Massachusetts Institute of Technology

An Automatic, Multi-Fidelity Framework for Optimizing the Performance of Super-Cavitating Hydrofoils Using Gaussian Process Regression and Bayesian Optimization

Author:
LT Benjamin W. Parker

Address:
MIT, Navy Postgraduate Office
77 Massachusetts Ave, Rm 5-317
Cambridge, MA 02139-4309
Phone: 360.286.8688
Website: http://web.mit.edu/2n/
Email: parkerbw@mit.edu

Advisor:
Dr. Luca Bonfiglio

Address:
MIT Sea Grant
12 Emily St
Cambridge, MA 02139
Phone: 617.253.7041
Website: http://seagrant.mit.edu/
Email: bonfi@mit.edu

July 2, 2018

Abstract

Computer automated design of complex engineering systems, such as hydrofoils for displacement support, is often limited by the computational resources required for the prediction of complex and competing quantities of interest driving the decision-making process. For hydrofoil supporting vessels operating at very high speeds, this task is complicated by the complex flow solution involving turbulent, multi-phase flow with phase change between vapor and liquid. In these operating conditions, determining an optimum design in a multi-dimensional input space necessitates an enormous amount of data often available only through expensive computer codes. In the specific case of hydrofoils, further complexities are introduced by the interconnectedness of the constraint and objective quantities. This paper applies recent advances in machine learning techniques to the development of an automated framework for iterating through a design loop that includes both physics-based computer simulations and surrogate model training. The key feature of this framework is the construction of probabilistic response surfaces based on Gaussian Process regression describing complex quantities of interest. The capabilities of the design framework are particularly suitable for the optimization of unconventional cavitating hydrofoil specifically designed to work in super-cavitating conditions, for which few experimental data or empirical formulations are available and the design must rely on the solution of challenging fluid dynamic problems. Prediction models based on potential flow assumptions frequently fail in predicting strong time-dependent phenomena such as cavity reattachment. An accurate prediction of the unsteady, turbulent, multi-phase flows requires the application of expensive solvers based on the solution of the Navier-Stokes equations. The necessity of fast, inexpensive objective function evaluations and the accuracy requirements represents a typical bottleneck in engineering optimization. To alleviate the computation burden and efficiently explore the design space, surrogate models are constructed using simulation data and interrogated for new design candidates. Each surrogate model is built using a multi-fidelity framework capable of producing response surfaces that retain the accuracy of the high-fidelity data used for their formulation while reducing their number by exploiting information coming from low-fidelity, cheaper models. The surrogate model is iteratively improved through Bayesian Optimization techniques and additional high-fidelity simulations automatically initiated within the design loop. In addition, Bayesian Optimization will be used to automatically determine the optimum kernel for the Gaussian regression model. This automated multi-fidelity Bayesian optimization framework can aid in taking the human out of the design loop, and therefore free manpower resources, remove potential human bias, and potentially enable the discovery of innovative designs.

An Automatic, Multi-Fidelity Framework for Optimizing the Performance of Super-Cavitating Hydrofoils Using Gaussian Process Regression and Bayesian Optimization

LT Benjamin W. Parker, USN
Graduate Student at Massachusetts Institute of Technology
Naval Construction and Engineering (Course 2N)
Email: parkerbw.edu

Computer automated design of complex engineering systems, such as hydrofoils for displacement support, is often limited by the computational resources required for the prediction of complex and competing quantities of interest driving the decision-making process. For hydrofoil supporting vessels operating at very high speeds, this task is complicated by the complex flow solution involving turbulent, multi-phase flow with phase change between vapor and liquid. In these operating conditions, determining an optimum design in a multi-dimensional input space necessitates an enormous amount of data often available only through expensive computer codes. In the specific case of hydrofoils, further complexities are introduced by the interconnectedness of the constraint and objective quantities. This paper applies recent advances in machine learning techniques to the development of an automated framework for iterating through a design loop that includes both physics-based computer simulations and surrogate model training. The key feature of this framework is the construction of probabilistic response surfaces based on Gaussian Process regression describing complex quantities of interest. The capabilities of the design framework are particularly suitable for the optimization of unconventional cavitating hydrofoil specifically designed to work in super-cavitating conditions, for which few experimental data or empirical formulations are available and the design must rely on the solution of challenging fluid dynamic problems. Prediction models based on potential flow assumptions frequently fail in predicting strong time-dependent phenomena such as cavity reattachment. An accurate prediction of the unsteady, turbulent, multi-phase flows requires the application of expensive solvers based on the solution of the Navier-Stokes equations. The necessity of fast, inexpensive objective function evaluations and the accuracy requirements represents a typical bottleneck in engineering optimization. To alleviate the computation burden and efficiently explore the design space, surrogate models are constructed using simulation data and interrogated for new design candidates. Each surrogate model is built using a multi-fidelity framework capable of producing response surfaces that retain the accuracy of the high-fidelity data used for their formulation while reducing their number by exploiting information coming from low-fidelity, cheaper models. The surrogate model is iteratively improved through Bayesian Optimization techniques and additional high-fidelity simulations automatically initiated within the design loop. In addition, Bayesian Optimization will be used to automatically determine the optimum kernel for the Gaussian regression model. This automated multi-fidelity Bayesian optimization framework can aid in taking the human out of the design loop, and therefore free manpower resources, remove potential human bias, and potentially enable the discovery of innovative designs.

1 Introduction

Engineering design problems generally include multiple design inputs and difficult to evaluate quantities of interest. The quantities of interest, which are the design objective and constraints, are determined by computer simulation or experimentation. Since both evaluation methods are often expensive, in terms of time, they often inadequately explore the design space. Thorough design space exploration often requires orders of magnitude more simulations than budgeted. Therefore, model-based optimization is used to "fill in the gaps" in the design space between simulation results.

Model-based optimization, also referred to as surrogate modeling or response surface modeling, is the method of creating models from training data that can recreate simulation results and make predictions at untested locations. The input training data is purposely chosen to span the design space and the black-box model will attempt to mimic the simulations outputs. If the design input is one dimensional, then the surrogate modeling is the process of fitting a curve to the data. If the design input is two dimensional, then the surrogate modeling is the process of fitting a surface to the

data. Higher order design inputs are more difficult to visualize, but the modeling concept is the same. Modeling allows the designers to optimize their design by more thoroughly exploring the design space. These simple examples don't show the model uncertainty, but this important factor will be addressed in Section 4. The theory behind of Gaussian process regression, also referred to as kriging, will be explained in Section 4, but this section will focus on the advantages of this response surface modeling technique. First, GP regression is flexible. GP modeling is capable of handling a large variety of engineering problems, including both regression and classification problems, and producing effective predictive models. Second, GP is user-friendly. Since GP is commonly used, there are many GP regression tools, training resources, and support venues. Because of the many resources available and the efficacy of GP for many engineering problems, I call GP user-friendly. Third, GP is a non-linear and non-parametric modeling paradigm. This enables more complex and better modeling solutions. Fourth, GP regression is computationally tractable. There are other regression techniques that are computationally cheaper, but GP gives advanced modeling solutions with an understood and stable computational cost.

One of the main goals of this paper is to demonstrate the feasibility of an automatic design loop, integrating physics-based solvers and machine learning principle. There are many advantages to having an automated design loop. These advantages include (1) freeing man-power resources to work on problems for which computers are ill-equipped, (2) decisions can be made quicker as humans and computers work simultaneously, (3) more thorough exploration of the design space is possible using surrogate models and digital twins, (4) removal of potential human bias which will increase the probability of discovering fresh design solutions and reduce the probability of inadvertently making design decisions earlier than necessary, and (5) easily allows for design bounds and constraints to be adjusted as decision makers learn in greater detail the scope of the engineering problem. There are many ways to attempt automation for an engineering design problems, but according to Shahriari et al. Bayesian optimization "promises greater automation so as to increase both product quality and human productivity" [25].

The paper first describes the design approach employed for the optimization of a super-cavitating hydrofoils, illustrating the main challenges to address to maintain stable cavitating regimes and prove a steady-state lift force. The second part of the paper will be dedicated to the description of the probabilistic framework used to construct response surfaces describing the stability of the cavitating regime, the lift development, the structural integrity of the hydrofoil and the hydrofoil efficiency which represents the quantity to optimize. The third part of the paper describes the details of the hydrodynamic problem and the multi-fidelity approach adopted to mitigate the computational burden while maintaining high accuracy in the solution of the super-cavitating flow. Finally, the super-cavitating hydrofoil design analysis and results are presented.

2 Hydrofoil Design

The goal of this project is to optimize the shape of hydrofoils specifically designed to sustain the displacement of ultra-fast small craft conceived to operate at very high speed (approximately 100 knots). The main idea is that higher speeds can be reached by completely or partially lifting the hull out of the water using hydrofoils capable to generate vertical forces completely or partially balancing the overall weight of the vessel. Lifting surfaces operating at speeds in excess of approximately 50 knots experience cavitation: the pressure on the hydrofoil suction surface drops below the surface tension, triggering water vaporization at ambient temperature (see [1, 2]).

The main challenge when designing ultra high-speed hydrofoil supported vessels is to ensure a stable cavitating regime on the supporting hydrofoils at the highest speeds (super-cavitating regime) while maintaining adequate performance at lower speeds, when the supporting hydrofoils operate in fully wet conditions. Figure 1 presents conventional hydrofoil shapes designed for different regimes [3].

