Optimized Hydrodynamics of Bio-Inspired Locomotive Swimmers

by

Ahmed Abouhussein

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2022 by the
Graduate Supervisory Committee:

Yulia Peet, Chair
Ronald Adrian
Jeonglae Kim
Mohamed Kasbaoui
Hans Mittelmann

ARIZONA STATE UNIVERSITY

December 2022

ABSTRACT

This work aims to address the design optimization of bio-inspired locomotive devices in collective swimming by developing a computational methodology which combines surrogate-based optimization with high fidelity fluid-structure interactions (FSI) simulations of thunniform swimmers. Three main phases highlight the contribution and novelty of the current work. The first phase includes the development and bench-marking of a constrained surrogate-based optimization algorithm which is appropriate to the current design problem. Additionally, new FSI techniques, such as a volume-conservation scheme, has been developed to enhance the accuracy and speed of the simulations. The second phase involves an investigation of the optimized hydrodynamics of a solitary accelerating self-propelled thunniform swimmer during start-up. The third phase extends the analysis to include the optimized hydrodynamics of accelerating swimmers in phalanx schools. Future work includes extending the analysis to the optimized hydrodynamics of steady-state and accelerating swimmers in a diamond-shaped school.

The results of the first phase indicate that the proposed optimization algorithm maintains a competitive performance when compared to other gradient-based and gradient-free methods, in dealing with expensive simulations-based black-box optimization problems with constraints. In addition, the proposed optimization algorithm is capable of insuring strictly feasible candidates during the optimization procedure, which is a desirable property in applied engineering problems where design variables must remain feasible for simulations or experiments not to fail. The results of the second phase indicate that the optimized kinematic gait of a solitary accelerating swimmer generates the reverse Karman vortex street associated with high propulsive efficiency. Moreover, the efficiency of sub-optimum modes, in solitary swimming, is

found to increase with both the tail amplitude and the effective flapping length of the swimmer, and a new scaling law is proposed to capture these trends. Results of the third phase indicate that the optimal midline kinematics in accelerating phalanx schools resemble those of accelerating solitary swimmers. The optimal separation distance in a phalanx school is shown to be around $2L$ (where L is the swimmer's total length). Furthermore, separation distance is shown to have a stronger effect, *ceteris paribus*, on the propulsion efficiency of a school when compared to phase synchronization.

DEDICATION

This work is dedicated to my late grandmother, Thanaa Mahgoub, who passed away earlier this year. May God shower you with His mercy, Nana.

ACKNOWLEDGMENTS

Praise be to the most high, Allah.

I would like to thank my thesis supervisor, Professor Yulia T. Peet, for her continued support and guidance during my time at the Integrative Simulations & Computational Fluids Lab. I thank the members of my thesis committee: Professor Ronald Adrian, Professor Hans Mittleman, Professor Mohamed Houssem Kasbaoui and Professor Jeonglae Kim for providing valuable insight through the course of my doctoral studies. I am grateful for the assistance of my lab mates: YiQin Xu, Daniel Coxe, Nusrat Islam and Fengrui Zhang. Additionally, I would like to acknowledge the National Science Foundation (NSF) for their generous support through the NSF CMMI grant # 1762827 grant.

My work would have not been possible without the endless encouragement from my family and friends. To my parents, Hesham and Mayada, I will always be indebted to you. To my siblings, Omar and Shahd, thank you for never failing to be there for me. To my grandfather, Abdulfattah, thank you for all that you have taught me. To my adoring wife, Karoline Joyce Davis, words cannot do justice to how much I cherish your love and kindness.

CONTENTS

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Chapter 1

INTRODUCTION

Biomimetic propulsion has triggered the interests of researchers for several decades, since it not only relates to the processes of evolution and natural selection in aquatic species, but also provides an inspiration for engineering and design of agile underwater devices (Triantafyllou, Triantafyllou, and Yue 2000; Fish and Lauder 2006; Coyle et al. 2018). In this aspect, identification of optimum, in some pre-defined sense, modes of locomotion is especially intriguing, since it gives a clear pathline towards design of bio-inspired vehicles that aim to provide a corresponding optimality (Paley and Wereley 2021). While parametric studies of aquatic biolocomotion can give some important insights towards understanding of swimming hydrodynamics (Bergmann, Iollo, and Mittal 2014; Zhu et al. 2002), in this work I am interested in developing formal optimization techniques that can automatically select and refine the candidates for optimality, given certain design criteria (Nocedal and Wright 2006). Specifically, the goal of this work is to develop efficient and accurate numerical framework for optimization of bio-inspired propulsion that utilizes fully-resolved computational fluid dynamics (CFD) simulations. This computational framework can then be used to investigate the optimized hydrodynamics of bio-inspired solitary and group underwater swimming devices.

There have been several studies devoted to optimization of aquatic swimmers in the previous literature (Kern and Koumoutsakos 2006; Gazzola, Van Rees, and Koumoutsakos 2012; Van Rees, Gazzola, and Koumoutsakos 2013; Eloy 2013; Tokić and Yue 2012; Maertens, Gao, and Triantafyllou 2017). All of them, with the

exception of Maertens, Gao, and Triantafyllou 2017, used the so-called evolutionary algorithms (EA), which rely on a nature-inspired method to select and "evolve" potentially favorable solution candidates towards optimality (Back 1996). Individuals, or solution candidates, in a population evolve using a variety of biologically inspired processes such as mutation, crossover and recombination (Hansen and Ostermeier 2001). However, most evolutionary, or genetic, algorithms demand a large number of function evaluations to arrive at a reasonable solution. For example, the EA used in Kern and Koumoutsakos 2006 required roughly 500 function evaluations to reach an optimum, and the same algorithm reached 8000 evaluations in Gazzola, Van Rees, and Koumoutsakos 2012. To cope with such substantial computational demands, the recent trend in bio-inspired aquatic optimization was to reduce the fidelity of the simulation method, with studies such as Eloy 2013; Tokić and Yue 2012 departing from CFD and employing simple hydrodynamic models based on a potential flow solution and empirical estimates (Michael J Lighthill 1960; Michael James Lighthill 1971; Webb 1975) to provide evaluation of hydrodynamic forces.

While reduced-order models based on potential theories and empirical correlations are valuable for providing basic scaling laws and dominant trends in biolocomotion (Saadat et al. 2017), their applicability to a quantitative evaluation of hydrodynamic parameters, especially during the optimization procedure, is questionable for several reasons: 1) They typically underestimate the total work and overestimate the speed, thrust, useful work and efficiency (Schultz and Webb 2002; Triantafyllou, Triantafyllou, and Grosenbaugh 1993); 2) They can not reliably predict the drag of an undulating swimmer (Barrett et al. 1999; Schultz and Webb 2002; Fish and Lauder 2006); 3) Due to a lack of experimental data on sub-optimum swimming, it is unclear to which extent they are applicable to the modes of locomotion, other

than the optimal; 4) They are largely based on steady-state swimming theories and data (Michael J Lighthill 1960; Schultz and Webb 2002; Gazzola, Argentina, and Mahadevan 2014), and might not be extendable to other hydrodynamic regimes, such as, e.g., acceleration (Akanyeti et al. 2017; Wise, Schwalbe, and Tytell 2018). For the above-mentioned reasons, I am interested in developing an optimization framework based on high-fidelity fully-resolved simulations (FRS), in order to put forward a robust and reliable methodology, which could be used through a variety of hydrodynamic regimes, and not just for a steady swimming, and could provide a predictive capability for the purposes of comparison, refinement and further development of reduced-order models. Furthermore, I am interested in applying the developed computational framework to uncover interesting trends in the optimal hydrodynamic behaviour of bio-inspired solitary and group swimmers. To this end, I divide this work into three main phases, discussed in Chapters 1, 2 and 3, respectively, which combine to address the current topics of interest. Details on the future work, which include investigations of optimal swimming in diamond-shaped fish schools, are discussed in Chapter 4. Finally, conclusions of the presented work are summarized in Chapter 5.

1.1 Phase 1

Phase 1 is primarily concerned with identifying an appropriate optimization algorithm to be used in the developed computational framework. To do so, however, one must first narrow down the search to a class of global optimization algorithms suitable to the current design problem. Global optimization accounts for the majority of problems in practical and engineering optimization. In terms of global optimization, gradient-based approaches can have faster convergence rates when compared to

gradient-free approaches. However, they are more likely to remain trapped at a local minimum. Additionally, a wide variety of gradient-based algorithms become inefficient when dealing with many real-world applications in the fields of science, medicine, and engineering design. The inefficiency arises when computing the objective function derivative becomes expensive or sometimes even infeasible. Derivative-free approaches such as evolutionary algorithms (EAs) or genetic algorithms (GAs) offer a lucrative alternative considering their ability in not requiring any assumption on the objective function landscape or its derivatives. As previously discussed, however, a high number of function evaluations required by such algorithms can be prohibitive. Attempts at alleviating the mentioned concerns gave rise to the field of a surrogate-based optimization (SBO) (Sobester, Forrester, and Keane 2008). This branch of optimization techniques makes use of a surrogate model built with the help of true function evaluations, when finding local or global optima. Function calls to the surrogate model are much cheaper than the true function, thereby cutting down the computational cost of optimization. Indeed, the presented benchmark study shows that SBO algorithms are particularly desirable when the optimization function is an expensive "black-box". I propose a new constrained variant of a SBO algorithm and present a second benchmark study to highlight it's relevance to the current design problem.

1.2 Phase 2

Phase 2 is focused on developing the physical and numerical modeling behind the CFD simulations and, subsequently, coupling those simulations to the SBO algorithm. The complete framework is then applied to optimize the kinematic gait of

an accelerating solitary swimmer during start-up. Phase 2 includes new developments that significantly improve efficiency of fluid-structure interaction (FSI) algorithms on body-fitted grids in incompressible flow. In particular, I introduce a new volume conservation approach for a geometry reconstruction of the undulating fish body with a widely used kinematic deformation model (Michael J Lighthill 1960; Barrett et al. 1999), which avoids, otherwise necessary for stability purposes, implicit iterations between the velocity and pressure solvers (Xu and Y. T. Peet 2017; Xu and Y. Peet 2021), and speeds up the calculations by three to five times. Another important aspect of the work highlighted in Phase 2 is establishing a grid convergence of the optimization procedure with a polynomial refinement of the CFD solution, and investigating a sensitivity of the optimization results to a numerical resolution of the FSI simulations, an important issue, which, to the authors' knowledge, seems to have been overlooked in the previous optimization studies (Kern and Koumoutsakos 2006; Gazzola, Van Rees, and Koumoutsakos 2012; Van Rees, Gazzola, and Koumoutsakos 2013; Maertens, Gao, and Triantafyllou 2017), at least in the context of bio-inspired propulsion. Additionally, I present a physical analysis of the optimum and sub-optimum modes of locomotion obtained with a high polynomial order of $N = 9$, and propose a new scaling law for the propulsive efficiency versus the kinematic gait parameters, applicable to both optimum and sub-optimum propulsion.

1.3 Phase 3

Phase 3 extends the modelling and analysis of optimized solitary swimming to optimized swimming in infinite phalanx-shaped schools. Phalanx schools have been investigated through different methods which include self-propelled rigid foils (Raspa,

Godoy-Diana, and Thiria 2013), self-propelled flexible foils (Godoy-Diana et al. 2019), numerical simulations of undulating swimmers (Hemelrijk et al. 2015; Gazzola et al. 2011), and fish tank experiments (Ashraf et al. 2016; Intesaaf Ashraf et al. 2017). Recent investigations suggest that the phalanx group swimming efficiency is sensitive to the separation distance between swimmers in the school (Hemelrijk et al. 2015; G. Li et al. 2019; Oza, Ristroph, and Shelley 2019), with varying conclusions reached in regards to the most beneficial swimming regime with respect to separation distance. For example, Hemelrijk et al. 2015 investigation of phalanx schools of mullet-shaped fish in steady flow revealed that the propulsive efficiency of the group is higher than a solitary swimmer as long as the separation distance within the school remains greater than or equal to one fish length. When the separation distance is smaller than one fish length, the authors noted that the group's swimming efficiency decreases below that of a steady solitary swimmer (Hemelrijk et al. 2015). Conversely, G. Li et al. 2019 notes that phalanx formations of red nose tetra fish have a high average cost of transport when the separation distance is close to one fish length. Instead, from their simulations, a separation distance of roughly half a fish length could provide a 2% improvement in the group's average cost of transport when compared to a solitary swimmer, assuming steady flow (G. Li et al. 2019). Another study, using an inviscid model, showed that tightly packed phalanx schools of rigid airfoils, with roughly half a chord length of separation distance, show a 5% improvement in the group steady swimming speed, when compared to a single airfoil (Oza, Ristroph, and Shelley 2019). This improvement is accompanied, however, by a 4% increase in the average cost of transport over that of a single airfoil. The observed speed-up and a higher cost of transport both decrease as the separation distance between the airfoils is increased until the system eventually approaches the single airfoil limit (Oza, Ristroph, and

Shelley 2019). The preferred separation distance in a school can depend on a range of different factors such as the swimming speed and/or Reynolds number (Intesaaf Ashraf et al. 2017), and the phase difference between swimmers (L. Li et al. 2020). It remains to be the case, however, that the group propulsive efficiency of fish schools in general, and for phalanx schools in particular, is directly related to the separation distance of swimmers within it.

Phase synchronization was likewise found to be important for phalanx schools, where, e.g., red nose tetra fish showed a preference towards either inphase or an antiphase swimming over all other synchronization modes (Ashraf et al. 2016). Several computational studies of a swimming fish pair demonstrated an increased stability of an antiphase swimming pattern (G. Li et al. 2019; Gazzola et al. 2011), while the trajectories diverged during an inphase swimming (Gazzola et al. 2011).

A different, but nevertheless important, schooling parameter to consider is the swimmers' gait (Maertens, Gao, and Triantafyllou 2017). Previous investigations of phalanx schools using undulating flexible fish-shape bodies rely on a fixed kinematic gait based on empirical data (Hemelrijk et al. 2015; G. Li et al. 2019; L. Li et al. 2020), which is often obtained by observing the midline envelope of real fish in steady swimming (Videler and Hess 1984; Ashraf et al. 2016). However, an optimized kinematic adjustment could result in improved swimming performance through schooling mechanisms such as wake capturing (Verma, Novati, and Koumoutsakos 2018). In fact, for swimmers to only maintain their relative position in a school, kinematic adjustment is required (G. Li et al. 2019). Additionally, kinematic gaits based on steady state swimming may not necessarily apply to unsteady swimming. Indeed, accelerating swimmers have distinctively different hydrodynamic and kinematic behaviour than steady swimmers. For example, a large survey of over 50 species of real fish revealed

that accelerating fish consistently swim with a higher tail amplitude when compared to steady swimming (Akanyeti et al. 2017). Similar studies report that eels and bluegill sunfish increase both their tail amplitude and undulation frequency during acceleration when compared to steady swimming (Tytell 2004; Wise, Schwalbe, and Tytell 2018). Additionally, investigations of the force field on the body of a robotic tuna-shaped swimmer revealed that during acceleration thunniform swimmers (such as tuna) generate a significant portion of forward thrust using their main body and may, consequently, adopt a more anguilliform-like body motion (Thandiackal et al. 2021). While linear acceleration may have been investigated for single swimmers, little is known about the collective behaviour of accelerating phalanx schools and the associated changes in kinematic motion. Accelerating collective motions play an important role in many biological functions of fish schools, such as an escape from predator, and also for the design of efficient maneuvers for autonomous underwater vehicles.

I investigate optimal propulsive performance in infinite accelerating phalanx schools in light of critical parameters such as separation distance, gait kinematics and phase synchronization. This is achieved by using the developed computational methodology to allow for a coupled fish array hydrodynamics to be fully optimized in terms of their midline kinematics, phase difference and undulation frequency. Three phalanx schools, with varying separation distances, are presented to highlight changes in optimal behaviour depending on the compactness of a school. The phalanx setups are also compared to an optimized solitary swimmer to highlight the differences between the school and single fish performance.

Chapter 2

IDENTIFYING AN OPTIMIZATION ALGORITHM

Identifying the correct algorithm is of paramount importance to maintaining an efficient and accurate optimization-based design study. I begin by introducing the general formulation of the optimization problem considered which can be expressed as follows,

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in \mathbb{R}^n \end{aligned} \tag{2.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is the function, $\mathbf{x}$ is a vector of design parameters and $\mathbf{x} \in S \cap C$. The set $S \subseteq \mathbb{R}^n$ contains the n-dimensional search space, which would define a rectangle in $\mathbb{R}^2$ or a rectangular cuboid in $\mathbb{R}^3$:

$$l(i) \leq x_i \leq u(i), \quad 1 \leq i \leq n \tag{2.2}$$

where $l(i)$ and $u(i)$ represent lower and upper bounds, respectively, on a design parameter in the ith dimension. The set $C \subseteq \mathbb{R}^n$ is the region defined by the set of $m \geq 0$ constraints:

$$\begin{aligned} &g_k(\mathbf{x}) \leq 0, \quad k = 1, ..., q, \\ &h_k(\mathbf{x}) = 0, \quad k = q+1, ..., m \end{aligned} \tag{2.3}$$

where $g_k(\mathbf{x})$ and $h_k(\mathbf{x})$ are referred to as the inequality and equality constraint sets, respectively, on the design parameter vector, $\mathbf{x}$. The intersection of the sets S and C defines the feasible domain, $D = S \cap C$.

Identifying an appropriate algorithm was performed by, first, choosing a suitable class of optimization algorithms and, second, designing a specialized algorithm suited

to the computational demands as well as the physical constraints involved in the numerical simulations of underwater swimming devices. Therefor, this chapter is broken into two general sections, where the first section is focused on the wide range of available optimization methods and the second is focused on the specifics of the final optimization algorithm incorporated into the current computational methodology.

2.1 Choosing a Suitable Class of Optimization Algorithms

The following definitions will be of use in the discussion of relevant categories of optimization algorithms. The best solution to an optimization problem, $\mathbf{x}^*$, is referred to as a *global minimizer*, namely:

Definition 2.1.1 (global minimizer)**.** $\mathbf{x}^*$ is a global minimizer if $f(\mathbf{x}^*) \leq f(\mathbf{x})$ for all $\mathbf{x} \in D$

A different solution could be found that is a minimizer only in the local vicinity of the objective function landscape. Although still a solution, a local solution may not necessarily be a global minimizer. This type of solution is referred to as a *local minimizer*, namely:

Definition 2.1.2 (local minimizer)**.** $\mathbf{x}^*$ is a local minimizer if there exists a neighborhood $\mathcal{N} \subseteq D$, such as $\mathbf{x}^* \in \mathcal{N}$ and $f(\mathbf{x}^*) \leq f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{N}$

Using the above definitions, optimization algorithms can be categories into two broad categories: gradient-free and gradient-based algorithms. As the name might imply, gradient-free algorithms seek to find an optimal solution without relying on the gradient of an objective function. This capability can be especially helpful when no gradient information is readily available or when a user is interested in identifying

a global minimizer. Gradient-based algorithms, on the other hand, rely on gradient information to identify an optimum, which means the solution is often a local optima. Consequently, this class of methods is especially popular when only a local minimizer is sufficient. The next two sections discuss gradient-based and gradient-free algorithms, respectively.

2.1.1 Gradient-based Algorithms

Gradient-based algorithms are a class of algorithms which relies on gradient information of the objective function to guide the search direction towards an optimum. Several gradient-based methods have been developed and implementations are widely available through python (SciPy), C++ (NLPOT), MATLAB (fmincon) and Dakota (COLIN & CONMIN) packages. Since MATLAB's fmincon (Toolbox et al. 1993) and Dakota's COLIN & CONMIN (Adams et al. 2020) packages include a wide variety of gradient-based methods, the discussion is limited to gradient-based algorithms commonly found in those packages. The general form of a gradient-based solver is:

1. Begin iteration k with computing a search direction $\mathbf{p}_k$ which defines the search for a lower objective function in the D search space.
2. Compute an appropriate step length α_k to be explored in direction $\mathbf{p}_k$
3. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k\alpha_k$ and $k = k + 1$
4. Evaluate $f(\mathbf{x}_k)$ and return to 1 if necessary

When information on the gradient is available, first and second order optimality conditions can be invoked to identify a local minimizer. More so, they are used in constructing meaningful estimates of α_k and $\mathbf{p}_k$. Theorems of the first and second

order optimally conditions are given below, and the reader can refer to any standard optimization textbook, such as Nocedal and Wright 2006 for complete proofs.

Theorem 2.1.1 (First-Order Necessary Condition). For a local minimizer $\mathbf{x}^*$, if $f(\mathbf{x})$ is $\mathcal{C}^1$ differentiable, then $\nabla f(\mathbf{x}^*) = 0$

Since this condition arises out of a first-order taylor series expansion of $f(\mathbf{x})$ around minizer $\mathbf{x}^*$, it is referred to as the *First-Order Necessary Condition*. Similarly one can derive the the *Second-Order Necessary Condition* by a second order taylor series expansion:

Theorem 2.1.2 (Second-Order Necessary Condition). For a local minimizer $\mathbf{x}^*$, if $\nabla^2 f(\mathbf{x})$ is $\mathcal{C}^0$ continuous in the neighborhood $\mathcal{N}$ of $\mathbf{x}^*$, then $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ is positive semidefinite.

Matrix M is said to be positive semidefinite if $\mathbf{x}^T M \mathbf{x} \geq 0$ and positive definite if $\mathbf{x}^T M \mathbf{x} > 0$ for all non-zero $\mathbf{x}$. The *Second-Order Sufficient Condition* infers the existence of a local minizmer by incorporating elements of the *First-Order Necessary Condition* and *Second-Order Necessary Condition*, namely:

Theorem 2.1.3 (Second-Order Sufficient Condition). If $\nabla^2 f(\mathbf{x}^*)$ is positive semidefinite, $\nabla^2 f(\mathbf{x})$ is $\mathcal{C}^0$ continuous in the neighborhood $\mathcal{N}$ of $\mathbf{x}^*$ and $\nabla f(\mathbf{x}^*) = 0$, then $\mathbf{x}^*$ is a local minizer.

The above theorems apply to *any* unconstrained optimization problem without loss of generality. Parallel theorems exist for constrained problems, which recast the objective function into a Lagrange function with the constraints being penalized by the Lagrange multiplier λ_i for each i constraint. The following five remarks are made to highlight the importance of optimally conditions in constrained optimization:

Remark 1. An equality constraint, $h_k(\mathbf{x}^A)$ is considered active for a solution candidate $\mathbf{x}^A$ if $h_k(\mathbf{x}^A) = 0$. While the inequality constraint, $g_k(\mathbf{x}^A)$ is considered active for a solution candidate $\mathbf{x}^A$ if $g_k(\mathbf{x}^A) \geq 0$

Remark 2. If the gradient of the set of active constraints is linearly independent for a solution candidate $\mathbf{x}^A$, then the linear independence constraint qualification (LICQ) holds at $\mathbf{x}^A$.

Remark 3. If $\mathbf{x}^*$ is a local minimzer to the constrained optimization problem and $f(\mathbf{x})$, $g(\mathbf{x})$, $h(\mathbf{x})$ are $\mathcal{C}^1$ differentiable and LICQ holds at $\mathbf{x}^*$, there exits a Lagrange multiplier λ_i^* vector which satisfies the *first-order necessary condition*. The set ($\mathbf{x}^*$, λ_i^*) is considered a solution to the Karush-Kuhn-Tucker (KKT) equations

Remark 4. The *second-order necessary condition* postulates that if the set ($\mathbf{x}^*$, λ_i^*) is a solution to the KKT equations, then the Hessian of the Lagrange function, with respect to ($\mathbf{x}$), is positive semidefinite.

Remark 5. The *second-order sufficient condition* postulates that if the set ($\mathbf{x}^*$, λ_i^*) is a solution to the KKT equations, and the Hessian of the Lagrange function, with respect to ($\mathbf{x}$), is positive definite, then $\mathbf{x}^*$ is a local minizer .

The KKT equations form the basis on which constrained nonlinear optimization problems are solved. After casting the optimization problem and relevant constraints into the Lagrangian form, several search methods could be used to predict α_k and $\mathbf{p}_k$. More information on the LICQ conditions and the KKT equations can be found in Nocedal and Wright 2006.

2.1.1.1 Modified Method of Feasible Directions

The Modified Method of Feasible Directions (MMFD) is an extension of the older conjugate gradient method to allow it to handle constrained optimization. The conjugate gradient method relies on a simple property, namely *conjugacy*, to generate a set of conjugate search directions which drive the optimization sequence. If we consider the unconstrained linear optimization problem

$$\text{minimize} \quad f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} \tag{2.4}$$

where A is a $n \times n$ symmetric positive definite matrix and the residual $\mathbf{r}_k$ of an $\mathbf{x}_k$ is defined as:

$$\mathbf{r}_k = A\mathbf{x}_k + \mathbf{b} \tag{2.5}$$

We generate the a set of conjugate gradient search directions $\{\mathbf{p}_0, \mathbf{p}_1, ..., \mathbf{p}_l\}$ by the following recursive formula:

$$\mathbf{p}_k = -\mathbf{r}_k + \beta_k \mathbf{p}_{k-1}, \tag{2.6}$$

where

$$\beta_k = \frac{\mathbf{r}_k^T A\mathbf{p}_{k-1}}{\mathbf{p}_{k-1}^T A\mathbf{p}_{k-1}}. \tag{2.7}$$

We initialize p_0 to the steepest descent direction, the direction where the gradient points to the minimum. The step length α_k is defined is given by

$$\alpha_k = \frac{-\mathbf{r}_k^T \mathbf{p}_k}{\mathbf{p}_k^T A\mathbf{p}_k}. \tag{2.8}$$

In this linear system, the conjugate gradient method converges to the solution x_* in n or less steps (the reader is refered to Nocedal and Wright 2006 for the full proof). While the MMFD methods follows similar steps for unconstrainted optimization, the algorithm will tweak the objective function in the presence of constraints. In other

words, MMFD finds a feasible search direction which aims to minimize the objective function and the constraints, respectively, by imposing the following requirements:

$$\nabla f(\mathbf{x}^k) \cdot \mathbf{S}^k \leq 0 \tag{2.9}$$

$$\nabla c_i(\mathbf{x}^k) \cdot \mathbf{S}^k \leq 0 \quad i = 1, ..., z \tag{2.10}$$

where $\mathbf{S}_k$ is the search vector at iteration k, $\nabla f(\mathbf{x}^k)$ is the analytical gradient of the objective function, $\nabla c_i(\mathbf{x}^k)$ is the analytical gradient of the ith active constraint and z is the total number of active constraints. A similar implementation of the presented MMFD technique is found in the Dakota MMFD solver.

2.1.1.2 Interior Point Method

The interior point method uses a logarithmic barrier function to transforms the original optimization problem, Eq. 2.1, into the following:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) - \mu \sum_{i=1}^{q} \ln(s_i) \\ \text{subject to} \quad & h(\mathbf{x}) = 0 \\ & g(\mathbf{x}) + s = 0 \end{aligned} \tag{2.11}$$

where $\mu > 0$ is the barrier parameter, s_i is the positive slack variables of the ith inequality constraint. One can note that all the inequalities have been replaced by the slack variables, s_i, which are now part of the KKT solution. This creates a sequence of equality constrained problems which are significantly easier to solve than inequality constrained problem. A line search using a conjugate gradient method or Newtons method can be used to identify the next iterate. This method has the added benefit of rejecting infeasible candidates by incorporating a trust region method in

the optimization process. Depending on the nature of the objective function, this could be an added benefit if the infeasible constraints could not be tolerated while it would unnecessarily increase function evaluations otherwise. An implementation of the interior point method is found in MATLAB's fmincon.

2.1.1.3 Sequential Quadratic Programming and Active Set

The Sequential Quadratic Programmming (SQP) and Active Set (AS) methods of MATLAB's fmincon fall under the same umbrella of nonlinear optimization algorithms. Consider an equality constrained problem where the constraint set C, from Eq. 2.3, contains only $h_k(\mathbf{x})$. Using a newton step to solve the KKT equations of the general nonlinear problem, we arrive at the following system of n (size of $\mathbf{x}$) + m (size of equality constraints):

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}_k & -A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} p_k \\ p_\lambda \end{bmatrix} = \begin{bmatrix} -\nabla f_k + A_k^T \lambda_k \\ -h_k \end{bmatrix} \tag{2.12}$$

where A_k is the Jacobian matrix of constraints $h_k(\mathbf{x})$ and p_λ and p_k are the steps used in the Newton iterations for the variables $\mathbf{x}$ and Lagrangian multiplier λ_k. Since the constraints are penalized by the always positive Lagrange multipler, it only suffices for the SQP approach to solve any constraint (be it equality or inequality) by casting it into an equality constraint. AS on the other hand takes the entire set to be the active set of constraints (equality and inequality), which is likely to be the reason for its MATLAB designation, and solves the inequality constrained problem (IQP). AS assigns $\lambda_k = 0$ to constraints outside the active set. Since the IQP is much more expensive to solve, this method does not scale well with the number of variables. However this AS method can insure strict feasibility of the solution candidates.

2.1.1.4 Limitations of Gradient-based Algorithms

Gradient-based methods remain to be formidable tools in the domain of linear and nonlinear constrained optimization, with especially lucrative aspects for design optimization. For example, the conjugate gradient search step has a fast convergence speed, promising a solution of the linear system $Ax = b$, where A is $n \times n$ symmetric matrix, in at most n steps. Moreover, certain algorithms such as the interior point methods are capable of insuring strictly feasible solution candidates during the design optimization process. Nevertheless, several drawbacks arise when the optimization function is a "black-box". Gradient-based methods invariably rely on the gradients of both the objective function and the constraint function when dealing with linear problems. The extension to nonlinear problems involves an additional requirement, stipulated by the KKT equations, of obtaining the Hessian of the Lagrangian function of the optimization problem. While numerical approximations could be provided (hereby increasing the function evaluation count and introducing numerical errors), the methods are still only "locally" accurate. Indeed, the "hill-climber" methods, as they are called, are often prone to converging to local minimzers. While remedies were introduced to make local gradient methods more "global" (György and Kocsis 2011), the attention often shifts away to gradient-free methods when considering "black-box" design optimization. As will be shown, these class of methods are specifically designed to suit the needs of simulation-based optimization studies, where no stipulations on the objective functions are required.

