed in natural carangiform and anguilliform swimmers. It does so by undulating at approximately the same constant Strouhal numbers, and by employing similar strategies for steady-state velocity control.

In addition to gait design, the optimization scheme is leveraged to study how energy recovery impacts optimal gait properties. It is found that energy recovery provides a significant increase in locomotion efficiency, and that this increase is accompanied by a distinct change in swimming kinematics. The swimmer with energy recovery favours an anguilliform gait while the swimmer without prefers the carangiform mode. The observed transition between swimming modes suggests that energy recovery may contribute to the extraordinary, yet poorly understood, efficiency of anguilliform swimmers. However, due to the simplicity of the simulation employed here a firm conclusion cannot be made. Further research is required to explore this issue. A complete answer will likely require a high order fluids model to confirm the energy extraction phenomenon and a full biomechanical model of the swimmer to understand how internal mechanics may allow it to exploit this fluid effect. Nonetheless, the results have identified a research avenue which could potentially provide insight on the anguilliform efficiency question and lead to significant improvements in future biomimetic swimmers.

In Chapter 5, the optimization process is used to produce high acceleration gaits which complement the Chapter 4 results. The efficient swimming parameterization is modified to represent an acceleration maneuver and then once again applied with the swimming model and PSO algorithm. The resulting gaits provide improved acceleration relative to those for optimized efficiency, however the performance of the mechanism is found to be severely restricted by its relatively low power density. The characteristics of the mechanism's power consumption suggests that acceleration performance could be improved by redistributing actuators to concentrate power in specific body regions, and by introducing elastic elements in the tail to reduce the cost of negative work. Upon comparing the acceleration gaits to those of the efficiency study, it is found that they can be naturally merged within a single control scheme. This property is used to develop simple control solution for velocity tracking which provides optimal efficiency during steady-state swimming and high acceleration during velocity transitions.

In terms of future optimization topics, additional maneuvers could certainly be investigated. For instance, in the process of developing the control algorithm both turning and braking are identified as problems which would benefit from further study. However, this task is complicated by the fact that the accuracy of the swimming model may not extend to these behaviours, and experimental validation for these situations would be quite difficult. Considering this, these topics are likely poor candidates for immediate work. Priority should be given to conducting a thorough validation of the swimming model for simple steady-state locomotion, and improving the optimization process. Once this has been accomplished, the resulting scheme can be applied to investigate the design issues highlighted within the present study. For instance, the optimization results suggest that energy recovery could significantly improve the efficiency of the simulated mechanism. A simplistic version of this is could be implemented by introducing elastic elements at each joint. However, the stiffness of each element would have to be properly tuned to ensure that it actually assists actuation rather than hindering it. In addition to this, a number of other design changes could explored to further improve performance. Possibilities include varying actuator size, cross-section, and link length, along the mechanism's body.

All of these issues are closely coupled to one another, rendering the system somewhat intractable to conventional design methods. The algorithmic approach implemented here provides a solution to this problem. If the problem statement were extended to include geometry and actuation variables in addition to gait kinematics, the optimization process could potentially output complex solutions which simultaneously optimize multiple design factors. In the past, our incomplete understanding of undulatory swimming has made it difficult to artificially replicate their locomotion strategies in an informed fashion, and the limitations of current technology make direct mimicry infeasible. A proven algorithmic design process would make it possible to generate optimal design solutions in the absence of a complete understanding of the problem at hand. The resulting design features could then be studied to reveal new insights on undulatory locomotion while highlighting key requirements for future actuation technologies. The work presented within this thesis represents a first step towards accomplishing this goal.