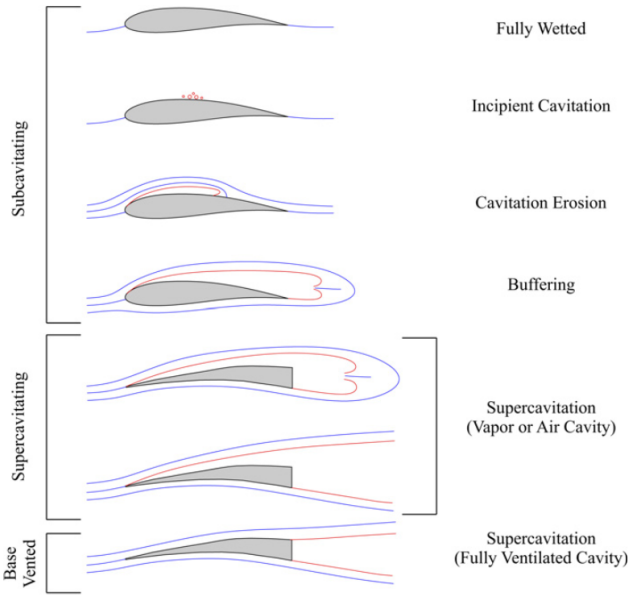


Fig. 1: Hydrofoil shapes designed for different cavitation environments. Red curves show the cavity contours and the blue lines show the flow streamlines [3].

The family of the super-cavitating hydrofoils analyzed in this study adopts an unconventional 2D section shape, specifically designed by Brizzolara [8] to ensure high hydrodynamic efficiency over a wide range of operating conditions (sub-cavitating and super-cavitating). Conventional super-cavitating hydrofoils feature a sharp leading edge and a blunt trailing edge ensuring cavity detachment at very high speeds. Their design is indeed conceived to efficiently operate only when a stable vapor cavity envelops the entire suction surface, enclosing many chords aft the trailing edge.

The shape of the Sub-Cavitating/Super-Cavitating (SBSC) unconventional hydrofoil presented in this study features a pointed leading edge ensuring cavity detachment at the operating conditions, and a sharp corner (face cavitator), triggering base cavitation at intermediate speeds or at high speeds and lower angles of attack.

At very high speeds (low cavitation numbers) and higher angles of attack the hydrofoil is fully enclosed in a vapor cavity (Super-Cavitating conditions). The trailing edge of the SBSC 2D hydrofoil has been tapered to provide a sufficient pressure recovery (hence higher lift) when working at lower speed (sub-cavitating conditions). As demonstrated by Brizzolara and Bonfiglio [9], this innovative 2D hydrofoil is able to ensure high performance also in sub-cavitating conditions, where conventional super-cavitating hydrofoils perform very poorly.

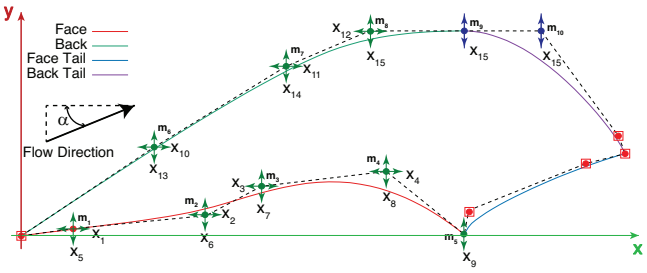


Fig. 2: Parametric model of the 2D hydrofoil section [26]. Four B-spline curves have been used to define the pressure and the suction surfaces on the forward part as well as the tail of the SCSB hydrofoil. These curves are generated from a control polygon indicated by black dashed lines connecting circular dots. Control points represented by green circular dots is parametrized through variables x_i. Red points represent fixed locations.

A first study aimed to shape an unconventional Sub-Cavitating/Super-Cavitating Hydrofoil (SB-SC) was presented in [26], where the 2-D hydrofoil has been described with a full parametric model and optimized using a differential evolution algorithm driven by hydrodynamic predictions obtained through a potential flow method. The optimized shape of the SB-SC hydrofoil was able to improve the lift-to-drag ratio of about 40%, but URANS simulations showed a consistent reduction of the cavity thickness on the hydrofoil suction surface, possibly affecting the stability of the cavitating flow in case of small deviations with respect to the reference shape. This problem was successively addressed by using a Risk-Adaptive Set Based Design (RASBD) approach able to include uncertainty in the manufacturing of the 2-D shape, identify candidate designs respecting constraints and providing quantification of the associated super-quantile risk [24].

The object of this paper is to develop a fully automated design framework capable of optimizing quantities of interest predicted using data-driven stochastic response surfaces generated from multi-fidelity simulations data. The main novelty of the proposed framework is the introduction of a self-learning procedure that exploits active learning algorithms and automatic simulation frameworks to improve the prediction of the response surface while simultaneously searching for the best design candidate.

2.1 Mission Requirements and Decision Framework

In the present study our goal is to optimize the shape of an unconventional 2-D super-cavtitating hydrofoil at constant operating conditions corresponding to the maximum speed of the hydrofoil supported SWATH vehicle patented by Brizzolara [10]. In this mission profile, the vessel's speed is $U = 61.67\ m/s$, corresponding to 120 knots, the hydrofoil angle of attack corresponds to $\alpha = 6^\circ$ and hydrodynamic chord length $c_H = 0.66m$. We assume constant thermodynamic characteristics of water corresponding to density of $\rho = 997.3$ kg/m^3 and saturation pressure of $p_{SAT} = 3064.68$ Pa.

When the lifting surface operates in the cavitating regime, their hydrodynamic characteristics are determined on the basis of the *cavitation index*, σ. Cavitation index is a non-dimensional parameter defined as the ratio between the difference of the local absolute pressure p and the saturation pressure, and the kinetic energy per unit of volume:

$$\sigma = \frac{p - p_{SAT}}{\frac{1}{2}\rho U_\infty^2}. \tag{1}$$

The objective quantity is to maximize the Lift-to-Drag ratio, L/D, where the lift force is expressed in terms of lift coefficient $C_L = \frac{L}{1/2\rho c U^2}$ and the drag force through the drag coefficient $C_D = \frac{D}{1/2\rho c U^2}$. The quantity to minimize is given by the ratio between drag and lift coefficient. While maximizing the hydrofoil efficiency in terms of drag-to-lift ratio it is important to guarantee the delivery of specific lift values, necessary structural characteristics and a stable flow regime through the introduction of specific constraints. In the present study four additional QoIs are considered and are shown in Equation (2). The hydrofoil must produce an appropriate lift force to support the displacement of the ship. Therefore, a minimum lift value is imposed to safeguard against sinkage, while to prevent the ship from experiencing over-lift, a maximum lift constraint is also applied. Stability of the cavitating regime is ensured by imposing a minimum value to the thickness of the cavitation pocket enveloping the suctions surface of the hydrofoil and measured at 10% of the chord length, $t_c^{10\%}$. Finally, to certify that the profile has adequate strength to undergo the high pressure loads necessary for developing high values of lift a minimum value to the profile inertia modulus, w, is imposed.

The most unique aspect of this decision framework is that decisions will be automatically taken within the design loop without human intervention. In fact, this framework has been developed to automatically train the surrogate models,

search the models for new designs, balance improving the design and improving the surrogate models, and initiate, run and post-process numerical simulations. The design problem is formally described as follow:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathcal{X}\subset\mathbb{R}^{15}}{\text{maximize}} \quad & \frac{C_L}{C_D}(\mathbf{x}) \\ \text{subject to} \quad & C_L(\mathbf{x}) \geq 0.3 \\ & C_L(\mathbf{x}) \leq 0.4 \\ & t_c^{10\%}(\mathbf{x}) \geq 0.004m \\ & w(\mathbf{x}) \geq 7.5 \times 10^{-6} m^3 \end{aligned} \tag{2}$$

Where **x** represents the design vector defining the shape of the hydrofoil through a parametric model specifically conceived to describe the shape of the unconventional supercavitating hydrofoil by means of 15 parameters.

2.2 Parametric Model

The foil shape is parametrically defined by a set of 4 B-Splines curves joining at specific functional points where C0, C1, C2 conditions are imposed (see Figure 2). Control points on the hydrofoil face have a direct impact on lift and drag forces while points on the suction side directly affect the cavity thickness while maintaining sufficient structural characteristics. Four control points (m_i, i= 1, 2, 3, 4) are employed for the definition of the hydrofoil pressure surface and allow for a double curvature profile. This generally leads to higher lift force while reducing flow separation at higher angles of attack. A fifth point (m_5) controls the vertical position of the face cavitator, regulating the cavity detachment position at the trailing edge. The shape of the suction surface is controlled by 5 points. Point m_6, regulates the thickness of the hydrofoil at the leading edge safeguarding against fluttering dynamic instability. At specific operating conditions, the shape of the vapor cavity depends on the curvature of the hydrofoil back controlled through the relative position of points m_7 and m_8. The tail of the hydrofoil is tapered by points m_9 and m_{10} position which is fixed in the chord-wise direction and vertically depends on point m_8. Lastly, red points in Figure 2 represent points having fixed position and not included in the parametric description. The non-dimensional geometry of the 2D section is kept constant along the span of the hydrofoil.

Table 1 presents minima and maxima for each component of the design vector ($x_i^{\min} \leq x_i \leq x_i^{\max}$ for $i = 1, ..., 15$) as well as values selected for the benchmark hydrofoil described in Figure 3.

Table 1: Benchmark design and bounds of design variables.

