Journal of *Marine Science and Engineering*

MDPI

Article

Optimising General Configuration of Wing-Sailed Autonomous Sailing Monohulls Using Bayesian Optimisation and Knowledge Transfer

Yang An [1,2,3,*], Feng Hu [1,2], Kuo Chen [1,2,3] and Jiancheng Yu [1,2,*]

[1] State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; hufeng@sia.cn (F.H.); chenkuo@sia.cn (K.C.)
[2] Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China
[3] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: anyang@sia.cn (Y.A.); yjc@sia.cn (J.Y.)

Abstract: Wing-sailed autonomous sailing monohulls are promising platforms used in various scenarios to provide data for marine science research. These platforms need to operate long-term in changing seas; their general configurations (size matching between sail, hull, and keel) necessitate careful trade-offs to balance safety and efficiency. Since autonomous sailboats are often designed for different observation missions, scientific pay-loads and target areas, their design space is considerably large. It is also challenging to obtain prior performance estimation from historical designs. Therefore, traditional offline surrogate-based simulation-driven design frameworks suffer from a large amount of sampling required, the computational cost of which remains too expensive for such ad hoc design tasks. This paper proposes an innovative, generalised simulation-driven framework combining Bayesian optimisation and knowledge transfer. It allows for high-quality, low-cost optimisation of autonomous sailing monohulls' general configuration without initial design and prior performance estimation. The proposed optimisation framework has been used to optimise the 'Seagull' prototype within the design constraints. The optimised design exhibits significant performance improvements. At the same time, the results show that the present method is significantly superior to traditional offline methods. The authors believe that the proposed framework promises to provide the autonomous sailing community with a solution for a general design methodology.

Keywords: autonomous sailboat; simulation-based design; Bayesian optimisation; surrogate-assisted evolutionary algorithm

Citation: An, Y.; Hu, F.; Chen, K.; Yu, J. Optimising General Configuration of Wing-Sailed Autonomous Sailing Monohulls Using Bayesian Optimisation and Knowledge Transfer. *J. Mar. Sci. Eng.* **2023**, *11*, 703. https://doi.org/10.3390/jmse11040703

Academic Editor: Gerasimos Theotokatos

Received: 7 March 2023
Revised: 20 March 2023
Accepted: 23 March 2023
Published: 24 March 2023

1. Introduction

Wing-sailed autonomous sailing monohulls are promising platforms for ocean data collection with good endurance, durability, and manoeuvrability [1]. In recent years, they have played vital roles in practical tasks such as ocean floor mapping [2], marine biological surveys [3], long-term ocean observations [4,5] and water mass tracking [6]. Without competent motivated crews and structurally reliable area-adjustable sails, small sails [7,8] and long and heavy keels [9,10] are often equipped to ensure the platform will not heel excessively changeable conditions. Ref. [11] states that general configurations (size matching between sail, hull, and keel) of wing-sailed autonomous sailing monohulls are usually designed empirically, resulting in excess overturning resistance (in terms of both overturning moment generation and stability) but weak speed performance. Since gales are infrequent [12], weak speed performance can weaken their passability in areas of strong currents, deteriorate the ability to track water masses and reduce the efficiency of long-distance observations [13,14] most of the time. Systematic optimisation of the general configuration is necessary to further unleash their speed potential while ensuring safety.

Optimising the general configuration has been an active topic of research in crewed (racing) sailboats. The basic framework is a double-loop process, as shown in Figure 1 [15,16]. For each general configuration, the inner loop–velocity prediction program (VPP) [17–20] was used to obtain its speed performance under (one or several) external conditions (including true wind speed, TWS and true wind angle, TWA) of interest. The design parameters are then adjusted iteratively in the outer loop according to the performance metrics to obtain a Pareto front or an optimal design. Depending on the source of the force model in the VPP, the framework can be divided into two categories: experimentally, with towing tanks and wind tunnels, or numerically, by computational fluid dynamics (CFD) simulations. Wing-sailed autonomous sailing monohull designs are usually an ad hoc task for mission scenarios, and the shape of the hull and keel may be "non-standard". So, it seems more sensible to adopt CFD simulations on hydrodynamic forces than to conduct a multitude of prohibitive tank tests.