Bibliography

- G. S. Chirikjian and J. W. Burdick. The kinematics of hyper-redundant robot locomotion. *Robotics and Automation, IEEE Transactions on*, 11(6):781–793, 1995.
- [2] S. Hirose and H. Yamada. Snake-like robots (tutorial). Robotics & Automation Magazine, IEEE, 16(1):88–98, 2009.
- [3] M. Tesch, K. Lipkin, I. Brown, R. Hatton, A. Peck, J. Rembisz, and H. Choset. Parameterized and scripted gaits for modular snake robots. *Advanced Robotics*, 23(9):1131–1158, 2009.
- [4] Shigeo Hirose. Biologically inspired robots : snake-like locomotors and manipulators. Oxford University Press, Oxford; New York, 1993.
- [5] G.S. Chirikjian and J.W. Burdick. A modal approach to hyper-redundant manipulator kinematics. *Robotics and Automation, IEEE Transactions on*, 10(3):343
 –354, jun 1994.
- [6] G.S. Chirikjian and J.W. Burdick. The kinematics of hyper-redundant robot locomotion. *Robotics and Automation, IEEE Transactions on*, 11(6):781–793, dec 1995.
- [7] G.S. Chirikjian and J.W. Burdick. Kinematically optimal hyper-redundant manipulator configurations. *Robotics and Automation, IEEE Transactions on*, 11(6):794–806, dec 1995.

- [8] J. Ostrowski and J. Burdick. Gait kinematics for a serpentine robot. In *Robotics and Automation*, 1996. Proceedings., 1996 IEEE International Conference on, volume 2, pages 1294 –1299 vol.2, apr 1996.
- [9] Jim Ostrowski, Joel Burdick, Andrew D. Lewis, and Richard M. Murray. The mechanics of undulatory locomotion: The mixed kinematic and dynamic case. In *In Proc. IEEE Int. Conf. Robotics and Automation*, pages 1945–1951, 1995.
- [10] H. Choset, B. Mirtich, and J. Burdick. Sensor based planning for a planar rod robot: incremental construction of the planar rod-hgvg. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 4, pages 3427 –3434 vol.4, apr 1997.
- [11] D. Rollinson F. Enner and H. Choset. Simplified motion modeling for snake robots. In Proc. 2012 IEEE International Conference on Robotics and Automation, May 2012.
- [12] C. Wright, A. Buchan, B. Brown, J. Geist, M. Schwerin, D. Rollinson, M. Tesch, and H. Choset. Design and architecture of the unified modular snake robot. In *Proc. 2012 IEEE International Conference on Robotics and Automation*, May 2012.
- [13] K. Lipkin, I. Brown, A. Peck, H. Choset, J. Rembisz, P. Gianfortoni, and A. Naaktgeboren. Differentiable and piecewise differentiable gaits for snake robots. In *Intelligent Robots and Systems*, 2007. IROS 2007. IEEE/RSJ International Conference on, pages 1864 –1869, 29 2007-nov. 2 2007.
- [14] M. Tesch, K. Lipkin, I. Brown, A. Hatton, R.and Peck, J. Rembisz, and H. Choset. Parameterized and scripted gaits for modular snake robots. Advanced Robotics, 23(9), 2009.
- [15] Carnegie Mellon University Biorobotics Laboratory. Biorobotics laboratory. http://www.biorobotics.org/, 2012.

- [16] H. Yamada S. Chigisaki M. Mori K. Takita K. Ogami S. Hirose. Development of amphibious snake-like robot acm-r5. In Proc. of 36th Int. Symposium on Robotics, 2005.
- [17] S. Hirose and E. F. Fukushima. Snakes and strings: new robotics components for rescue operations. In *41st SICE Annual Conference*, volume 1, pages 338–343, August 2002.
- [18] Kenneth A. McIsaac and James P. Ostrowski. Geometric approach to anguilliform locomotion: modelling of an underwater eel robot. Proceedings - IEEE International Conference on Robotics and Automation, 4:2843–2848, 1999.
- [19] K. A. McIsaac and J. P. Ostrowski. A geometric approach to gait generation for eel-like locomotion. In *Intelligent Robots and Systems, 2000. (IROS 2000). Pro*ceedings. 2000 IEEE/RSJ International Conference on, volume 3, pages 2230– 2235 vol.3, 2000.
- [20] K. A. McIsaac and J. P. Ostrowski. Motion planning for anguilliform locomotion. Robotics and Automation, IEEE Transactions on, 19(4):637–652, 2003.
- [21] Alessandro Crespi, Andr Badertscher, Andr Guignard, and Auke Jan Ijspeert. Amphibot i: an amphibious snake-like robot. *Robotics and Autonomous Systems*, 50(4):163–175, 2005.
- [22] A. Crespi and A.J. Ijspeert. Online optimization of swimming and crawling in an amphibious snake robot. *Robotics, IEEE Transactions on*, 24(1):75–87, feb. 2008.
- [23] EPFL Biorobotics Laboratory. Biorobotics laboratory. http://biorob.epfl.ch/, 2012.
- [24] F. Boyer, L. Damien, and P. Wenger. The eel-like robot. In ASME Design Engineering Technical Conferences, San Diego, 2009.
- [25] W. Khalil, G. Gallot, O. Ibrahim, and F. Boyer. Dynamic modeling of a 3-d serial eel-like robot. In Proceedings of the 2005 IEEE International Conference