Variable	Benchmark	$x_i^{\min}$	$x_i^{\max}$
x_1	7.764E-2	7.835E-2	8.806E-2
x_2	2.756E-1	2.471E-1	2.777E-1
x_3	3.581E-1	4.003E-1	4.499E-1
x_4	5.438E-1	5.274E-1	5.927E-1
x_5	2.58E-3	3.803E-3	4.274E-3
x_6	7.23E-3	9.3E-3	1.045E-2
x_7	1.767E-2	1.386E-2	1.558E-2
x_8	2.297E-2	8.803E-3	9.893E-3
x_9	5.3E-4	2.563E-3	2.881E-3
x_{10}	1.981E-1	1.774E-1	1.994E-1
x_{11}	3.946E-1	3.693E-1	4.151E-1
x_{12}	5.19E-1	5.266E-1	5.918E-1
x_{13}	3.161E-2	2.817E-2	3.166E-2
x_{14}	6.041E-1	5.658E-2	6.359E-2
x_{15}	7.299E-2	6.975E-2	7.839E-2

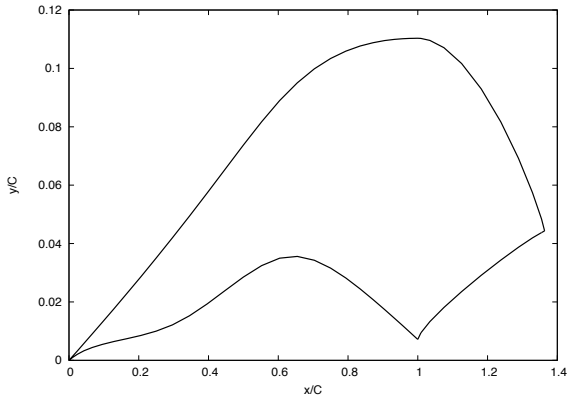


Fig. 3: Benchmark Hydrofoil Shape. The 3D hydrofoil features a sharp leading edge ensuring cavity detachment and a sharp edge on the pressure surface that ensures that the hydrofoil tail is enveloped in a stable cavity pocket at low cavitation index. The pressure surface presents a double curvature profile meant to increase the lift generated in supercavitating conditions.

All quantities of interest in Equation (2) are computed by Unsteady Reynolds Averaged Navier-Stokes supercavitating simulations developed by [4]. Additionally, tor the initial training dataset, low-fidelity data is generated by solving a two dimensional flow around super-cavitating hydrofoils using a potential flow model developed by [26]. These two numerical models provide two data sources characterized by different fidelity that are blended together to speed

up the formulation of stochastic response surfaces.

3 Performance assessment through multi-fidelity predictions

The proposed multi-fidelity simulation framework relies on two different fidelity information sources describing hydrofoil performance in terms of transient-free lift and drag coefficients as well as cavity thickness at specific hydrofoil locations. These metrics are provided by a low-fidelity Boundary Element Method (BEM) formulated around the assumption of potential flow and a URANS simulation framework specifically tailored to this particular application. The main advantages of these framework are the simple interface with external tools for geometry generation and the minimal requirement of input data which reduce the simulation set-up burden, provide a parametric mesh for any hydrofoil shape, and effectively speed up the systematic runs of the numerical solver while ensuring consistency and robustness of the results. Given the same computational resources, the CPU time required for a low-fidelity prediction is about 600 times faster than the time required to obtain a high-fidelity prediction.

3.1 Low-fidelity CFD

The low-fidelity CFD solver is a BEM potential flow simulator accurately described by Vernengo et al. [27]. Potential flow implies irrotational and inviscid flow. Moreover incompressibility is assumed. With these premises, the Laplace equation for the scalar velocity potential, φ, is shown in Equation 3. For pressure, p, the Bernoulli equation for unsteady flow is used and is shown in Equation 4. The Laplace and Bernoulli equations replace the continuity and momentum equations as the governing equations for fluid motion.

$$\nabla^2\varphi(x) = 0 \tag{3}$$

$$p = -\rho[\frac{\partial\varphi}{\partial t} + 0.5|\nabla\varphi|^2 + gz] + f \tag{4}$$

To solve the system of equations in cavitating conditions, two boundary conditions are applied to the problem. The first uses the bubble thickness t_c to define the cavity surface as a flow streamline. This boundary condition is expressed in Equation 5 where n is the local boundary normal. The second boundary condition is shown in Equation 6 and enforces saturation pressure to the fluid on the cavity pocket surface. These governing equations and boundary conditions were used in an iterative process to converge on the hydrodynamic forces. Numerical integration and extrapolation techniques were applied to calculate the forces on all faces of the hydrofoil. Finally, viscous effects are considered through a flat plate approximations for local frictional coefficients predictions (see [27]).

$$\frac{D}{Dt}(n - t_c(x)) = 0 \tag{5}$$

$$p = p_{vapor} \tag{6}$$

3.2 High-fidelity CFD

Possible cavity instabilities leading to harmful lift unsteadiness are addressed by solving a more realistic fluid dynamic problem, governed by the continuity and momentum equations, solved in a Cartesian reference frame for the pressure and velocity unknowns:

$$\begin{aligned} &\frac{\partial(\rho u_i)}{\partial x_i} = 0, \\ &\frac{\partial u_i}{\partial t} + \frac{\partial(u_i u_j)}{\partial x_j} = \frac{\partial}{\partial x_j}(\nu\frac{\partial u_i}{\partial x_j}) - \frac{1}{\rho}\frac{\partial p}{\partial x_i} + g_i. \end{aligned} \tag{7}$$

The solution of this problem allows designers to consider viscous effects through a more rigorous approach based on the solution of the turbulent boundary layer flow. The numerical solution of this system of PDEs is complicated by the high Reynolds number flow involved in our application: $Re = \frac{U_c}{\nu} \approx 5e6$. This requires a very high resolution in both temporal and spatial discretization, significantly increasing the computational complexity of the problem. The solution of the turbulent flow around the hydrofoil is achieved applying a Reynolds averaging technique to the N-S equations. More specifically, we employed a k-ε turbulence model to provide the closure equations required for the solution of the Reynolds stresses [19].

Cavitation occurs when the pressure field drops below the vapor pressure value, causing water vaporization at constant temperature. Unsteady vaporization and condensation of the fluid interacts with the turbulent boundary layer in a strongly time varying flow pattern. The individual bubble dynamics is not captured by the solver, but volumes of fluid intersected by cavitation bubbles are treated as homogeneous fluid having a variable density and viscosity, which depends on the fraction of liquid volume occupied by vapor bubbles. The solution of unsteady RANS equations is therefore complicated by the presence of a fluid interface due to the simultaneous presence of two different water phases: liquid and vapor. Additional equations are required for the solution of the non-linear fluid dynamic problem. The determination of the interface between vapor and liquid phase has been obtained using the Volume of Fluid surface capturing technique [14]. This method relies on the definition of a scalar function γ describing the balance of two different phases within each control volume. In such way, each cell is filled with water at two different phases: liquid L and vapor V, whose characteristics are defined by means of the scalar function γ:

$$\rho = (1-\gamma)\rho_L + \gamma\rho_V, \qquad \nu = (1-\gamma)\nu_L + \gamma\nu_V. \tag{8}$$

The VOF function γ is numerically calculated in cavitating flow conditions by solving the following transport equation:

$$\frac{\partial\gamma}{\partial t} + \nabla\cdot(\gamma U) + \nabla\cdot[\gamma(1-\gamma)U_r] = \frac{\dot{m}}{\rho_V}. \tag{9}$$

In the context of the present study we employed the cavitation model proposed by Kunz et al. [18], describing the mass transfer term through the following equation:

$$\dot{m} = \dot{m}^+ - \dot{m}^-. \tag{10}$$

Here the vaporization term $\dot{m}^+$ is proportional to the liquid volume fraction in the amount by which the pressure is below the vapor pressure while the transformation of vapor to liquid, $\dot{m}^-$, is a polynomial function of the vapor volume fraction γ:

$$\dot{m}^+ = \frac{C_V \rho_V \gamma \min[0, p - p_{SAT}]}{\left(\frac{1}{2}\rho_L U_\infty^2\right) t_\infty}, \qquad \dot{m}^- = \frac{C_C \rho_V \gamma^2 (1-\gamma)}{t_\infty}. \tag{11}$$

Standard values of $C_C = 1000$ and $C_V = 1000$ have been used.

The numerical solution of the flow governing equations is provided using the open source libraries of OpenFOAM based on a finite volume approach with a collocated arrangement of variables (see [21]). Forces are numerically calculated integrating the pressure distribution and the shear stresses on the hydrofoil surface.

4 Modeling: Gaussian Process Regression

Gaussian process regression is an inductive, non-parametric method of modeling unknown functions from a finite set of training points for the purpose of making predictions. The authoritative reference for GP is *Gaussian Processes for Machine Learning* by C. E. Rasmussen and C. K. I. Williams [23].

To perform a design by optimization problem of the super-cavitating hydrofoil parametrized as shown in section 2.2, we consider a dataset of n observation, $\mathcal{D}$, of inputs and corresponding outputs. Here, independent inputs are different design vectors $\mathbf{x}_i \in \mathbb{R}^D$ (Where D=15) and dependent output are scalar quantities of interest y_i, obtained through numerical solution of the hydrodynamic problem (for instance D/L, C_L, etc.).