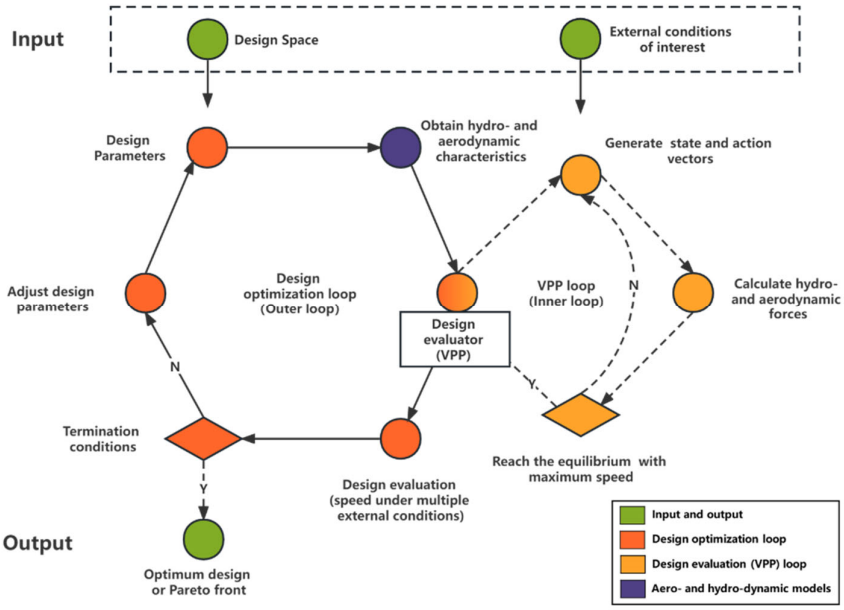

Figure 1. General configuration optimisation framework for a typical sailboat. Note: the same colours will be used in later flowcharts to aid better understanding.

Compared to other design problems (e.g., ships or AUVs design optimisation), CFD-based simulation-driven optimisation of sailboats is naturally more computationally costly. This is because in the former, the merit of a design can be obtained with only one simulation (one resistance assessment), whereas in the latter, the merit evaluation process (VPP itself) is a loop, with more simulations required. Furthermore, autonomous sailboat design suffers more from computational cost problems than crewed sailboats. The main reason is that most performance-sensitive crewed sailboat design requirements come from regattas, class rules limit racing boats, and many historical regatta data are available for reference. Therefore, the optimisation process only needs to explore limited design space by performing several simulations concentrated around the performance predictions per design. Then, offline surrogates that require no update were constructed by sampling the cross matrix to mimic aerodynamic and hydrodynamic characteristics [21–23], as Figure 2 shows. However, for wing-sailed autonomous sailing monohulls, the specific mission requirements, the "non-standard" hull and keel, and the scale effects [24,25] make it difficult for designers to give both a good baseline design and a rough performance estimation. If offline surrogate-assisted optimisation is employed, a massive number of simulations are required to construct an accurate global model for each design. Our previous work proposed a CFD-compatible VPP of wing-sailed autonomous sailing monohulls [26]; in the case study, we used about 150 CFD simulations to model the specific (one) design of

our prototype. Considering that each simulation can take several to tens of hours for an optimisation task, the computational costs are clearly not practical for engineering with a large design space.

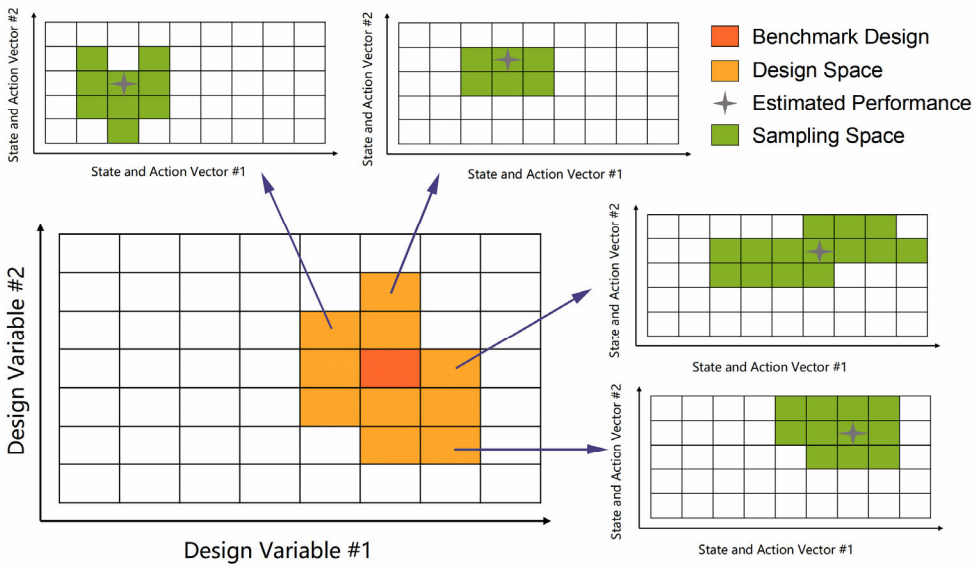

Figure 2. Schematic diagram of samplings of general configuration optimisation of a crewed racing sailboat. The design space of a crewed sailboat is limited. For each design, there is also good enough prior information on performance. Thus, only a relatively small amount of CFD sampling is required.