on Robotics and Automation, pages 1270–1275, 2005.

- [26] G. Gallot, O. Ibrahim, and W. Khalil. Dynamic modeling and simulation of a 3-D hybrid structure eel-like robot. In *Robotics and Automation*, 2007 IEEE International Conference on, pages 1486–1491, 2007.
- [27] F. Boyer, M. Porez, A. Leroyer, and M. Visonneau. Fast dynamics of an eel-like robot Comparisons with navier-stokes simulations. *Robotics, IEEE Transactions* on, 24(6):1274–1288, 2008.
- [28] M. Alamir, M. El Rafei, G. Hafidi, N. Marchand, M. Porez, and F. Boyer. Feedback design for 3d movement of an eel-like robot. In *Robotics and Automation*, 2007 IEEE International Conference on, pages 256–261, 2007.
- [29] D. S. Barrett, M. S. Triantafyllou, D. K. P. Yue, M. A. Grosenbaugh, and M. J. Wolfgang. Drag reduction in fish-like locomotion. *Journal of Fluid Mechanics*, 392:183–212, 1999.
- [30] J. Gray. Studies in animal locomotion: I. the movement of fish with special reference to the eel. J Exp Biol, 10(1):88–104, 1933.
- [31] G. Taylor. Analysis of the swimming of long and narrow animals. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 214(1117):pp. 158–183, 1952.
- [32] M. J. Lighthill. Note on swimming of slender fish. Journal of Fluid Mechanics, 9(Part 2):305–317, 1960.
- [33] M J Lighthill. Hydromechanics of aquatic animal propulsion. Annual Review of Fluid Mechanics, 1(1):413–446, 1969.
- [34] M. J. Lighthill. Aquatic animal propulsion of high hydromechanical efficiency. Journal of Fluid Mechanics, 44(02):265–301, 1970.
- [35] M. J. Lighthill. Large-amplitude elongated-body theory of fish locomotion. Proceedings of the Royal Society of London. Series B. Biological Sciences, 179(1055):125–138, 1971.

- [36] M. P. Paidoussis, editor. 8 Solitary cylindrical structures in axial flow, volume Volume 2, pages 787–1032. Academic Press, 2003.
- [37] C. E. Jordan. Coupling internal and external mechanics to predict swimming behavior: A general approach. American Zoologist, 36(6):710–722, 1996.
- [38] Th Y. T. Wu. Swimming of a waving plate. Journal of Fluid Mechanics, 10:321– 344, 1961.
- [39] Th. Yao-Tsu Wu. Hydromechanics of swimming of fishes and cetaceans. volume 11 of Advances in Applied Mechanics, pages 1 – 63. Elsevier, 1971.
- [40] Jian-Yu Cheng, Li-Xian Zhuang, and Bing-Gang Tong. Analysis of swimming three-dimensional waving plates. *Journal of Fluid Mechanics*, 232:341–355, 1991.
 Cited By (since 1996): 63 Export Date: 30 November 2010 Source: Scopus.
- [41] J. Carling, T. L. Williams, and G. Bowtell. Self-propelled anguilliform swimming: Simultaneous solution of the two-dimensional navier-stokes equations and newton's laws of motion. *Journal of Experimental Biology*, 201(23):3143–3166, 1998.
- [42] H. Liu and K. Kawachi. A numerical study of undulatory swimming. Journal of Computational Physics, 155(2):223–247, 1999. ISI Document Delivery No.: 254AX Times Cited: 56 Cited Reference Count: 29 Liu, H Kawachi, K Academic press inc San diego.
- [43] R. Cortez, L. Fauci, N. Cowen, and R. Dillon. Simulation of swimming organisms: coupling internal mechanics with external fluid dynamics. *Computing in Science & Engineering*, 6(3):38–45, 2004.
- [44] S. Kern and P. Koumoutsakos. Simulations of optimized anguilliform swimming. Journal of Experimental Biology, 209(24):4841–4857, 2006.
- [45] I. Borazjani and F. Sotiropoulos. On the role of form and kinematics on the hydrodynamics of self-propelled body/caudal fin swimming. J Exp Biol, 213(1):89– 107, 2010.