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) | i = 1, ..., n\} \tag{12}$$

The input design vectors are collected to form a $D \times n$ design matrix, $\mathbf{X}$, while output scalars are collected to form a output vector, $\mathbf{y}$, of length n. The underlying assumption when building a surrogate model is that there exists an unknown function that relates the input vector and the output value, and this function can be learned with quantified uncertainty even in the presence of noisy data. This relationship is shown in Equation 13.

$$y_i = f(\mathbf{x}_i) + \varepsilon \qquad \text{for} \quad i = 1, 2, ..., n \tag{13}$$

Here, we model the unknown function using Gaussian Process (GP) regression. GP regression assigns an *a priori*, also referred to as prior, probability to every function within the design space. After training data $\mathcal{D}$ is considered, the probability of each function conditioned on the training data can be determined. This is the posterior distribution of functions. However, the set of possible functions within the design space is infinite and each needs to have its probability quantified. To solve this hurdle, the function distribution is assumed to be a specific stochastic process called a Gaussian process. A GP is a generalization of the joint Gaussian probability distribution. Being $f(\mathbf{x})$ a GP, we characterize it by means of its expected value, $m(\mathbf{x})$, and covariance function, (kernel) $k(\mathbf{x}, \mathbf{x}')$. The implication of this is that at any location $\mathbf{x}_i$, $f(\mathbf{x}_i)$ is a random variable with a normal distribution, and for any finite set of locations $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m$, the random variables $f(\mathbf{x}_1), f(\mathbf{x}_2), f(\mathbf{x}_m)$ will have a joint Gaussian distribution.

$$f(\mathbf{x}) \sim \mathcal{N}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \tag{14}$$

In Equation 13, ε is the noise in the observation data and is assumed to be an independent, identically distributed Gaussian distribution with zero mean and variance σ^2.

$$\varepsilon \sim \mathcal{N}(0, \sigma^2) \tag{15}$$

4.1 Prediction

GP regression is initialized by selecting appropriate mean and covariance functions for the prior function space. For convenience, the mean function will be set equal to zero. For the covariance function, any kernel that produces a positive semidefinite covariance matrix (of size $n \times n$) is valid. The mean and covariance function, and the assumed error distribution, are combined to form the GP prior for the target values, as shown in Equation 16.

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, K + \sigma^2 I) \tag{16}$$

The GP prior over all possible functions is then conditioned on the training dataset to produce the posterior.

Since a GP prior is combined with Gaussian noise, the posterior is also a GP that can be analytically determined. In such a way we can infer predictions at the test point $\mathbf{x}_\star$. The joint distribution of the training data and the test data is shown in Equation 17.

$$\begin{bmatrix} \mathbf{y} \\ f(\mathbf{x}_\star) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K + \sigma^2 I & K(X, \mathbf{x}_\star) \\ K(\mathbf{x}_\star, X) & K(\mathbf{x}_\star, \mathbf{x}_\star) \end{bmatrix}\right) \tag{17}$$

The expanded covariance matrix is a composed of blocks: (1) the original covariance matrix of the training data, K, which is size $n \times n$, (2) two covariance vectors between the training data inputs and the testing data inputs, $K(X, \mathbf{x}_\star)$ and $K(\mathbf{x}_\star, X)$, both of which are length n, and (3) the kernel function evaluated at the test point, $K(\mathbf{x}_\star, \mathbf{x}_\star)$, which is a scalar.

To simplify the notation, the vector $K(X,\mathbf{x}_\star)$ will be written as $\mathbf{k}_\star$, the vector $K(\mathbf{x}_\star,X)$ as $\mathbf{k}_\star^T$, the scalar $K(\mathbf{x}_\star,\mathbf{x}_\star)$ as $k_\star$, and the unknown random function $f(\mathbf{x}_\star)$ as $f_\star$. The predictive distribution is conditioned on $\mathbf{x}_\star$ as well as X and $\mathbf{y}$, which is a straightforward operation for Gaussian processes. Therefore, the GP fundamental predictive equation is shown in Equation 18 and is defined by a mean and variance function which are shown in Equations 19, and 20.

$$f_\star|\mathcal{D},\mathbf{x}_\star \sim \mathcal{N}(\mathbb{E}[f_\star],\mathbb{V}(f_\star)) \tag{18}$$

$$\mathbb{E}[f_\star] = \mathbf{k}_\star^T(K+\sigma^2 I)^{-1}\mathbf{y} \tag{19}$$

$$\mathbb{V}(f_\star) = k_\star - \mathbf{k}_\star^T(K+\sigma^2 I)^{-1}\mathbf{k}_\star \tag{20}$$

The predicted quantity of interest value with input $\mathbf{x}_\star$ is found by evaluating Equation 19 and the variance at the test point is found by evaluating Equation 20.

4.2 Covariance Functions

The critical components when training GPs is the choice of the specific form of the covariance function and the vector of hyperparameters that defines it. Covariance function must be produce a covariance matrix that is positive semidefinite and symmetric. The choice of specific kernel functions is based on the specific knowledge that we have of the training data. Frequently used functions include Radial Basis Function (RBF), Rational Quadratic, Periodic, Linear and Constant kernel. The kernel used in this paper is called the automatic relevance determination (ARD) function and is the product of the RBFs.

The Radial Basis Function, also known as the Squared Exponential or Gaussian kernel, is the most ubiquitous kernel. This is because RBF kernels have a number of convenient properties. First, RBF kernels only have two hyperparameters, the signal variance, σ^2, and the lengthscale, l. The signal variance, also called the output variance, is a hyperparameter common to all kernels and magnifies (if greater than one) or reduces (if less than one) the correlation between inputs (see [11]).

$$k_{RBF}(x,x') = \sigma^2 exp\left(-\frac{(x-x')^2}{2l^2}\right) \tag{21}$$

Another kernel property that is used is the fact that the product of two valid kernels is also a valid kernel. To prove this, let $f_1(x)$ and $f_2(x)$ be two independent GPs. The product of two GPs is another GP $f_{product}(x) = f_1(x) \times f_2(x)$ and this GP has the kernel $k_{product}(x,x') = k_1(x,x') \times k_2(x,x')$. Therefore, the GP $f_{product}(x)$ has a valid kernel which is composed by multiplying two other kernels. Therefore, there is no end to the number of or depth of the kernels available for modeling.

The ARD kernel used for this analysis is the product of RBF kernels. This common training method for multi-dimensional problems uses an RBF kernel for each design input and multiplies all these together to compose a single kernel and subsequent covariance matrix. The ARD kernel is shown in Equation 22.

$$k_{ARD}(x,x') = \prod_{i=1}^{D} k_{RBF}(x_i,x_i') \tag{22}$$

ARD gets its name from the lengthscale hyperparameters, l_i which automatically determine how important each feature is with regards to affecting change on the quantity of interest. When the lengthscales are relatively large, then the exponent approaches zero and that feature become irrelevant compared with the other design features. Often, the learned hyperparameters are rewritten as $\lambda_i = \frac{1}{l_i}$ so that irrelevant features will have hyperparameters near zero.

4.3 Optimizing Hyperparameters

Different predictive models can be obtained using different combinations of hyperparameters, $\boldsymbol{\theta}$. Therefore, by varying hyperparameters the model can be improved. The metric used to compare predictive models is the Negative Log Marginal Likelihood (NLML), also called the *evidence*. Integrating the product of the likelihood and the prior over all prior functions, $\mathbf{f}$, yields the Marginal Likelihood, shown in Equation 23. Here, a new notation is introduced to show which variables are dependent on the hyperparameters.

$$p(\mathbf{y}|X,\boldsymbol{\theta}) = \int p(\mathbf{y}|\mathbf{f},X,\boldsymbol{\theta})p(\mathbf{f}|X,\boldsymbol{\theta})d\mathbf{f} \tag{23}$$

The prior is a GP, $\mathbf{f}|X,\boldsymbol{\theta} \sim \mathcal{N}(\mathbf{0},K_\theta)$. The likelihood is also a GP, $\mathbf{y}|\mathbf{f},X,\boldsymbol{\theta} \sim \mathcal{N}(\mathbf{f},\sigma^2 I)$. Two Gaussians multiplied together yields a new Gaussian and the Marginal Likelihood can be determined. Finally, by taking the logarithmic of both sides of the Marginal Likelihood equation and multiplying through by negative one Equation 24 is produced, which is used to calculate the $NLML = -log(p(\mathbf{y}|X,\boldsymbol{\theta}))$.

$$NLML = \frac{1}{2}\mathbf{y}^T(K_\theta+\sigma^2 I)^{-1}\mathbf{y} + \frac{1}{2}log|K_\theta+\sigma^2 I| + \frac{n}{2}log(2\pi) \tag{24}$$

The hyperparameter space is searched for hyperparameters that minimize the NLML, or, conversely, to maximize the Marginal Likelihood.

$$\boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} NLML(\boldsymbol{\theta}) \tag{25}$$

For this paper, the search for hyperparameters was conducted using a quasi-Newton, gradient descent optimization algorithm. This method essentially goes through the following optimization loop: (1) with the current (or initial) hyperparameters determine NLML using Equation 24, (2) calculate

the direction of steepest descent of the NLML using Equation **??**, (3) take step in this direction in the hyperparameter space, and (4) check for convergence. This method depends on the step-size, the initial value of the hyperparameters, and the convergence criteria. While this optimization technique doesn't ensure that the global optimum will be found, the method is well understood, efficient, and can be compared against results found with different initial hyperparameters and random restarts.