To maximise the speed performance of wing-sailed autonomous sailing monohulls within the constraints of safety (heeling), the paper outlines a novel optimisation framework for the general configuration. The framework effectively reduces the computational cost by employing two surrogates. Knowledge transfer surrogates reuse CFD-generated data to provide prior information, while the Kriging surrogate combined with Bayesian optimisation methods ensures that the added sampling is concentrated around the equilibrium points of interest. The sail design will be optimised on the prototype 'Seagull' to verify the effectiveness of the method and to improve the speed performance of the platform. The authors believe that the proposed framework holds the promise of providing the autonomous sailboat community with a generic, systematic and engineering-implementable design and optimisation method. Additionally, it will lead to significant improvements in the performance of subsequent platforms in the future, making autonomous sailboats a more effective data collection platform for marine science.

This paper is organised into five sections. Following this introduction, Section 2 describes the optimisation problem and expounds on the motivation to apply Bayesian optimisation and knowledge transfer. The implementation of the optimisation framework is detailed in Section 3. Section 4 presents the case study of the wing sail design of the 'Seagull' prototype and illustrates the results. Finally, conclusions are drawn in Section 5.

2. Motivations

2.1. Problem Description

The optimisation aims to find the Pareto front of the speed metric in a bounded design space S under a set of conflicting external conditions (e.g., upwind with large TWS and downwind with small TWS) $C = \{c_i\}$, as shown in Equation (1), where d is the set of design variables for a particular design, and c_i is the set of specific TWA and TWS in the inertia system of the platform.

Ref. [11] states that to ensure overturning resistance in headwinds with large TWS, most designers sacrifice the overall speed metric of the platform, especially in downwind

conditions with small TWS. Therefore, we optimise two different operating points while limiting the maximum heel angle, i.e., $i = 2$ in this paper.

$$P(d) = \underset{d \in S}{\operatorname{argmax}}\, v_{s\_i} \tag{1}$$

The process is shown in Figure 3. For each design d, the speed metric v_{s_i} under c_i can be obtained from the design evaluator VPP, as shown in Equation (2). More specifically, the VPP obtains the forces (and moments) by aero- and dynamical models of specific design d under a specific external environment c_i, speed v_s, attitude $attitude$, and control quantities denoted $action$, as shown in Equations (3) and (4):

$$v_{s\_i} = f_{VPP}(d, c_i) \tag{2}$$

$$F_{areo} = f_{areo}(d, c_i, v_s, attitude, action) \tag{3}$$

$$F_{hydro} = f_{hydro}(d, c_i, v_s, attitude, action) \tag{4}$$

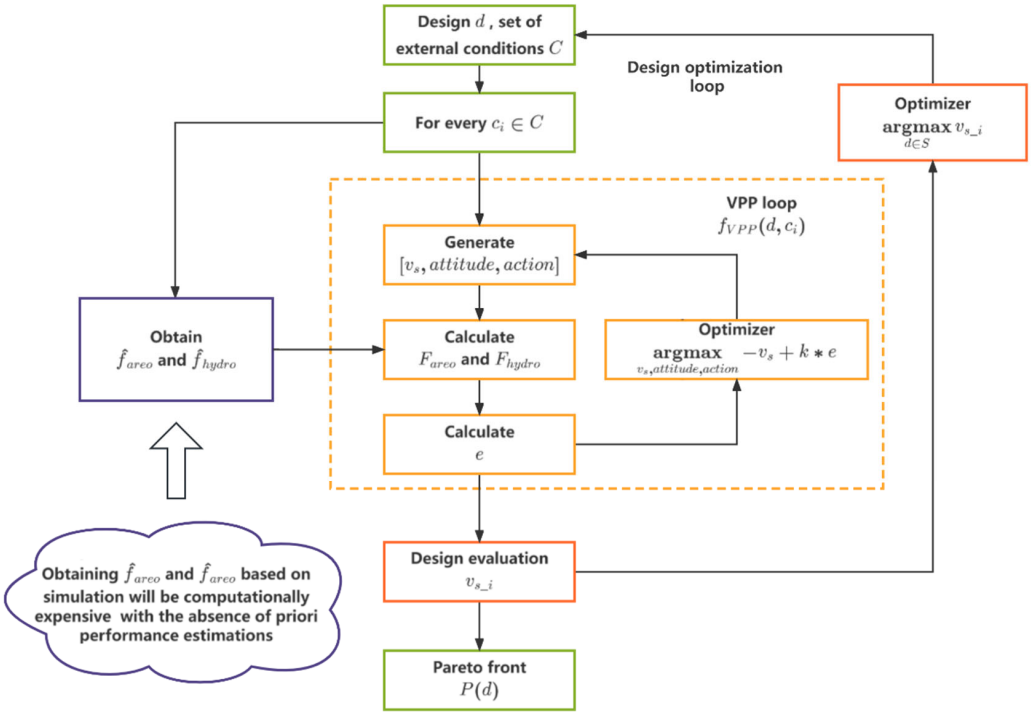

Figure 3. Problem Description. The core issue is the difficulty of obtaining surrogates of a particular design's hydro- and aerodynamic characteristics in the absence of prior performance estimates. Note: The same colouring scheme is used as in Figure 1 for comparison.