2.1.2 Gradient-free Algorithms

As previously stated, the main appeal of gradient-free algorithms lies in their ability to identify a solution when no *a priori* information is available. Instead of an analytical description of the objective function, this class of methods relies on a measure of fitness, used to asses the competitiveness of a solution candidate, as the optimization goal. For example, the fitness of a candidate may be evaluated through an expensive numerical simulation. In such cases, the numerical simulation acts as a "black-box" function which receives a certain input and produces a corresponding output. Gradient-free algorithms typically share a few common features (Stork, Eiben, and Bartz-Beielstein 2020). For example, gradient-free algorithms can take inspiration from natural evolutionary processes such as mutation and recombination as candidates evolve towards an optimum. Additionally, candidate search can involve stochastic processes such as sampling from a probability distribution. Perhaps the most common feature is the inclusion of control parameters. Since gradient-free algorithms are meant to apply to a wide range of optimization problems, control parameters can be used to "tweak" the algorithms accordingly. I limit the discussion of gradient-free algorithms to evolutionary and surrogate-based methods, as these two approaches are the most relevant to the study of design optimization through computer simulations (Booker et al. 1999; Kern and Koumoutsakos 2006). The reader is referred to Stork, Eiben, and Bartz-Beielstein 2020 for a detailed discussion on other gradient-free algorithms.

2.1.2.1 Evolutionary Algorithms

Evolutionary algorithms (EAs), sometimes referred to as genetic algorithms (GAs), are a branch of bio-inspired gradient-free optimization algorithms which mimic natural processes such as crossover, mutation and reproduction of individuals within a population. Individuals are ranked based on their fitness and the best parents are chosen to breed (crossover) and then mutate according to a fixed probability (Michalewicz, Hinterding, and Michalewicz 1997). The children are then entered into the population and the individuals are ranked based on fitness again, and the least fit solutions are discarded as the population "evolves". Taking μ_k as individuals within a population at iteration k, τ_k as parents chosen from a population at iteration k, λ_k as offsprings at iteration k, a generic EA can be described as shown in Table 1.

Table 1: A Generic Evolutionary Algorithm

EA pseudocode
1. Initialize population of μ_k individuals.
2. Evaluate fitness of μ_k individuals.
for $k = 0, 1, \ldots,$ **do**:
3. Choose best individuals as parents for next generation τ_k.
4. Apply variation operator to generate λ_k offspring from τ_k parents.
5. Evaluate fitness of λ_k.
6. Select λ_k to replace previous individuals to create generation μ_{k+1}.
7. If convergence is achieved, **break** and report best individual(s).
end

Variation operators, such as mutation, crossover and selection, can vary between implementations of EAs and GAs.

2.1.2.2 Surrogate Based Optimization

A surrogate based optimization (SBO) algorithm is a non-gradient based global search method which seeks to identify an optima by building surrogate approximation of the objective function, thereby requiring no derivative information or approximation. An initial surrogate of the objective function is created by sampling objective function values according to a space filling strategy. An infill criteria is then used to assess fitness of inexpensive surrogate function evaluations to determine one or more points to be evaluated using the true objective function. Those points are used to refine the surrogate and the process is repeated till an optima is identified. SBO encompasses a wide range of design choices during each phase: the initial construction plan of the surrogate, the infill criteria which can incorporate any derivatrive-free or derivative-based optimzier and the surrogate construction technique. For this reason it is referred to as a "meta-modeling" approach (Stork, Eiben, and Bartz-Beielstein 2020). Let M_k and $s_k(\mathbf{x})$ be defined as the set of candidates used for "true" function evaluations and the surrogate model, respectively, at iteration k. Further, define the maximum number of allowable iterations, k_{max}. A generic surrogate-based optimization algorithm can then be described as shown in Table 2.

2.1.3 Gradient-free Versus Gradient-based Study

Despite the fact that a choice of an optimization algorithm is an important decision in engineering practice, there are remarkably few documented studies that compare different (gradient-based and gradient-free) optimization algorithms with respect to an analytical or a black-box function. Rios and Sahinidis 2013 performed a comparison

Table 2: A Generic Surrogate-Based Optimization Algorithm

SBO pseudocode

1. Construct an initial candidate set, T, preferably using a space filling strategy (Tenne 2015; Morris and Mitchell 1995) and set $M_0 = T$
2. Evaluate the fitness of all initial candidates, $f(T)$.
3. Construct the initial surrogate model, $s_0(\mathbf{x})$, using T and $f(T)$.

for $k = 0, 1, \ldots, k_{max}$ **do**:

4. Generate a candidate set, X_k, according to an infill criteria (Sobester, Forrester, and Keane 2008).
5. Evaluate $s_k(X_k)$.
6. Identify next promising point, $\mathbf{x}_k$, by applying the infill criteria.
7. Evaluate $f(\mathbf{x}_k)$.
8. Identify the current best point, $\mathbf{x}_k^*$. If convergence is achieved, **break** and report best individual(s).
9. Set $M_{k+1} = M_k \cup \mathbf{x}_k$. Re-fit s_{k+1} with M_{k+1}.

end

of gradient-free solvers including SBO and EAs, on a large set of test problems with available solutions. Their work however was limited to gradient-free solvers with bound constraints. They did not show a comparison of gradient-based approaches to gradient-free (SBO and EAs) solvers, nor did the authors consider changes in optimization performance related to more complex nonlinear constraints on design variables. Other authors considered different variants of the problem such as evaluating a computationally cheap objective function with computationally expensive black-box constraints with gradient-free solvers (Müller and Woodbury 2017) or gradient-free optimization in tuberculosis treatment (Cicchese et al. 2017) and sensor placement (Szalay and Nagy 2014). I aim to fill this gap by analyzing the performance of 5 gradient-based and 9 gradient-free algorithms with respect to two test problems: 1) unconstrained and constrained optimization of the analytical Rosenbrock function. Details of the 5 gradient-based and 9 gradient-free algorithms are can be found in

Table 3. More information on the specific implementation of each algorithm can be found in the "Comments" column of Table 3, with the exception of the GA, SBL, SBG, and EGO algorithms which are natively supported through the MATLAB (Toolbox et al. 1993) and DAKOTA (Adams et al. 2020) software packages.

Table 3: Optimization Algorithms included in the gradient-free versus gradient-based benchmark

	Software Package	Solver	Comments
Gradient-based	MATLAB	Interior Point (IP)	FMINCON
	MATLAB	Sequential Quadratic Programming (SQP)	FMINCON
	MATLAB	Active Set (AS)	FMINCON
	DAKOTA	Conjugate Gradient (FRCG)	CONMIN
	DAKOTA	Method of Feasible Directions (MFD)	CONMIN
Derivative-free	MATLAB	Genetic Algorithm (GA)	
	MATLAB	Covariance Matrix Adaptation Evolution Strategy (CMAE-ES)	Hansen 2016
	DAKOTA	Evolutionary Algorithm (EA)	COLINY Hart et al. 2010
	DAKOTA	Single Objective Genetic Algorithm (SOGA)	JEGA
	DAKOTA	Multiple Objective Genetic Algorithm (MOGA)	JEGA
	DAKOTA	Surrogate based local optimizer (SBL)	
	DAKOTA	Surrogate based global optimizer (SBG)	
	DAKOTA	Efficient global optimization (EGO)	

Details on the optimization problem, computational set-up and results are discussed in the following chapters, respectively.

2.1.3.1 Optimization Benchmark Function

The Rosenbrock function was selected as the canonical analytical objective function of the optimization problem (De Jong 1975). The Rosenbrock function is a non-convex function known for containing a global minimum within a wide basin. The two-dimensional form of the function is described as follows:

$$f(x_1, x_2) = (a - x_1)^2 + b(x_2 - x_1^2)^2 \tag{2.13}$$

with global minimum located at $(x_1, x_2) = (a, a^2)$, where $f(x_1, x_2) = 0$. The parameters were chosen to be $a = 0.35$ and $b = 100$. Two sets of constraints were considered.

1. Bound constraints, i.e. $S \cap C = S$ (referred to as unconstrained):

$$-0.2 \leq x_i \leq 0.5, \quad 1 \leq i \leq 2 \tag{2.14}$$

2. A constraint set, C, closely related to the natural constraints encountered in the black-box optimization study involving a solitary swimmer:

$$\begin{aligned} g_1(\mathbf{x}) &= x_2 + 2.5x_1^2 - 0.5 \leq 0 \\ g_2(\mathbf{x}) &= -x_2 - x_1 + 0.4 \leq 0 \end{aligned} \tag{2.15}$$

The specific combination of the function parameters and the constraint set insures that the global minimum resides within the constrained domain, as shown in Figure 2.1.3.1.

2.1.3.2 Computational Set-up

Numerical simulations were performed on 3.20GHz Intel processors on a Linux environment. The MATLAB and Dakota solvers were run on MATLAB 2018b

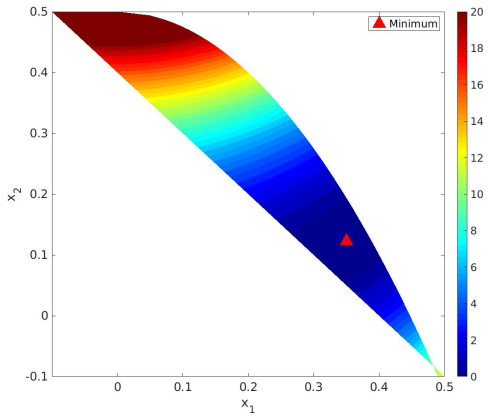


Figure 1: 2-D Rosenbrock over the constrained domain

and Dakota 6.8 respectively. All solvers were started from the feasible point of $(x_1, x_2)_0 = (0.2, 0.3)$. The convergence criteria was set to 1×10^{-4} on the absolute error whenever possible, (an exception is SBG, because it can only be terminated by specifying a maximum iteration count). The function evaluations count as well as the absolute error, achieved after convergence, are reported in Table 4 and Table 3 5. A margin of error (MOE) is calculated with 95% confidence for appropriate solvers as follows:

$$\begin{aligned} SD &= \sqrt{\frac{\sum_{i=1}^{m}(z_i - \mu_z)^2}{m-1}} \\ MOE &= \frac{SD}{\sqrt{m}} * 1.96 \end{aligned} \tag{2.16}$$

where z_i is a random variable at iteration i, μ_z is the sample mean, m is the number of realizations taken to be 1000, SD is the standard deviation and MOE is the margin of error. The exceptions are the deterministic gradient-based solvers and the SBG solver. The population size of all Dakota EAs/GAs were chosen to be 20, while the others did not provide an option for population limit, within a generation. Similarly, the crossover rate and mutation rate were set to be 50% for all the three Dakota GA/EA solvers, while those rates were not specified for other algorithms. The SBO algorithms

used 20 function evaluations to construct an initial surrogate of the function. SBG and SBL used a derivative-free algorithm (COLINY_EA) when minimizing. The Dakota SOGA & MOGA solvers were unable to provide solutions within tolerance for many of the runs, which is reflected by a poor solution average. In that case, a stricter convergence criteria had little effect on improving solution quality.

2.1.3.3 Results

Table 4: Benchmark Results for Unconstrained Optimization

Algorithm	Function Evaluation	Function Evaluation MOE	Absolute Error	Absolute Error MOE
FMINCON-IP	22	N/A	3.92×10^{-5}	N/A
FMINCON-SQP	50	N/A	7.90×10^{-5}	N/A
FMINCON-AS	43	N/A	4.00×10^{-5}	N/A
CONMIN_FRCG	37	N/A	4.62×10^{-7}	N/A
MATLAB_GA	7302	541	7.91×10^{-5}	2.18×10^{-6}
CMA-ES	181	4	4.19×10^{-5}	1.78×10^{-6}
COLINY_EA	1031	10	1.07×10^{-4}	7.03×10^{-6}
SOGA	864	18	3.11×10^{-3}	2.62×10^{-4}
MOGA	4518	227	8.21×10^{-4}	7.65×10^{-5}
SBG	25	N/A	5.89×10^{-5}	N/A
SBL	204	1	1.72×10^{-5}	1.09×10^{-6}
EGO	23	0	2.72×10^{-5}	4.08×10^{-6}

The gradient decent solvers displayed superior performance by having not only low function evaluations but also the shortest wall clock time, with around $1s$ on average compared to an average of $2.5s$ and $15s$ for GAs/EAs and SBO respectively. Solver performance across different packages (MATLAB's fmincon and Dakota's CONMIN)

Table 5: Benchmark Results for Constrained Optimization

Algorithm	Function Evaluation	Function Evaluation MOE	Absolute Error	Absolute Error MOE
FMINCON-IP	24	N/A	6.95×10^{-5}	N/A
FMINCON-SQP	26	N/A	3.59×10^{-5}	N/A
FMINCON-AS	22	N/A	2.12×10^{-5}	N/A
CONMIN_MFD	25	N/A	1.54×10^{-6}	N/A
MATLAB_GA	2772	42	4.80×10^{-6}	9.70×10^{-7}
CMA-ES	104	3	4.78×10^{-5}	1.88×10^{-6}
COLINY_EA	965	14	7.11×10^{-5}	4.85×10^{-6}
SOGA	1034	24	2.48×10^{-3}	3.96×10^{-4}
MOGA	2346	44	3.34×10^{-3}	6.62×10^{-3}
SBG	22	N/A	3.71×10^{-5}	N/A
SBL	203	1	2.00×10^{-5}	1.24×10^{-6}
EGO	23	0	2.13×10^{-5}	4.36×10^{-6}

did not show notable differences, albeit featuring different algorithms. The solvers converged to a local optima which is the global optimum for this test problem.

While most GA algorithms converged at fewer iterations when the optimization problem was altered from unconstrained to constrained, the Dakota SOGA and MOGA solvers were an exception. This is likely due to the GAs dependence on the variation operators relative to the objective function landscape, which remains an active area of research in genetic programming (Laumanns, Zitzler, and Thiele 2001). In terms of unconstrained optimization, the best performing GA/EA, CMA-ES, showed faster convergence than the worst performing SBO algorithm, SBL, with an absolute error that is double in magnitude. However, the CMA-ES algorithm performance is unpredictable near boundaries and should not be used when a constrained optimization problem's minimzer is expected to be on or relatively close to a boundary (Hansen 2016). Additionally, the best performing SBO algorithm, EGO, required roughly 13%

of the function evaluations needed for CMA-ES and roughly 0.3% of the function evaluations needed for the worst performing GA algorithm, MATLAB's GA. In terms of constrained optimization, the best performing GA, CMA-ES, required only 50% of iterations needed for the worst performing SBO, SBL, with an error that is double in magnitude. However SBG required roughly 21% of the iterations needed by CMA-ES, with a comparable absolute error.

For both constrained and unconstrained optimization problems, SBO algorithms showed performance similar to gradient descent algorithms in terms of not only function evaluations but also solution accuracy. Moreover the function evaluation MOE reported is consistently lower in the case of SBO algorithms relative to GA algorithms, suggesting that convergence "stochasticity" is much lower in SBO. The absolute error MOE was roughly one order of magnitude lower than the absolute error for the algorithms considered. This suggests that solution "stochasticity" has remained relatively consistent between all derivative-free algorithms, with the exception of MOGA applied to the unconstrained problem. Consequently, the results of this study seem to suggest that the a suitable algorithm may be found within the family of SBO algorithms. However, applied engineering problems often involve one or more constraints which can not be violated. In other words, design variables, in certain situations, must remain feasible for simulations or experiments not to fail. Therefor, attention must be paid to an optimization algorithm's ability to handle constraints. A more detailed discussion on this topic can be found in the next section.

2.2 The Proposed Constrained Optimization Algorithm

There exists a wide spectrum of algorithms under the umbrella of SBO with varying performance depending on the choices made in developing the surrogate optimization framework (Razavi, Tolson, and Burn 2012; Wang et al. 2014). The choices can vary in many regards including the constraint handling technique (CHT) used to deal with available constraints. Multiple previous studies indicate that the CHT of an optimization algorithm can have a drastic effect on the quality and accuracy of the resulting solution (Mezura-Montes and Coello 2011; Woldesenbet, Yen, and Tessema 2009). More importantly, the CHT of the algorithm chosen to solve the bio-inspired locomotion design problem must be able to adequately handle constraints without generating any infeasible candidates. In other words, the algorithm must guarantee strictly feasible candidates during the optimization procedure so the numerical simulations of the thunniform swimmer(s) do not fail. This section starts with a discussion of the available CHT techniques for use with an SBO algorithm. Following that is a discussion on optimizing over highly constrained, or "thin", domains. Next, a new constrained SBO algorithm is proposed to handle inexpensive constraint functions by ensuring strictly feasible candidates while keeping the number of function evaluations low. The proposed algorithm is benchmarked against several other algorithms with different CHTs to asses it's accuracy and efficiency.

2.2.1 On Constraint Handling

A popular class of CHTs involves casting the constrained optimization problem as a multi-objective optimization problem where the objectives minimize the fitness function and constraint violations. One way to solve the multi-objective optimization is to use a "filter" approach (Audet et al. 2000). The objective function and the constraints form a Pareto front, where candidates that are non-dominated, those which lay outside the Pareto front, are accepted through the filter and dominated candidates, those which lay within the Pareto front, remain unfiltered. The filter approach, along with other multi-objective optimization techniques (Müller and Woodbury 2017; Regis 2014), relax constraint requirements during the optimization procedure, and therefore may not be easily adapted to practical problems where the constraints must be strictly enforced (Nazemian and Ghadimi 2021). The requirement for strictly feasible candidates can arise, for example, in optimization problems where constraint failures result in non physically realizable parameters in a CFD simulation (Gebraad et al. 2017; Neufeld, Chung, and Behdinian 2011) or a physical experiment. When infeasible candidates are strictly not allowed, it is not possible to use the information contained within them to guide the optimization process, which might be a beneficial strategy in other circumstances (Ray et al. 2009; Orvosh and Davis 1994).

One approach to avoid infeasible solutions is to cast the constrained optimization problem into a single unconstrained objective with an added term which penalizes constraint violations. An example is the interior point penalty method, which adds a barrier function to the objective function that approaches infinity as the point approaches the boundary of the feasible region (Dikin 1967; Karmarkar 1984). With this method, however, it becomes problematic to find the optimum solutions that may

be located on the boundary of the feasible region. A modification of the method that avoids this problem is an "extreme barrier" treatment, where the objective function is not modified in the feasible domain, but is set to infinity in the infeasible domain (Audet and Dennis Jr 2006; Diouane, Gratton, and Vicente 2015). Convergence and accuracy of these methods is highly dependent on the threshold of the penalty parameter (Queipo et al. 2005). A high threshold can dominate the objective function and deteriorate convergence speed, while a low threshold could produce infeasible candidates.

Other CHTs have been developed around different infill strategies. For example, it has been suggested that the Expected Improvement (EI) infill criteria, used in the Efficient Global Optimization (EGO) algorithm by Jones, Schonlau, and Welch 1998, be set to zero when any constraint is violated (Gardner et al. 2014; Sobester, Forrester, and Keane 2008). When the infill strategy is solely guided by the EI criteria, there will be no exploration of the infeasible regime. Another possible solution arises when considering optimization problems where simulation failure can not be determined *a priori* (Forrester, Sóbester, and Keane 2006). The suggested treatment, by Forrester, Sóbester, and Keane 2006, is to impute a value that is proportional to the sum of the surrogate response prediction and the prediction uncertainty where the simulation fails. Although other CHTs, aimed at problems with expensive constraints, rely on creating surrogate predictions of the constraint functions (Parr et al. 2012), they are generally inadvisable when constraint functions are inexpensive (Sasena, Papalambros, and Goovaerts 2001).

I propose a simple rejection based CHT for inexpensive constraints, capable of insuring strictly feasible candidates, for the Metric Stochastic Response Surface (MSRS) infill sampling criterion (Regis and Shoemaker 2007), which has been shown

to converge to the global minimum under certain conditions. Ordinary Kriging (OK) surface (Eldeiry and Garcia 2010) is employed as the surrogate model to exploit with the constrained MSRS infill strategy, and the resulting SBO algorithm variant is referred to as OK-CMSRS. Ordinary Kriging was chosen for its ability to allow for a flexible and reliable prediction method (Sobester, Forrester, and Keane 2008; Eldeiry and Garcia 2010). However, the response surface model need not always be Ordinary Kriging and can, if desired, be substituted with a different approximation model (Regis and Shoemaker 2007). Inexpensive constraints can at times create a extremely small feasible region, referred to as a "thin" domain, which increases the challenge of finding a suitable feasible solution. The next subsection provides a formal definition of a "thin" feasible domain and discuses approaches to appropriately deal with this challenge.

2.2.2 Optimizing over "Thin" Domains

In some cases the feasible domain may be restricted to a "thin" region. I formally define a "thin" feasible domain if for any $\mathbf{x} \in D$, we have:

$$P[\mathbf{x} + \mathbf{d}\alpha \in D] \simeq 0 \tag{2.17}$$

where $P[E]$ is the probability of event E, $\mathbf{d}$ is an arbitrary direction and $\alpha > 0$ is an arbitrary scalar (Martinez and Sobral 2013). It follows that a singular equality constraint is sufficient to generate a "thin" feasibility region. To allow for optimization over a "thin" feasible region, I follow Kumar et al. 2020; Runarsson and Yao 2000; Mezura-Montes and Coello 2011; Liu et al. 2016; Liang et al. 2006 and transform each equality constraint into two inequality constraints using a small positive margin,

$\epsilon > 0$, as follows:

$$|h_r(\mathbf{x})| - \epsilon \leq 0, \quad r = q+1, ..., m. \tag{2.18}$$

Likewise, multiple inequality constraints could result in a "thin" feasible region that is geometrically similar to that of an equality constraint. For example, consider the case where set C is defined by:

$$\begin{aligned} g_1(\mathbf{x}) &= x_1^2 + x_2^2 - 1 \leq 0, \\ g_2(\mathbf{x}) &= -x_1^2 - x_2^2 + 1 \leq 0. \end{aligned} \tag{2.19}$$

The intersection of these two inequality constraints defines an equality constraint,

$$h_1(\mathbf{x}) = x_1^2 + x_2^2 - 1 = 0, \tag{2.20}$$

which can be treated in the same manner (equation 2.18). I assume that treatment of equality and inequality constraints in this manner generates admissible points, for which the objective function can be evaluated. If this is not the case, the margin of error ϵ may be adjusted. To quantify the measure of "thinness" of a feasible domain, I introduce the feasibility ratio, ρ, defined as

$$\rho = \frac{m(D)}{m(S)}, \tag{2.21}$$

where $m(X)$ refers to the measure of the set X. The feasibility ratio thus provides an estimate of the size of the feasibility region relative to the search space. The attention is now turned to a new constrained SBO algorithm capable of generating strictly feasible candidates, even for some problems with a "thin" feasible domain or, alternatively, a small ρ.

2.2.3 Optimization Method Description

SBO algorithms rely on approximating solutions based on a surrogate model of the objective function. First, an initial surrogate model of the objective function is created using a data set of true function evaluations sampled with a space filling strategy. Then, with each iteration an infill criterion, which attempts to balance global and local exploration, is used to refine the surrogate with multiple surrogate function evaluations and one true function evaluation. In this work, I choose a sampling technique and a surrogate model from the DACE MATLAB toolbox (Lophaven, Nielsen, Søndergaard, et al. 2002). I define an infill strategy according to the Metric Stochastic Response Surface (MSRS) method proposed by Regis and Shoemaker 2007. Finally, I extend the MSRS method to produce strictly feasible candidates at each iteration step. The resulting variant algorithm is referred to as OK-CMSRS.

2.2.3.1 Metric Stochastic Response Surface (MSRS) Algorithm

Let M_k and $s_k(\mathbf{x})$ be defined as the set of candidates used for "true" function evaluations and the surrogate model, respectively, at iteration k. I define maximum number of iterations, k_{max}, and a tolerance, tol.

Step 1: Create an initial sampling set $T \subset D$, containing only feasible points, using Latin Hypercube Sampling (LHS) (McKay, Beckman, and Conover 1979) and a constraint handling technique as described in section 2.2.3.2. I require that $\text{card}(T) \geq n + 1$, where 'card' stands for cardinality, and n is the dimension of the search space. Evaluate $f(T)$, where $f(\mathbf{x})$ is the true objective function. Set $M_0 = T$.

Step 2: Fit a Kriging model surrogate, $s_0(\mathbf{x})$, with a Gaussian correlation function and 0-th order regression polynomial using M_0, commonly referred to as Ordinary Kriging (OK), as follows:

$$s_0(\mathbf{x}) = \hat{\mu} + \psi^T \mathbf{\Psi}^{-1}(f(M_0) - \mathbf{1}\hat{\mu}) \tag{2.22}$$

where

$$\hat{\mu} = \frac{\mathbf{1}^T \mathbf{\Psi}^{-1} f(M_0)}{\mathbf{1}^T \mathbf{\Psi}^{-1} \mathbf{1}}. \tag{2.23}$$

ψ is a correlation vector of the untried point, $\mathbf{x}$, to the sampled data and, therefore, has a length equal to card(M_0). The correlation equation, $\psi^{(i)}$, is given by:

$$\psi^{(i)} = \exp\left(- \sum_{p=1}^{n} |x_p^{(i)} - x_p|^2 \right), \tag{2.24}$$

where $\mathbf{x}^{(i)} \in M_0$, $i = 1, \ldots,$ card(M_0). Finally, $\mathbf{\Psi}$ is a square matrix of size card(M_0) $\times$ card(M_0) , with each matrix element $\mathbf{\Psi}_{ij}$ given by the correlation between the vectors $\{\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\}$ from the set M_0, as in the equation 2.24. For a full derivation of the Ordinary Kriging predictor, the reader is referred to Forrester and Keane 2009; Jones 2001.

Set $k = 0$.

Step 3: While $(k \leq k_{max})$

a) Create a set of strictly feasible candidate points, X_k, according to the proposed CHT (section 2.2.3.2) and evaluate $s_k(X_k)$.

b) Use the MSRS scoring method which assigns a weighted score to each point in set X_k based on two criteria: 1) the distance of points in X_k to M_k, and 2) the surrogate response values, $s_k(X_k)$. The weighted score insures that the next candidate point has a low objective value that is far away

from previously sampled points. The point with the best weighted score is identified as the next evaluation point, $\mathbf{x}_k$.

c) Evaluate $f(\mathbf{x}_k)$.

d) Identify the current best point $\mathbf{x}_k^*$. If *tol* is met: **break**.

e) Set $M_{k+1} = M_k \cup \mathbf{x}_k$. Re-fit s_{k+1} with M_{k+1}. Set $k = k + 1$.

Step 4: Return $\mathbf{x}_{best} = \mathbf{x}_k^*$.

2.2.3.2 Constrained Candidate Sampling

In the MSRS method, the candidate points are split into two categories (Regis and Shoemaker 2007):

1) Uniformly sampled global points: The first set, U_k, is generated by a random sampling with a uniform distribution of points from the box-constrained domain such that $U_k \subset S$. I set $\text{card}(U) = 2000\,n$, where n is the dimension of the search space.

2) Normally sampled local points: The second set N_k is generated by adding perturbations to $\mathbf{x}_k^*$ drawn from a random normal distribution with zero mean and a variance s^2:

$$N_k^j = \mathbf{x}_k^* + \mathcal{N}(0, s^2), \quad j = 1, ..., \text{card}(N_k), \tag{2.25}$$

where N_k^j are components of N_k, $\mathcal{N}(0, s^2)$ is the normal distribution with 0 mean and s^2 variance, s is chosen as

$$s = \gamma\big[\min_{i\in[1,n]} (u(i) - l(i))\,\big], \tag{2.26}$$

and γ is the "perturbation rate" that can assume one of the three values, 0.1, 0.01, or 0.001, assigned randomly, with equal probability (in other words, γ is a

uniformly distributed discrete random variable over the set $\{0.1, 0.01, 0.001\}$).I set $\text{card}(N) = 2000\,n$.

I define the possibly infeasible candidate set as $X'_k = U_k \cup N_k$. It follows that $\text{card}(X'_k) = 4000\,n$. I note that $X'_k \subset S$, however it may be that $X'_k \not\subset S \cap C$. I then enforce all, equality and/or inequality, constraints through the following CHT algorithm:

Step 1: Evaluate $g_r(X'_k)$ for $r = 1, ..., q$ and $h_r(X'_k)$ for $r = q+1, ..., m$.

Step 2: Define 'penalty' vector, J, for each candidate point:

$$J_i = \sum_{r=1}^{q} \max(0, g_r(\mathbf{x}_i)) + \sum_{r=q+1}^{m} \max(0, |h_r(\mathbf{x}_i)| - \epsilon), \quad i = 1, ..., \text{card}(X'_k), \tag{2.27}$$

where J_i are the components of the vector J.

Step 3: The feasible set, X_k, is simply defined as $X_k = \{X'_k : J = 0\}$. In other words, the X_k entries are the candidates in X'_k with a zero penalty.

This method does not guarantee that the selected set X_k will always be of the nominal $\text{card}(X'_k)$ but it will surely contain only feasible points. If it happens that $X_k \subset M_k$, then the candidates are resampled.

To construct the initial LHS sampling set, T, the same 3-step process to generate feasible candidates is followed. Originally, a possibly infeasible set T' is generated using LHS sampling over the box domain S, consisting of Ceiling $[(n+1)/\rho]$ candidate points. Infeasible candidates are then rejected using the presented CHT algorithm. If it happens that $\text{card}(T) < n + 1$, a new set of candidate points is generated to add to T', until the condition on $\text{card}(T)$ is satisfied.

2.2.4 On the Convergence of the Proposed Method

To prove convergence of the OK-CMSRS algorithm, we first need the following lemma.

Lemma 2.2.1. Let $\mathcal{E}_k = \{T, X'_1, ..., X'_k\}$. Then the two following conditions are satisfied:

$[C1^0]$ For all k, elements of X'_k are conditionally independent given the random vectors in $\mathcal{E}_{k-1}$.