4.4 Multi-Fidelity Regression

The flexibility of Gaussian process regression allows for different fidelity levels for the training data. A multi-fidelity dataset may be the result of different types of physics-based computer simulations, varied mesh granularities for the simulations, or combining simulation data with physical experiment results. In general, it is rightly assumed that fidelity level is inversely proportional to "cost." Cost can be measured in computational, temporal, or monetary resources. Overall, the goal is to make high accuracy predictions while training a model with multiple fidelity levels of data. Cheaper low-fidelity data is used to make a better model than that produced solely with expensive high fidelity data. In this paper we consider training data obtained by prediction models characterized by two different fidelity levels. The multi-fidelity training dataset consists of high-fidelity with $\mathcal{D}_H$ and low-fidelity data $\mathcal{D}_L$.

$$\mathbf{y}_H = f_H(\mathbf{x}_H) + \varepsilon_H \tag{26}$$
$$\mathbf{y}_L = f_L(\mathbf{x}_L) + \varepsilon_L \tag{27}$$

Equations (28) and (29) show the zero-mean Gaussian processes that are used. These Gaussian processes are independent of each other, $f_L \perp \delta_H(\mathbf{x})$.

$$f_L(\mathbf{x}) \sim \mathcal{GP}(0, k_L(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}_L)) \tag{28}$$
$$\delta_H(\mathbf{x}) \sim \mathcal{GP}(0, k_H(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}_H)) \tag{29}$$

Equation (30) shows the assumed function distribution relationships in terms of the independent, zero-mean GPs defined above. These model relationships were used by Kennedy and O'Hagan [17], Bonfiglio et al. [6], [7], and Perdikaris and Karniadakis [22]. For these relationship, a regression scaling parameter, ρ, is also introduced.

$$f_H(\mathbf{x}) = \rho f_L(\mathbf{x}) + \delta_H(\mathbf{x}) \tag{30}$$

The high-fidelity input/output equation, Equations 27, is combined with Equation 30 to produce Equation 31.

$$\mathbf{y}_H = \rho f_L(\mathbf{x}) + \delta_H(\mathbf{x}) + \varepsilon_H \tag{31}$$

Therefore, the joint distribution with both levels of data is shown in Equation 32, where $\mathbf{y}^T = [\mathbf{y}_L^T \mathbf{y}_H^T]$.

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_L \\ \boldsymbol{y}_H \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \boldsymbol{K}_{LL} & \boldsymbol{K}_{LH} \\ & \boldsymbol{K}_{HH} \end{bmatrix} \right) \tag{32}$$

Where the blocks within the matrix described Equation 32 are defined as:

$$\begin{aligned} \boldsymbol{K}_{LL} &= k_L(\boldsymbol{x}_L, \boldsymbol{x}_L'; \theta_L) + \sigma_{\varepsilon_L}^2 \boldsymbol{I} \\ \boldsymbol{K}_{LH} &= \rho k_L(\boldsymbol{x}_L, \boldsymbol{x}_H; \theta_L) \\ \boldsymbol{K}_{HH} &= \rho^2 k_L(\boldsymbol{x}_H, \boldsymbol{x}_H'; \theta_L) + k_H(\boldsymbol{x}_H, \boldsymbol{x}_H'; \theta_H) + \sigma_{\varepsilon_H}^2 \boldsymbol{I} \end{aligned} \tag{33}$$

Let K_{mf} be the size $(n_L + n_H) \times (n_L + n_H)$ multi-fidelity block covariance matrix of $\mathbf{y}$. Each diagonal block of the covariance matrix is used to model data from a single fidelity data class. The non-diagonal blocks are used for modeling correlations between classes of data. The optimum set of hyperparameters and model parameters (see Equation (34)), $\boldsymbol{\Theta}$, are learned using the training data set.

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_L, \boldsymbol{\theta}_H, \rho, \sigma_L^2, \sigma_H^2] \tag{34}$$

The NLML, predicted function value, and predictive variance equations now use the block covariance matrix, K_{mf}, and are shown below.

$$\mathbb{E}[f_\star] = \mathbf{k}_\star^T K_{mf}^{-1} \mathbf{y} \tag{35}$$
$$\mathbb{V}(f_\star) = k_\star - \mathbf{k}_\star^T K_{mf}^{-1} \mathbf{k}_\star \tag{36}$$

$$\begin{aligned} \boldsymbol{\Theta}^\star &= \arg\min_{\boldsymbol{\Theta}} NLML(\boldsymbol{\Theta}) \\ &:= \frac{1}{2} \mathbf{y}^T K_{mf}^{-1}(\boldsymbol{\Theta}) \mathbf{y} + \frac{1}{2} log|K_{mf}(\boldsymbol{\Theta})| + \frac{n}{2} log(2\pi) \end{aligned} \tag{37}$$

5 Optimizing: Bayesian Optimization

Once GP surrogate models for each quantity of interest considered for the design of the super-cavitating hydrofoil are obtained, we aim to efficiently search the design space in order to discover optimal candidate designs characterized by maximum hydrodynamic efficiency and respecting mission constraints. Global optimization is often computationally expensive requiring many function evaluations. The goal is to use the predictive model instead of physics-based solver to explore the design space. A potent tool for exploring the design space and searching for the global minimum while observing a computational budget is Bayesian optimization [20]. This block-box optimizer uses the predictive response surface, in terms of posterior mean and uncertainty, and an acquisition function to find a minimum within the design space. Instead of minimizing the quantity of interest directly, the acquisition function, also referred to as the

utility function, is maximized (or minimized, based on utility function). There are a number of available acquisition functions, but the acquisition function used for the analysis described in this paper is a variation of the Expected Improvement (EI) acquisition function called the Constrained Expected Improvement (CEI) acquisition function. The EI acquisition function balances exploring and exploiting the response surface [16]. It explores the model by giving credit to areas within the design space that have high uncertainty. Simultaneously, it exploits the model by giving credit to areas within the design space that have desirable quantity of interest values. Section 5.1 gives the theory for the general EI utility without constraints and Section 5.2 explains how the EI utility function can be used when inequality constraints are present [12].

5.1 Expected Improvement Acquisition Function

As stated above the goal of Bayesian optimization with an EI acquisition function is to search the predictive model for the global minimum and for areas within the model with high uncertainty. The engineering design problem is to minimized the objective value, as shown in Equation 38, but BO will instead search for a new design that maximizes the utility function, as shown in Equation 39

$$\mathbf{x}^{\star} = \arg\min_{\boldsymbol{x}} f_{\star}(\mathbf{x}) \tag{38}$$

$$\mathbf{x}_{n+1} = \arg\max_{\boldsymbol{x}} \alpha_{EI}(\mathbf{x}) \tag{39}$$

To develop the EI utility function, first the Improvement function is defined in Equation 40. Since the prediction from the surrogate model is a random variable, the Improvement function is also a random variable. Therefore, taking the expected value of the Improvement function yields the Expected Improvement acquisition function. Let $\mathbf{x}^{\triangle}$ be the design from the current dataset with the best observed quantity of interest, referred to as the "best" design, and $\mathbf{x}^{+}$ be the candidate design being tested.

$$I(\mathbf{x}^{+}) = max(0,\, f(\mathbf{x}^{\triangle}) - f_{\star}(\mathbf{x}^{+})) \tag{40}$$

$$\alpha_{EI}(\mathbf{x}^{+}) = \mathbb{E}[I(\mathbf{x}^{+})] \tag{41}$$

A closed form of Equation 41 was developed by Jones et al. [16] and is shown in Equation 42. In Equation 42, ϕ is the standard Gaussian probability density function and Φ is the standard Gaussian cumulative distribution function. Also, let Z_f be the quotient of the difference between the mean predicted value at $\mathbf{x}^{+}$ and the observed "best" value and the variance at the predicted value. Z_f is defined in Equation 43.

$$\begin{aligned}\alpha_{EI}(\mathbf{x}^{+}) = \Phi(Z_f)\left(\mathbb{E}[f_{\star}(\mathbf{x}^{+})] - f(\mathbf{x}^{\triangle})\right) \\ + \phi(Z_f)\mathbb{V}[f_{\star}(\mathbf{x}^{+})]\end{aligned} \tag{42}$$

$$Z_f = Z_f(\mathbf{x}^{+}) = \frac{\mathbb{E}[f_{\star}(\mathbf{x}^{+})] - f(\mathbf{x}^{\triangle})}{\mathbb{V}[f_{\star}(\mathbf{x}^{+})]} \tag{43}$$

In Equation 42 note the two components that are combined to form the EI utility function. The first term gives credit for improving upon the best design: *exploitation term*. The second term gives credit where there is high uncertainty. At first this may seem odd, but this is the *exploration term*. The exploration component gives credit to regions which need more data to improve the model.

To explore the tradespace using BO, the design space is sampled, the EI utility function and gradient are calculated, and the design follows the path of greatest ascent until a maximum is reached. In each iteration, the maximum EI reached for each restart is stored, the maximums are compared, and the design with the greatest EI is the next design simulated with the physics-based solver. This new input/output pair is added to the training dataset and the model is re-trained. BO can be considered a "greedy" optimization methodology, since with each iteration either the "best" design or the model is improved.