Then, the maximum speed that satisfies the multi-degree-of-freedom equilibrium equation is found iteratively, shown as Equations (5) and (6). Since the constraint $e = 0$ is too strict, it is usually relaxed to $e < \epsilon$ in practice. Then, the objective function can be rewritten as Equation (7), where k is a predefined parameter that guarantees the sensitivity of the optimisation to v_s.

Refs. [25,27] state that the predictive performance of VPP depends heavily on the aero- and hydrodynamic model of specific design f_{areo} and f_{hydro}. These models should be extensive, covering feedback corresponding to all the states of interest. For engineering

realizability, how to reasonably obtain f_{areo} and f_{hydro} without accurate prior performance estimates is the key to the problem.

$$v_{s\_i} = \underset{v_s, attitude, action}{\operatorname{argmax}} \quad v_s \text{ s.t. } e = 0 \tag{5}$$

$$e = \sum_{DOF=1}^{6} |F_{areo} + F_{hydro}| \tag{6}$$

$$\underset{v_s, attitude, action}{\operatorname{argmin}} \quad -v_s + k * e \tag{7}$$

2.2. Constructing Surrogate Adaptively Using Bayesian Optimisation

An essential aspect of reducing the computational cost is to note that the VPP in the autonomous sailboat design flow only needs to evaluate the performance of a particular design under two conflicting (upwind with large TWS and downwind with small TWS) rather than various external environments as in a general VPP [28]. Thus, for each design, a fine-grained global surrogate is not necessary. As Figure 4 shows, sampling is expected to be concentrated around the unknown equilibrium state (i.e., the actual operating state of the platform satisfying the multi-degree-of-freedom equilibrium equation under c_i).

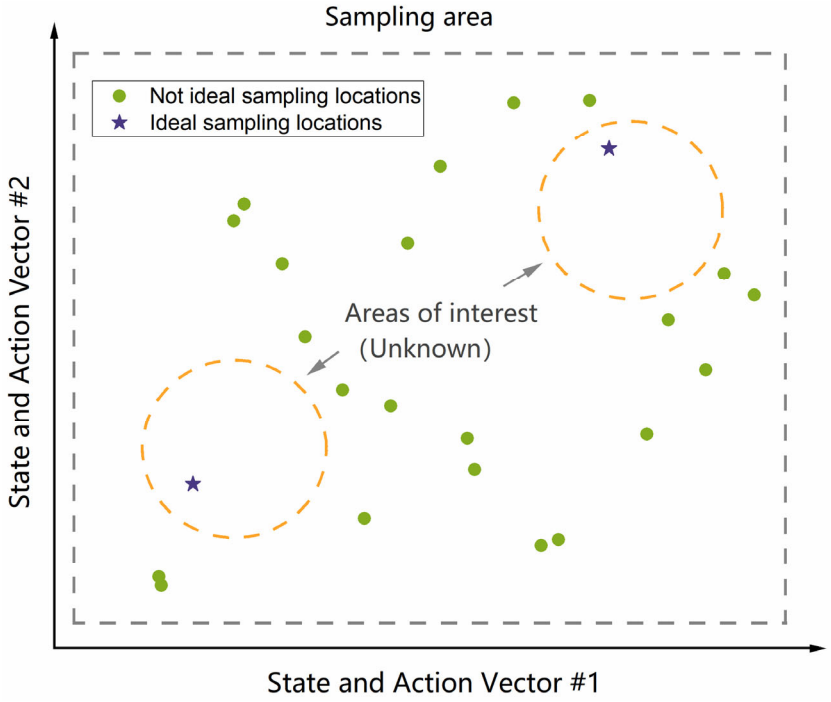

Figure 4. Schematic diagram of ideal sampling. Ideally, it is only necessary to sample near the equilibrium point under two conflicting states for each design.

Bayesian optimisation (BO) [29] has been widely used in similar non-convex optimisation problems requiring extensive and expensive simulations [30–32]). Unlike the classical space-filling sampling pattern (offline surrogate-assisted design with static data input), BO collects new data during the evolutionary search, creating a dynamic and symbiotic sampling coupling with optimisation. More specifically, BO first constructs a prior belief using data collected so far, and defines an acquisition function to guide exploration by leveraging the uncertainty in the posterior. Then, through evolution, the maximum (minimum) point of the acquisition function becomes the following sampling point. Finally, the simulation is executed at the sampling point, and the prior belief is refined as data are observed via Bayesian posterior updating [33]. This process is repeated until the allowed number of

simulations is used up [34,35]. Thus, BO was able to complete the optimisation process with as little sampling as possible [36–38]. Part of our work, therefore, focuses on how to design (by designing a proper acquisition function) a framework for optimising the general configuration of wing-sailed autonomous sailing monohulls.