$[C2^0]$ For any $\mathbf{x} \in S, j = 1, ..., \text{card}(X'_k)$ and $\delta > 0$, there exists $\mu_j(\mathbf{x}, \delta) > 0$ such that

$$P[X'_{k,j} \in S \cap B(\mathbf{x}, \delta) | \sigma(\mathcal{E}_{k-1})] \geq \mu_j(\mathbf{x}, \delta), \tag{2.28}$$

where $P[E]$ is the probability of event E, $B(\mathbf{x}, \delta)$ is the open ball of radius δ and center $\mathbf{x}$, and $\sigma(\mathcal{E}_{k-1})$ is the σ-algebra generated by the random vectors in $\mathcal{E}_{k-1}$.

Proof. See Ref. Regis and Shoemaker 2007 for the proof (p. 500). □

Note that the condition $[C2^0]$ ensures that no region of S is left unexplored by the MSRS infill criterion. I now extend this lemma to the constrained variant of MSRS, i.e. CMSRS.

Lemma 2.2.2. Let $\mathcal{E}_k = \{T, X'_1, ..., X'_k\}$, and $X_k \subseteq X'_k \; \forall \, k$. Then the two following conditions are satisfied:

[C1] For all k, elements of X_k are conditionally independent given the random vectors in $\mathcal{E}_{k-1}$.

[C2] Let $D = S \cap C$. For any $\mathbf{x} \in D, i = 1, ..., \text{card}(X_k)$ and $\delta > 0$, there exists $\nu_i(\mathbf{x}, \delta) > 0$ such that

$$P[X_{k,i} \in D \cap B(\mathbf{x}, \delta) | \sigma(\mathcal{E}_{k-1})] \geq \nu_i(\mathbf{x}, \delta). \tag{2.29}$$

Proof. To prove the condition [C1], I note that, by Lemma 1, for all k, elements of the set X'_k are conditionally independent. Since $X_k \subseteq X'_k$, it follows that the elements of X_k are conditionally independent as well, which proves the condition [C1]. Furthermore, by Lemma 1, the condition $[C2^0]$ holds. Consider $i = 1, \ldots, \text{card}(X_k)$. Since $X_{k,i} \subset X'_k$, we can associate an index $j(i) \in 1, \ldots, \text{card}(X'_k)$, such that $X_{k,i} = X'_{k,j}$. Thus, we have

$$\begin{aligned} & P[X_{k,i} \in D \cap B(\mathbf{x}, \delta) | \sigma(\mathcal{E}_{k-1})] \\ & = P[X_{k,i} \in S \cap C \cap B(\mathbf{x}, \delta) | \sigma(\mathcal{E}_{k-1})] \\ & = \ P[X'_{k,j} \in S \cap B(\mathbf{x}, \delta) | \sigma(\mathcal{E}_{k-1})] \geq \mu_j(\mathbf{x}, \delta) \quad [C2^0] \\ & = \nu_i(\mathbf{x}, \delta), \end{aligned} \tag{2.30}$$

which proves the condition [C2]. □

I note that in the case when the set C contains strict equality constraints, that is D is a "thin" feasible region, condition [C2] may not be satisfied, since probability in equation 2.29 may not be strictly positive:

$$P[X_{k,i} \in D \cap B(\mathbf{x}, \delta) | \sigma(\mathcal{E}_{k-1})] \geq 0. \tag{2.31}$$

However, this situation will not occur when the equality constraints are treated with a small positive margin $\epsilon > 0$, as specified in equation 2.18. Consequently, the conditions [C1] and [C2] are satisfied for any feasible domain D considered in the current method.

Following that conditions [C1] and [C2] hold, the theorem of convergence for the CMSRS method is given as:

Theorem 2.2.3. Let f be a continuous function defined on $D \subseteq \mathbb{R}^n$, $D = S \cap C$, and suppose that $\mathbf{x}^*$ is the unique global minimizer of f on D in the sense that $f(\mathbf{x}^*) = \inf_{\mathbf{x} \in D} f(\mathbf{x}) > -\infty$ and $\inf_{\mathbf{x} \in D, ||\mathbf{x} - \mathbf{x}^*|| \geq \eta} f(\mathbf{x}) > f(\mathbf{x}^*)$ for all $\eta > 0$. Define

the sequence of random vectors $\{\mathbf{x}_k^*\} := \mathbf{x}_k^* = \mathbf{x}_k$ if $f(\mathbf{x}_k) < f(\mathbf{x}_{k-1}^*)$ while $\mathbf{x}_k^* = \mathbf{x}_{k-1}^*$ otherwise. Then $\mathbf{x}_k^* \to \mathbf{x}^*$ almost surely.

Proof. Theorem 2.2.3 is a special case of Theorem 1 in Ref. Regis and Shoemaker 2007 for a constrained domain D. Since the conditions [C1] and [C2] are satisfied for the constrained domain D, the proof in Regis and Shoemaker 2007, page 500, can be applied to the current case without any modifications. Consequently, the reader is referred to Regis and Shoemaker 2007 for the proof. □

2.2.5 Constrained Handling Technique Study

I compare the OK-CMSRS algorithm with several other algorithms which include: 1) The EGO algorithm with zero EI for infeasible candidates, referred to as EGO-Z, 2) an SBO algorithm with an OK model, a MSRS infill strategy and a data imputation treatment (Parr et al. 2012), referred to as OK-IMSRS, 3) an SBO algorithm with an OK model, a MSRS infill strategy, and a simple constant penalty of 1×10^6 outside the feasible domain, referred to as OK-PMSRS and, finally, 4) an evolutionary algorithm, CMA-ES, which treats infeasible solutions by resampling (Hansen 2016). The five considered algorithms are summarized in Table 6. The algorithms are benchmarked on the following six analytical test problems: 1) the Rosenbrock function (De Jong 1975), 2) the Shifted Rotated Rastrigin's Function (Suganthan et al. 2005), 3) Weight Minimization of a Speed Reducer (RC15) (Kumar et al. 2020), 4) the Tension/Compression Spring Design (RC17) (Kumar et al. 2020), 5) 10-Bar Truss Design (RC 27) (Kumar et al. 2020) and 6) Test Problem 3 in Chapter 4 of Floudas and Pardalos 1990. Algorithm performance metrics include a function evaluation count and a solution error to assess efficiency and accuracy of each optimization solver.

Table 6: Optimization Algorithms

Software Language	Algorithm	Comments
MATLAB	EGO-Z	EGO-DACE (Lophaven, Nielsen, Søndergaard, et al. 2002)
MATLAB	OK-IMSRS	MSRS-MATSuMoTo (Mueller 2014)
MATLAB	OK-PMSRS	MSRS=MATSuMoTo (Mueller 2014)
MATLAB	CMA-ES	Hansen (Hansen 2016)
MATLAB	OK-CMSRS	See Section 2.2.3

2.2.5.1 Optimization Benchmark Functions

All test problems employed in the CHT study are summarized in Table 7. Below I provide a more detailed description of each test problem.

Table 7: Problem specifications for design problems 1 - 6. n is total number of dimensions, g is the total number of inequality constraints, h is the total number of equality constraints, $f(\mathbf{x}^*)$ is the function minimum and ρ is the estimated feasibility ratio.

Problem	n	g	h	$f(\mathbf{x}^*)$	ρ
1. Rosenbrock	2	2	0	0	2.00×10^{-1}
2. Rastrigin	2	2	0	-33.0	2.40×10^{-1}
3. Speed reducer	7	11	0	2.99×10^{3}	9.76×10^{-4}
4. Spring	3	4	0	1.27×10^{-2}	7.50×10^{-3}
5. Truss	10	3	0	5.24×10^{2}	1.16×10^{-1}
6. p43	4	2	1	-4.51	8.80×10^{-6}

2.2.5.1.1 Test Problem 1

Test Problem 1 is the constrained Rosenbrock function introduced earlier (see Section 2.1.3.1). A summary of the design parameters, together with the feasibility ratio for this problem, can be found in Table 7.

2.2.5.1.2 Test Problem 2

The second analytical function considered is the shifted rotated Rastrigin function used as one of the benchmark functions for CEC2005 competition (Suganthan et al. 2005). This variant of the Rastrigin function provides a non-linear, non-separable, multi-modal challenging test function (Figure 3). The 2-D form used in this study is given as follows:

$$f(x_1, x_2) = \sum_{i=1}^{2} (z_i^2 - 10\cos(2\pi z_i)) - 33, \tag{2.32}$$

where $\mathbf{z} = (\mathbf{x} - \mathbf{x}^*)\mathbf{M}$, $\mathbf{x} = [x_1, x_2]$, $\mathbf{x}^* = [x_1^*, x_2^*]$ is the shifted global optimum, with $f(\mathbf{x}^*) = -33$, $\mathbf{M}$ is a linear transformation matrix with condition number = 2. Set S is given by:

$$\begin{aligned} -2 \leq x_0 \leq 10, \\ -10 \leq x_1 \leq 2. \end{aligned} \tag{2.33}$$

The C constraint set in this case is:

$$\begin{aligned} g_1(\mathbf{x}) = x_2 + 0.15x_1^2 - 2 \leq 0, \\ g_2(\mathbf{x}) = -x_2 - x_1 - 1 \leq 0. \end{aligned} \tag{2.34}$$

A summary of the design parameters, together with the feasibility ratio for this problem, can be found in Table 7.

2.2.5.1.3 Test Problems 3-6

The third, fourth and fifth were chosen to be the speed reducer (RC15), spring (RC17) and truss (RC27) problems, respectively, from the 2020 CEC constrained optimization competition (Kumar et al. 2020). The RC15, RC17 and RC27 problems

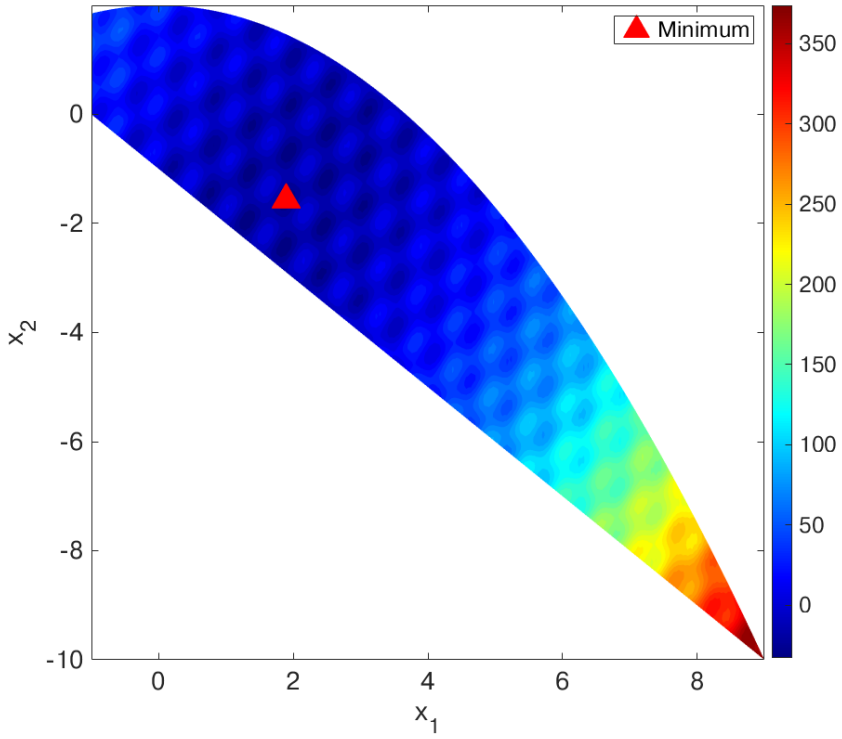


Figure 2: Rastrigin

Figure 3: The 2-D Rastrigin functions over the constrained domain. The triangle represents the optimum solution.

with rigid constraint requirements are of practical relevance to mechanical engineering design. The sixth problem is Test Problem 3 in Chapter 4 of Floudas and Pardalos 1990, referred to as "p43" in Table 7. The sixth problem has one equality constraint and the feasible space is, therefore, restricted to a "thin" region. In a treatment of this constraint, I set $\epsilon = 0.0001$ in equation 2.18, commensurate with the optimization competition (Liang et al. 2006). The dimensional space, number of constraints, the objective function minimum value and the feasibility ratio for test problems 3-6 can be found in Table 7.

2.2.5.2 Results

Numerical simulations were performed on a desktop with a CPU featuring six 3.20GHz Intel processors on a Linux environment. The optimization algorithms were run on MATLAB 2020a. The convergence criteria was based on an absolute error tolerance, $tol = |f_k(\mathbf{x}_{\text{best}}) - f(\mathbf{x}^*)| \leq 0.1|f(\mathbf{x}^*)|$, where $f_k(\mathbf{x}_{\text{best}})$ is the best found solution at iteration k and $f(\mathbf{x}^*)$ is the *a priori* identified global function minimum. An exception is Test Problem 1 where $tol = |f_k(\mathbf{x}_{\text{best}}) - f(\mathbf{x}^*)| \leq 1 \times 10^{-3}$ since $f(\mathbf{x}^*)$ in this case is 0. All algorithms were set to terminate at 1000 iterations, regardless of tol. Each algorithm was run for $N = 50$ realizations and the average function evaluation count as well as the absolute error of the average best function value are reported in Table 8. Additionally, a margin of error (MOE) on the function evaluation count is calculated with 95% confidence for all solvers using Equation 2.16.

The EGO-Z algorithm, with an EI-based CHT, typically demands much higher function evaluations than OK-CMSRS. Moreover, the EGO-Z algorithm could not generate any feasible candidates for Test Problem 6, likely due to the "thin" region of feasibility. Algorithms with an OK surrogate function and a MSRS search technique displayed varying performance based on the selected CHT. For example, the OK-IMSRS required more than double the function evaluations for the Rosenbrock function when compared to OK-CMSRS. The OK-PMSRS required more than 13 times the amount of function evaluations for the same problem, which could be due to the "cliff" effect associated with penalty functions Sobester, Forrester, and Keane 2008. Five out of the six benchmarks showed that CMA-ES can be at times more efficient at identifying a feasible solution within tolerance when compared to other SBO algorithms, with the one exception being OK-CMSRS.

Table 8: Optimization results for test problems 1 - 6. FE: Average Function Evaluation Count, FE MOE: Function Evaluations Margin of Error, and BF: Average Best Function Value.

Algorithm	EGO-Z			OK-IMSRS			OK-PMSRS			CMA-ES			OK-CMSRS		
	FE	FE MOE	BF	FE	FE MOE	BF	FE	FE MOE	BF	FE	FE MOE	BF	FE	FE MOE	BF
I. Rosenbrock	97	50	2.85×10^{-4}	69	13	4.88×10^{-4}	460	84	5.11×10^{-4}	60	6	4.40×10^{-4}	33	2	5.54×10^{-4}
II. Rastrigin	191	22	-30.9	290	43	-30.8	317	60	-30.8	130	51	-30.6	127	15	-30.9
III. Speed Reducer	20	1	3.07×10^{3}	59	26	3.24×10^{3}	59	26	3.22×10^{3}	15	2	3.21×10^{3}	14	2	3.21×10^{3}
IV. Spring	943	63	2.22×10^{-2}	889	84	1.66×10^{-2}	968	45	1.97×10^{-2}	243	90	1.36×10^{-2}	105	45	1.28×10^{-2}
V. Truss	138	35	5.26×10^{2}	695	109	5.93×10^{2}	772	94	5.95×10^{2}	256	42	5.69×10^{2}	74	4	5.61×10^{2}
VI. p43[1]	NA	NA	NA	1000	0	-3.72	1000	0	-3.72	212	7	0	131	1	-4.06

[1] EGO-Z could not generate any feasible candidates for this problem.

Additionally, the CMA-ES is typically associated with a higher MOE, as seen in the Spring design benchmark, when compared to the OK-CMSRS. When the constrained region is "thin" (Test Problem 6), the CMA-ES may not find the global optima. The OK-CMSRS maintained competitive performance by showing the least variation in the best solution reported (lowest MOE) as well as the lowest average function evaluations across all optimization benchmarks, including high dimensional ($D = 10$) and highly constrained ($g = 11$) design problems, with error levels remaining within tolerance. Results from the six analytical benchmarks show that the OK-CMSRS is a promising tool capable of insuring strictly feasible candidates while maintaining competitive convergence properties when dealing with expensive simulations-based black-box optimization problems with inexpensive constraints. Consequently, the OK-CMSRS algorithm is chosen to optimize the locomotion of bio-inspired robotic swimmers for achieve efficient propulsion. The next chapter describes the details of the physical and numerical framework used to predict the hydrodynamic behaviour of bio-inspired thunniform swimmers.

Chapter 3

MODELING AND ANALYSIS OF AN OPTIMAL SOLITARY BIO-INSPIRED THUNNIFORM SWIMMER

This chapter introduces the physical and numerical modeling details of CFD simulations of a solitary thunniform swimmer during start-up. An optimization study is conducted to investigate the optimal kinematic gait of a solitary thunniform swimmer as it accelerates from rest. Section 1 describes the physical model of a solitary self-propelled undulatory swimmer, including the equations governing the swimmer's shape, deformation kinematics and the volume conservation scheme, as well as the self-propulsion mechanism. In section 2, the Navier-Stokes equations governing the fluid flow are formulated and discretized using a spectral element method on moving body-fitted grids. In section 3, the simulation setup, including the computational grid and the boundary conditions are introduced. In Section 5, a formulation of the optimization problem, concerning a solitary accelerating thunniform swimmer, is introduced. This chapter is concluded with a discussion of the results given in Section 4.

3.1 Model of Self-Propelled Undulatory Swimmer

3.1.1 Swimmer Shape

The computational model of a thunniform swimmer employed in the current work considers its two-dimensional approximation in a streamwise-lateral plane. The

geometrical features of the streamlined body of a thunniform swimmer in its static configuration in this two-dimensional approximation have been described using real fish morphology data (Du et al. 2015; Valdivia y Alvarado 2007),

$$y_r^s(x) = r_1 \sin(r_2 x) + r_3 \sin(r_4 x), \tag{3.1}$$

where $r_1 = 0.055 l_b$, $r_2 = \frac{2\pi}{1.25 l_b}$, $r_3 = 0.08 l_b$, $r_4 = \frac{2}{l_b}$, and l_b is the length of the swimmer's body up to but not including the tail. I take equation (3.1) to be the curve corresponding to the right boundary of the fish, $y_r^s(x)$, and the left boundary is given by the symmetry, such as $y_l^s(x) = -y_r^s(x)$. The total dimensional length, $L = l_b + l_t$, where l_t is the length of the tail, is chosen to be $L = 0.3$ m to mimic the length of a realistic soft robotic thunniform swimmer (Valdivia y Alvarado 2007). I model a sharp tail by linearly extending the body of the swimmer across l_t, which is chosen to be $l_t = 0.05$ m, yielding $l_b = 0.25$ m. The static swimmer geometry can be seen in Figure 4.

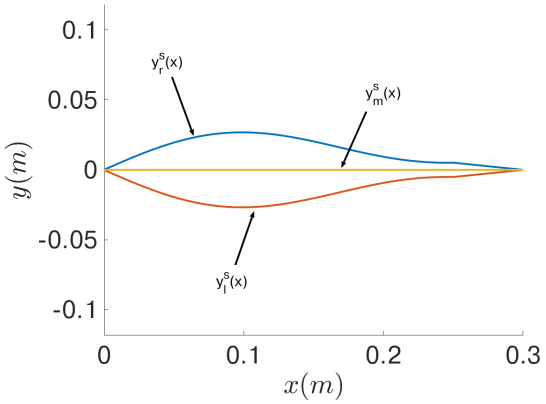


Figure 4: Static thunniform swimmer. $y_r^s(x)$ and $y_l^s(x)$ are the right and left boundaries of the swimmer in its static configuration, respectively; $y_m^s(x) = 0$ is the static midline.

3.1.2 Swimmer Kinematics

The majority of robotic swimmers are designed to mimic the body and/or caudal fin (BCF) movement of aquatic animals (Du et al. 2015). BCF movement characterizes a range of swimming motions that are used by fast swimmers like tuna, sharks and pikes, and hence provides superior propulsive techniques to be utilized by robots. The form of the BCF flapping motion for a slender body undulating in a lateral direction with respect to a direction of motion can be described by a traveling wave equation (Michael J Lighthill 1960; Barrett et al. 1999) for the body midline deformation of thunniform (tuna) swimmers:

$$y_m(x,t) = \left[c_1 \frac{x}{L} + c_2 \left(\frac{x}{L}\right)^2\right] \sin(kx - \omega t), \tag{3.2}$$

where y and x are the lateral and streamwise coordinates of the midline points, respectively, t is the time variable, L is the fish length, c_1 and c_2 are the linear and quadratic wave amplitudes, k is the wave number associated with the body motion, and w is the wave frequency. Subsequent biological studies revealed that the thunniform swimming mode of the BCF family can be characterized by the following body wave number relation (Donley and Dickson 2000),

$$k = \frac{2\pi}{\lambda L},$$

where λ, the body wave length, was measured to be ~ 1.1. I fix λ to be 1.1 and I note that while the tail-beat frequency is observed to have a positive correlation with the swimming speed (Bainbridge 1958; McMasters et al. 2008), I fix $\omega = 2\pi$ rad/s in this study.

The midline of a swimmer, which is simply $y_m^s(x) = 0$ in its static configuration, is discretized with the corresponding midline points with x-coordinates x_m. During

the motion described by Eq. (3.2), the x-coordinates of the midline discretization points remain unchanged, while y-coordinates update according to Eq. (3.2). After the midline of the swimmer is deformed, the next step is to deform the right and left curves of the body. Two methods for the body deformation are considered: a non-conservative, and a conservative method, that are described below.

3.1.2.1 Non-conservative Body Deformation

The simplest way to deform the body of the fish is to adjust the boundary curves, $y_r(x,t)$ and $y_l(x,t)$, to maintain orthogonality of the body cross-sections with respect to the midline while keeping the lateral distance between the corresponding points on the fish surface and the midline constant (Bergmann, Iollo, and Mittal 2014).

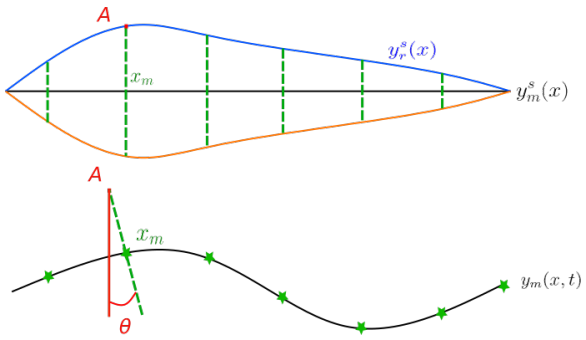


Figure 5: Point A on the right boundary of a swimmer is shown in its static configuration (top view), and at a later point in time when the swimmer is moving (bottom view).

For example, consider a point, A, associated with a certain midline point with the x-coordinate of x_m, which lies on the boundary of a swimmer as depicted in its static configuration in the top view of Figure 5. As the midline deforms, point A on the boundary moves to maintain orthogonality of the connecting segment with the

midline as shown in the bottom view of Figure 5. The geometrical location of the point A, lying, for example, on the right boundary of the swimmer, $\{x_r^A(t), y_r^A(t)\}$, can be described at any instance in time by the following equations:

$$x_r^A(t) = x_m - \sin(\theta(x_m, t))y_r^s(x_m), \tag{3.3a}$$

$$y_r^A(t) = y_m(x_m, t) + \cos(\theta(x_m, t))y_r^s(x_m), \tag{3.3b}$$

where the angle $\theta(x_m, t)$ is such that

$$\theta(x_m, t) = \arctan\left(\frac{d\, y_m(x,t)}{dx}\Big|_{x=x_m}\right). \tag{3.4}$$

The corresponding surface velocity, $\mathbf{v}_r^A(t) = \{v_{x\,r}^A(t), v_{y\,r}^A(t)\}$, for the point A on the right boundary can be obtained by taking the time derivative of Eq. (3.3):

$$v_{x\,r}^A(t) = \frac{d\, x_r^A}{dt} = -\frac{d}{dt}(\sin(\theta(x_m, t)))y_r^s(x_m), \tag{3.5a}$$

$$v_{y\,r}^A(t) = \frac{d\, y_r^A}{dt} = \frac{d}{dt}(y_m^s(x_m, t)) + \frac{d}{dt}(\cos(\theta(x_m, t)))y_r^s(x_m). \tag{3.5b}$$

Similar expressions can be written for the left boundary.

It is easily seen that holding the width of each orthogonal segment distance constant is not sufficient to maintain the volume conservation of the swimmer as the arc length of the midline fluctuates during swimming. To further illustrate this point, I plot the volume of the fish over one time period for the non-volume conserving deformation scheme (Figure 6). It is not physiologically realizable for a neutrally-buoyant swimmer, robotic or organic, to change volume during swimming, unless other buoyancy aids are engaged, which would result in a change of the swimming altitude (Lindsey, Smith, and Croll 2010; Alexander 2013). Additionally, a lack of body volume conservation leads to a sudden change in the volume of the surrounding fluid in the enclosed domain, which violates the incompressibility constraint and leads to numerical instabilities (Xu

and Y. T. Peet 2017). This has been previously alleviated by using additional iterations on the fluid pressure-velocity solve in the moving mesh framework (Xu and Y. T. Peet 2017; Xu and Y. Peet 2021), but this technique is expensive and is hardly suitable within the optimization procedure, where every instance of the CFD solver needs to be executed multiple times. For example, the results obtained with a non-conservative scheme in Figure 6 required, on average, three to five fluid solver iterations per time step (Xu and Y. T. Peet 2018). However, the number of iterations can be significantly larger depending on the problem in hand. For example, 10 to 15 fluid sub-iterations have been reported in (Xu and Y. T. Peet 2017) when computing the fluid-structure interaction cylinder/cantilever beam benchmark of Turek and Hron 2006 with a similar non-conservative solid deformation update in a spectral-element method. While the problem of volume conservation has been previously explored in relation to free-surface flows and fluid-structure interaction with passive solid objects (Franci and Cremonesi 2017; Khayyer et al. 2019), volume conservation of swimmers, to our best knowledge, has not received similar attention in the literature related to fish propulsion (Bergmann, Iollo, and Mittal 2014; Kern and Koumoutsakos 2006; Iman Borazjani and Sotiropoulos 2008). An exception is a study of Shirgaonkar, MacIver, and Patankar 2009, where they imposed divergence-free deformations on an undulatory eel motion, albeit in their deformation scheme the body cross-sections were moved strictly laterally, rather than orthogonal to the midline, in which case the volume conservation is straightforward. An importance and a physiological relevance of orthogonal body deformations was discussed in Bergmann, Iollo, and Mittal 2014. Therefore, in this work, I develop a volume-conservation method for a fish undulatory motion while keeping the fish body cross-sections orthogonal to the midline at all times.

3.1.2.2 Conservative Body Deformation

With the conservative approach, the body surface velocities obtained via Eq. (3.5), which by themselves do not guarantee the volume conservation, are augmented by a correction velocity $\mathbf{v_c}(t) = v_c(t) \cdot \mathbf{n}$, with the magnitude constant over the body surface, which acts in a surface-normal direction $\mathbf{n}$ (with $\mathbf{n}$ denoting the surface outer unit normal). With the correction velocity, Eqs. (3.5) can be re-written as:

$$v^A_{x\,r}(t) = -\frac{d}{dt}(\sin(\theta_m(x_m,t)))y^s_r(x_m) + v_c(t)\cos(\theta(x_m,t)), \tag{3.6a}$$

$$v^A_{y\,r}(t) = \frac{d}{dt}(y^s_m(x_m,t)) + \frac{d}{dt}(\cos(\theta_m(x_m,t)))y^s_r(x_m) + v_c(t)\sin(\theta(x_m,t)). \tag{3.6b}$$

The correction velocity, $v_c(t)$, is found as the solution to the following optimization problem:

$$\begin{aligned} \text{minimize} \quad & \frac{d\mathcal{V}\left(v_c(t)\right)}{dt} \\ \text{subject to} \quad & v_c(t) \in \mathbb{R}, \end{aligned} \tag{3.7}$$

where $\mathcal{V}\left(v_c(t)\right)$ is the volume of the swimmer that depends on the correction velocity $v_c(t)$ (area in the current two-dimensional approximation). To solve the optimization problem, I use the Newton-Raphson secant method at every time step iteration, with the initial value of $v_c(t) = 0$. At each time step, the Newton-Raphson secant procedure converges in roughly 6–8 iterations to machine precision. I note that the iterations on the system of Eqs. (3.6)–(3.7) are purely geometrical and are exceedingly cheaper in comparison to the full fluid pressure-velocity iterations of the Navier-Stokes equations that would have been performed without the volume-conservation scheme (Xu and Y. T. Peet 2017; Xu and Y. Peet 2021). With the new developed correction procedure, the volume of the swimmer is conserved, see Figure 6, and no additional fluid iterations are required on the solver to yield a stable solution. This leads to at least three to

five times of the computational speed-up in the current problem, and up to 10–15 potentially (Xu and Y. T. Peet 2017). The conservative approach will be used in all the computations presented for a solitary swimmer.

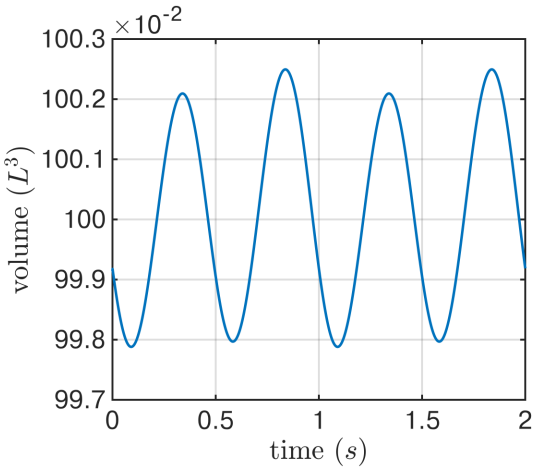


Figure 6: Volume of a swimmer undergoing an undulatory body motion over one cycle period with non-conservative and conservative deformation methods.