5.2 Constrained Expected Improvement Acquisition Function

When constraints are imposed upon the design selection, the original EI acquisition function will most likely nominate infeasible designs. Therefore the probability of satisfying all the constraints must be factored into the BO EI methodology. Let c_k be constraint k and assume all constraints are of the form $c_k \geq 0$. Constraints not originally expressed in this manner can be mathematically rewritten into this form, as shown below. The EI acquisition function value found using the predictive model for the quantity of interest is then adjusted based on the constraint surrogate model values. However, the constraint models provides mean and variance function values, just as the object regression model. Therefore, there will never be certainty whether a design is feasible or infeasible using the predictive models, but rather a probability of satisfy all the constraints. These probabilities are multiplied with the EI acquisition function defined in 42 to produce the Constrained Expected Improvement (CEI) acquisition function. The probabilities are able to be simply multiplied together because the models, and therefore the probabilities, are assumed to be independent.

$$\alpha_{CEI}(\mathbf{x}^{+}) = \alpha_{EI}(\mathbf{x}^{+}) \prod_{k=1}^{K} Pr\left(c_k(\mathbf{x}^{+}) \geq 0\right) \tag{44}$$

There is now a Z_k value for each constraint k, as shown in Equation 46, in addition to the Z_f defined in 43. Finally, while the unconstrained EI function defined the "best" design as the design with the minimum engineering objective value, $\mathbf{x}^{\triangle}$, with the inclusion of constraints in design problem, the "best" design is redefined as the observed design with the minimum engineering objective value that satisfies all constraints. While they are defined differently, the same

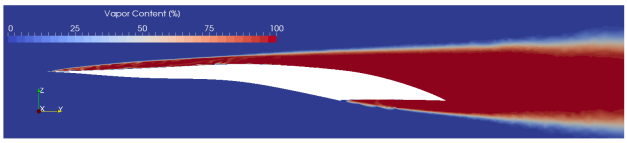
notation will be used for the "best" observed design. The Constrained Expected Improvement acquisition function is defined in Equation 45.

$$\alpha_{CEI}(\mathbf{x}^+) = \alpha_{EI}(\mathbf{x}^+) \prod_{k=1}^{K} \Phi(Z_k) \tag{45}$$

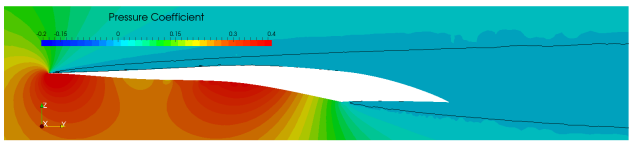
$$Z_k = Z_k(\mathbf{x}^+) = \frac{\mathbb{E}[c_k(\mathbf{x}^+)] - 0}{\mathbb{V}[c_k(\mathbf{x}^+)]} = \frac{\mathbb{E}[c_k(\mathbf{x}^+)]}{\mathbb{V}[c_k(\mathbf{x}^+)]} \tag{46}$$

6 Super- Cavitating Hydrofoil Design Analysis

The multi-fidelity framework for super-cavitating hydrofoils performance predictions was leveraged to generate a multi-fidelity training data set used to formulate GP regression models. In particular, the design space was sampled using a Latin Hypercube Sampling (LHS) algorithm (see [15]) identifying 442 design suitable for low-fidelity predictions and 111 hydrofoils to be evaluated using high-fidelity URANS simulations. From the initial dataset, a benchmark design was established. The benchmark design was the design with the greatest Lift-Drag ratio and satisfied all the constraints. The benchmark design has a Lift-Drag ratio of 10.76 and is shown in Figure 4 in terms of vapor content (Figure 4a) and pressure (Figure 4b) contours.



(a) Vapor Content of benchmark hydrofoil



(b) Pressure Distribution of benchmark hydrofoil

Fig. 4: Benchmark hydrofoil pressure and vapor distributions

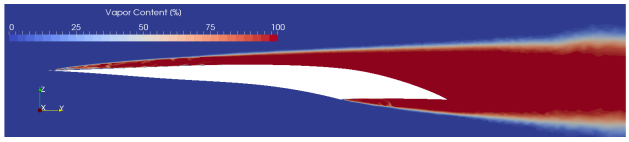
6.1 Automatic Design Loop

The 2D super-cavitation hydrofoil automatic design loop is shown in Figure 13, providing an overview of the automatic design process, including the input and output files for each step and a short description. We initiated the design loop by training surrogate models using the multi-fidelity training dataset. Training the surrogate models requires four inputs: 1) a low-fidelity design matrix, X_L, 2) a high-fidelity design matrix, X_H, 3) a low-fidelity simulation results matrix, y_L, and 4) a high-fidelity simulation results matrix, y_H. As described in Section 4, training a model means identifying the set of hyperparameters that minimizes the negative log-marginal likelihood of the model. Once the optimum hyperparameters are determined, the surrogate model has been found for that QoI.

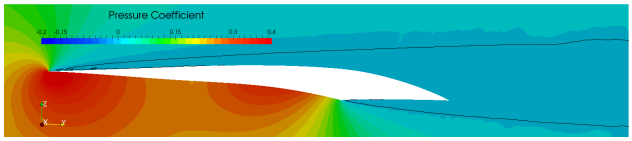
Using the most recently trained surrogate models, the design space is explored for new candidate designs using the Bayesian Optimization method described in section 5. Figure 13 shows the design loop with the Bayesian optimization search method. When initializing the automatic design loop, the human designer must choose which search paradigm to utilize. Regardless of the search method, the top candidate is saved and represents the input into the Simulation Environment.

In this study, the simulation environment is the high-fidelity URANS solver (low-fidelity solver was only used to generate part of the initial training set). The simulation environment has two inputs: 1) the candidate design vector, **x**, and 2) an operating conditions vector, **v**. The output for the simulation environment are the time-averaged QoIs over steady-state time intervals.
In the first module, Geometry Setup, the inputs x and v are used to generate the input files for meshing engine (GMSH, see [13]), which are a 2D geometry data file (geometry.dat) and the mesh instructions. Next, the mesh instructions, 2D geometry, and operating condition vector are input into the Mesh Generation module. This module will produce a mesh file (geometry.msh) compatible with the CFD solver (OpenFOAM, see [21]). Next, the CFD Interface module will use the mesh file and the operating condition vector to automatically set up the initial and boundary conditions, the fluid characteristics, the flow model, and the numerical solvers and schemes necessary for the flow prediction. These output will be compiled into a single data file (inputCFD.dat). The inputCFD.dat file is the input into the CFD Solution module which will calculate the CFD solution, storing each QoI data at each time step, by utilizing OpenFOAM libraries. The outputs from this module are time-step folders containing field variables and integral quantities, such as forces (force.dat and forceCoeffs.dat). Finally, the QoI Prediction module is a post-processing tool that takes the CFD Solution outputs and produces the QoIs for the design vector, x, under the operating conditions, v. More details on the simulation environment such as mesh generation strategy and solver settings can be found in [5] and [4].

After each iteration, the "best" design is compared with the simulation environment output. If the newest design that was simulated has in improved Lift-to-Drag ratio and satisfies all the constraint, it replaces the "best" design and is used in future iterations.

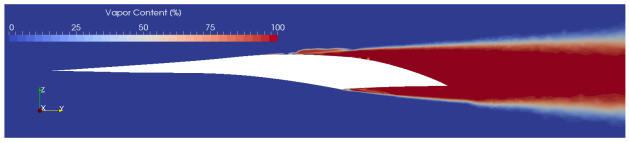


(a) Vapor Content Contours

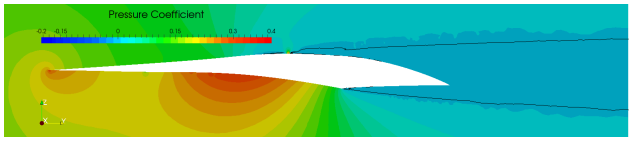


(b) Pressure Distribution Contours

Fig. 5: Best hydrofoil candidate: Iteration 4-5



(a) Vapor Content Contours

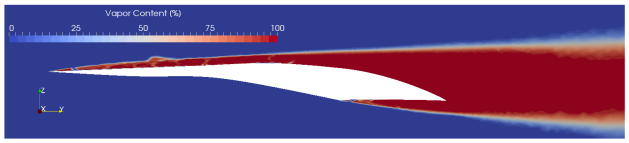


(b) Pressure Distribution Contours

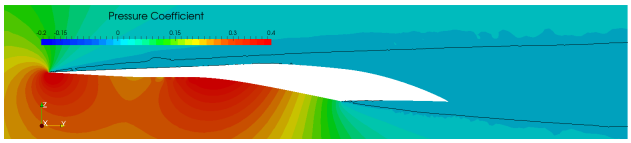
Fig. 6: Best hydrofoil candidate: Iteration 6

Table 2: Results from Phase 1

Iteration	D/L	L/D	*PercentImprovement*
0	0.092902789	10.76	0.00%
⋮	⋮	⋮	⋮
4	0.091376341	10.94	1.67%
⋮	⋮	⋮	⋮
6	0.090972396	10.99	2.12%
7	0.088483169	11.30	4.99%
⋮	⋮	⋮	⋮
48	0.088079071	11.35	5.48%



(a) Vapor Content Contours



(b) Pressure Distribution Contours

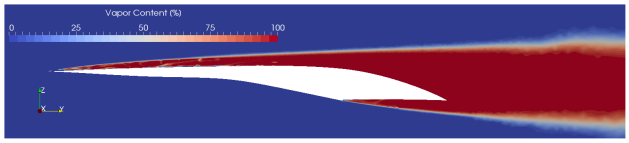
Fig. 7: Best hydrofoil candidate: Iteration 7-47

6.2 Results

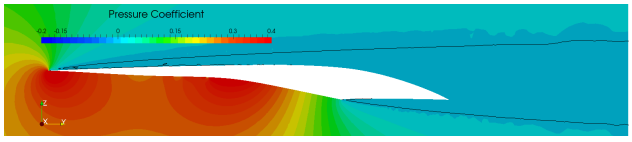
The automatic design loop was implemented in two phases: Phase 1 is a coarse design search and Phase 2 is a fine design search.