2.3. Reusing High-Fidelity CFD Data by Knowledge Transfer

In sailboat design practice, hydro- or aerodynamic characteristics may follow the same trend between designs, even if the design parameters are different. In the literature [22], this phenomenon is described thusly: "the curves, surfaces or hypersurfaces that represent the forces will share a topological between different designs,", i.e., "geometry topology is maintained" and therefore, "data topology is maintained" (Figure 5). Thus, in theory, the potential connections between designs can be exploited to obtain surrogates $\hat{f}_{areo}$ and $\hat{f}_{hydro}$ for each design with fewer simulations.

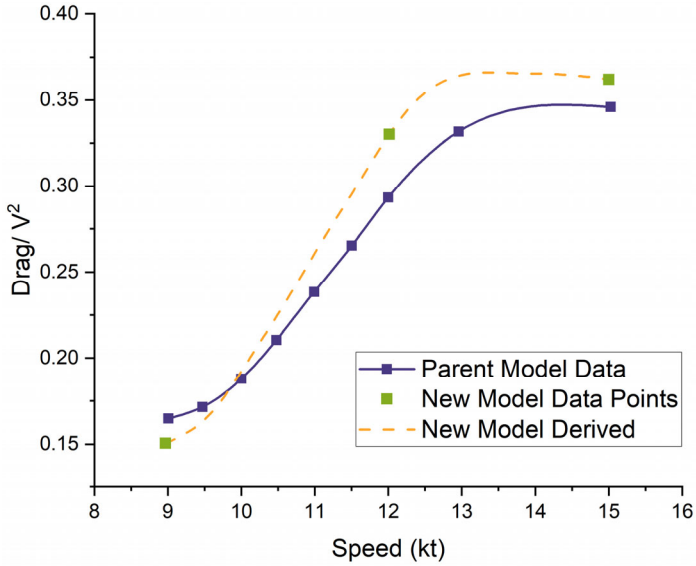

Figure 5. The " Model with different design parameters but the same topology" can be estimated with fewer data points (data from [22]).

The second aspect of reducing the computational cost is introducing knowledge transfer techniques [39,40] to extract the intrinsic connection between different designs and to maximise the reuse of the high fidelity CFD data obtained during the optimisation process. Knowledge transfer aims to transfer knowledge from the source task to the target task to address three significant challenges that most traditional ML algorithms face: insufficient data, incompatible computing capacity and mismatched distribution [41,42]. The main challenge in achieving knowledge transfer is determining whether there is an essential similarity between the source and target tasks. In the context of the present problem, we have shown above that this similarity is naturally guaranteed. Although the collected CFD data cannot be reused directly, certain parts of the data can still be reused together with a few labelled data in the new design. Therefore, another part of our work is focused on incorporating knowledge transfer into our optimisation framework to improve the reusability of CFD simulation data.

3. Optimisation Framework for General Configuration of Wing-Sailed Autonomous Sailboats

In the proposed framework, the outer loop calls the inner loop to obtain the speed metric $v_{s\_j}(j = 1, 2)$ for each design under the external environment of interest, thus finding the Pareto set on the design space S. Any multi-objective optimisation method such as MOEA/D [43] or NSGA-II [44] is optional. In this paper, we have chosen the commonly

used NSGA-II algorithm, which has competitive computational complexity and diversity in solution [44]. The flowchart is shown in Figure 6.