3.1.3 Self-propulsion

I model self-propulsion by considering the Newton's second law of motion for the undulating swimmer, while calculating the corresponding viscous and pressure forces directly from the fully-resolved simulations of the fluid-body interactions. Newton's second law of motion for the self-propelling fish can be written as

$$m \frac{d\,\mathbf{U}(\mathbf{t})}{dt} = \mathbf{F}(t), \tag{3.8}$$

where m is the mass of the fish, $\mathbf{U}(t)$ is its translational velocity, and $\mathbf{F}(t)$ is the sum of the viscous and pressure forces acting on its body. The force vector $\mathbf{F}(t)$ consists of

a streamwise, $F_x(t)$, and lateral, $F_y(t)$, components, defined as

$$F_x(t) = \oint_{\Gamma_f} -(\sigma\,\mathbf{n}) \cdot \mathbf{i}\, d\,\Gamma, \tag{3.9}$$

$$F_y(t) = \oint_{\Gamma_f} -(\sigma\,\mathbf{n}) \cdot \mathbf{j}\, d\,\Gamma, \tag{3.10}$$

where σ is the total Cauchy stress tensor, which includes viscous and pressure contributions, Γ_f is the curvilinear boundary of the fish, and $\mathbf{i}$, $\mathbf{j}$ are the unit vectors in the streamwise and lateral directions, respectively. The streawise force $F_x(t)$ consists of the sum of the thrust and drag contributions, while the lateral force $F_y(t)$ will sometimes be referred to as the lift force, to conform to the standard definitions in the airfoil literature (Anderson Jr 2010).

In the current formulation, I only consider the contribution of the streawise force, $F_x(t)$, i.e. thrust and drag, to the motion, and thereby the swimmer is confined to move in the streamwise direction and does not travel in the lateral direction. I also neglect propulsion effects due to moments. The same assumptions were made in (Borazjani and Sotiropoulos 2010; Yu, Lu, and Huang 2021). The adopted motion mimics the case of a swimmer which leverages surrounding fluid to maintain its straight heading direction, and is beneficial for consideration of stability (Borazjani and Sotiropoulos 2010; Yu, Lu, and Huang 2021). Therefore, I assume

$$m\,\frac{d\,U(t)}{dt} = F_x(t), \tag{3.11}$$

where $U(t)$ is the fish forward velocity, and $F_x(t)$ is the streamwise force defined in Eq. (3.9).

3.2 Numerical Method

3.2.1 Arbitrary Lagrangian-Eulerian formulation

In order to solve the moving fluid-structure interaction (FSI) problem, an arbitrary Lagrangian-Eulerian (ALE) formulation (Ho 1989; Deville et al. 2002) of the incompressible Navier-Stokes equations is considered:

$$\rho \left(\frac{\delta \mathbf{u}}{\delta t} + (\mathbf{u} - \mathbf{w}) \cdot \nabla \mathbf{u} \right) + \nabla p = \mu \Delta \mathbf{u}, \tag{3.12a}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{3.12b}$$

where ρ, $\mathbf{u} = \{u_x, u_y\}$, p, μ and $\mathbf{w} = \{w_x, w_y\}$ are the fluid density, velocity, pressure, dynamic viscosity and the mesh velocity, respectively, while the derivative $\delta/\delta\, t$ represents the ALE derivative, please, refer to (Deville et al. 2002; Merrill and Peet 2019) for more details. Mesh velocity has to satisfy the kinematic boundary condition

$$\mathbf{w} = \begin{cases} \mathbf{v}_{\Gamma_m}, & \text{on } \Gamma_m, \\ 0, & \text{on } \Gamma_s, \end{cases} \tag{3.13}$$

where Γ_m is the moving boundary of the fluid domain, Γ_s is the stationary boundary, and $\mathbf{v}_{\Gamma_m}$ is the velocity of the moving boundary.

3.2.2 Spatial Discretization

The spectral element method (SEM) is used to solve Eqs. (3.12) by applying a weighted residual technique that casts the governing equations into a weak, or variational form (Maday and Patera 1989). The weak formulation of Eqs. (3.12)

is given by: find $\mathbf{u} \in \mathcal{H}^1(\Omega(t))^2$ and $p \in \mathcal{L}^2(\Omega(t))$ such that $\forall \mathbf{v} \in \mathcal{H}_0^1(\Omega(t))^2$ and $\forall q \in \mathcal{L}^2(\Omega(t))$,

$$\frac{\delta}{\delta t}\int_{\Omega(t)} \mathbf{v}\cdot\rho\,\mathbf{u}\;d\,\Omega + \int_{\Omega(t)} \mathbf{v}\cdot\rho\,\nabla\cdot(\mathbf{u}\mathbf{u}-\mathbf{u}\mathbf{w})\;d\,\Omega = \int_{\Omega(t)} (p\,\nabla\cdot\mathbf{v} - \mu\nabla\mathbf{v}:\nabla\mathbf{u})\;\;d\Omega, \tag{3.14a}$$

$$-\int_{\Omega(t)} q\nabla\cdot\mathbf{u}\;d\,\Omega = 0, \tag{3.14b}$$

where $\nabla\mathbf{v} : \nabla\mathbf{u}$ refers to a double-dot product between the two tensors, $\mathcal{L}^2(\Omega(t))$ denotes the Hilbert space of square-integrable functions, $\mathcal{H}^1(\Omega(t))^2$ denotes the Sobolev space of 2D vector functions which possess a square-integrable first derivative, while $\mathcal{H}_0^1(\Omega(t))^2$ further constraints the functions to vanish on Dirichlet boundaries.

In a spectral element method, the computational domain $\Omega(t)$ is subdivided into E non-overlapping rectilinear elements, or $\Omega(t) = \cup_{e=1}^{E}\{\Omega_e(t)\}$. Additionally, each element $\Omega_e(t)$ is mapped to the reference domain, $\hat{\Omega} = [-1,1]^2$, such that $\mathbf{x}^e(\boldsymbol{\delta}) \in \Omega_e(t) \rightarrow \boldsymbol{\delta} \in \hat{\Omega}$. The trial functions $(\mathbf{u}, p)$ and the test functions $(\mathbf{v}, q)$ are discretized on the reference domain using high-order polynomial expansions, which, in 2D, take the form

$$\mathbf{s}(x,y) = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} \hat{\mathbf{s}}_{ij}\phi_i(x)\phi_j(y). \tag{3.15}$$

Here, ϕ_i and ϕ_j are the basis functions in the x and y coordinates with degree N_x and N_y, respectively, and $\hat{\mathbf{s}}_{ij}$ are the vector-valued basis coefficients. I specify the Lagrange interpolating polynomials as basis functions, which satisfy the cardinality property $\phi_i(\xi_k) = \delta_{ik}$, $\phi_j(\xi_m) = \delta_{jm}$, with δ_{ik}, δ_{jm} being the Kronecker delta functions, and $\{\xi_k, \xi_m\}$, $k = 1 \ldots N_x$, $m = 1 \ldots N_y$, the interior collocation points. I set the polynomial order $N_x = N_y = N$, with the discretization points as Gauss-Lobatto-Legendre (GLL) points for the velocity, and $N_x = N_y = N - 2$, with the discretization

points as Gauss-Lobatto (GL) points for the pressure, to arrive at the $\mathbb{P}_N - \mathbb{P}_{N-2}$ formulation known for its ability to suppress spurious pressure modes (Bernardi and Maday 1988). The spectral element discretization ensures numerical stability and minimizes dissipative and dispersive errors (Deville et al. 2002). For additional details, including extension of Eq. (3.15) to curvilinear coordinates and handling a global assembly across the elements, the reader is referred to Deville et al. 2002; Merrill and Peet 2019; Fischer 1997.

3.2.3 Temporal Discretization

3.2.3.1 ALE Equations

Equations (3.14) are discretized in time using explicit Adams-Bashforth scheme for nonlinear terms, and implicit backward-difference (BDF) scheme for viscous and pressure terms, yielding the following discretization

$$\begin{aligned} \frac{1}{\delta t}\left[\left(\int_{\Omega(t^n)} \mathbf{v}\cdot\rho\,\mathbf{u}\;d\Omega\right)^n - \left(\int_{\Omega(t^{n-1})} \mathbf{v}\cdot\rho\,\mathbf{u}\;d\Omega\right)^{n-1}\right] \\ + \sum_{j=1}^{n} a_j \left(\int_{\Omega(t^{n-j})} \mathbf{v}\cdot\rho\,\nabla\cdot(\mathbf{uu}-\mathbf{uw})\;d\Omega\right)^{n-j} \\ -\int_{\Omega(t^n)} (p\nabla\cdot\mathbf{v}d\Omega)^n = -\left(\int_{\Omega(t^n)} \mu\nabla\mathbf{v}:\nabla\mathbf{u}\right)\;d\Omega)^n \end{aligned} \tag{3.16}$$

where a_j is the j-th coefficient of the n^{th}-order Adams-Bashforth scheme ($n = 3$ is used in the current setup), and δt denotes the time step. To ensure a divergence-free velocity field, the standard pressure-velocity decoupling approach resulting from the operator splitting is utilized (Deville et al. 2002; Quarteroni, Saleri, and Veneziani 2000). As mentioned above, with a non-conservative solid deformation scheme, a nonlinear term has to be discretized implicitly via an iterative approach, while iterations are not

required when the volume of the moving solid body is conserved. The SEM ALE algorithm was previously validated extensively on a variety of benchmark problems involving moving and deforming grids in its explicit and implicit formulation (Merrill and Peet 2019; Patel et al. 2019; Xu and Y. Peet 2021).

3.2.3.2 Mesh Velocity

To compute the mesh velocity, at the end of each time step t^n, the fish kinematics is first updated according to Eqs. (3.6). In solving for the correction velocity via Eq. (3.7), a volume change is discretized as $d\mathcal{V}/dt \approx (\mathcal{V}^n - \mathcal{V}^{n-1})/\delta t$.

In the current situation of only one swimmer, $\Gamma_m = \Gamma_f$, and the velocity of the undulating fish surface, in accordance with Eq. (3.13), is set as the boundary condition for the mesh velocity at the fluid-body interface as

$$\mathbf{v}^n_{\Gamma_m} = \begin{cases} \mathbf{v}^n_r & \text{on } \Gamma_f : y^s \geq 0, \\ \mathbf{v}^n_l & \text{on } \Gamma_f : y^s < 0, \end{cases} \tag{3.17}$$

where $\mathbf{v}^n_r, \mathbf{v}^n_l$ are velocities of the right and left boundaries of the swimmer, respectively, computed at time step t^n, and y^s is a static configuration of the fish as in Figure 4. As noted above, the other boundary conditions on the mesh velocity are set to zero at the stationary surface Γ_s, which includes the inlet, the outlet, and the lateral boundaries of the fluid domain.

To find the interior mesh velocity $\mathbf{w}^n(x, y)$ while maintaining a smooth mesh movement, a blending function, $f_{blend}(x^s(x, y), y^s(x, y))$, is designed, based on a static mesh configuration, to blend the mesh movement from the fish boundary Γ_f to the rest of the domain. In order to do that, I first define the undamped interior mesh

velocity, by extruding the fish surface velocity into the interior of the domain as

$$\mathbf{w}_\Gamma^n(x,y) = \begin{cases} 0 & \text{for } (x,y) : x^s < 0, \\ \mathbf{v}_r^n(x) & \text{for } (x,y) : 0 \le x^s \le 0.3, y^s \ge 0, \\ \mathbf{v}_l^n(x) & \text{for } (x,y) : 0 \le x^s \le 0.3, y^s < 0, \\ \mathbf{v}_r^n(x_t) & \text{for } (x,y) : x^s > 0.3, \end{cases} \tag{3.18}$$

where x_t is the current axial position of the tail tip, and then smooth it with the help of a blending function as

$$\mathbf{w}^n(x,y) = f_{blend}\left(x^s(x,y), y^s(x,y)\right) \mathbf{w}_\Gamma^n(x,y). \tag{3.19}$$

The blending function, $f_{blend}(x^s, y^s)$, takes the following form:

$$f_{blend}(x^s,y^s) = \begin{cases} -y^s + 0.2, & (x^s,y^s) \in X^+ \\ \quad y^s + 0.2, & (x^s,y^s) \in X^- \\ \frac{-(-y^s+0.2)(x^s-0.3)^2}{0.09} + 1, & (x^s,y^s) \in Y^+ \\ \quad \frac{-(y^s+0.2)(x^s-0.3)^2}{0.09} + 1, & (x^s,y^s) \in Y^- \\ 1, & (x^s,y^s) \in Z \\ 0, & \text{otherwise}, \end{cases} \tag{3.20}$$

where the regions X^+, X^-, Y^+, Y^-, Z are defined according to the following rules:

$$\begin{aligned} X^+ &= \{y^s \le 0.2, y^s \ge 0.1, x^s \ge 0, x^s \le 0.3\}, \\ X^- &= \{y^s \ge -0.2, y^s \ge -0.1, x^s \ge 0, x^s \le 0.3\}, \\ Y^+ &= \{y^s \le 0.2, y^s \ge 0.1, x^s > 0.3, x^s \le 0.6\}, \\ Y^- &= \{y^s \ge -0.2, y^s \le -0.1, x^s > 0.3, x^s \le 0.6\}, \\ Z &= \{y^s \ge y^s(\Gamma_f), |y^s| < 0.1, x^s > 0.3, x^s < 0.6\}. \end{aligned} \tag{3.21}$$

The blending function ensures that only the mesh in the vicinity of the undulating fish deforms, while the rest of the mesh remains stationary. Once the mesh velocity $\mathbf{w}^n$ is calculated, the fluid domain geometry is updated implicitly as $\mathbf{x}^n = \mathbf{x}^{n-1} + \mathbf{w}^n \, \delta t$.

3.2.3.3 Self Propulsion

The Newton's second law of motion that is utilized for a fish self propulsion in the current work, given by Eq. (3.11), is temporally discretized with the second-order implicit Adams-Moulton scheme:

$$U^n = U^{n-1} + \frac{\delta t}{2m}(F_x^n + F_x^{n-1}), \tag{3.22}$$

where $F_x^n = F_x(t^n)$ is the total propelling force acting on the fish in the x direction at a time t^n given by Eq. (3.9). Instead of physically moving the fish through the domain, I model the swimmer's propulsion by adjusting the fluid velocity at the inlet, $\mathbf{u}_{inlet}^n = \{U^n, 0\}$. To achieve an implicit update of the Eq. (3.22) that depends on the force F_x^n, which subsequently depends on the fluid velocity field, a fixed-point iteration approach with Aitken relaxation is performed (Aitken 1927; Küttler and Wall 2008). In the current application, the fixed-point method typically converges in 3–4 iterations to a set tolerance of 1×10^{-4}.

The need for an implicit update between the fluid-body interaction of the fish motion stems from the previous FSI studies involving incompressible flow and solid structures, which indicated that a simple sequentially partitioned coupling between the fluid and the solid (which would be equivalent here to an explicit discretization of Newton's second law) can be unstable irrespective of the time step due to the so-called "added-mass" effect (Causin, Gerbeau, and Nobile 2005; Förster, Wall, and Ramm 2007). It was shown in Küttler and Wall 2008 that the use of an implicit

predictor-corrector scheme based on fixed-point subiterations with Aitken relaxation alleviates the added-mass effect and stabilizes a numerical solution to the FSI problem. An adaption of this fixed-point iteration technique in a spectral-element method shows high accuracy and fast convergence for classical FSI benchmarks (Xu and Y. T. Peet 2017; Xu and Y. Peet 2021), and I refer the reader to these studies for a complete numerical description.

3.3 Simulation Setup

3.3.1 Physical Parameters

In the current simulations, the fluid and fish density are both taken to be 1×10^3 kg/m^3, which corresponds to a neutrally-buoyant swimmer. Dynamic viscosity of the fluid, μ, is set to 1×10^{-3} kg/(m $\cdot$ s). The mass of the fish m is specified as 0.8606 kg, which corresponds to the parameters of a soft robotic thunniform swimmer prototype (Valdivia y Alvarado 2007).

3.3.2 Computational Grid

The computational domain is specified as a rectangle with dimensions $12L$ x $4L$. The fish leading edge is fixed a distance L away from the inlet and a distance $2L$ away from the right and left lateral boundaries. Note that the leading edge (nose) of the fish does not move in the current formulation, while the trailing edge (tip of the tail) moves, see Eq. (3.2). A body fitted structured hexahederal mesh with $N_{el} = 8887$ total elements is constructed to discretize the domain, with the element refinement

along the streamwise direction and the lateral direction close to the fish boundary (See Figure 7a for an illustration of the spectral element mesh in the swimmer's static configuration). In the current study, I perform numerical simulations for two values of the polynomial order N, $N = 5$ and $N = 9$, for the purposes of grid refinement, and to establish a sensitivity of optimization results to the degree of polynomial approximation in a high-order solver. The total gridpoint count based on the GLL velocity points is 3.2×10^5 for $N = 5$ and 8.9×10^5 for $N = 9$. Figure 7b shows a close-up of the numerical grid with the interior GLL points included for the polynomial order $N = 5$. The inset demarcated by the red rectangle in Figure 7b further illustrates the details of the grid for the two polynomial orders, $N = 5$ and $N = 9$, in Figure 8. The static mesh shown in Figures 7, 8 undergoes a dynamic remeshing at every time step to conform to the undulating fish geometry, according to the rules described in Section 3.2.3.2.

3.3.3 Boundary Conditions

Fluid velocity at the fluid-body interface is set equal to the velocity of the moving boundary. A velocity inlet, a pressure outlet, and symmetry boundary conditions are prescribed at the inflow, outflow, and lateral boundaries, respectively. An inlet velocity is obtained at each time step from the self propulsion calculation as described in Section 3.2.3.3.

3.3.4 Initial Conditions

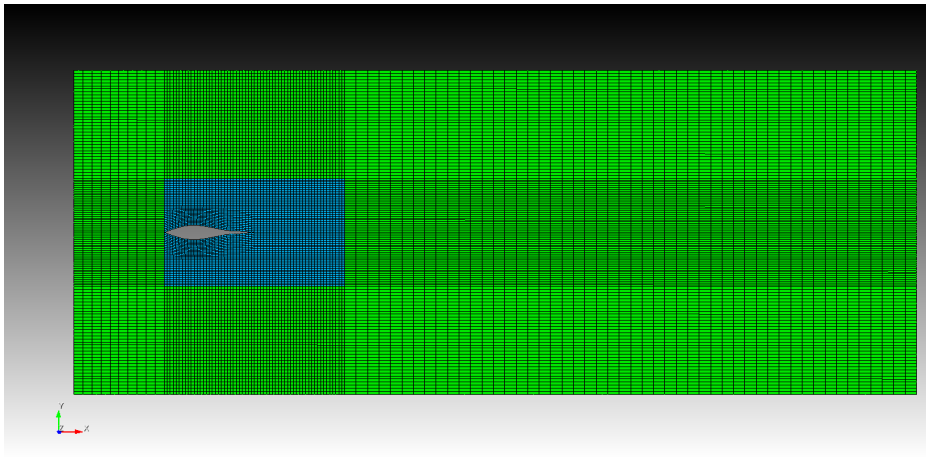
A static configuration of the swimmer represented by Eq. (3.1) and depicted in Figures 4, 7 does not actually correspond to any realizable fish position during swimming. To begin the undulations, I first need to move the swimmer into its starting, but deformed, position. The initial position of the swimmer midline is given by Eq. (3.2) evaluated at $t = 0$, while the position of fish surface at $t = 0$ can be reconstructed via Eqs. (3.3), (3.4) evaluated at $t = 0$. Once the fish is moved into its starting position, the static mesh of Figure 7 is correspondingly deformed via applying the blending function f_{int} given by Eq. (3.20) to the mesh deformation $\delta\mathbf{x}$ instead of the mesh velocity as in Eq. (3.19). The initial fluid velocity and the mesh velocity at $t = 0$ are all set to zero. Once the domain geometry and the numerical mesh are deformed, the simulations commence at $t = 0$ by applying a midline deformation curve of Eq. (3.2) to the time $t^1 = \delta t$ and following all the algorithmic steps as described above, while using the deformed start-up geometry as the initial configuration at $t = 0$.

3.4 Optimization

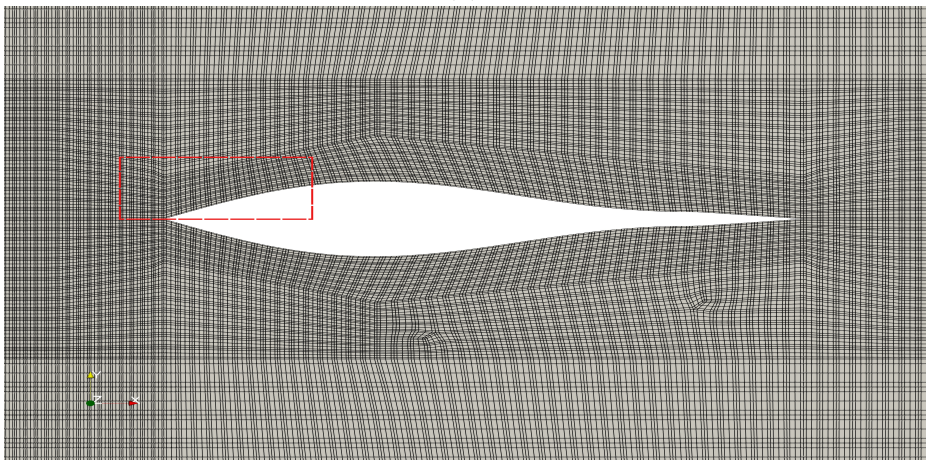
3.4.1 Problem Formulation

The optimization problem is stated as

$$\begin{aligned} &\text{maximize} \quad f(\mathbf{z}) \\ &\text{subject to} \quad \mathbf{z} \in \mathbb{R}^n, \end{aligned} \tag{3.23}$$



(a)

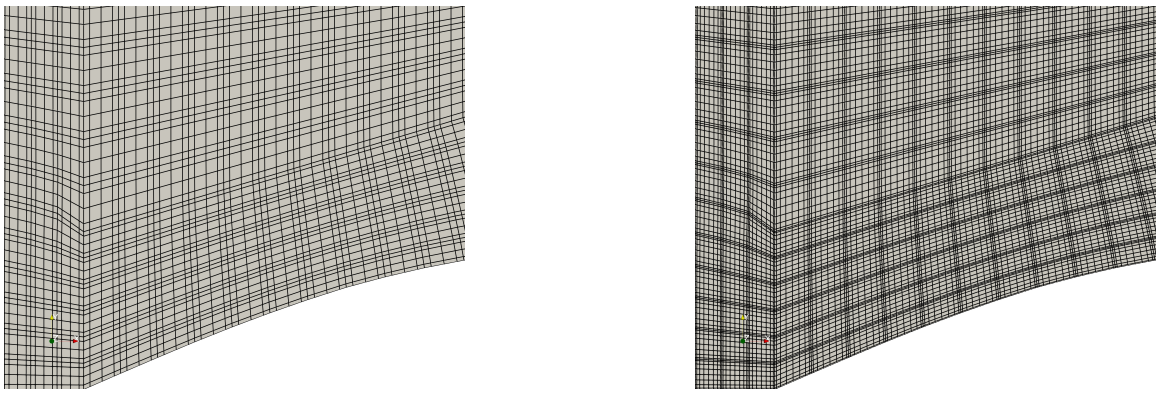


(b)

Figure 7: a) Spectral element mesh with refinement highlighted by black lines; only element boundaries are shown; The region defined by the union of X^+, X^-, Y^+, Y^- and Z is highlighted in blue; b) Numerical grid with GLL discretization points with N=5. Red rectangle indicates an inset for zoom-in in Figure 8.

where $f : \mathbb{R}^n \to \mathbb{R}$ is the objective function, and $\mathbf{z} \in S \cap C$ is a vector of design parameters. The set $S \subseteq \mathbb{R}^n$ contains the n-dimensional search space:

$$l(i) \leq z_i \leq u(i), \quad 1 \leq i \leq n, \tag{3.24}$$



(a) $N = 5$ (b) $N = 9$

Figure 8: An inset of the numerical grid corresponding to a red rectangle in Figure 7 with N-th order polynomial discretization (GLL points shown). a) $N = 5$ and b) $N = 9$

where $l(i)$ and $u(i)$ represent the lower and the upper bounds, respectively, on a design parameter in the ith dimension. The set $C \subseteq \mathbb{R}^n$ contains a set of $m \geq 0$ constraints:

$$g_r(\mathbf{z}) \leq 0, \quad r = 1, ..., m,$$

where $g_r(\mathbf{z})$ is referred to as an inequality constraint on the design parameter vector, $\mathbf{z}$.

3.4.2 Design Parameters

The design parameters are taken to be the amplitude coefficients $\{c_1, c_2\}$ of the midline undulation function $y_m(x, t)$, see Eq. (3.2).

3.4.3 Constraints

In order to allow for physically realizable swimming configurations, I limit the midline movement, Eq. (3.2), by imposing the following constraint set, $\mathcal{C}$, on the design parameters $\{c_1, c_2\}$,

$$\mathcal{C} = \begin{cases} |c_1 + c_2| \leq 0.1L, \\ |\frac{c_1^2}{4c_2}| \leq 0.1L, \\ c_2 \leq 0. \end{cases} \tag{3.25}$$

The constraint set is derived from a physical intuition used in soft swimmer design (Du et al. 2015; Valdivia y Alvarado 2007; Borazjani and Sotiropoulos 2010). The first and second constraints ensure that the maximum tail and body motion amplitudes do not exceed $0.1\,L$. The design domain governed by the first two constraints is symmetric about the origin $\{c_1, c_2\} = \{0, 0\}$, with the pairs $\{c_1, c_2\}$ flipping sign resulting in identical swimming motions that are left-to-right reflections of each other. To avoid redundancy and to save 50% of the computational time, I introduce the third constraint restricting the design domain to one half of the coordinate space.

3.4.4 Objective Function: Propulsive Efficiency

I define the objective function, $f(\mathbf{z})$, as a propulsive swimming efficiency of an undulatory swimmer. A propulsive efficiency can be defined as the ratio of a "useful" energy gained over the total work done by the swimmer over a certain time period

(Webb 1975; Tytell and Lauder 2004; Akanyeti et al. 2017), which can be stated as

$$\eta(c_1, c_2, t_0) = \frac{W_{useful}}{W_{total}} = \frac{\int_0^{t_0} \left(\oint_{\Gamma_f} -(\sigma\,\mathbf{n}) \cdot \mathbf{i}\, d\,\Gamma \right) U(t) d\,t}{\int_0^{t_0} \oint_{\Gamma_f} -(\sigma\,\mathbf{n}) \cdot \mathbf{v}(x, y, t)\, d\,\Gamma d\,t} \tag{3.26}$$

where t_0 is an *a priori* specified variable indicating a time period, and $\mathbf{v}(x, y, t)$ is the fish surface velocity due to undulation. I set t_0 equal to $2T$, where $T = 2\pi/\omega = 1$ s is the tail-beat period. By setting $f(\mathbf{z}) = \eta(c_1, c_2, t_0)$ and fixing t_0 I seek a kinematic gait corresponding to a swimming mode capable of efficiently transferring undulation work to propulsive energy over a brief time span, which is referred to as the start-up propulsive efficiency. Since the objective function, Eq. (3.26), does not explicitly depend on the design parameters, and instead its evaluation requires a solution of the Navier-Stokes equation, this model is referred to as a "black-box" optimization problem (See Chapter 2. I can define a maximum number of iterations, k_{max}, and a tolerance bound, $\varkappa_{opt}$, such that the iterations terminate if the computed tolerance

$$tol = \frac{||f_{\text{opt}} - f_{\text{old-opt}}||}{f_{\text{old-opt}}} \tag{3.27}$$

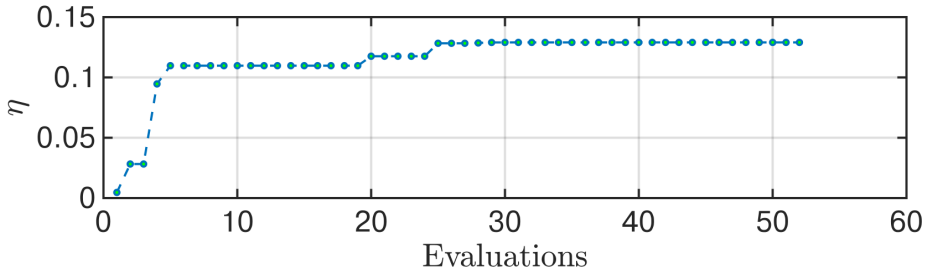
reaches the value

$$tol \leq \varkappa_{opt}, \tag{3.28}$$

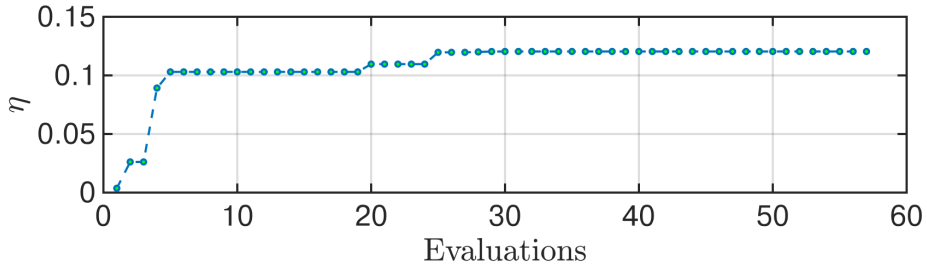
where f_{opt} and $f_{\text{old-opt}}$ are the values of the objective function from the best and the next best iterations. I specify the following termination parameters for the optimization algorithm: $k_{max} = 700, \varkappa_{opt} = 1 \times 10^{-4}$. The optimization problem is solved with the constrained SBO algorithm introduced and bench-marked in Chapter 2.

Table 9: Results for the swimmer propulsive efficiency optimization with two SEM polynomial orders.