- [46] E. D. Tytell, C. Hsu, T. Williams, A. Cohen, and L. Fauci. Interactions between internal forces, body stiffness, and fluid environment in a neuromechanical model of lamprey swimming. *Proceedings of the National Academy of Sciences*, 107(46):19832–19837, 2010.
- [47] H. Frith and R. Blake. The mechanical power output and hydromechanical efficiency of northern pike (esox lucius) fast-starts. *Journal of Experimental Biology*, 198(9):1863–73, 1995.
- [48] Chris E. Jordan. A model of rapid-start swimming at intermediate reynolds number: undulatory locomotion in the chaetoknath sagitta elegans. *Journal of Experimental Biology*, 163(1):119–137, 1992.
- [49] B. Morgan. Stochastic Optimization for Learning Swimming Robot Gaits. PhD thesis, Washington University, St. Louis, 2009.
- [50] P. D Kuo and D. Grierson. Genetic algorithm optimization of escape and normal swimming gaits for a hydrodynamical model of carangiform locomotion. In *Genetic and Evolutionary Computing Conference Late Breaking Papers*, pages 170–177, July 2003.
- [51] D Barrett, M Grosenbaugh, and M Triantafyllou. The optimal control of a flexible hull robotic undersea vehicle propelled by an oscillating foil. *Proceedings* of Symposium on Autonomous Underwater Vehicle Technology, pages 1–9, 1996.
- [52] M. Sato, M. Fukaya, and T. Iwasaki. Serpentine locomotion with robotic snakes. Control Systems Magazine, IEEE, 22(1):64–81, 2002.
- [53] M. Sfakiotakis, D.M. Lane, and J.B.C. Davies. Review of fish swimming modes for aquatic locomotion. Oceanic Engineering, IEEE Journal of, 24(2):237 –252, April 1999.
- [54] C.C. Lindsey. 1 form, function, and locomotory habits in fish. In W.S. Hoar and D.J. Randall, editors, *Locomotion*, volume 7 of *Fish Physiology*, pages 1 – 100. Academic Press, 1979.

- [55] K. A. McIsaac and J. P. Ostrowski. Motion planning for anguilliform locomotion. *IEEE Transactions on Robotics and Automation*, 19(4):637–652, 2003.
- [56] O. Ekeberg. A combined neuronal and mechanical model of fish swimming. Biological Cybernetics, 69(Copyright 1994, IEE):363-74, 1993.
- [57] S. Van Wassenbergh, B. Flammang J. Strother, L. Ferry-Graham, and P. Aerts. Extremely fast prey capture in pipefish is powered by elastic recoil. *Journal of The Royal Society Interface*, 5(20):285–296, 2008.
- [58] N.Yu Guskova, G. Makhortykh, and M. Shcheglova. Inertia and drag of elliptic cylinders oscillating in a fluid. *Fluid Dynamics*, 33:91–95, 1998. 10.1007/BF02698165.
- [59] F. Hess. Bending moments and muscle power in swimming fish. In Eighth Australasian Fluid Mechanics Conference, volume 2, 1983.
- [60] F. Hess and J. J. Videler. Fast continuous swimming of saithe (pollachius virens): a dynamic analysis of bending moments and muscle power. J Exp Biol, 109(1):229–251, 1984.
- [61] K. Dowling. Limbless Locomotion: Learning to Crawl with a Snake Robot. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 1997.
- [62] Melanie Mitchell. An Introduction to Genetic Algorithms. MIT Press, Cambridge, MA, USA, 1998.
- [63] John Dalton. Genetic algorithms. http://www.edc.ncl.ac.uk/highlight/rhjanuary2007.php, 2007.
- [64] Baker J. E. Adaptive selection methods for genetic algorithms. In Proceeding of the First International Conference on Genetic Algorithms and Their Applications, pages 101–111, 1985.
- [65] K. A. De Jong. An Analysis of the Behavior of a Class of Genetic Adaptive Systems. PhD thesis, University of Michigan, Ann Arbor, 1975.