Phase 1 Phase 1 started with the initial dataset and trained surrogate models, and iterated through the automatic design loop 100 times. In each iteration, the number of Bayesian optimization restarts was 1000. The number of BO restarts is the number of random design points that are used to explore and search the surrogate models. A higher number of BO restarts indicates a more refined search of the model. In Phase 1, the number of BO restarts is moderate, since the goal of Phase 1 is to improve the models and make obvious improvements upon the benchmark design. The results from Phase 1 are shown in Table 2.

Phase 2 Phase 2 started with the final dataset and trained surrogate models from Phase 1, and iterated through the design loop five times. In each iteration, the number of Bayesian optimization restarts was 1,000,000. The number of BO restarts in Phase 2 is 99900% greater than the Phase 1 value. This is consistent with the goal of Phase 2, which is to use the improved models created in Phase 1 to finely search the models for global improvements of the hydrofoil design. The results from Phase 2 are shown in Table 3.



(a) Vapor Content Contours

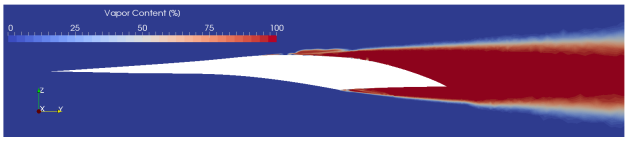


(b) Pressure Distribution Contours

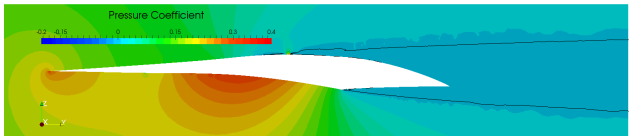
Fig. 8: Best hydrofoil candidate: Iteration 48-100

Table 3: Results from Phase 2

Iteration	D/L	L/D	*PercentImprovement*
0	0.092902789	10.76	0.00%
⋮	⋮	⋮	⋮
100	0.088079071	11.35	5.48%
101	0.085346869	11.717	8.85%
102	0.085283969	11.726	8.93%
103	0.085283969	11.726	8.93%
104	0.085283969	11.726	8.93%
105	0.085283969	11.726	8.93%

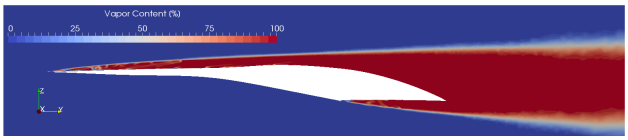


(a) Vapor Content Contours

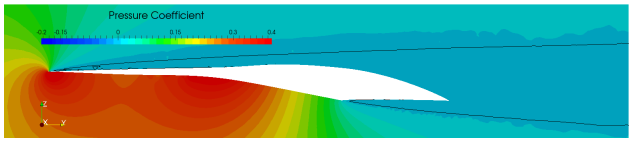


(b) Pressure Distribution Contours

Fig. 9: Best hydrofoil candidate: Iteration 101



(a) Vapor Content Contours



(b) Pressure Distribution Contours

Fig. 10: Best hydrofoil candidate: Iteration 102-105

Figures 5 - 10 presents vapor content and pressure coefficient contours for best-so-far hydrofoils obtained during the automatic optimization loop. During the first iterations the thickness of the hydrofoil close to the leading edge is gradually reduced, as evident from comparing Figure 5, 6, and 7. Figure 6 demonstrates the probabilistic nature of the design framework: even though one constrain acts on the thickness of the cavitation pocket, the best-so-far design identified using surrogate models do not properly predict the quantity of interest corresponding to the cavity thickness.

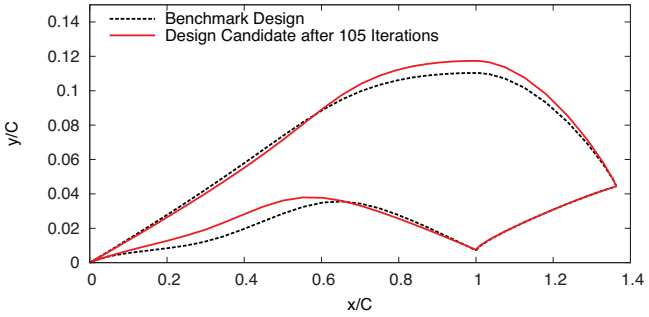


Fig. 11: Comparison between benchmark (black dashed line) and optimal design (red solid line). The optimal design has been identified after 105 iterations of the Bayesian Optimization loop presented in the present study.

Figure 11 presents a comparison between the shape of the benchmark hydrofoil, best performing design among the initial training dataset and the candidate design, obtained after 105 iteration of the Bayesian Optimization loop previously described. The shape of the optimal hydrofoil presents a thinner leading edge that has the effect of reducing the drag on the pressure surface while producing high pressure over an extensive region on the hydrofoil face downstream the leading edge. The reduced thickness of the hydrofoil close to the leading edge causes a reduction of the inertia of the 2-D section, which is recovered on the hydrofoil suction surface, close to the trailing edge. In design conditions, the hydrofoil suction surface is fully enveloped in a vapor cavity, hence its shape does not significantly affect the hydrodynamic efficiency of the hydrofoil.

Figure 12 presents quantitative results in terms of hydrodynamic efficiency improvement achieved at different iteration of the automatic loop presented in this study. Blue solid line shows the optimization path followed during phase 1, in which we used a coarse search with the number of BO restarts has been set to 1,000. Red solid line presents the optimization path followed during phase 2, in which we used a finer search with the number of BO restarts set to 1,000,0000. As evident in figure 12, phase 1 allowed an efficiency increase larger than 5% with respect to the best hydrofoils of the training data set, while a finer search allowed to further increase the accuracy of approximately 3%. The trade-off between exploration and exploitation is clearly evident in figure 12, where regions of constant hydrodynamic efficiency improvements (horizontal segments) correspond to exploration iterations in which the BO algorithm select new designs with the goal of reducing prediction uncertainty. The vertical paths correspond to exploitation iterations in which the learning algorithm requires a URANS simulation after identifying a promising region of the design space.

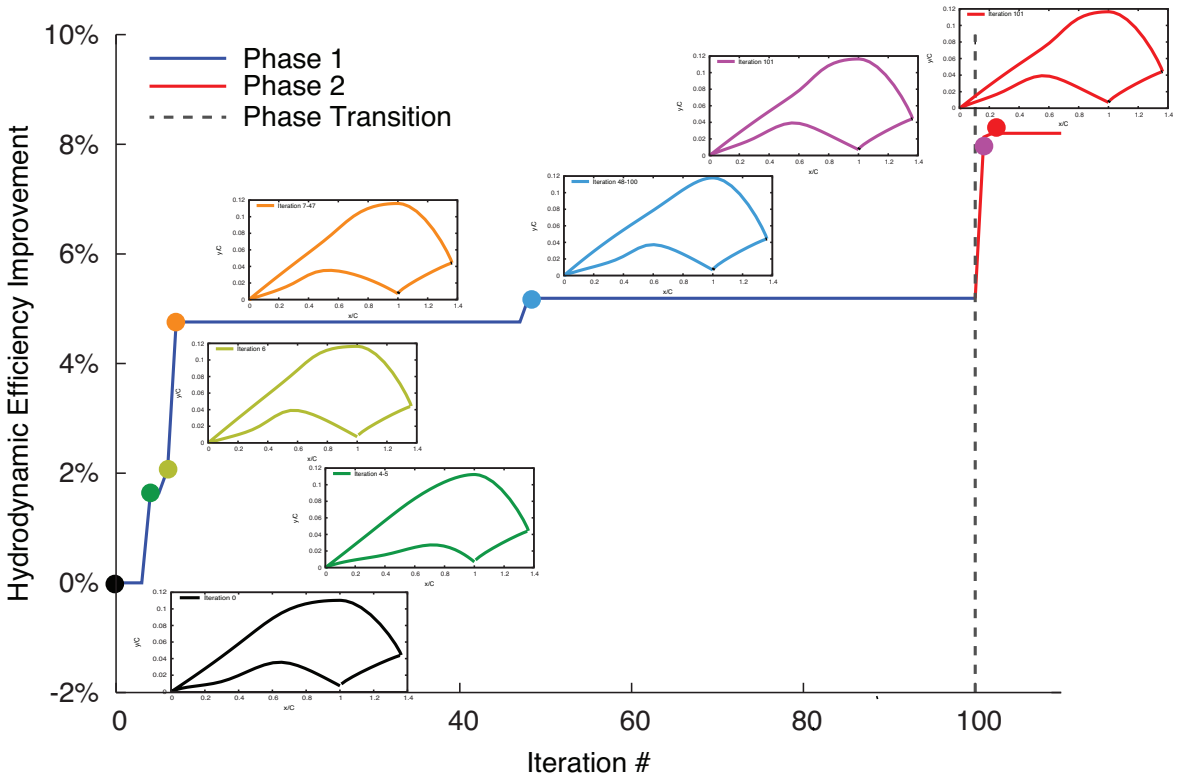


Fig. 12: Phases 1 and 2: Percent Improvement

7 Conclusion

This paper demonstrates the effectiveness of using an automated engineering design loop which includes machine learning techniques and physics-based solvers to improve the design of a super-cavitating hydrofoil. The machine learning techniques applied to the hydrofoil design challenge included Gaussian process regression, multi-fidelity regression, and Bayesian optimization.