Case	Evaluations	Efficiency (η_{opt})	Optimum $\{c_1/L, c_2/L\}$	*tol*
$N = 5$	52	12.90%	$\{0.2106, -0.1108\}$	5.9×10^{-5}
$N = 9$	57	12.04%	$\{0.2104, -0.1106\}$	4.5×10^{-5}



(a) $N = 5$



(b) $N = 9$

Figure 9: Propulsive efficiency η versus the number of function evaluations for two SEM polynomial orders: (a) $N = 5$ and (b) $N = 9$. The η value reported corresponds to the best solution found over the considered number of iterations.

3.5.1 Optimum Parameters

In this section, results of the optimization procedure as applied to an accelerated self-propulsion of a thunniform swimmer that maximizes its propulsive efficiency are presented for two polynomial orders, $N = 5$ and $N = 9$. Figure 9 documents the evolution of the best identified value of the propulsive efficiency η as the function evaluations progress, and Table 9 lists the final results of the optimization procedure, including the total number of evaluations, the final value of the propulsive efficiency, the optimum motion parameters $\{c_1/L, c_2/L\}$, and the actual error tolerance norm *tol* from Eq. (3.27) at the end of optimization. In both cases the tolerance bound $\varkappa_{opt} = 1 \times 10^{-4}$ is reached with less than 60 iterations. The $N = 5$ case results in a slightly higher tolerance norm but a lower function evaluation count, and yields a slightly higher value of an optimum propulsive efficiency $\eta_{\text{opt}} \approx 12.90\%$ as compared to $N = 9$ answer of $\eta_{\text{opt}} \approx 12.04\%$. While the efficiency values are slightly different, both polynomial orders produce essentially identical solutions in terms of the optimum mode of locomotion $\{c_1/L, c_2/L\}$, differing only in the fourth significant digit. The computed optimum efficiency agrees well with the biological data for the propulsive efficiency measured in the live fishes for the given tailbeat amplitude (Akanyeti et al. 2017).

For comparison, I also perform a similar optimization study using a popular evolutionary optimization approach (Kern and Koumoutsakos 2006; Tokić and Yue 2012; Eloy 2013), utilizing COLINY-EA algorithm from the state-of-the-art DAKOTA optimization framework (Adams et al. 2009), which, after $k_{max} = 700$ iterations reached a tolerance *tol* $=1.4 \times 10^{-3}$ and produced a mode with only 11.6% efficiency (Abouhussein, Islam, and Peet 2021). This, again, confirms that the SBO optimization procedure

might be a more efficient alternative to use with expensive function evaluations, at least within the scope of the considered design problem.

3.5.2 Analysis of the Optimum Versus Sub-optimum Modes of Locomotion

In this section, I perform an analysis of the optimum as compared to sub-optimum modes of locomotion, using $N = 9$ case as a reference.

3.5.2.1 Hydrodynamic Variables

Figure 10a shows a contour plot of the objective function, the propulsive efficiency η, using Kriging interpolation between evaluated data points. Figure 10b documents the final attainable swimming speed at the end of the considered period $2\,T$, $U_{max} = U(t = 2\,T)$. The quantities, whose ratio defines the propulsive efficiency in Eq. (3.26), namely W_{useful} and W_{total}, are plotted in Figures 10c and 10d. To draw a complete picture, Figures 10e and 10f present the time-averaged streamwise and lateral forces, $\bar{F}_x$, $\bar{F}_y$, acting on the swimmer, defined as

$$\bar{F}_x = \frac{1}{t_0} \int_0^{t_0} F_x(t)\, d\,t, \tag{3.29}$$

$$\bar{F}_y = \frac{1}{t_0} \int_0^{t_0} F_y(t)\, d\,t, \tag{3.30}$$

with the forces $F_x(t)$, $F_y(t)$ given by Eqs. (3.9), (3.10), respectively, and $t_0 = 2\,T$.

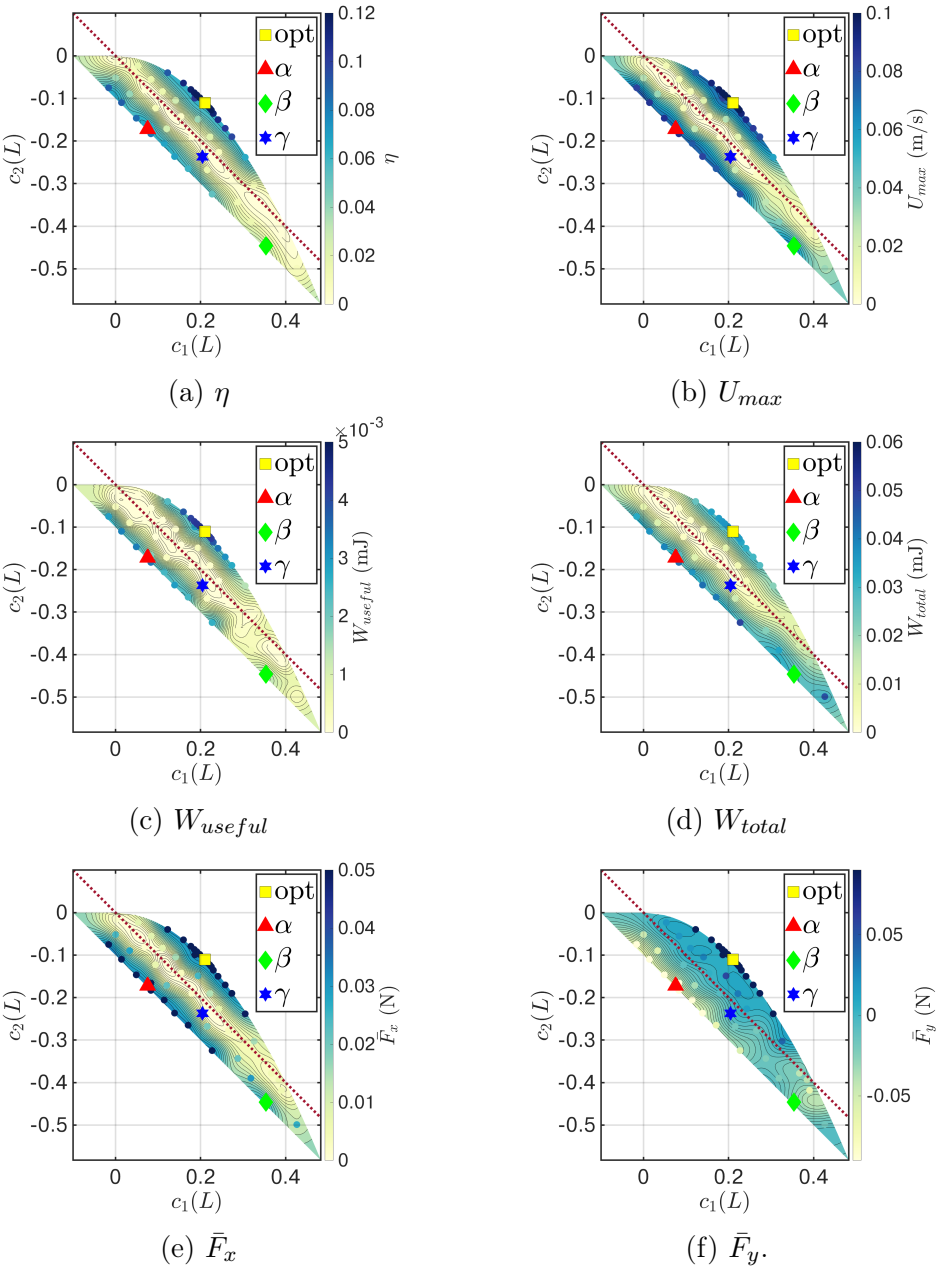
At a glance, the contours of all the five hydrodynamic variables, apart from the lift force, seem rather similar. Another interesting observation is that efficiency is very low in the interior of the design domain, with the line $c_1 + c_2 = 0$ corresponding to a near-zero efficiency. Indeed, as can be seen from Eq. (3.2), the line $c_1 + c_2 = 0$

represents a zero tail amplitude motion, and thus corresponds to a zero theoretical propulsive efficiency (Gazzola, Argentina, and Mahadevan 2014; Saadat et al. 2017; Gibouin et al. 2018).

To better understand the relationships and the correlation properties between different hydrodynamic variables associated with propulsive swimming, Figure 11 presents the correlation matrix of the six considered hydrodynamic variables, and documents the respective Pearson correlation coefficient R. It can be seen that the correlation between efficiency η and the useful (propulsive) work W_{useful} ($R = 0.98$), and, consequently, between efficiency η and the propulsive force $\bar{F}_x$ ($R = 0.97$) are the highest. It is interesting to note that, despite the fact that the total work W_{total} enters the denominator of Eq. (3.26), the correlation between efficiency η and the total work W_{total} is still positive ($R = 0.71$), mainly because the increased total work leads to an increased useful work (correlation of $R = 0.79$) and an increased propulsive force ($R = 0.85$). For the lateral force, I present the correlations with its absolute value, since its sign (indicating whether the total lift acts to the right or to the left with respect to the fish motion) does not matter in regard to efficiency. I note that the correlation between η and $|\bar{F}_y|$ is high ($R = 0.98$), mainly, because high lift is associated with more energetic undulations, which also positively correlate with the propulsive force and the useful work.

3.5.2.2 Relation of Efficiency to Kinematics

I further analyze the details of the optimum and sub-optimum swimming by selecting three specific kinematic modes, named as the modes α, β and γ (Table 10), and comparing their kinematic and hydrodynamic trends with the optimum mode.



(a) η (b) U_{max}

(c) W_{useful} (d) W_{total}

(e) $\bar{F}_x$ (f) $\bar{F}_y$.

Figure 10: Hydrodynamic quantities: (a) η; (b) U_{max}; (c) W_{useful}; (d) W_{total}; (e) $\bar{F}_x$; (f) $\bar{F}_y$; evaluated for the entire design domain using Kriging interpolation between the computed CFD data points with $N = 9$, shown as circles color-coded by the corresponding values. The colored symbols in each figure represent the optimum (square), α (triangle), β (diamond) and γ (star) modes selected for comparison in Table 6. Red dotted line corresponds to $c_1 + c_2 = 0$.

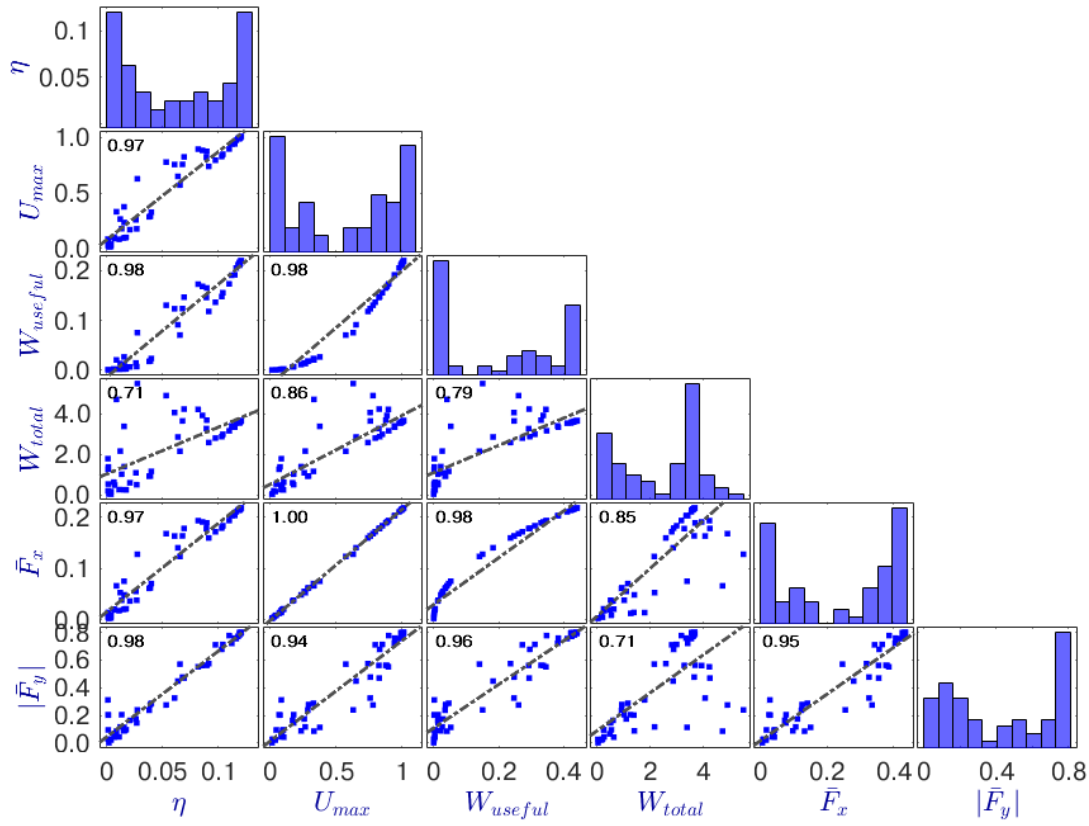
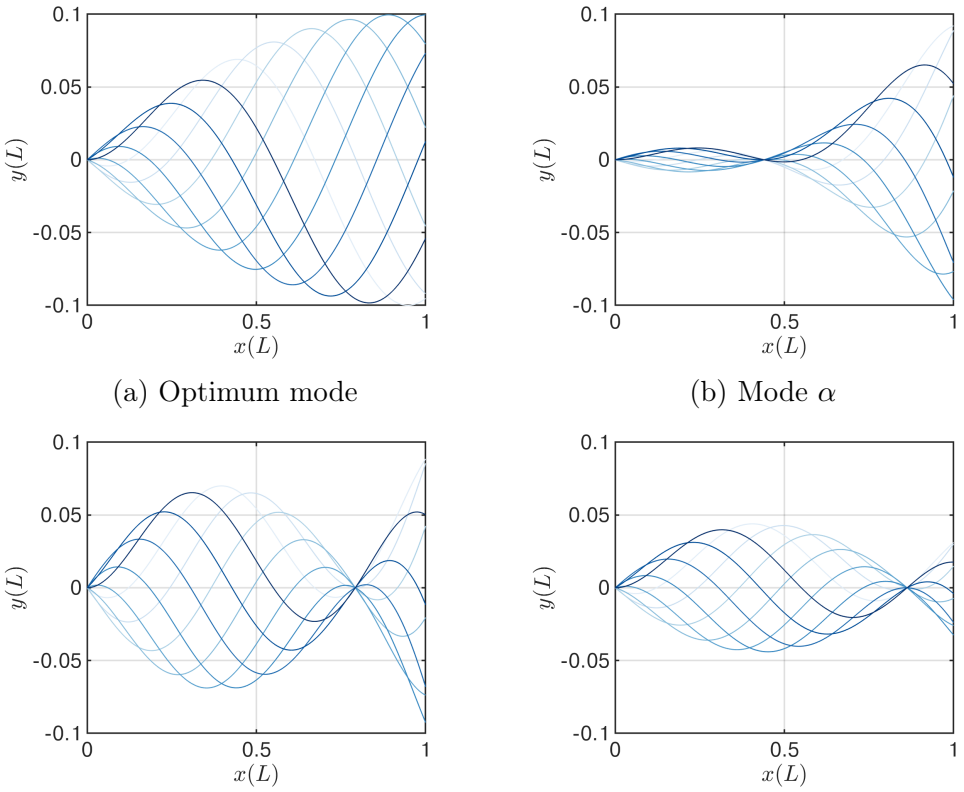


Figure 11: Correlation matrix of hydrodynamic quantities with the respective Pearson correlation coefficient (R) in black inside the corresponding squares. Histograms of each variable are found in the diagonal elements. Histograms are composed of 10 uniformly-distributed bins between the minimum and the maximum value for each variable.

The kinematics of the four selected modes is shown in Figure 12, where the shape of the swimmer's midline is illustrated for each one-tenth of a period during one undulation cycle T. It can be seen that the three more efficient modes, optimum, α and β, all have comparably large tail amplitudes, while the least efficient mode, γ, out of the four analyzed, has a very small tail amplitude motion. The first three modes, however, apart from having similar tail amplitudes, exhibit other marked differences in the body line kinematics, which, as will be seen below, effect their efficiency.

Table 10: Kinematic modes selected for comparison

Mode	$\{c_1/L, c_2/L\}$	η
opt	$\{0.2104, -0.1106\}$	12.04%
α	$\{0.0758, -0.1722\}$	7.99%
β	$\{0.3534, -0.4461\}$	2.74%
γ	$\{0.2046, -0.2372\}$	0.64%



(a) Optimum mode

(b) Mode α

(c) Mode β

(d) Mode γ

Figure 12: Swimmer midline deformation across one time cycle for four propulsive modes: (a) Optimum; (b) α; (c) β; and (d) γ. Deformations of the midline in time are encoded every one-tenth of the period in the different shades of blue from lightest ($t = 0$) to darkest ($t = T$).

Table 11 reports on the hydrodynamic quantities of the optimum and α, β, γ modes with $N = 9$. In addition to the absolute quantities for each mode, Table

11 also documents the relative quantities of the sub-optimum modes normalized by the respective quantity of the optimum mode. In Table 11, A denotes the tail amplitude of a corresponding kinematic mode, with the other hydrodynamic quantities as defined previously. The results indicate that, indeed, a small tail amplitude ratio ($A_\gamma/A_{\text{opt}} \approx 0.33$) is associated with a reduced efficiency ($\eta_\gamma/\eta_{\text{opt}} \approx 0.05$), consistent with the findings in Triantafyllou, Triantafyllou, and Yue 2000. Regarding the relation of efficiency to the amplitude, it was previously established that the thrust coefficient scales with the amplitude squared, A^2, for the flapping foil propulsion (Theodorsen 1935; Garrick 1937; Moored and Quinn 2019), and these arguments were extended to derive the scaling laws for the fish locomotion (M. Lighthill 1969; Gazzola, Argentina, and Mahadevan 2014; Saadat et al. 2017).

However, quadratic scaling with the tail amplitude can not explain the efficiency of all the realizable modes. For example, if we look back at Table 11, we recognize that the optimum, α and β modes all have a comparable tail amplitude, however, their efficiency varies by as much as five times.

To illustrate this phenomenon, I consider the mode β, which has a tailbeat amplitude of $A_\beta/A_{\text{opt}} \approx 0.93$, however only one fifth of efficiency of the optimum mode ($\eta_\beta/\eta_{\text{opt}} \approx 0.23$). To understand where the efficiency losses might come from, I look at the total work and the useful work ratios of the mode β provided in Table 11. We see that the efficiency losses are not fully attributed to a larger amount of the total work ($W_{\text{total}_{\beta_r}} = 1.50$), but are instead mostly effected by a lower propulsive work ($W_{\text{useful}_{\beta_r}} = 0.34$). The lower propulsive work in the sub-optimum β-swimmer corresponds to a lower propulsive force ($\bar{F}_{x_{\beta_r}} = 0.58$) and a lower maximum speed ($U_{\max_{\beta_r}} = 0.62$). This is consistent with the results of the correlation analysis in Figure 11.

3.5.2.3 New Scaling Law for Optimum and Sub-optimum Propulsion

As seen from the previous section, while scaling of the thrust with the tailbeat square amplitude A^2 might explain the optimum propulsion between the species, it is not necessarily applicable to a sub-optimum motion, as the efficiency results might be very different for the same tailbeat amplitude. To understand this phenomenon, I look back at the shape of the midline deformation of the swimmer over the undulation cycle presented in Figure 12. It can be seen that, between the optimum, α and β modes, while the tail amplitudes are similar, there are marked differences in the body line kinematics. I note, for example, an existence of a "fixed point" in the midline deformation curves of sub-optimum modes, where the amplitude of motion is zero. This fixed point essentially separates the motion of the tail from the motion of the body. Since in thunniform swimming, most of the propulsion comes from the tail (Donley and Dickson 2000; Lauder and Tytell 2005; Fish et al. 2021), I argue that the existence of this fixed point essentially reduces the effective propulsive length of the fish from its full length L to a "flapping length" L_f, defined as the distance between the fixed point and the tip of the tail. Conferring with the equation of the midline deformation, Eq. (3.2), I identify the axial coordinate where the deformation is zero (fixed point) as

$$x_f = -\frac{c_1}{c_2} L, \tag{3.31}$$

and therefore I can define the flapping length as

$$L_f = (1 + \frac{c_1}{c_2}) L. \tag{3.32}$$

If the fixed point defined by Eq. (3.31) lies outside of the fish body (such as, in the optimum mode), I constraint the flapping length to be equal to the fish length $L_f = L$.

Following the scaling arguments presented by Gazzola, Argentina, and Mahadevan 2014, I define the mass of the fluid set in motion by the fish tail propulsion as ρL_f^2 per unit depth, acceleration of the fluid as $A\,\omega^2$, and the local angle between the tail and the direction of motion as A/L_f. Thus, I arrive at the thrust force scaling as $T \sim \rho\,\omega^2 L_f A^2$. Since drag is independent of the tail amplitude (Gazzola, Argentina, and Mahadevan 2014; Gibouin et al. 2018), propulsion velocity U is proportional to the net thrust, and the total work in the denominator of Eq. (3.26) roughly scales with the hydrodynamic forces, I can conclude that the efficiency η should obey the scaling $\eta \sim C_\eta\, L_f A^2$, where C_η is some dimensional constant of proportionality.

Indeed, using a standard propulsion law $\eta \sim\ L\, A^2$, I obtain a relative efficiency for the mode α with respect to the optimum mode as $\eta_{\alpha\,1} = 0.94$ instead of the observed $\eta_\alpha = 0.66$. However, scaling with $\eta \sim\ L_f\, A^2$ results in $\eta_{\alpha\,2} = 0.56$, which is much closer to the observed efficiency. Likewise, for the mode β, we would have $\eta_{\beta\,1} = 0.86$ from the original scaling law, and $\eta_{\beta\,2} = 0.18$ with the modified scaling, which is much closer to the actual $\eta_\beta = 0.23$.

Plotting a correlation of efficiency with LA^2 (unmodified propulsion law) in Figure 13a shows that a correlation with $L\,A^2$ is a reasonable rule of thumb ($R = 0.91$), but it still has a lot of outliers, such as the modes α and β discussed here. However, correlating efficiency with $L_f A^2$ in Figure 13b provides a perfect agreement with the correlation coefficient of $R = 0.99$.

This result, first, shows that simple fluid mechanics scaling laws, with a slight modification, are applicable to a wide range of realizable (prescribed) undulatory motions of the aquatic swimmer in the viscous flow, optimum as well as sub-optimum. And, second, it provides the guidance for the optimum design of the undulatory motions of a swimmer: while maximizing the tailbeat amplitude under allowable

constraints, it should also maximimze the propulsive area and avoid the fixed points in the body deformation curve.

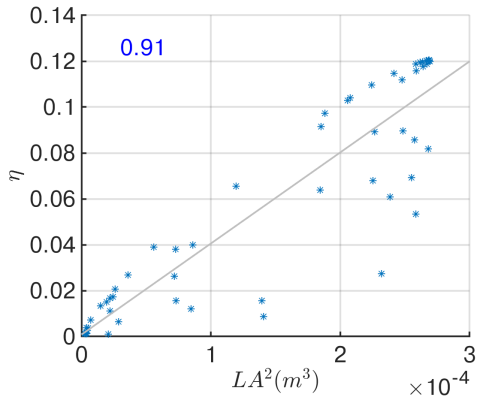
	η	W_{useful}(mJ)	W_{total}(mJ)	$U_{\max}$(m/s)	$\bar{F}_x$(N)	$\bar{F}_y$(N)	A(m)
opt	12.04%	4.411	36.64	0.101	0.044	0.080	0.030
α	7.99%	3.132	39.20	0.085	0.037	−0.054	0.029
β	2.74%	1.506	54.99	0.063	0.026	−0.024	0.028
γ	0.64%	0.040	6.25	0.010	0.004	−0.004	0.010

(a)

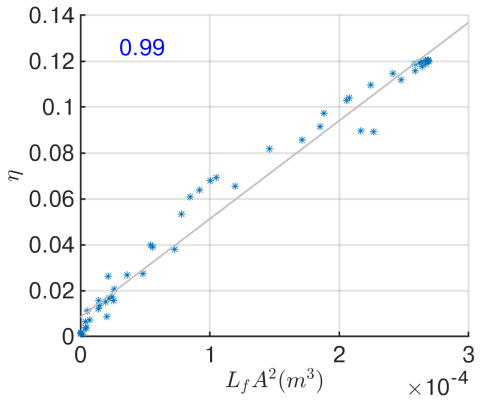
	η_r	W_{useful_r}	W_{total_r}	$U_{\max_r}$	$\bar{F}_{x_r}$	$\bar{F}_{y_r}$	A_r
α	0.66	0.71	1.07	0.84	0.84	−0.67	0.97
β	0.23	0.34	1.50	0.62	0.58	−0.30	0.93
γ	0.05	0.01	0.17	0.10	0.10	−0.05	0.33

(b)

Table 11: Hydrodynamic quantities of selected kinematic modes with $N = 9$: a) Absolute quantities, and b) Relative quantities of the sub-optimum modes normalized by the respective quantity of the optimum mode.



(a) Correlation versus LA^2



(b) Correlation versus L_fA^2

Figure 13: Correlation of efficiency η versus a) LA^2 and b) L_fA^2. Pearson correlation coefficient (R) in shown in blue inside each plot.

3.5.2.4 Vortex Wakes

I now analyze the vortex wakes produced by the four selected kinematic modes, to understand the relationships between the swimming efficiency and vorticity dynamics. The vortex wakes produced by the four modes are shown in Figure 14. I present two snapshots, at $t = T = 1$ s and $t = 2T = 2$ s, respectively, to show the dynamics of the pair of positive and negative vortices resulting from each stroke. For the optimum mode, after 1 s, the first pair of primary positive (P-1-P) and negative (P-1-N) vortices behind the optimum swimmer is advected at an angle, which is expected since the swimmer generates its first stroke from an equilibrium position. After 2 s, the P-1-P and P-1-N vortices are of comparable magnitude and they behave similarly to a pair of point vortices, which propel each other in the negative streamwise direction, thus propelling the fish in the positive streamwise direction. The second primary vortex pair (P-2-P and P-2-N) begins to shed at $t = 2$ s in a fashion similar to the reverse Karman street wake associated with high propulsive efficiency (Triantafyllou, Triantafyllou, and Grosenbaugh 1993; Triantafyllou, Triantafyllou, and Yue 2000; Van Buren, Floryan, and Smits 2018).

Out of the three additional modes considered, only mode α, second in rank with respect to efficiency after the optimum mode, results in a similar reverse Karman street. However, a smaller secondary negative vortex (S-1-N) is observed after the first stroke of the α, β and γ swimmers. Effect of the S-1-N vortex varies between the modes. For the α mode, S-1-N is first attached to P-1-P at 1 s but is observed to be isolated at 2 s. However for the β mode, the secondary vortex effects are more prominent, and in fact disrupt the wake by separating and redirecting of the primary vortices (note, for example, two instances of P-1-N and P-2-P vortices at

the end of the second cycle, which got split up and steered away from each other during the undulation). Despite having a sufficient tail amplitude ($A_{\beta_r} = 0.93$), mode β fails to produce a reverse Karman street wake and consequently results in a low propulsive efficiency. This might be associated with a lower effective flapping length, as discussed above. Mode γ leaves the wake with minute vortices due to small tail oscillations that lack a propulsive power. These small oscillations result in the lowest propulsive efficiency between the four modes ($\eta_{\gamma_r} = 0.05$). Since the secondary vortex formation is observed in all the considered modes except for the optimum one, I can conclude that secondary vortices act to disrupt the formation of an efficient reverse Karman vortex street and thus interfere negatively with the efficient propulsion. A negative effect on propulsion can also be attributed to a formation of the head vortices, especially noticeable in the two least efficient, β and γ, modes. While head vortices do not interfere with the wake, they presumably effect the hydrodynamic forces and the energy expenditure of the swimmer. Finally, it is worth noting that secondary tail vortex formation is only observed during the first cycle for all the modes, and might be associated with the onset of acceleration, while the shedding of the head vortices is noticed for both the first and the second cycles.

3.5.2.5 Strouhal Numbers

It has been previously observed that the stable existence of a wake conducive to an efficient propulsion is largely dependent on the non-dimensional oscillating frequency of the swimmer, or the Strouhal number, St (Triantafyllou, Triantafyllou, and Grosenbaugh 1993; Triantafyllou, Triantafyllou, and Yue 2000). If I consider the

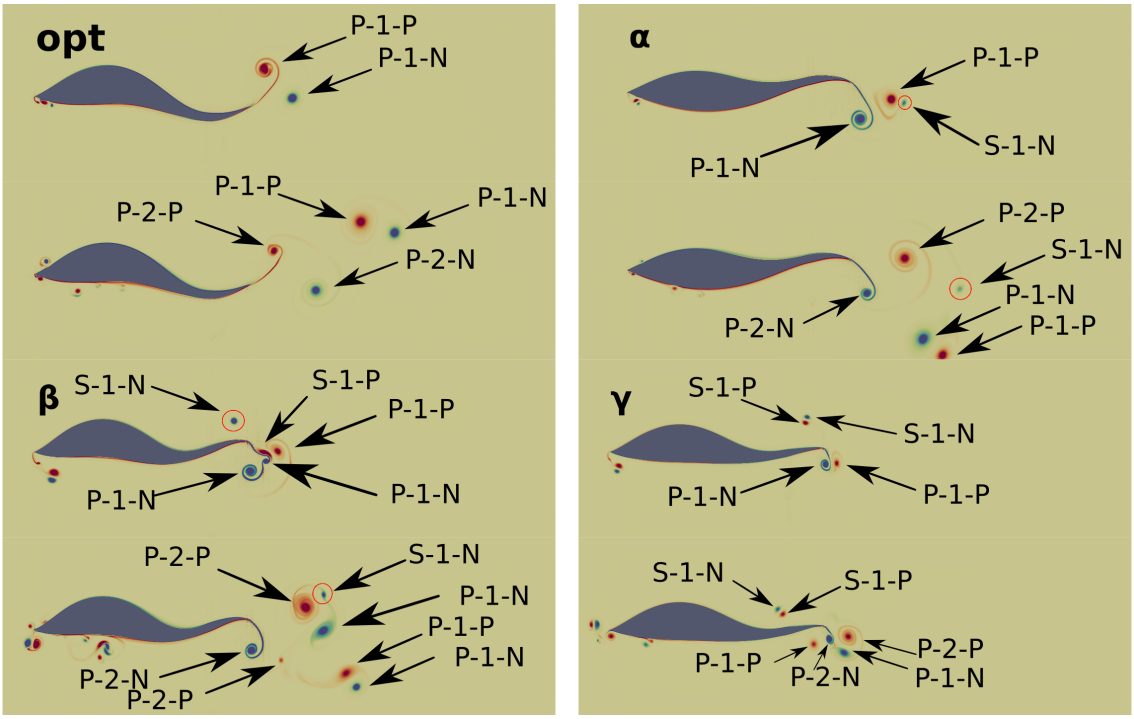


Figure 14: Vorticity (red – positive, blue – negative) in the wake of the fish for the four modes with $N = 9$. Maximum and minimum values of vorticity are 100 s^{-1} and –100 s^{-1}, respectively. Top row is at $t = T = 1$ s and bottom row is at $t = 2, T = 2$ s for each mode. Vortices are named with a 'A-B-C' convention, where A is either a 'P' primary or 'S' secondary vortex, B is an indication that a vortex appears after either the '1' first or the '2' second time cycle and 'C' is either a 'P' positive or 'N' negative vortex.