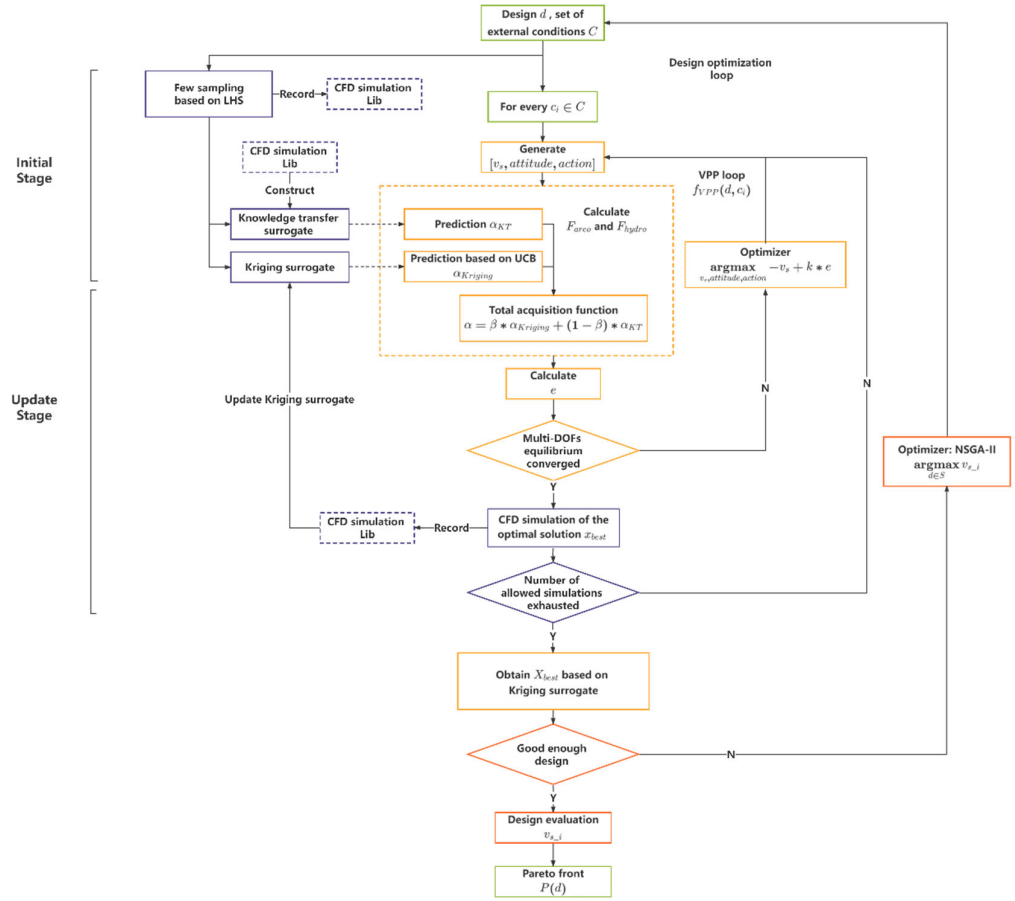

Figure 6. The flowchart of the proposed framework. Note: The same colouring scheme is used as in Figure 1 for comparison.

In the inner loop, the speed evaluations v_{s_i} for each design d under $C = \{c_i\}$ are obtained in a dual-surrogate process. In the initial stage, we first obtain a small amount of CFD data on f_{areo} and f_{hydro} using latin hypercube sampling (LHS) [45]. Then, based on these data, we construct the rough global kriging surrogate and train the knowledge transfer surrogate in combination with the historical data from the optimisation process (thus, the first few designs were evaluated without a knowledge transfer surrogate). In the update stage, we first construct a weighted acquisition function based on both surrogates to improve the weak prediction performance of the kriging surrogates due to insufficient samplings in the initial stage. Thereafter, based on the BO method, the process iteratively samples based on the acquisition function until the allowed number of samplings is exhausted or the intended accuracy is achieved. Throughout the entire process, all the data generated are recorded for reuse.

3.1. Initial Stage

As Algorithm 1, since there is no prior performance estimation for a design d, it is necessary to sample a predefined number of points to obtain their global characteristics. Therefore, an initial sampling is performed using the LHS within two searching spaces for each external condition c_i. Separate searching spaces help improve the effectiveness of sampling. These samples are then first used to construct a crude global kriging surrogate, which, in contrast to other surrogate models, gives an estimate and a prediction error for the non-test sample points. The crude kriging surrogate will be used to update the surrogate during BO.

However, inadequate sampling may not capture features accurately. Therefore, the samples obtained from the LHS are used to train the knowledge transfer surrogate $\hat{f}_{KT}$ together with the observed data from the CFD simulation library to provide priori information for optimisation. More specifically, for each design that has been evaluated, the RBF surrogates are constructed using the observed high-fidelity data d_i stored in the library. When evaluating a new design, a predefined number of the RBF surrogates of the closest designs (in the design space) are selected and the network parameters are frozen. These RBFs are then trained as part of the network to obtain the so-called knowledge transfer surrogate with the LHS samples. The network structure of the knowledge transfer surrogate is illustrated below in Figure 7.

Algorithm 1 Pseudo code for initial stage

Input Design d, specific external condition $c_i (i = 1, 2)$, number of expensive CFD simulations allowed FE_{ini}. CFD simulation library Lib.
Output $\hat{f}_{Kri}$ and $\hat{f}_{KT}$, new Lib

Set searching space SP_i for $X = [v_s, attitude, action]$ considering $c_i (i = 1, 2)$
$j = 1$
//sampling under each external condition separately
for $j \leq 2$ **do**
//sampling to obtain rough global features
Generation of $\frac{FE_{ini}}{2}$ sampling points in the SP_j range using LHS.
Perform CFD simulation on sampling points $X_i (i = 1, 2, \ldots, \frac{FE_{ini}}{2})$, obtain observation Y_i including hydrodynamic and aerodynamic observations.
Deposit $[X_i, Y_i]$ into Lib.
end for
//Constructing crude kriging surrogate
Construct the Kriging surrogate $\hat{f}_{Kri}$ by $[X_i, Y_i]$ $(i = 1, 2, \ldots, FE_{ini})$
//extracting prior information from characteristics of related designs
Load $RBF_i (i = 1, 2, \ldots, 5)$ of the 5 most similar designs to d from Lib
Train knowledge transfer surrogate $\hat{f}_{KT}$ by $[X_i, Y_i]$ $(i = 1, 2, \ldots, FE_{ini})$ and RBF_i $(i = 1, 2, \ldots, 5)$