A new hydrofoil design was discovered that improved the Lift-to-Drag ratio by more than 8.8%. This was done automatically, without spending human resources or introducing potential human bias to the final solution. This analysis was accomplished in 105 iterations, with more than 5.1 million designs being explored in the simulation environment. The final design results are shown in Figure 11.

Since this field of research is promising and continuing to grow, there are plenty areas for future study. Specific to the topics discussed in this these, the following concepts are worthy of future study: (1) include an optimized kernel search within an automatic design loop, (2) incorporate multi-disciplinary optimization in an automatic design loop, and (3) compare the results of automatic design loops with different modeling and optimization methodologies.

Acknowledgements

I am grateful to Luca Bonfiglio and Paris Perdikaris from MIT's Sea Grant for being encouraging, understanding, and approachable. This paper would not have been possible without their teaching and guidance. I am also thankful for Professor George Em Karniadakis from Brown University for his big-picture ideas and direction, and for my thesis advisor Prof. Michael Triantafyllou for his flexibility and patience.

Biography

LT Benjamin Parker is an Engineering Duty Officer in the United States Navy. He is currently assigned to the Massachusetts Institute of Technology for graduate education in Naval Engineering and will be transferring to Puget Sound Naval Shipyard and Intermediate Maintenance Facility in Bremerton, WA upon graduation. He received a master's degree in Systems Engineering from Colorado State University in 2014, a master's degree in Applied Mathematics from University of Washington in 2010, and a bachelor's degree in Mathematics from Northwest University in 2007.

References

[1] A. J. Acosta. A note on partial cavitation of flat plate hydrofoils. Technical Report E-19.9, California Institute of Technology, 1955.

[2] A. J. Acosta. Hydrofoils and hydrofoil craft. *Annual Review of Fluid Mechanics*, 5:161–184, 1973.

[3] J. Auslaender. Low drag supercavitating hydrofoil sections. *Technical Report Tech. Rep. 001-7, Hydronautics Inc., Laurel, MD.*, 1962.

[4] L. Bonfiglio and S. Brizzolara. A multiphase RANSE-based computational tool for the analysis of supercavitating hydrofoils. *Naval Engineers Journal*, 128(1):47–64, 2015.

[5] Luca Bonfiglio and Stefano Brizzolara. A Numerical Investigation over the Cavitating Flow Regime of a 2D-Hydrofoil. In *13th International Conference on Fast Sea Transportation, FAST 2015. 1-4 September, Washington D.C., USA*, 2015.

[6] Luca Bonfiglio, Paris Perdikaris, and Stefano Brizzolara. Multi-Fidelity Optimization of high speed SWATHs . 2004.

[7] Luca Bonfiglio, Paris Perdikaris, Stefano Brizzolara, and George Em Karniadakis. A multi-fidelity framework for investigating the performance of supercavitating hydrofoils under uncertain flow conditions. *AIAA Non-Deterministic Approaches*, pages 1–13, 2017.

[8] S Brizzolara. A new family of dual-mode supercavitating hydrofoils. In *Proceedings of the Fourth International Symposium on Marine Propulsors*, 2015.

[9] S Brizzolara and L Bonfiglio. Comparative cfd investigation on the performance of a new family of supercavitating hydrofoils. In *Journal of Physics: Conference Series*, volume 656, page 012147. IOP Publishing, 2015.

[10] Stefano Brizzolara. Watercraft device, 2014. US Patent 8,763,546.

[11] Alexander Forrester, Andy Keane, et al. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

[12] Jacob R. Gardner, Matt J. Kusner, Zhixiang Eddie Xu, Kilian Q. Weinberger, and John P. Cunningham. Bayesian Optimization with Inequality Constraints. *Proceedings of the 31st International Conference on Machine Learning (ICML 2014)*, 32:937–945, 2014.

[13] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331, 2009.

[14] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981.

[15] Ronald L Iman, James M Davenport, and Diane K Zeigler. Latin hypercube sampling (program user's guide).[lhc, in fortran]. Technical report, Sandia Labs., Albuquerque, NM (USA), 1980.

[16] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization*, 13:455–492, 1998.

[17] M. C. Kennedy and A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000.

[18] Robert F Kunz, David A Boger, David R Stinebring, Thomas S Chyczewski, Jules W Lindau, Howard J Gibeling, Sankaran Venkateswaran, and T. R. Govindan. A preconditioned Navier–Stokes method for two-phase flows with application to cavitation prediction. *Computers & Fluids*, 29(8):849–875, 2000.

[19] B.E. Launder and D.B. Spalding. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3(2):269–289, 1974.

[20] J Mockus. Bayesian Heuristic approach to discrete and global optimization: Algorithms, visualization, software, and applications. *Technometrics*, 41(4):378–379, 2010.

[21] OpenFOAM Foundation. *OpenFOAM user guide*. OpenFOAM Foundation, 2014.

[22] Paris Perdikaris and George Em Karniadakis. Model inversion via multi-fidelity Bayesian optimization: a new paradigm for parameter estimation in haemodynamics, and beyond. *Journal of The Royal Society Interface*, 13(118):20151107, 2016.

[23] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. 2006.

[24] J. O. Royset, L. Bonfiglio, G. Vernengo, and S. Brizzolara. Risk-Adaptive Set-Based Design and Applications to Shaping a Hydrofoil. *Journal of Mechanical Design*, 139(10):101403, 2017.

[25] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

[26] Giuliano Vernengo, Luca Bonfiglio, Stefano Gaggero, and Stefano Brizzolara. Physics-based design by optimization of unconventional supercavitating hydrofoils. *Journal of Ship Research*, 60(4):187–202, 2016.

[27] Giuliano Vernengo, Luca Bonfiglio, Stefano Gaggero, and Stefano Brizzolara. Physics-Based Design by Optimization of Unconventional Supercavitating Hydrofoils. *Journal of Ship Research*, 60(4):187–202, 2016.

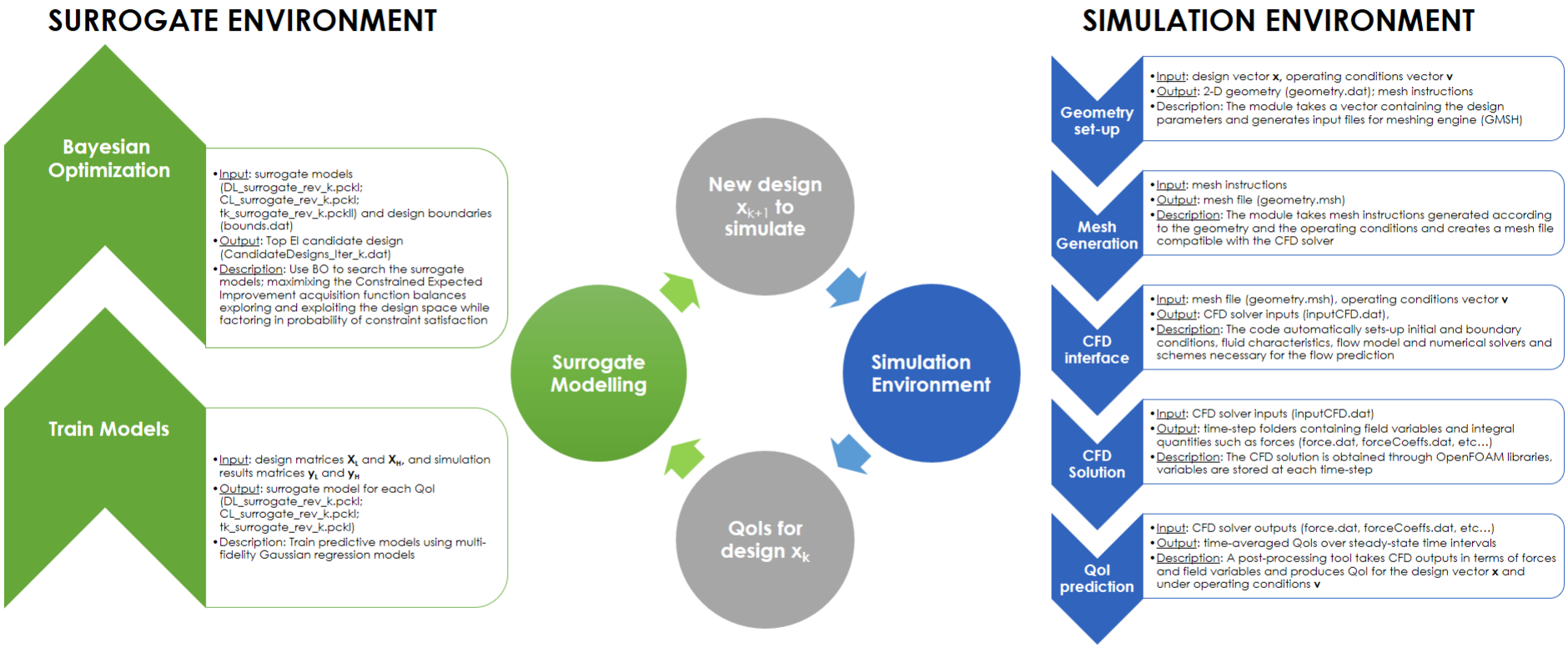


Fig. 13: Automatic Design Loop for 2D Super-Cavitating Hydrofoil