Strouhal number of each of the four modes, defined as:

$$St = fA/U_{\max} \tag{3.33}$$

where A is the tail amplitude and U_{max} is the velocity at the end of the $2\,T$ period, I arrive at:

1. $St_{\text{opt}} = 1 * 0.03/0.10 = 0.30$.
2. $St_{\alpha} = 1 * 0.029/0.085 = 0.34$.
3. $St_{\beta} = 1 * 0.028/0.063 = 0.44$.
4. $St_{\gamma} = 1 * 0.010/0.010 = 1$.

It is worth noting that the Strouhal numbers for the first three modes (optimum, α and β), which all exhibit a high tail amplitude, fall well within the St range $(0.25 - 0.45)$ associated with optimal propulsion in a wide range of experimental observations of various fish (Triantafyllou, Triantafyllou, and Grosenbaugh 1993; Akanyeti et al. 2017), despite having very different actual efficiencies within the three modes. It is only the fourth, the least efficient γ mode, with very small tail amplitude, which yields a St number clearly out of range for the optimal propulsion, due to a very low attained velocity. It might be that the optimal Strouhal scaling is only applicable to the undulation motions that follow the rules of the optimal propulsion (like the optimum mode here), and the realistic fishes typically swim in the near-optimal regime.

3.5.3 Comparison Between $N = 5$ and $N = 9$ Cases

A lower polynomial order $N = 5$ case is compared here to $N = 9$ case to establish grid convergence and to analyze the sensitivity of the optimization results to a polynomial degree in a high-order method.

3.5.3.1 Hydrodynamic Quantities

The correlations of the hydrodynamic quantities between $N = 5$ and $N = 9$ cases are presented in Figure 15. To compute the correlations, I interpolate the data obtained from the CFD evaluation points of $N = 9$ case to the evaluation points of $N = 5$ case yielded by the optimization procedure.

I observe an excellent agreement between the hydrodynamic data of $N = 5$ and $N = 9$ cases, with the correlation coefficient of $R \geq 0.99$. Among all presented

quantities, lift exhibits the most scatter, although it still yields a correlation coefficient of $R = 0.99$. The scatter is primarily in the region close to zero, where the actual value of the lift is so low, that it is most prone to numerical uncertainties.

Table 13 includes the hydrodynamic quantities for the optimum and the three selected sub-optimum modes, see Table 10, for $N = 5$ case, which can be compared with the corresponding values for $N = 9$ case presented in Table 11. A most notable difference is a slight upward shift of the efficiency values computed for all the four modes with $N = 5$, accompanied by (and perhaps attributed to) a slight downward shift of the total work. The other quantities, including propulsive and lift forces, useful work, and maximum velocity, do not show consistent deviations. The fact that the viscous forces are predicted well by the lower polynomial order solution but the total work deviates slightly might be an indication that it is the resolution of the interaction between the viscous forces and the local deformations of the moving surface, which defines the total work, that benefits the most from a higher-order approximation. Note that, since the resulting shift in efficiency is monotonic, the lower-order $N = 5$ approximation predicts the correct relative efficiency rankings between the modes, illustrates a perfect correlation with the high-order $N = 9$ case for all the hydrodynamic quantities considered, and identifies the same optimum mode, as seen from Table 9.

Table 12: Hydrodynamic quantities of chosen low order modes

Mode	η_r	W_{useful_r}	W_{total_r}	U_{max_r}	F_{x_r}	$\overline{L}_{f_r}$
α	0.24	0.33	1.38	0.59	0.57	-0.11
β	0.06	0.01	0.17	0.11	0.10	0.00
γ	0.66	0.67	1.05	0.83	0.83	-0.62

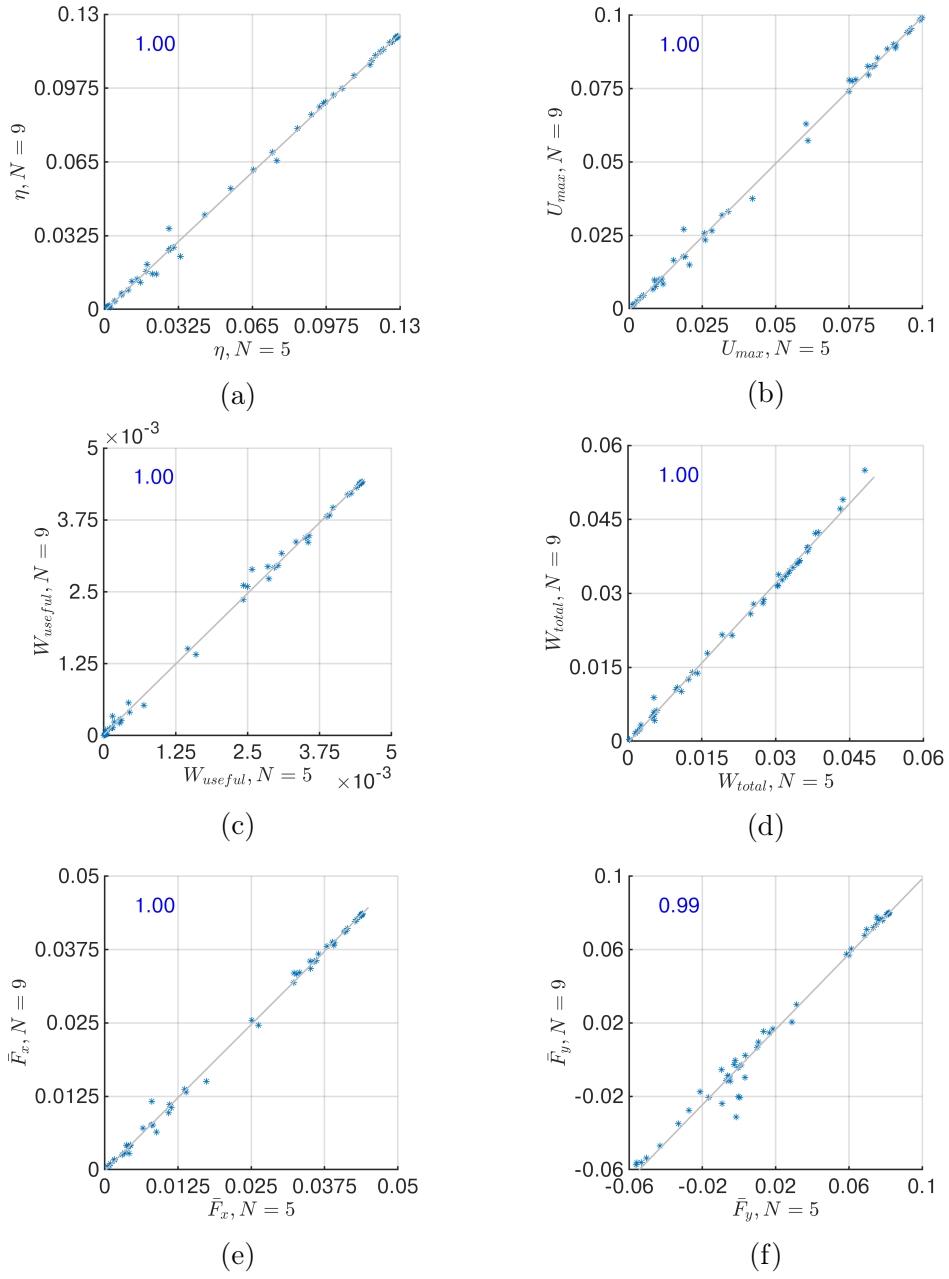


Figure 15: Correlation between hydrodynamic quantities obtained with $N = 9$ (y-axis) and $N = 5$ (x-axis) spectral element method. Pearson correlation coefficient is highlighted in blue on each plot.

The deviance from the reported efficiency values is attributed to marginal deviations in the propulsive work, $\mathrm{W_{useful}}$, which increases by $\approx 2\%$ and total work, $\mathrm{W_{total}}$, which

decreases by $\approx 5\%$. However, the deviations are mode dependent (for example mode α is shown to have a decrease of $\approx 3\%$ in its propulsive work), and the largest of which remains under $\approx 15\%$. The notable exception is the lift approximation of mode α. Even though large deviations are noted in this particular case, lift has not been observed to correlate with propulsive performance as the lift force, in the setup of this experiment, is generated in a direction that is perpendicular to the swimming direction. The persistence of the overall trends as well as the robustness of the optimum mode, between the different fidelity levels, across the allowable range of kinematic modes suggests that lower order approximations fail to capture small scale events which ultimately do not effect the choice of efficient gait parameters.

	η	W_{useful}(mJ)	W_{total}(mJ)	$U_{\max}$(m/s)	$\bar{F}_x$(N)	$\bar{F}_y$(N)
opt	12.90%	4.492	34.81	0.102	0.044	0.082
α	8.49%	3.091	36.42	0.085	0.037	−0.055
β	3.04%	1.462	48.80	0.060	0.025	−0.010
γ	0.75%	0.044	5.90	0.011	0.005	−0.004

Table 13: Hydrodynamic quantities of selected kinematic modes with $N = 5$.

3.5.3.2 Vortex Wakes

Figure 16 shows the wakes produced by the optimum mode and the mode α at $t = 1$ s and $t = 2$ s computed with $N = 5$ and $N = 9$ polynomial orders. For the optimum mode, at $t = 1$ s, the wakes behind each swimmer appear nearly identical. Towards the head, however, there are less small vortices resolved using a lower order approximation. At $t = 2$ s, a similar pattern is observed where the smaller vortices are missing at $N = 5$. The reverse von Karman street wake appears after the second

stroke both for $N = 5$ and $N = 9$ cases, and the first pair of vortices are able to propel each other due to closely matched circulation strengths in both cases. A better capture of small vortices in a higher-order approximation case is in line with the prediction of a higher total work of the optimum swimmer and, consequently, a lower maximum efficiency by $N = 9$ solution. Since these small vortices do not interfere with the wake, an almost identical kinematic gait mode is recovered as the optimum with both polynomial resolutions.

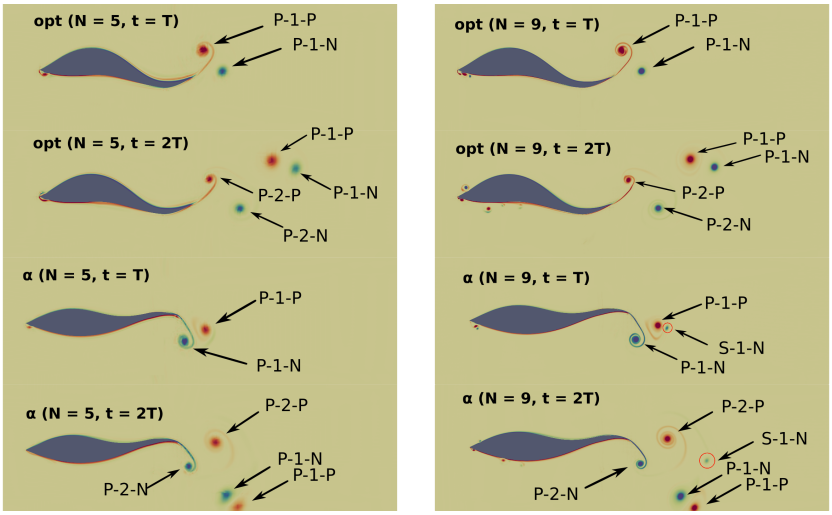


Figure 16: Vorticity (red – positive, blue – negative) in the wake of the fish for the optimum and α modes with $N = 5$ (left) and $N = 9$ (right). Maximum and minimum values of vorticity are 100 s^{-1} and –100 s^{-1}, respectively. The vortices follow the naming convention of Figure 14.

The wake of the α mode similarly remains unchanged with one notable exception: the negative secondary vortex (S-1-N) is less visible in $N = 5$ case at both time instances. Before $t = 1$ s, the smaller negative vortex (S-1-N) generates and stays attached to the primary positive vortex (P-1-P) as the swimmer moves from its static position to an upward stroke position (not shown here). As the swimmer completes its stroke, at $t = 1$ s, the secondary negative vortex (S-1-N) appears to have been

stretched around the positive vortex (P-1-P) and is barely visible. At $t = 2$ s, the first pair of vortices (P-1-N and P-1-P) float in the negative streamwise direction while the second pair (P-2-N and P-2-P) initiate a reverse von Karman street wake, as in the $N = 9$ case. Since the appearance of S-1-N vortex in α, β and γ modes was previously associated with a lower efficiency, its under-resolution in $N = 5$ case might, again, explain the increased efficiency trends with a lower polynomial order.

In the β mode, shown in Figure 17, while a weak S-1-N vortex can be seen at $t = 1$ s with $N = 5$, the overall vorticity pattern is notably simpler as compared to the $N = 9$ case, where a complicated pattern of vortex breakdown and merging was observed. While the S-1-N vortex is primarily dissipated by 2 s, its original formation still caused the primary vortex (P-1-P) to separate from the wake. Therefore, the β mode was similarly unable to produce an efficient reverse von Karman wake with $N = 5$. As before, a somewhat weaker S-1-N vortex and an under-resolution of a vortex breakdown with a low polynomial order resulted in a slightly higher predicted efficiency in $N = 5$ case. The same trend persists for the γ mode, where, while the overall wake structure is similar between $N = 5$ and $N = 9$ cases, the $N = 5$ case resolves significantly less details regarding the secondary vorticity both around the head and the tail of the fish, resulting in a slightly higher predicted efficiency.

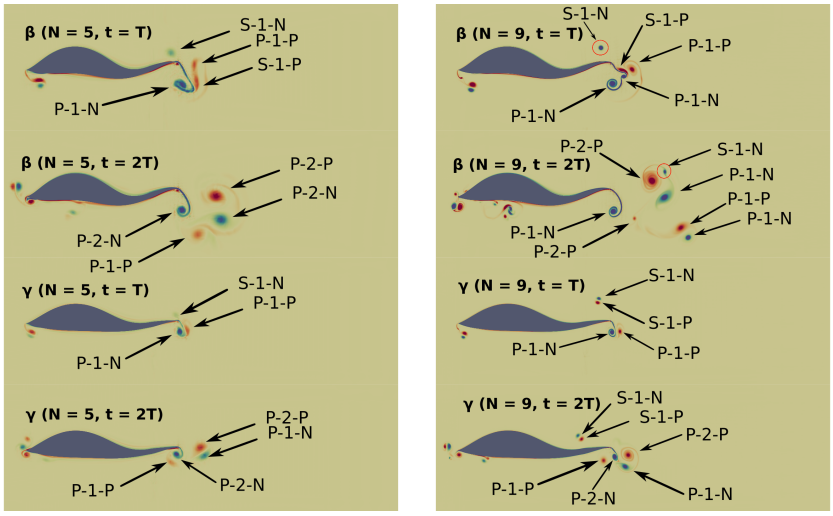


Figure 17: Vorticity (red – positive, blue – negative) in the wake of the fish for the β and γ modes with $N = 5$ (left) and $N = 9$ (right). Maximum and minimum values of vorticity are 100 s^{-1} and –100 s^{-1}, respectively. The vortices follow the naming convention of Figure 14.

Chapter 4

MODELING AND ANALYSIS OF OPTIMAL PHALANX SCHOOLS OF BIO-INSPIRED THUNNIFORM SWIMMERS

This chapter extends the physical and numerical modeling details of a solitary thunniform swimmer during start-up to accelerating schools of thunniform swimmers in phalanx-shaped schools during start-up. Multiple optimization studies are conducted to investigate the optimal kinematic gait, phase lag and separation distance within phalanx schools as the swimmers accelerate from rest. Additionally, an optimization study involving a solitary swimmer, with the same numerical set-up as with the phalanx schools, is presented for comparison. Section 1 includes the modeling details of different phalanx schools investigated. Section 2 describes how the optimization cases are set up. The results are discussed in section 3.

4.1 Methods

4.1.1 Physical Model

The swimmer model are similar to those of a single thunniform swimmer discussed in Section 3.1. However, for simulations of phalanx schools, a second swimmer, swimming with a phase lag, is introduced with the following midline kinematics:

$$y_m(x,t) = \left[c_0 + c_1\frac{x}{L} + c_2\left(\frac{x}{L}\right)^2\right]\sin(kx - \omega t + \phi), \tag{4.1}$$

where ϕ represents phase lag in radians. Self-propulsion is modeled by considering Newton's second law of motion for undulating swimmers, while calculating the corre-

sponding viscous and pressure forces directly from the fully-resolved simulations of the fluid-body interactions:

$$m\,\frac{d\,U(t)}{dt} = F'_x(t), \tag{4.2}$$

where m is the mass of the fish, $U(t)$ is the fish forward velocity and $F'_x(t)$ is the streamwise self-propelling force acting on the swimmers' body. For a solitary swimmer, the streamwise self-propelling force, $F_{x_s}(t)$, can be described as follows:

$$F_{x_s}(t) = \oint_{\Gamma_s} -(\sigma\,\mathbf{n})\cdot\mathbf{i}\,d\,\Gamma, \tag{4.3}$$

where σ is the total Cauchy stress tensor, which includes viscous and pressure contributions, Γ_s is the curvilinear boundary of the solitary swimmer, and $\mathbf{i}$ is the unit vector in the streamwise direction. For a phalanx school, the average streamwise self-propelling force, $F_{x_p}(t)$ is defined as follows:

$$F_{x_p}(t) = \frac{1}{2}\left(\oint_{\Gamma_{p_1}} -(\sigma\,\mathbf{n})\cdot\mathbf{i}\,d\,\Gamma + \oint_{\Gamma_{p_2}} -(\sigma\,\mathbf{n})\cdot\mathbf{i}\,d\,\Gamma\right), \tag{4.4}$$

where Γ_{p_1} is the curvilinear boundary of one phalanx swimmer with no phase lag (as defined in Eq. 3.2) and Γ_{p_2} is the curvilinear boundary of a second phalanx swimmer with phase lag (as defined in Eq. 4.1). $F'_x(t)$ can then be formally defined as:

$$F'_x(t) = \begin{cases} F_{x_s}(t), & \text{for a solitary swimmer} \\ F_{x_p}(t), & \text{otherwise} \end{cases} \tag{4.5}$$

In other words, the self propulsion speed, $U(t)$, is proportional to the average streamwise self-propelling force, $F_{x_p}(t)$, in a phalanx school. Only the contribution of the streamwise force, $F'_x(t)$, i.e. thrust and drag, is considered in regards to motion, and thereby the swimmer(s) are confined to move in the streamwise direction and do not travel in the lateral streamwise. Similar assumptions were made in previous studies as they can be beneficial for stability (Borazjani and Sotiropoulos 2010; Yu, Lu, and Huang 2021).

4.1.2 Numerical Model

The numerical modeling in the current study follows the method previously outlined in Section 3.2. Fluid-body interactions of the modeled swimmer are solved by considering the Arbitrary Lagrangian-Eulerian (ALE) formulation of the incompressible Navier Stokes (NS) Equation on a moving mesh (Ho 1989; Deville et al. 2002):

$$\rho\left(\frac{\delta \mathbf{u}}{\delta t} + (\mathbf{u} - \mathbf{w}) \cdot \nabla \mathbf{u}\right) + \nabla p = \mu \Delta \mathbf{u}, \tag{4.6a}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{4.6b}$$

where ρ, $\mathbf{u} = \{u_x, u_y\}$, p, μ and $\mathbf{w} = \{w_x, w_y\}$ are the fluid density, velocity, pressure, dynamic viscosity and the mesh velocity, respectively, while the derivative $\delta/\delta\, t$ represents the ALE derivative. A higher order spectral element method (Maday and Patera 1989; Deville et al. 2002) is used to solve the NS equations and further details on the temporal and spatial discretization are found in Section 3.2. Additionally, a predictor-corrector scheme is used to solve an implicit backwards Euler discretization of the self propulsion equation (Eq. 4.2):

$$U^n = U^{n-1} + \frac{\delta t}{m} {F'}_{x^n}, \tag{4.7}$$

where δt is the time step and ${F'}_{x^n} = F'_x(t^n)$ is the total propelling force acting on the swimmer(s) in the x direction at a time t^n. The fluid and fish density are both taken to be 1×10^3 kg/m^3, which corresponds to a neutrally-buoyant swimmer. Dynamic viscosity of the fluid, μ, is set to 1×10^{-3} kg/(m $\cdot$ s). The mass of a single swimmer m is specified as 0.8606 kg, which corresponds to the parameters of a soft robotic thunniform swimmer prototype (Valdivia y Alvarado 2007).

The computational domain is specified as a rectangle with dimensions $9.4L \times 16L$ for the solitary swimmer simulations and $9.4L \times 2SD$ for the phalanx school simulations

with SD being the separation distance between swimmers in a phalanx school. For solitary swimmer simulations, the fish leading edge is fixed a distance L away from the inlet and a distance $8L$ away from the right and left lateral boundaries. For phalanx school simulations, one swimmer is placed at $0.5SD$ away from the right boundary and the second swimmer is placed at $0.5SD$ away from left boundary, while both swimmers are fixed at a distance of L away from the inlet. I set the Gauss-Lobatto-Legendre (GLL) integration order (Deville et al. 2002) as $N_x = N_y = N = 6$. Fluid velocity at the fluid-body interface is set equal to the velocity of the moving boundary. A fluid velocity is prescribed at the inlet where the inlet velocity at each time step is set to U^n (Eq. 4.7). A pressure outflow boundary is used at the outlet. Symmetry conditions are used at the lateral boundaries for a solitary swimmer while periodic boundaries are used at the lateral boundaries in the phalanx schools to simulate an infinite array of swimmers. The initial conditions are similar to those discussed in Section 3.3.

4.2 Optimization Cases Set-up

The general optimization problem is stated as

$$\begin{aligned} &\text{maximize} \quad f(\mathbf{z}) \\ &\text{subject to} \quad \mathbf{z} \in \mathbb{R}^n, \end{aligned} \tag{4.8}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is the objective function, and $\mathbf{z} \in S \cap C$ is a vector of design parameters. The set $S \subseteq \mathbb{R}^n$ contains the n-dimensional search space and the set $C \subseteq \mathbb{R}^n$ contains a set of $m \geq 0$ inequality constraints, as presented in Section 3.4. For a solitary swimmer, the traveling wave amplitude coefficients, that is $\{c_0, c_1, c_2\}$, along with undulation frequency f, make up the design parameters, $\mathbf{z}_s = \{c_0, c_1, c_2, f\}$.

The set, S_s in this case is given by:

$$\begin{aligned} -0.0500L \leq c_0 \leq 0.0500L, \\ -0.1000L \leq c_1 \leq 0.8333L, \\ -0.7333L \leq c_2 \leq 0L, \\ 0 \leq f \leq 3. \end{aligned} \tag{4.9}$$

In order to allow for only physically realizable modes as deduced from biological data, the following constraint set, C_s, is imposed (Du et al. 2015; Valdivia y Alvarado 2007; Borazjani and Sotiropoulos 2010):

$$\begin{aligned} g_1(\mathbf{x}) &= c_0 + c_1 + c_2 - 0.1L \leq 0, \\ g_2(\mathbf{x}) &= -c_0 - c_1 - c_2 - 0.1L \leq 0, \\ g_3(\mathbf{x}) &= c_0 - \frac{c_1^2}{4c_2} - 0.1L \leq 0, \\ g_4(\mathbf{x}) &= -c_0 + \frac{c_1^2}{4c_2} - 0.1L \leq 0. \end{aligned} \tag{4.10}$$

The propulsive efficiency of a solitary swimmer, η_s, is taken to be the objective function, $f_s(\mathbf{z}_s)$. The propulsive efficiency is defined as the ratio of a "useful" energy gained over the total work done by the swimmer over a certain time period (Webb 1975; Tytell and Lauder 2004; Akanyeti et al. 2017), which can be stated as

$$\eta_s(c_0, c_1, c_2, f) = \frac{W_{\text{useful}_s}}{W_{\text{total}_s}} = \frac{\int_0^{2T} \left(\oint_{\Gamma_{f_s}} -(\sigma\,\mathbf{n}) \cdot \mathbf{i}\, d\,\Gamma \right) U(t) d\,t}{\int_0^{2T} \oint_{\Gamma_{f_s}} -(\sigma\,\mathbf{n}) \cdot \mathbf{v}(x,y,t)\, d\,\Gamma d\,t} \tag{4.11}$$

where $T = 1s$ and $\mathbf{v}(x, y, t)$ is the swimmer surface velocity due to undulation.

For phalanx schools, the design parameters include an extra dimension to account for phase lag, $\mathbf{z}_p = \{c_0, c_1, c_2, f, \phi\}$. Note that all swimmers in a phalanx school have the same amplitude coefficients and undulation frequency, which allows for definitions

Table 14: Optimization Results

Case	Evaluations	Optimal Parameter Set	Efficiency	Tolerance
SD = 1L	51	$\{0.0001L, 0.2096L, -0.1107L, 3, 1.037\pi\}$	21.49%	4.8×10^{-4}
SD = 2L	42	$\{0.0L, 0.2105L, -0.1107L, 3, 0.9995\pi\}$	22.05%	9.9×10^{-4}
SD = 3L	39	$\{-0.0001L, 0.2104L, -0.1108L, 2.9998, 0.9613\pi\}$	21.83%	2.4×10^{-4}
solitary	34	$\{0.0L, 0.2105L, -0.1107L, 3, 0.9995\pi\}$	21.73%	4.0×10^{-4}

of averaged quantities within a school, such as Equation 4.4. Set S_p, in a school, is given by:

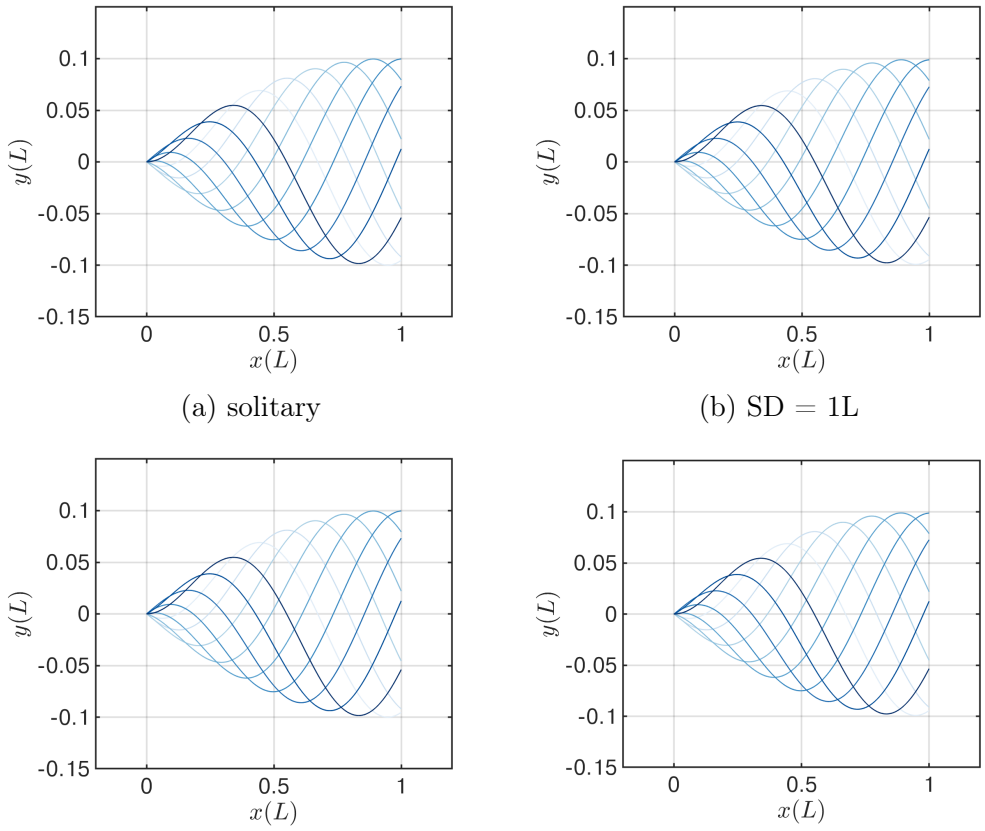
$$\begin{aligned} -0.0500L \leq c_0 \leq 0.0500L, \\ -0.1000L \leq c_1 \leq 0.8333L, \\ -0.7333L \leq c_2 \leq 0L, \\ 0 \leq f \leq 3, \\ 0 \leq \phi \leq 2\pi, \end{aligned} \tag{4.12}$$

Set C_p in this case is equal to C_s (Eq. 4.10). The objective function, $f_p(\mathbf{z}_p)$ is taken to be the average propulsive efficiency of a phalanx school:

$$\eta_p(c_0, c_1, c_2, f, \phi) = \frac{W_{\text{useful}_p}}{W_{\text{total}_p}} = \frac{\int_0^{2T} \left(\oint_{\Gamma_{p_1}} -(\sigma\,\mathbf{n}) \cdot \mathbf{i}\, d\,\Gamma \right) U(t) d\,t + \int_0^{2T} \left(\oint_{\Gamma_{p_2}} -(\sigma\,\mathbf{n}) \cdot \mathbf{i}\, d\,\Gamma \right) U(t) d\,t}{\int_0^{2T} \oint_{\Gamma_{p_1}} -(\sigma\,\mathbf{n}) \cdot \mathbf{v}(x,y,t)\, d\,\Gamma d\,t + \int_0^{2T} \oint_{\Gamma_{p_2}} -(\sigma\,\mathbf{n}) \cdot \mathbf{v}(x,y,t)\, d\,\Gamma d\,t} \tag{4.13}$$

A total of four optimization cases are considered to represent three phalanx schools ($\mathbf{z} = \mathbf{z}_p, f(\mathbf{z}) = f_p(\mathbf{z}_p), S = S_p, C = C_p$) and a solitary swimmer ($\mathbf{z} = \mathbf{z}_s, f(\mathbf{z}) = f_s(\mathbf{z}_s), S = S_s, C = C_s$). The phalanx school cases only differ in respect to SD, which is varied between $1L$, $2L$ and $3L$, respectively. The optimization problem is solved with the constrained SBO algorithm introduced and bench-marked in Chapter 2. 30 data points are chosen to construct the initial surrogate for all cases. Additionally,

the maximum iterations k_{max} and the tolerance termination criteria, $\varkappa_{opt}$ are set to 1000 and 1×10^{-3}, respectively.