3.2. Update Stage

The pseudo-code for the update stage is shown in Algorithm 2. In the update stage, the kriging surrogate is alternately updated by Bayesian optimisation until the pre-determined number of simulations is exhausted. In Bayesian optimisation, the balance between explore and exploit is achieved by employing the acquisition function. The acquisition function is essentially a weighted sum of the predicted value (mean) and uncertainty (variance). It is usually constructed based on strategies such as the probability of improvement (PI) [46], expected improvement (EI) [47], lower/upper confidence bound (LCB/UCB) [48], etc. In this paper, the acquisition function value α of state X under c_i is constructed considering the optimal value, the predicted value and the uncertainty, as in Equation (8). k_1, k_2, k_3 are pre-determined weights, $v_{best}, v_{mean}, v_{uncertainty}$, and are dependent on the values below.

$$\alpha(X, c_i) = -v_s + k * \sum_{DOF} k_1 * v_{best} + k_2 * v_{mean} + k_3 * v_{uncertainty} \tag{8}$$

For each degree of freedom (DOF), considering the uncertainties in aerodynamic and hydrodynamic forces (moments) will be largely due to the coarseness of the kriging model in the initial stage, the cumulative sum of $\left| F_{areo} + F_{hydro} \right|$ may have a more considerable uncertainty and may lead to a lack of sufficient heuristic in the early stages of optimisation.

Therefore, we add the predicted value of the knowledge transfer surrogate to the acquisition function to provide a heuristic. F_{areo} and F_{hydro} can be obtained by kriging and knowledge transfer surrogate. The weight β is set as Equation (9) according to the number of current and maximum iterations N_{now} and N_{max}. The setting ensures that the prior information from the historical design is fully utilised to improve efficiency in the beginning of the optimisation without affecting the convergence of the mid and late stages of the process. F_{areo} and F_{hydro} can be expressed as mean m, uncertainty inv and the upper and lower boundary as ub and lb, as in Equations (9)–(12).

$$\beta = \begin{cases} 2 * N_{now}/N_{max} & \text{if } N_{now} < N_{max}/2 \\ 1 & \text{else} \end{cases} \tag{9}$$

$$m = \beta * m_{Kri} + (1 - \beta) * m_{TL} \tag{10}$$

$$inv = \beta * inv_{kri} \tag{11}$$

$$lb = m - inv/2 \tag{12}$$

$$ub = m + inv/2 \tag{13}$$

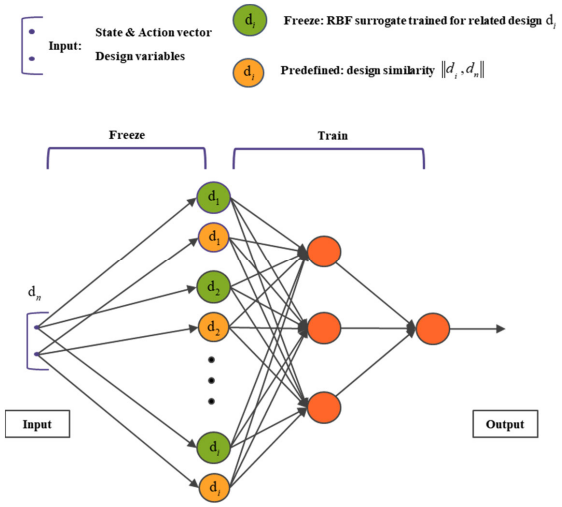

Figure 7. Network structure of the knowledge transfer surrogate.

Thus, the $v_{best}, v_{mean}, v_{uncertainty}$ of $\left|F_{areo} + F_{hydro}\right|$ are defined in Equations (14)–(20). While v_{best} represents the percentage of minimum difference between hydrodynamic and aerodynamic forces (moments), v_{mean} characterises the percentage deviation of the best value from the mean. Both of the above imply exploration (or convergence). Additionally, $v_{uncertainty}$ is negatively correlated with the deviation between the best and the mean as a percentage of the uncertainty interval, implying exploration, i.e., global optimality-seeking performance.