- [66] J. N. Kennedy and R. C. Eberhart. Particle swarm optimization. 1995.
- [67] Y. Shi and R. Eberhart. A modified particle swarm optimizer. The 1998 IEEE International Conference on Evolutionary Computation Proceedings, pages 69– 73, 1998.
- [68] X. Shenheng and Y. Rahmat-Samii. Boundary conditions in particle swarm optimization revisited. Antennas and Propagation, IEEE Transactions on, 55(3):760 -765, March 2007.
- [69] J. Tan, Y. Gu, G. Turk, and C. K. Liu. Articulated swimming creatures. ACM Trans. Graph., 30:58:1–58:12, August 2011.
- [70] J. Cortes, S. Martinez, J. P. Ostrowski, and K. A. McIsaac. Optimal gaits for dynamic robotic locomotion. *The International Journal of Robotics Research*, 20(9):707–728, 2001.
- [71] G. K. Taylor, R. L. Nudds, and A. L. R. Thomas. Flying and swimming animals cruise at a Strouhal number tuned for high power efficiency. *Nature*, 425(6959):707–711, 2003.
- [72] C. Eloy. Optimal Strouhal number for swimming animals. ArXiv e-prints, February 2011.
- [73] U Muller, B Heuvel, E Stamhuis, and J Videler. Fish foot prints: morphology and energetics of the wake behind a continuously swimming mullet (chelon labrosus risso). Journal of Experimental Biology, 200(22):2893–906, 1997.
- [74] J. Yu and L. Wang. Design framework and motion control for biomimetic robot fish. In Intelligent Control, 2005. Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation, pages 1435 -1440, june 2005.
- [75] D. Korkmaz, G. Ozmen Koca, and Z. H. Akpolat. Robust forward speed control of a robotic fish. In *Mechatronics*, 2011. Proceedings of the 6th International Advanced Technologies Symposium, May 2011.

- [76] Tytell E. D. and Lauder G. V. The hydrodynamics of eel swimming. Journal of Experimental Biology, 207:1825–1841, March 2004.
- [77] V. van Ginneken, E. Antonissen, U. K. Mller, R. Booms, E. Eding, J. Verreth, and G. van den Thillart. Eel migration to the Sargasso: remarkably high swimming efficiency and low energy costs. *Journal of Experimental Biology*, 208:1329–1335, april 2005.
- [78] W. W. Schultz and P. W. Webb. Power requirements of swimming: Do new methods resolve old questions? *Integrative and Comparative Biology*, 42(5):1018– 1025, 2002.
- [79] P. Domenici and R. Blake. The kinematics and performance of fish fast-start swimming. *Journal of Experimental Biology*, 200(8):1165–78, 1997.
- [80] D. G. Harper and R. W. Blake. Prey capture and the fast-start performance of northern pike esox lucius. *Journal of Experimental Biology*, 155(1):175–192, 1991.
- [81] Hobbico Inc. Futaba servos. http://www.futaba-rc.com/servos/digital.html, 2012.
- [82] E. D. Tytell. Kinematics and hydrodynamics of linear acceleration in eels, anguilla rostrata. Proceedings of the Royal Society of London. Series B: Biological Sciences, 271(1557):2535-2540, 2004.
- [83] E. D. B. Medagoda and P. W. Gibbens. Synthetic-Waypoint Guidance Algorithm for Following a Desired Flight Trajectory. *Journal of Guidance, Control, and Dynamics*, 33(2):601–606, March 2010.
- [84] Tamara Knutsen. Designing an underwater eel-like robot and developing anguilliform locomotion control. Technical report, University of Pennsylvania, 2000.