(a) solitary (b) SD = 1L

(c) SD = 2L (d) SD = 3L

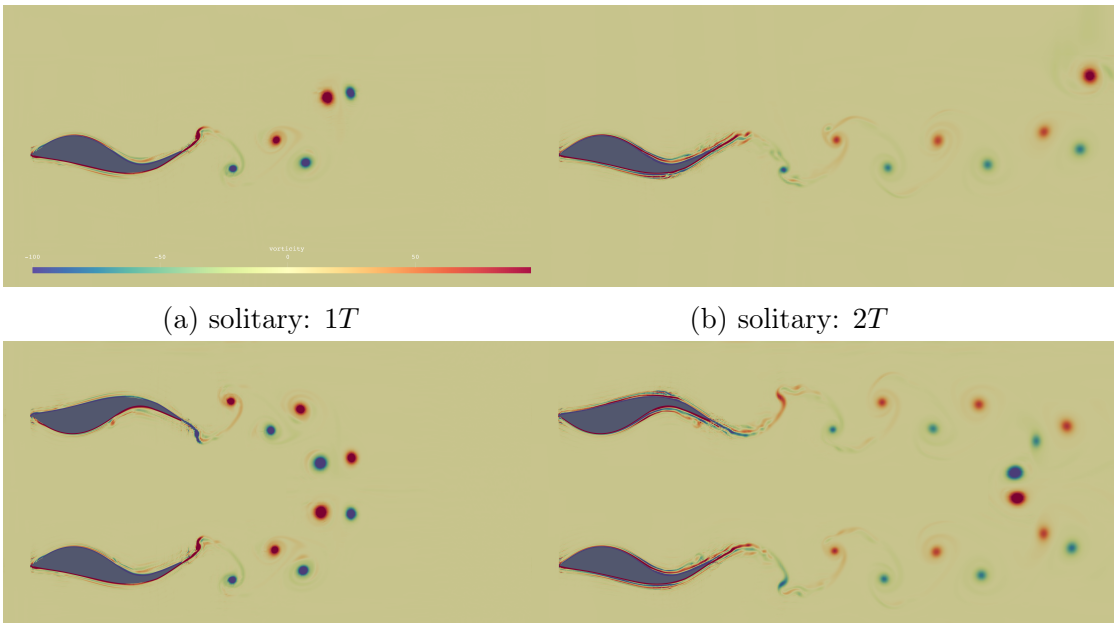
Figure 18: Swimmer midline deformation across one period for four optimum propulsive modes: (a) solitary (b) SD = 1L; (c) SD = 2L; and (d) SD = 3L. Deformations of the midline in time are encoded every one-tenth of the period in the different shades of blue from lightest ($t = 0$) to darkest ($t = 1/3$).

4.3 Results

4.3.1 Optimization Results

All optimization case results can be found in Table 14. All cases terminated within tolerance with varying total evaluation counts with ranged from 34 to 51 evaluations, with the first 30 coming from the initial sampling scheme and 4-21 optimization iterations. The first three parameters related to midline deformations, namely $\{c_0, c_1, c_2\}$, are found to be closely similar to that of the optimum solitary swimmer. In other words, all three fish pair cases report similar optimum kinematics to that of a solitary swimmer (Figure 18). These kinematics show no noticeable head motion along with growing body undulations to reach a tail amplitude of roughly $0.1L$. The $0.1L$ is the maximum allowable tail amplitude according to the present constraints (Eq. 4.10). Similarly the optimum undulation frequency remains close to the upper bound of 3 Hz for the three schools. The optimum phase lag parameter, ϕ, is shown to be close to π, which represents an antiphase lag between swimmers in an infinite array of a phalanx fish school. The vorticity field created by the solitary and $SD = 1L$ optimum swimming modes is shown after $1T$ and $2T$ in Figure 19. The wakes behind optimum swimmers are quite similar to the reverse von Karman street typically associated with high propulsive swimming efficiency in real fish (Triantafyllou, Triantafyllou, and Yue 2000), with no significant interactions noted between the wakes of swimmers in the phalanx school.

The reported optimum efficiency is slightly different for each fish school, where the highest reported efficiency, $\eta^*_{2L} = 22.05\%$, is found at a SD of 2L. The dense school ($SD = 1L$) and the sparse school ($SD = 3L$) report lower optimum efficiencies



(a) solitary: $1T$ (b) solitary: $2T$

(c) $SD = 1L$: $1T$ (d) $SD = 1L$: $2T$

Figure 19: Vorticity (color bar shown in solitary: $1T$ subfigure) in the wake of the swimmers for the optimum mode in solitary swimming (top row) and the $SD = 1L$ phalanx school (bottom row) after $1T$ (left) and $2T$ (right), respectively.

of $\eta^*_{1L} = 21.49\%$ and $\eta^*_{3L} = 21.83\%$, respectively. In contrast to a solitary swimmer, I observe that phalanx fish schools with $SD = 2L$ ($\frac{\eta^*_{2L} - \eta^*_s}{\eta^*_s} \approx 1.5\%$) and $SD = 3L$ ($\frac{\eta^*_{3L} - \eta^*_s}{\eta^*_s} \approx 0.5\%$) are slightly more efficient, while the $SD = 1L$ school ($\frac{\eta^*_{1L} - \eta^*_s}{\eta^*_s} \approx -1.1\%$) is slightly less efficient.

3-D stem plots of optimization cases is shown in Figure 20. 3-D space is used to represent each data point according to it's respective parameter value and is colored by it's respective propulsive efficiency value. Consequently, multiple stem plots are presented for each case to account for all four and five dimensions for solitary and school swimming, respectively. For the phalanx school cases, data points with the highest efficiencies are located within one region of the 3-D space in each stem plot. This optimal efficiency region is characterized by a small head motion ($c_0 \approx 0$), a high frequency ($f \approx 3$ Hz) and a phase lag, $\phi = \pi$, commonly referred to as antiphase

swimming. In other words, efficient swimmers in all three phalanx schools rarely deviated from kinematics exhibited by their respective optimum mode.

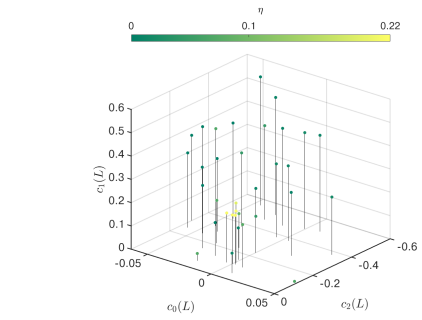
I define the maximum attainable swimming speed at the end of the considered period $2\,T$ as $U_{max} = U(t = 2\,T)$. Additionally, I define the time-averaged streamwise $\bar{F}'_x$ acting on the swimmer:

$$\bar{F}'_x = \frac{1}{t_0} \int_0^{t_0} F'_x(t)\, d\, t \tag{4.14}$$

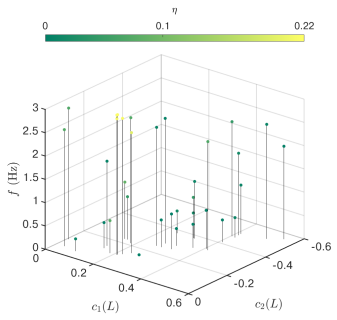
with $t_0 = 2\,T$. I formally define η, W_{useful} and W_{total} as:

$$\eta = \begin{cases} \eta_s, & \text{solitary} \\ \eta_p, & \text{school} \end{cases}, \quad W_{\text{useful}} = \begin{cases} W_{\text{useful}_s}, & \text{solitary} \\ W_{\text{useful}_p}, & \text{school} \end{cases}, \quad W_{\text{total}} = \begin{cases} W_{\text{total}_s}, & \text{solitary} \\ W_{\text{total}_p}, & \text{school} \end{cases}. \tag{4.15}$$

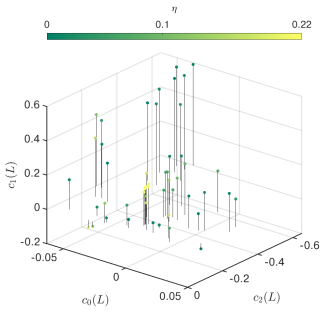
where 'solitary' and 'school' refer to quantities relevant to solitary and phalanx school swimming, respectively. Table 15 includes hydrodynamic quantities such as W_{useful}, W_{total}, $\bar{F}'_x$ and U_{max} for the four optimum swimming modes.



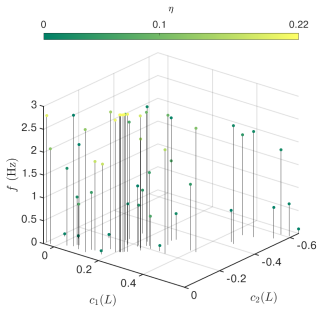
(a) s: $c_0 - c_1 - c_2$



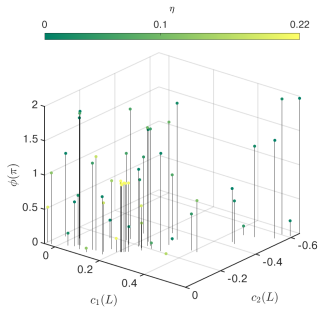
(b) s: $c_1 - c_2 - f$



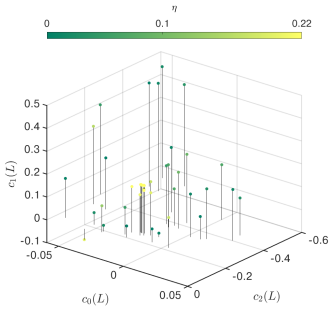
(c) $SD = 1L$: $c_0 - c_1 - c_2$



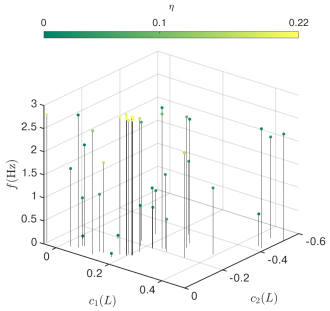
(d) $SD = 1L$: $c_1 - c_2 - f$



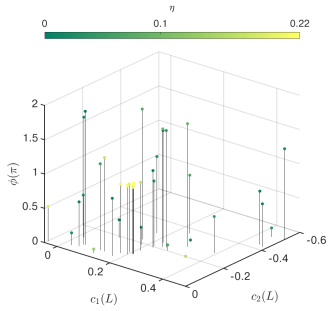
(e) $SD = 1L$: $c_1 - c_2 - \phi$



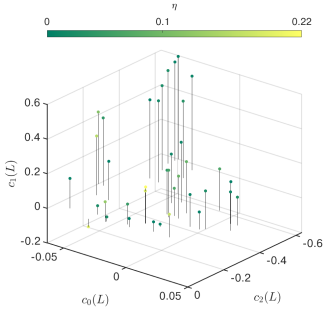
(f) $SD = 2L$: $c_0 - c_1 - c_2$



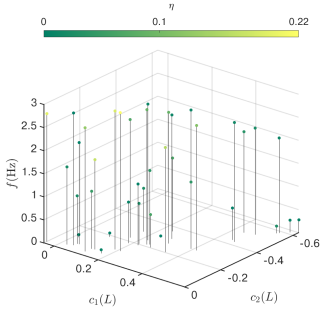
(g) $SD = 2L$: $c_1 - c_2 - f$



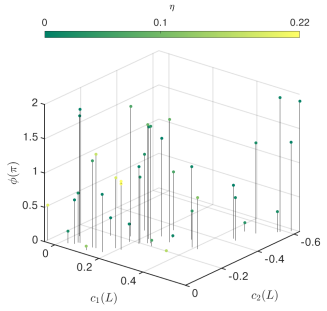
(h) $SD = 2L$: $c_1 - c_2 - \phi$



(i) $SD = 3L$: $c_0 - c_1 - c_2$



(j) $SD = 3L$: $c_1 - c_2 - f$



(k) $SD = 3L$: $c_1 - c_2 - \phi$

Figure 20: 3-D stem plots of the four dimensions for a solitary swimmer (first row) and five dimensions for phalanx schools with $SD = 1L$ (second row), $SD = 2L$ (third row) and $SD = 3L$ (fourth row).

4.4 Discussion

4.4.1 Effects of Body Kinematics on the Propulsive Efficiency of a Solitary Swimmer

The optimum undulation of a solitary thunniform swimmer, shown in Figure 18, differs slightly from the reported midline kinematics for solitary thunniform swimmers in other studies (Shadwick et al. 1999; Di Santo et al. 2021). While in any case there is no significant head motion, the case presented here shows growing body and tail undulations, while thunniform swimmers are shown to maintain a relatively straight body and only using their tail for propulsion (Di Santo et al. 2021; Li, Liu, and Su 2017). This is because most previous studies (Shadwick et al. 1999; Di Santo et al. 2021; Li, Liu, and Su 2017; Knower et al. 1999; Shadwick and Syme 2008) focus on steady thunniform swimming, with unsteady swimming trends gaining traction only recently. For example one previous study investigated the acceleration of a solitary thunniform swimmer from rest (Thandiackal et al. 2021). Investigation of the pressure forces around a solitary accelerating robotic tuna suggests that the posterior main body generates a significant portion of forward thrust as adjacent fluid is pushed backwards. This mechanism is similar to drag-based propulsion mechanisms found in anguilliform swimming and hence the study suggests that solitary thunniform swimmers may adopt an anguilliform-like motion as they accelerate from rest. Indeed, current optimization results are in line with this trend, with the optimal point falling in an optimum region in the $c_0 - c_1 - c_2$ space which is characterized by large posterior body and tail undulations (see Figure 18).

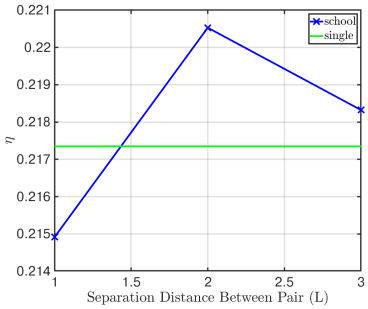
Table 15: Hydrodynamic Quantities of Optimum Swimmers

Case	W_{useful} (mJ)	W_{total} (mJ)	$\bar{F}'_x$ (N)	$U_{\max}$(m/s)
SD = 1L	184.11	856.67	0.28139	0.65205
SD = 2L	183.69	833.97	0.28108	0.65127
SD = 3L	181.78	832.64	0.27961	0.64779
solitary	181.19	833.68	0.27916	0.64899

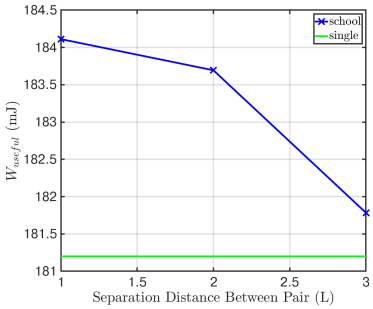
4.4.2 Effects of Body Kinematics on the Propulsive Efficiency of Phalanx Schools

To understand the similarity in the optimum body kinematics between all three phalanx schools and the solitary swimmer, the main energy saving mechanisms in phalanx fish schools are discussed, namely: channeling and pulsating jet effects. When swimmers in a phalanx school are within close lateral proximity to one another, an area of augmented flow develops between them. This augmented flow arises as a result of each swimmer generating a velocity field in its swimming direction due to the no-slip condition. Since the augmented velocity field imparts momentum in the swimming direction, swimmers need less thrust force to reach a specific swimming speed (Daghooghi and Borazjani 2015). This is typically referred to as the channeling effect (Weihs 1973) and can been observed in fish schools which range from phalanx (Intesaaf Ashraf et al. 2017) to rectangular (Daghooghi and Borazjani 2015) and diamond (Pan and Dong 2020) schools. When swimmers in a phalanx school swim with an antiphase motion, counter rotating vortices are shed by the two neighbors in each half period. These counter rotating vortices combine to form dipoles, producing a pulsating jet behind the swimmers (Godoy-Diana et al. 2019). While in phase swimmers can similarly benefit from the pulsating jet affect, antiphase swimmers produce higher thrust by creating a more compact jet (Godoy-Diana et al. 2019).

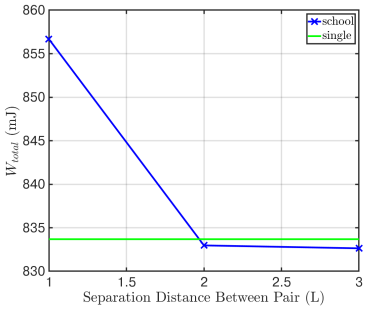
Therefore, it can be observed that swimming with a higher undulation frequency will enhance both energy saving mechanisms, since higher undulation frequency will result in faster flow around each swimmer (Thandiackal et al. 2021) as well as a higher frequency pulsating jet in their wake. Similarly, high body and tail amplitudes result in faster flow and larger vortices (See Section 3.5). It is then of no surprise that the optimal midline kinematics in an accelerating phalanx school exhibit similar behaviour to that of a solitary swimmer including a maximized tail displacement as well as an undulation frequency.



(a) η^* versus SD



(b) W_{useful} versus SD



(c) W_{total} versus SD

Figure 21: Optimum Efficiency, W_{useful} and W_{total} as a function of separation distance within an infinite phalanx school.

4.4.3 Effects of Separation Distance on the Propulsive Efficiency of Phalanx Schools

Figure 21 shows plots of the optimal propulsive efficiency of each fish school optimization case as a function of the separation distance within a school. Additionally, the two work quantities, namely W_{useful} and W_{total}, are included to investigate what is driving the changes in efficiency across different separation distances. As previously noted, the optimal efficiency of a phalanx school peaks at a separation distance of $2L$ with a value of 22.05%, which amounts to a small 1.5% increase over the solitary swimmer optimum. This increase is largely driven by a higher useful work quantity $\frac{W_{\text{useful}_{2L}} - W_{\text{useful}_s}}{W_{\text{useful}_s}} \approx 1.4\%$. The relative change in total work is $\frac{W_{\text{total}_{2L}} - W_{\text{total}_s}}{W_{\text{total}_s}} \approx 0.03\%$, which is an order of magnitude lower than the relative change in W_{useful}. Since the optimum mode uses an antiphase gait, the increased W_{useful} could likely be attributed to the presence of a pulsating jet behind the swimmers. The presence of a pulsating jet behind the swimmers would increase their time-averaged streamwise force, $\bar{F}'_x$, (as seen in Table 15) and consequently contribute to more work in the swimming direction or W_{useful}. Indeed, W_{useful} is shown to increase with a decreasing separation distance across the three schools which all employ the same midline and phase kinematics. This trend suggests that the increase in W_{useful} is directly related to thrust enhancing mechanisms, such as the pulasting jet and channeling effects, in the phalanx school.

While the school with a $SD = 1L$ results in the highest W_{useful}, it's propulsive efficiency is the lowest, even when compared to a solitary swimmer. This happens because W_{total} similarly increases to reach the highest value between all phalanx schools and the solitary swimmer. Since the direction of swimming and the separation distance is fixed within each school, swimmers in a school would have to exert additional effort, if needed, to maintain the same fixed distance during swimming. While this may not

present a challenge at more sparse schools, such as $SD = 2L$ and $SD = 3L$, this can result in an increased W_{total} in dense schools ($SD \leq 1L$). Indeed, the challenge of dense phalanx schools has been investigated in previous studies. For example, Hemerijk *et al.* observed a deteriorated steady group swimming efficiency, when compared to solitary swimming, for simulated mullets in phalanx schools with $SD < 1L$ (Hemelrijk et al. 2015). The authors' comment:"This is probably due to an increased resistance (on average per fish) of the phalanx to oncoming flow due to close proximity of lateral neighbours". Gazolla *et al.* performed simulations, using a vortex particle method, of multiple phalanx schools consisting of pairs of 2-D self-propelled anguilliform swimmers (Gazzola et al. 2011). In these simulations, where swimmers were free to move in both streamwise and lateral directions and only their displacement (undulation) was fixed, the pairs in a phalanx school diverged after 8 periods. In 3-D, a bigger phalanx school (consisting of 5 members) diverged at an earlier time of 6 periods. Both these studies suggest that swimming in an infinite phalanx school requires active adjustment as well as added effort to maintain a fixed separation distance, or at the least cohesion, within a school. Inviscid models of rigid wings in steady flow similarly suggest that while densely packed phalanx schools ($SD = 0.66L$) show an 5% increase in the swimming speed over a solitary swimmer (similar qualitative trends can be observed for U_{max} in Table 15), the associated cost of transport increases by 4% (Oza, Ristroph, and Shelley 2019).

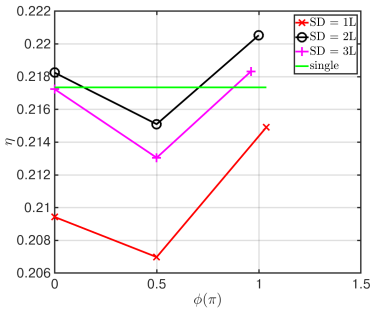
Finally the most sparse school ($SD = 3L$) displayed similar propulsive trends to the solitary swimmer, where the η, W_{useful} and W_{total} quantities remaining relatively unchanged. This is unsurprising since all the hydrodynamic quantities are expected to approach the solitary swimmer limit with increasing separation distance within a school (Oza, Ristroph, and Shelley 2019). The presented optimization cases at different

separation distances suggest the presence of three schooling regimes of accelerating phalanx thunniform swimmers in the separation distance region characterized by $1L \leqslant SD \leqslant 3L$:

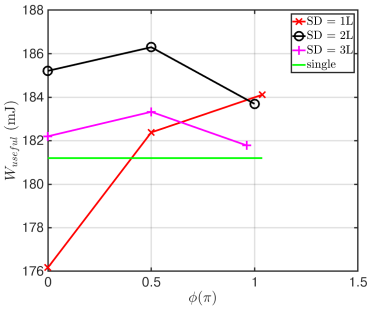
1. $1L \leqslant SD < 2L$: highest thrust, highest W_{useful} and highest W_{total} in a phalanx school can be achieved compared to solitary swimming. However, propulsive efficiency could suffer in a dense school.
2. $SD \approx 2L$: improved thrust, improved W_{useful} and relatively constant W_{total} compared to solitary swimming. Highest gains in efficiency, when compared to a solitary swimmer, as the school benefits from energy saving mechanisms without providing additional work to maintain it's position.
3. $2L < SD \leqslant 3L$: relatively constant thrust, W_{useful}, W_{total} and η as compared to solitary swimming as the school begins to approach the solitary swimmer limit.

4.4.4 Effects of Phase Synchronization on the Propulsive Efficiency of a Phalanx School

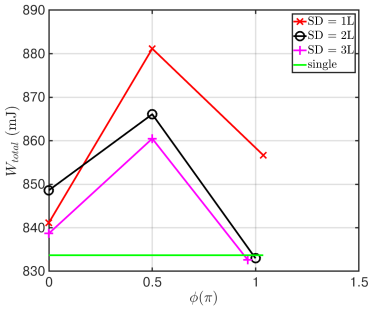
To isolate the effects of phase synchronization, I compare the optimal point, in each case, to two other points where all other kinematics $\{c_0, c_1, c_2, f\}$ are fixed to optimum values and the phase is changed to 0 and 0.5π, respectively, as shown in Figure 22. Between all three schools, antiphase motion ($\phi = \phi^* \approx \pi$) consistently shows the best swimming efficiency, with the inphase motion ($\phi = 0$) being second best and the $\phi = 0.5\pi$ being the worst. This is in line with previous studies which investigated phase synchronization behaviour in steady swimming of self-propelled foils. For example, Raspa, Godoy-Diana and Thiria showed inphase motion (resulting



(a) η^* versus ϕ



(b) W_{useful} versus ϕ



(c) W_{total} versus ϕ

Figure 22: Optimum Efficiency, W_{useful} and W_{total} as a function of phase lag and separation distance within an infinite phalanx school.

in asymmetric flow) leads to higher transverse velocity fluctuation, when compared to motion which results in symmetric flow (arising from antiphase swimming) (Raspa, Godoy-Diana, and Thiria 2013). As a result, for a given momentum input, the antiphase motion generated more thrust. A similar study with flexible foils showed that while inphase motion was shown to benefit swimmers from the pulsating jet effect, it results in a wake where the average propulsive jet is much more laterally spread, when compared to the wake of an antiphase motion, and thus does not contribute efficiently to propulsion (Godoy-Diana et al. 2019). This effect was also observed for pairs of real fish in steady swimming (Ashraf et al. 2016) where the was a slight preference for antiphase motion over inphase. Additionally, a preference of antiphase

and inphase motion over an intermediate phase lag (such as $\phi = 0.5\pi$) has been previously shown in steady swimming simulations of tetra fish (G. Li et al. 2019).

In the $SD = 2L$ and $SD = 3L$ schools, the most efficient phase lag does not necessarily result in high W_{useful}, indeed it is the lowest for $\phi = \phi^*$, when compared to other ϕ values. Instead the most efficient phase lag reduces the W_{total} for the school. As previously discussed, inphase motion results in more adverse flow conditions by introducing higher velocity fluctuations. Consequently, a more adverse flow field would require higher work input to maintain cohesion within a phalanx school and hence can contribute to an increased W_{total}. Alternatively, the most efficient phase in the $SD = 1L$ school results in the highest W_{useful}. This is likely due to phase synchronization effects playing a larger role in dense schools, when compared to sparse schools, and can be further explored in a future study.

Finally, I observe that the relative ranking of efficiency based on separation distance within a school does not change regardless of phase lag, with the most preferable distance remaining $SD = 2L$. This suggests that the separation distance within a phalanx school plays a larger role in the schooling efficiency than phase synchronization, for the parameters explored in this study.

Chapter 5

FUTURE WORK

Diamond-shaped schools generated much interest in fields such as bio-locomotion and flow physics. Using a two-dimensional inviscid flow model, Weihs 1973 showed that fish swimming diamond-shaped configurations can exploit the wake of upstream swimmers to their advantage, thereby enhancing the hydrodynamic performance of the entire school. Subsequent viscous numerical simulations showed that bio-inspired swimmers can improve their efficiency, up to 20% when compared to solitary swimming, when swimming in a diamond shaped formation (Hemelrijk et al. 2015). Although multiple studies (Hemelrijk et al. 2015; G. Li et al. 2019) emphasize the importance of free control available to the swimmers to be able to adjust their kinematic gain or phase to exploit the background flow, generated by diamond formations, few studies actually attempt to investigate the effects of free control. One such example is the work of Verma, Novati, and Koumoutsakos 2018 where, through a reinforcement learning policy, a single 'small' follower exploits the wake of a single leader. To fill this gap, I am investigating the effects of free control of the kinematic gait and phase synchronization of swimmers swimming in diamond-shaped schools. Optimization experiments are currently being conducted to characterize and elucidate the optimized hydrodynamics of diamond-shaped schools of swimmers swimming in both accelerating and steady flows.

Chapter 6

CONCLUSIONS

A computational framework for the design optimization of bio-inspired locomotive devices has been developed. This framework combines the use of an efficient constrained SBO algorithm and high fidelity spectral fluid simulations to investigate the optimal swimming motion and synchronization behaviour in solitary and group swimming. Three main phases describe the contribution and novelty of the current work.

The first phase starts with a careful investigation of different classes of optimization algorithms. The purpose of the investigation was to compare the general performance of 5 state-of-the-art gradient-based algorithms to 9 state-of-the-art gradient-free algorithms on a constrained and an unconstrained optimization problem. The results showed that the SBO family of the gradient-free algorithms were the most appropriate when the design objective function is an expensive "black-box" function. Additionally, a new constrained SBO algorithm, OK-CMSRS, was developed and bench-marked to asses it's appropriateness to the current design problem. The bench-mark included a total of 5 constrained optimization algorithms tested on 6 analytical constrained problems. Results from the analytical benchmarks show that the OK-CMSRS algorithm is a promising tool capable of insuring strictly feasible candidates while maintaining competitive convergence properties when dealing with expensive simulations-based "black-box" optimization problems with inexpensive constraints.

The second phase focuses on the development and coupling of FSI simulations of a solitary self-propelled swimmer to the OK-CMSRS SBO algorithm. An iterative implicit FSI scheme is shown to yield stable simulations of a swimmer swimming

in place using its self-propulsion velocity. The stability of the simulations is further enhanced by introducing a novel volume conserving scheme of the swimmer to maintain a constant volume without having to change it's midline deformations. This volume conserving scheme enables faster and more accurate predictions of the flow field around a self-propelled swimmer. By coupling the SBO algorithm to the FSI simulations, interesting new insights around the optimal kinematic behaviour of an accelerating solitary swimmer are discovered. For example, the optimal kinematic gait of a solitary accelerating swimmer results in a reverse von Karman street wake, which is typically associated with efficient propulsion in steady state fishlike swimming (Triantafyllou, Triantafyllou, and Yue 2000). Moreover, a new propulsive efficiency scaling law, based on the "effective" flapping length, is presented. The scaling law suggests that an efficient bio-inspired swimmer seeks to avoid any points of zero undulation across it's midline to maximize its propulsive efficiency. While the predicted value of efficiency may depend on numerical factors, such as the order of the approximation polynomials, the relative efficiency ranking between different swimming modes does not change.

The third phase extends the modeling of a solitary swimmer to schools in a phalanx formation. Critical parameters relevant to an infinite phalanx fish school, such as midline kinematics, separation distance and phase synchronization, were investigated in light of efficient propulsion in fish accelerating from rest. The optimal midline kinematics of swimmers in an accelerating phalanx formation did not differ much from those of an accelerating solitary swimmer. This is primarily due to the fact that hydrodynamic schooling benefits present in phalanx schools, which arrise from phenomena such as the "channeling effect" (Daghooghi and Borazjani 2015) and the "pulsating jet" mechanism (Godoy-Diana et al. 2019), are enhanced by same optimal midline kinematics of a solitary swimmer, such as a high undulation frequency and

tail amplitude. Moreover, the optimal separation distance of $2L$ is shown to strike a balance between the gained propulsive work and the total work input of swimmers in a phalanx school. Finally, an analysis of the phase lag effects shows that the the relative ranking of efficiency based on separation distance within a school does not change regardless of phase lag. This trend suggests that the separation distance within an accelerating infinite phalanx school has a more significant effect on its propulsive efficiency when compared to the phase lag between the swimmers, assuming all else is constant. In the future, I aim to extend the modeling and analysis to include optimal accelerating and steady state propulsion in diamond-shaped schools.

REFERENCES

Abouhussein, A., N. Islam, and Yulia Peet. 2021. *A constraint handling approach with guaranteed feasibility for surrogate based optimization.* ARXIV PREPRINT ARXIV:2107.10190. Arxiv.org/abs/2107.10190.

Adams, Brian, William Bohnhoff, Keith Dalbey, Mohamed Ebeida, John Eddy, Michael Eldred, Russell Hooper, Patricia Hough, Kenneth Hu, John Jakeman, et al. 2020. *Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.13 User's Manual.* Technical report. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).

Adams, Brian M, William J Bohnhoff, KR Dalbey, JP Eddy, MS Eldred, DM Gay, K Haskell, Patricia D Hough, and Laura P Swiler. 2009. "DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: version 5.0 User's Manual." *Sandia National Laboratories, Tech. Rep. SAND2010-2183.*

Aitken, Alexander Craig. 1927. "On Bernoulli's Numerical Solution of Algebraic Equations." *Proceedings of the Royal Society of Edinburgh* 46:289–305.

Akanyeti, Otar, Joy Putney, Yuzo R Yanagitsuru, George V Lauder, William J Stewart, and James C Liao. 2017. "Accelerating fishes increase propulsive efficiency by modulating vortex ring geometry." *Proceedings of the National Academy of Sciences* 114 (52): 13828–13833.

Alexander, R McNeill. 2013. *Principles of animal locomotion.* Princeton University Press.

Anderson Jr, John David. 2010. *Fundamentals of aerodynamics.* Tata McGraw-Hill Education.

Ashraf, I, R Godoy-Diana, J Halloy, B Collignon, and B Thiria. 2016. "Synchronization and collective swimming patterns in fish (Hemigrammus bleheri)." *Journal of the Royal Society Interface* 13 (123): 20160734.

Ashraf, Intesaaf, Hanaé Bradshaw, Thanh-Tung Ha, José Halloy, Ramiro Godoy-Diana, and Benjamin Thiria. 2017. "Simple phalanx pattern leads to energy saving in cohesive fish schooling." *Proceedings of the National Academy of Sciences* 114 (36): 9599–9604.

Audet, Charles, J Denni, Douglas Moore, Andrew Booker, and Paul Frank. 2000. "A surrogate-model-based method for constrained optimization." In *8th symposium on multidisciplinary analysis and optimization,* 4891.

Audet, Charles, and John E Dennis Jr. 2006. "Mesh adaptive direct search algorithms for constrained optimization." *SIAM Journal on optimization* 17 (1): 188–217.

Back, Thomas. 1996. *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms.* Oxford University Press.

Bainbridge, Richard. 1958. "The speed of swimming of fish as related to size and to the frequency and amplitude of the tail beat." *Journal of Experimental Biology* 35 (1): 109–133.

Barrett, DS, MS Triantafyllou, DKP Yue, MA Grosenbaugh, and MJ Wolfgang. 1999. "Drag reduction in fish-like locomotion." *Journal of Fluid Mechanics* 392:183–212.

Bergmann, Michel, Angelo Iollo, and Rajat Mittal. 2014. "Effect of caudal fin flexibility on the propulsive efficiency of a fish-like swimmer." *Bioinspiration & biomimetics* 9 (4): 046001.

Bernardi, Christine, and Yvon Maday. 1988. "A collocation method over staggered grids for the Stokes problem." *IJNMF* 8:537–557.

Booker, Andrew J, John E Dennis, Paul D Frank, David B Serafini, Virginia Torczon, and Michael W Trosset. 1999. "A rigorous framework for optimization of expensive functions by surrogates." *Structural optimization* 17 (1): 1–13.

Borazjani, I, and Fotis Sotiropoulos. 2010. "On the role of form and kinematics on the hydrodynamics of self-propelled body/caudal fin swimming." *Journal of Experimental Biology* 213 (1): 89–107.

Borazjani, Iman, and Fotis Sotiropoulos. 2008. "Numerical investigation of the hydrodynamics of carangiform swimming in the transitional and inertial flow regimes." *Journal of experimental biology* 211 (10): 1541–1558.

Causin, Paola, Jean-Frédéric Gerbeau, and Fabio Nobile. 2005. "Added-mass effect in the design of partitioned algorithms for fluid–structure problems." *Computer methods in applied mechanics and engineering* 194 (42-44): 4506–4527.

Cicchese, Joseph M, Elsje Pienaar, Denise E Kirschner, and Jennifer J Linderman. 2017. "Applying optimization algorithms to tuberculosis antibiotic treatment regimens." *Cellular and molecular bioengineering* 10 (6): 523–535.

Coyle, Stephen, Carmel Majidi, Philip LeDuc, and K Jimmy Hsia. 2018. "Bio-inspired soft robotics: Material selection, actuation, and design." *Extreme Mechanics Letters* 22:51–59.

Daghooghi, Mohsen, and Iman Borazjani. 2015. "The hydrodynamic advantages of synchronized swimming in a rectangular pattern." *Bioinspiration & biomimetics* 10 (5): 056018.

De Jong, Kenneth Alan. 1975. *An analysis of the behavior of a class of genetic adaptive systems.* University of Michigan.

Deville, Michel O, Paul F Fischer, Paul F Fischer, EH Mund, et al. 2002. *High-order methods for incompressible fluid flow.* Vol. 9. Cambridge university press.

Di Santo, Valentina, Elsa Goerig, Dylan K Wainwright, Otar Akanyeti, James C Liao, Theodore Castro-Santos, and George V Lauder. 2021. "Convergence of undulatory swimming kinematics across a diversity of fishes." *Proceedings of the National Academy of Sciences* 118 (49).

Dikin, II. 1967. "Iterative solution of problems of linear and quadratic programming." In *Doklady Akademii Nauk,* 174:747–748. 4. Russian Academy of Sciences.

Diouane, Youssef, Serge Gratton, and Luis Nunes Vicente. 2015. "Globally convergent evolution strategies for constrained optimization." *Computational Optimization and Applications* 62 (2): 323–346.

Donley, Jeanine M, and Kathryn A Dickson. 2000. "Swimming kinematics of juvenile kawakawa tuna (Euthynnus affinis) and chub mackerel (Scomber japonicus)." *Journal of Experimental Biology* 203 (20): 3103–3116.

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

Eldeiry, Ahmed A, and Luis A Garcia. 2010. "Comparison of ordinary kriging, regression kriging, and cokriging techniques to estimate soil salinity using LANDSAT images." *Journal of Irrigation and Drainage Engineering* 136 (6): 355–364.

Eloy, Christophe. 2013. "On the best design for undulatory swimming." *Journal of Fluid Mechanics* 717:48.

Fischer, Paul F. 1997. "An overlapping Schwarz method for spectral element solution of the incompressible Navier–Stokes equations." *Journal of Computational Physics* 133 (1): 84–101.

Fish, FE, and George V Lauder. 2006. "Passive and active flow control by swimming fishes and mammals." *Annu. Rev. Fluid Mech.* 38:193–224.

Fish, Frank E, Natalia Rybczynski, George V Lauder, and Christina M Duff. 2021. "The Role of the Tail or Lack Thereof in the Evolution of Tetrapod Aquatic Propulsion." *Integrative and Comparative Biology.*

Floudas, Christodoulos A, and Panos M Pardalos. 1990. *A collection of test problems for constrained global optimization algorithms.* Vol. 455. Springer Science & Business Media.

Forrester, Alexander IJ, and Andy J Keane. 2009. "Recent advances in surrogate-based optimization." *Progress in aerospace sciences* 45 (1-3): 50–79.

Forrester, Alexander IJ, András Sóbester, and Andy J Keane. 2006. "Optimization with missing data." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 462 (2067): 935–945.

Förster, Christiane, Wolfgang A Wall, and Ekkehard Ramm. 2007. "Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows." *Computer methods in applied mechanics and engineering* 196 (7): 1278–1293.

Franci, Alessandro, and Massimiliano Cremonesi. 2017. "On the effect of standard PFEM remeshing on volume conservation in free-surface fluid flow problems." *Computational Particle Mechanics* 4 (3): 331–343.

Gardner, Jacob R, Matt J Kusner, Zhixiang Eddie Xu, Kilian Q Weinberger, and John P Cunningham. 2014. "Bayesian optimization with inequality constraints." In *ICML,* 2014:937–945.

Garrick, IE. 1937. *Propulsion of a flapping and oscillating airfoil.* NACA Report No. 567.

Gazzola, Mattia, Médéric Argentina, and Lakshminarayanan Mahadevan. 2014. "Scaling macroscopic aquatic locomotion." *Nature Physics* 10 (10): 758–761.

Gazzola, Mattia, Philippe Chatelain, Wim M Van Rees, and Petros Koumoutsakos. 2011. "Simulations of single and multiple swimmers with non-divergence free deforming geometries." *Journal of Computational Physics* 230 (19): 7093–7114.

Gazzola, Mattia, Wim M Van Rees, and Petros Koumoutsakos. 2012. "C-start: optimal start of larval fish." *Journal of Fluid Mechanics* 698 (9): 5–18.

Gebraad, Pieter, Jared J Thomas, Andrew Ning, Paul Fleming, and Katherine Dykes. 2017. "Maximization of the annual energy production of wind power plants by optimization of layout and yaw-based wake control." *Wind Energy* 20 (1): 97–107.

Gibouin, Florence, Christophe Raufaste, Yann Bouret, and Médéric Argentina. 2018. "Study of the thrust–drag balance with a swimming robotic fish." *Physics of Fluids* 30 (9): 091901.

Godoy-Diana, Ramiro, Jérôme Vacher, Veronica Raspa, and Benjamin Thiria. 2019. "On the fluid dynamical effects of synchronization in side-by-side swimmers." *Biomimetics* 4 (4): 77.

György, András, and Levente Kocsis. 2011. "Efficient multi-start strategies for local search algorithms." *Journal of Artificial Intelligence Research* 41:407–444.

Hansen, Nikolaus. 2016. "The CMA evolution strategy: A tutorial." *arXiv preprint arXiv:1604.00772.*

Hansen, Nikolaus, and Andreas Ostermeier. 2001. "Completely derandomized self-adaptation in evolution strategies." *Evolutionary computation* 9 (2): 159–195.

Hart, William, Cynthia Phillips, Lee Riesen, John Siirola, Jean-Paul Watson, and Jonathan Esckstein. 2010. *Acro (A Common Repository for Optimizers) v. 2.0.* Technical report. Sandia National Laboratories.

Hemelrijk, CK, DAP Reid, H Hildenbrandt, and JT Padding. 2015. "The increased efficiency of fish swimming in a school." *Fish and Fisheries* 16 (3): 511–521.

Ho, Lee Wing. 1989. "A Legendre spectral element method for simulation of incompressible unsteady viscous free-surface flows." PhD diss., Massachusetts Institute of Technology.

Jones, Donald R. 2001. "A taxonomy of global optimization methods based on response surfaces." *Journal of global optimization* 21 (4): 345–383.

Jones, Donald R, Matthias Schonlau, and William J Welch. 1998. "Efficient global optimization of expensive black-box functions." *Journal of Global optimization* 13 (4): 455–492.

Karmarkar, Narendra. 1984. "A new polynomial-time algorithm for linear programming." In *Proceedings of the sixteenth annual ACM symposium on Theory of computing,* 302–311.

Kern, Stefan, and Petros Koumoutsakos. 2006. "Simulations of optimized anguilliform swimming." *Journal of Experimental Biology* 209 (24): 4841–4857.

Khayyer, Abbas, Naoki Tsuruta, Yuma Shimizu, and Hitoshi Gotoh. 2019. "Multi-resolution MPS for incompressible fluid-elastic structure interactions in ocean engineering." *Applied Ocean Research* 82:397–414.

Knower, TORRE, ROBERT E Shadwick, STEPHEN L Katz, JEFFREY B Graham, and CLEMENT S Wardle. 1999. "Red muscle activation patterns in yellowfin (Thunnus albacares) and skipjack (Katsuwonus pelamis) tunas during steady swimming." *Journal of Experimental Biology* 202 (16): 2127–2138.

Kumar, Abhishek, Guohua Wu, Mostafa Z Ali, Rammohan Mallipeddi, Ponnuthurai Nagaratnam Suganthan, and Swagatam Das. 2020. "A test-suite of non-convex constrained optimization problems from the real-world and some baseline results." *Swarm and Evolutionary Computation* 56:100693.

Küttler, Ulrich, and Wolfgang A Wall. 2008. "Fixed-point fluid–structure interaction solvers with dynamic relaxation." *Computational mechanics* 43 (1): 61–72.

Lauder, George V, and Eric D Tytell. 2005. "Hydrodynamics of undulatory propulsion." *Fish Physiology* 23:425–468.

Laumanns, Marco, Eckart Zitzler, and Lothar Thiele. 2001. "On the effects of archiving, elitism, and density based selection in evolutionary multi-objective optimization." In *International Conference on Evolutionary Multi-Criterion Optimization,* 181–196. Springer.

Li, Gen, Dmitry Kolomenskiy, Hao Liu, Benjamin Thiria, and Ramiro Godoy-Diana. 2019. "On the energetics and stability of a minimal fish school." *PLoS One* 14 (8): e0215265.

Li, Liang, Máté Nagy, Jacob M Graving, Joseph Bak-Coleman, Guangming Xie, and Iain D Couzin. 2020. "Vortex phase matching as a strategy for schooling in robots and in fish." *Nature communications* 11 (1): 1–9.

Li, Ningyu, Huanxing Liu, and Yumin Su. 2017. "Numerical study on the hydrodynamics of thunniform bio-inspired swimming under self-propulsion." *PloS one* 12 (3): e0174740.

Liang, Jing J, Thomas Philip Runarsson, Efren Mezura-Montes, Maurice Clerc, Ponnuthurai Nagaratnam Suganthan, CA Coello Coello, and Kalyanmoy Deb. 2006. "Problem definitions and evaluation criteria for the CEC 2006 special session on

constrained real-parameter optimization." *Journal of Applied Mechanics* 41 (8): 8–31.

Lighthill, Michael J. 1960. "Note on the swimming of slender fish." *Journal of fluid Mechanics* 9 (2): 305–317.

Lighthill, Michael James. 1971. "Large-amplitude elongated-body theory of fish locomotion." *Proceedings of the Royal Society of London. Series B. Biological Sciences* 179 (1055): 125–138.

Lighthill, MJ. 1969. "Hydromechanics of aquatic animal propulsion." *Annu. Rev. Fluid Mech.* 1 (1): 413–446.

Lindsey, Benjamin W, Frank M Smith, and Roger P Croll. 2010. "From inflation to flotation: contribution of the swimbladder to whole-body density and swimming depth during development of the zebrafish (Danio rerio)." *Zebrafish* 7 (1): 85–96.

Liu, Jianjun, Kok Lay Teo, Xiangyu Wang, and Changzhi Wu. 2016. "An exact penalty function-based differential search algorithm for constrained global optimization." *Soft Computing* 20 (4): 1305–1313.

Lophaven, Søren Nymand, Hans Bruun Nielsen, Jacob Søndergaard, et al. 2002. *DACE: a Matlab kriging toolbox.* Vol. 2. Citeseer.

Maday, Yvon, and Anthony T Patera. 1989. "Spectral element methods for the incompressible Navier-Stokes equations." In *IN: State-of-the-art surveys on computational mechanics (A90-47176 21-64). New York, American Society of Mechanical Engineers, 1989, p. 71-143. Research supported by DARPA.* 71–143.

Maertens, Audrey P, Amy Gao, and Michael S Triantafyllou. 2017. "Optimal undulatory swimming for a single fish-like body and for a pair of interacting swimmers." *Journal of Fluid Mechanics* 813:301–345.

Martinez, Jose Mario, and FNC Sobral. 2013. "Constrained derivative-free optimization on thin domains." *Journal of Global Optimization* 56 (3): 1217–1232.

McKay, Michael D, Richard J Beckman, and William J Conover. 1979. "Comparison of three methods for selecting values of input variables in the analysis of output from a computer code." *Technometrics* 21 (2): 239–245.

McMasters, Robert L, Casey P Grey, John M Sollock, Ranjan Mukherjee, Andre Benard, and Alejandro R Diaz. 2008. "Comparing the mathematical models of

Lighthill to the performance of a biomimetic fish." *Bioinspiration & Biomimetics* 3 (1): 016002.

Merrill, Brandon E, and Yulia T Peet. 2019. "Moving overlapping grid methodology of spectral accuracy for incompressible flow solutions around rigid bodies in motion." *Journal of Computational Physics* 390:121–151.

Mezura-Montes, Efrén, and Carlos A Coello Coello. 2011. "Constraint-handling in nature-inspired numerical optimization: past, present and future." *Swarm and Evolutionary Computation* 1 (4): 173–194.

Michalewicz, Zbigniew, Robert Hinterding, and Maciej Michalewicz. 1997. "Evolutionary algorithms." In *Fuzzy evolutionary computation,* 3–31. Springer.

Moored, Keith W, and Daniel B Quinn. 2019. "Inviscid scaling laws of a self-propelled pitching airfoil." *AIAA Journal* 57 (9): 3686–3700.

Morris, Max D, and Toby J Mitchell. 1995. "Exploratory designs for computational experiments." *Journal of statistical planning and inference* 43 (3): 381–402.

Mueller, Juliane. 2014. "MATSuMoTo: The MATLAB surrogate model toolbox for computationally expensive black-box global optimization problems." *arXiv preprint arXiv:1404.4261.*

Müller, Juliane, and Joshua D Woodbury. 2017. "GOSAC: global optimization with surrogate approximation of constraints." *Journal of Global Optimization* 69 (1): 117–136.

Nazemian, Amin, and Parviz Ghadimi. 2021. "Multi-objective optimization of trimaran sidehull arrangement via surrogate-based approach for reducing resistance and improving the seakeeping performance." *Proceedings of the institution of mechanical engineers, part M: journal of engineering for the maritime environment* 235 (4): 944–956.

Neufeld, Daniel, Joon Chung, and Kamran Behdinian. 2011. "Aircraft conceptual design optimization considering fidelity uncertainties." *Journal of aircraft* 48 (5): 1602–1612.

Nocedal, Jorge, and Stephen Wright. 2006. *Numerical optimization.* Springer Science & Business Media.

Orvosh, David, and Lawrence Davis. 1994. "Using a genetic algorithm to optimize problems with feasibility constraints." In *Proceedings of the First IEEE Con-*

ference on Evolutionary Computation. IEEE World Congress on Computational Intelligence, 548–553. IEEE.

Oza, Anand U, Leif Ristroph, and Michael J Shelley. 2019. "Lattices of hydrodynamically interacting flapping swimmers." *Physical Review X* 9 (4): 041024.

Paley, Derek A, and Norman M Wereley. 2021. *Bioinspired Sensing, Actuation, and Control in Underwater Soft Robotic Systems.* Springer.

Pan, Yu, and Haibo Dong. 2020. "Computational analysis of hydrodynamic interactions in a high-density fish school." *Physics of Fluids* 32 (12): 121901.

Parr, James M, Andy J Keane, Alexander IJ Forrester, and Carren ME Holden. 2012. "Infill sampling criteria for surrogate-based optimization with constraint handling." *Engineering Optimization* 44 (10): 1147–1166.

Patel, Saumil, Paul Fischer, Misun Min, and Ananias Tomboulides. 2019. "A characteristic-based spectral element method for moving-domain problems." *Journal of Scientific Computing* 79 (1): 564–592.

Quarteroni, Alfio, Fausto Saleri, and Alessandro Veneziani. 2000. "Factorization methods for the numerical approximation of Navier–Stokes equations." *CComputer Methods in Applied Mechanics and Engineering* 188 (1-3): 505–526.

Queipo, Nestor V, Raphael T Haftka, Wei Shyy, Tushar Goel, Rajkumar Vaidyanathan, and P Kevin Tucker. 2005. "Surrogate-based analysis and optimization." *Progress in aerospace sciences* 41 (1): 1–28.

Raspa, V, R Godoy-Diana, and B Thiria. 2013. "Topology-induced effect in biomimetic propulsive wakes." *Journal of Fluid Mechanics* 729:377–387.

Ray, Tapabrata, Hemant Kumar Singh, Amitay Isaacs, and Warren Smith. 2009. "Infeasibility driven evolutionary algorithm for constrained optimization." In *Constraint-handling in evolutionary optimization,* 145–165. Springer.

Razavi, Saman, Bryan A Tolson, and Donald H Burn. 2012. "Review of surrogate modeling in water resources." *Water Resources Research* 48 (7).

Regis, Rommel G. 2014. "Constrained optimization by radial basis function interpolation for high-dimensional expensive black-box problems with infeasible initial points." *Engineering Optimization* 46 (2): 218–243.

Regis, Rommel G, and Christine A Shoemaker. 2007. "A stochastic radial basis function method for the global optimization of expensive functions." *INFORMS Journal on Computing* 19 (4): 497–509.

Rios, Luis Miguel, and Nikolaos V Sahinidis. 2013. "Derivative-free optimization: a review of algorithms and comparison of software implementations." *Journal of Global Optimization* 56 (3): 1247–1293.

Runarsson, Thomas P., and Xin Yao. 2000. "Stochastic ranking for constrained evolutionary optimization." *IEEE Transactions on evolutionary computation* 4 (3): 284–294.

Saadat, M, Frank E Fish, AG Domel, V Di Santo, GV Lauder, and H Haj-Hariri. 2017. "On the rules for aquatic locomotion." *Physical Review Fluids* 2 (8): 083102.

Sasena, Michael J, Panos Y Papalambros, and Pierre Goovaerts. 2001. "The use of surrogate modeling algorithms to exploit disparities in function computation time within simulation-based optimization." *Constraints* 2 (5).

Schultz, William W, and Paul W Webb. 2002. "Power requirements of swimming: Do new methods resolve old questions?" *Integrative and Comparative Biology* 42 (5): 1018–1025.

Shadwick, Robert E, STEPHEN L Katz, Keith E Korsmeyer, Torre Knower, and JAMES W Covell. 1999. "Muscle dynamics in skipjack tuna: timing of red muscle shortening in relation to activation and body curvature during steady swimming." *J. Exp. Biology* 202 (16): 2139–2150.

Shadwick, Robert E, and Douglas A Syme. 2008. "Thunniform swimming: muscle dynamics and mechanical power production of aerobic fibres in yellowfin tuna (Thunnus albacares)." *Journal of Experimental Biology* 211 (10): 1603–1611.

Shirgaonkar, Anup A, Malcolm A MacIver, and Neelesh A Patankar. 2009. "A new mathematical formulation and fast algorithm for fully resolved simulation of self-propulsion." *Journal of Computational Physics* 228 (7): 2366–2390.

Sobester, András, Alexander Forrester, and Andy Keane. 2008. *Engineering design via surrogate modelling: a practical guide.* John Wiley & Sons.

Stork, Jörg, Agoston E Eiben, and Thomas Bartz-Beielstein. 2020. "A new taxonomy of global optimization algorithms." *Natural Computing,* 1–24.

Suganthan, Ponnuthurai N, Nikolaus Hansen, Jing J Liang, Kalyanmoy Deb, Ying-Ping Chen, Anne Auger, and Santosh Tiwari. 2005. "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization." *KanGAL report* 2005005 (2005): 2005.

Szalay, Zoltán, and Lajos Nagy. 2014. "Performance analysis of an open ended coaxial resonator based displacement sensor." In *The 8th European Conference on Antennas and Propagation (EuCAP 2014),* 3648–3651. IEEE.

Tenne, Yoel. 2015. "Initial sampling methods in metamodel-assisted optimization." *Engineering with Computers* 31 (4): 661–680.

Thandiackal, Robin, Carl H White, Hilary Bart-Smith, and George V Lauder. 2021. "Tuna robotics: hydrodynamics of rapid linear accelerations." *Proceedings of the Royal Society B* 288 (1945): 20202726.

Theodorsen, Theodore. 1935. *General theory of aerodynamic instability and the mechanism of flutter.* NASA Technical Report No. 46.

Tokić, Grgur, and Dick KP Yue. 2012. "Optimal shape and motion of undulatory swimming organisms." *Proceedings of the Royal Society B: Biological Sciences* 279 (1740): 3065–3074.

Toolbox, Symbolic Math, et al. 1993. "Matlab." *Mathworks Inc.*

Triantafyllou, George S, MS Triantafyllou, and MA Grosenbaugh. 1993. "Optimal thrust development in oscillating foils with application to fish propulsion." *Journal of Fluids and Structures* 7 (2): 205–224.

Triantafyllou, Michael S, GS Triantafyllou, and DKP Yue. 2000. "Hydrodynamics of fishlike swimming." *Annual review of fluid mechanics* 32 (1): 33–53.

Turek, S., and J. Hron. 2006. "Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow." In *Fluid-Structure Interaction,* edited by H.-J. Bungartz and M. Schäfer, 53:371–385. Springer Berlin Heidelberg.

Tytell, Eric D. 2004. "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.

Tytell, Eric D, and George V Lauder. 2004. "The hydrodynamics of eel swimming: I. Wake structure." *Journal of Experimental Biology* 207 (11): 1825–1841.

Valdivia y Alvarado, Pablo Pablo Alvaro. 2007. "Design of biomimetic compliant devices for locomotion in liquid environments." PhD diss., Massachusetts Institute of Technology.

Van Buren, Tyler, Daniel Floryan, and Alexander J Smits. 2018. "Bioinspired underwater propulsors." In *Bioinspired Structures and Design, eds. L. Daniel and and W. Soboyejo.* Cambridge University Press, Cambridge.

Van Rees, Wim M, Mattia Gazzola, and Petros Koumoutsakos. 2013. "Optimal shapes for anguilliform swimmers at intermediate Reynolds numbers." *Journal of Fluid Mechanics* 722:R3.

Verma, Siddhartha, Guido Novati, and Petros Koumoutsakos. 2018. "Efficient collective swimming by harnessing vortices through deep reinforcement learning." *Proceedings of the National Academy of Sciences* 115 (23): 5849–5854.

Videler, JJ, and F Hess. 1984. "Fast continuous swimming of two pelagic predators, saithe (Pollachius virens) and mackerel (Scomber scombrus): a kinematic analysis." *Journal of experimental biology* 109 (1): 209–228.

Wang, Chen, Qingyun Duan, Wei Gong, Aizhong Ye, Zhenhua Di, and Chiyuan Miao. 2014. "An evaluation of adaptive surrogate modeling based optimization with two benchmark problems." *Environmental Modelling & Software* 60:167–179.

Webb, Paul W. 1975. "Hydrodynamics and energetics of fish propulsion." *Bulletin—Fisheries Research Board of Canada.*

Weihs, D. 1973. "Hydromechanics of fish schooling." *Nature* 241 (5387): 290–291.

Wise, Tyler N, Margot AB Schwalbe, and Eric D Tytell. 2018. "Hydrodynamics of linear acceleration in bluegill sunfish, Lepomis macrochirus." *Journal of Experimental Biology* 221 (23): 1–12.

Woldesenbet, Yonas Gebre, Gary G Yen, and Biruk G Tessema. 2009. "Constraint handling in multiobjective evolutionary optimization." *IEEE Transactions on Evolutionary Computation* 13 (3): 514–525.

Xu, YiQin, and Yulia Peet. 2021. "Verification and convergence study of a spectral-element numerical methodology for fluid-structure interaction." *Journal of Computational Physics: X* 10:100084.

Xu, YiQin, and Yulia T Peet. 2017. "Accuracy and performance of fluid-structure interaction algorithms with explicit versus implicit formulations of the fluid

solver." In *23rd AIAA Computational Fluid Dynamics Conference,* 3449. 2017–3449. Denver, CO.

Xu, YiQin, and Yulia T Peet. 2018. "Optimum gaits of 2D thunniform locomotion for efficient swimming and performance of fish pair." In *2018 Fluid Dynamics Conference,* 2915. 2018–2915. Atlanta, GA.

Yu, Huiyang, Xi-Yun Lu, and Haibo Huang. 2021. "Collective locomotion of two uncoordinated undulatory self-propelled foils." *Physics of Fluids* 33 (1): 011904.

Zhu, Q, MJ Wolfgang, DKP Yue, and MS Triantafyllou. 2002. "Three-dimensional flow structures and vorticity control in fish-like swimming." *Journal of Fluid Mechanics* 468:1–28.