$$m_{sum} = m_{areo} + m_{hydro} \tag{14}$$

$$ub_{sum} = ub_{areo} + ub_{hydro} \tag{15}$$

$$lb_{sum} = lb_{areo} + lb_{hydro} \tag{16}$$

$$inv_{sum} = ub_{sum} - lb_{sum} \tag{17}$$

$$v_{best} = \begin{cases} \frac{min\ (\ |ub_{sum}|, |lb_{sum}|\ )}{min\ (\ \left|m_{areo}\right|, \left|m_{hydro}\right|\ )} & \text{if } sign\ (ub_{sum} * lb_{sum}) = 1 \\ 0 & \text{else} \end{cases} \tag{18}$$

$$v_{mean} = \frac{|m_{sum}|}{min\ (\ \left|m_{areo}\right|, \left|m_{hydro}\right|\ )} \tag{19}$$

$$v_{uncertainty} = \frac{1}{inv_{sum}} \tag{20}$$

The termination condition for the update stage is convergence for all external environments c_i ($i = 1, 2$ in this study). When the termination condition is met, all observations of the design are saved, and an RBF surrogate for the design is constructed so that it can be called to form a knowledge transfer surrogate when evaluating incoming designs.

Algorithm 2 Pseudo code for update stage

Input Design d, specific external condition $c_i (i = 1, 2)$, predefined search space SP_i, surrogates $\hat{f}_{Kri}$ and $\hat{f}_{KT}$, optimisation history Lib, number of expensive CFD simulations allowed FE_{up}, predefined tolerance δ, minimum sampling interval Δ.

Output v_{s_i} $(i = 1, 2)$, new Lib

$count_{sim} = 0$
$flag_i = 0 (i = 1, 2)$ *//termination identifier*
while $count_{sim} < FE_{up}$ **do**
    //sampling under each external condition in turn
    $i = mod(count_{sim}, 2) + 1$
    if $flag_i = 1$
        $i = mod(i + 1, 2) + 1$
    end if
    Set searching space SP_i for $X = [v_s, attitude, action]$ considering $c_i (i = 1, 2)$
    //observing and saving observations according to the acquisition function
    Obtain X_{best} that maximises $\alpha(X, c_i)$, as in Equations (8)–(16), using optimiser.
    Perform a simulation on X_{best} and obtain the observation Y_{best}.
    Calculate e, deposit $[X_{best}, Y_{best}]$ into Lib
    //determine if this external condition converges
    if $e < \delta$ or distance of nearest sample in Lib $dis < \Delta$ do
        Obtain $v_{s\_i} \in X_{best\_i}$
        $flag_i = 1$
    end if
    //updating kriging surrogate
    Update $\hat{f}_{Kri}$ with $[X_{best}, Y_{best}]$
    $count_{sim} = count_{sim} + 1$
    //determining if all external conditions converges
    if all $flag_i = 1$
        break
    end if
end while
//extracting and saving characteristics from observations of current design
Train RBF network with all samples of the current design deposited into Lib

4. Application to the 'Seagull' Prototype

The 'Seagull' (Figure 8) was our first autonomous sailboat prototype to verify the general configuration, structural durability, hardware reliability, control strategy and power management and was a testbed for essential functions such as path-following, tracking/gybing, and virtual mooring. Sea trials have shown that the capsize resistance of the original design has not been fully utilised, and therefore there is potential for larger sails to improve speed performance. Changes to the hull will result in the need to re-arrange the cabin equipment, whilst modifications to the keel will result in redesigning the supporting equipment, so the general configuration optimisation work is limited to the sails only. The new sail will have a trapezoidal configuration instead of the original rectangular configuration, effectively increasing the sail area while lowering the centre of gravity and the aerodynamic centre under the wind gradient effect.

Figure 8. The 'Seagull' prototype.

4.1. Setup of Design Space, External Condition, and Sampling Space

The design variables are the root chord length RC, taper ratio TR and the span SPAN of the sail. In order to limit the design variables to a reasonable range and avoid oversized and overweight sails, variables are given in the form of a range of derived dimensionless numbers: the sail area-displacement ratio and the (geometric) aspect ratio, as in Equations (21) and (22). The respective ranges are $SA/D \in [4, 8]$; for the original design, the sail area was 4, which seems too small. $AR \in [1, 7]$, because most autonomous sailboats have wing sails in this range [11]. TR is defined within [0.4, 1]. The sails are mainly made of balsa wood, their density is set at 120 KG/m^3. The weight of the mast and sail drive mechanism is accounted for in the hull.

$$SA/D = \frac{RC * (1 + TR) * SPAN}{2 * V_{dis}{}^{\frac{2}{3}}} \tag{21}$$

$$AR = \frac{2 * SPAN}{RC * (1 + TR)} \tag{22}$$

From meteorological data, the seasonal wind characteristics of the target area are an average 5 m/s in spring and 12 m/s in winter. Therefore, the performance at 5 m/s TWS and 150° TWA (downwind, in the inertial coordinate system) and 12 m/s TWS 50° TWA (upwind) are typical external conditions of interest.

The Froude number is set within [0, 0.4] to ensure that the search space covers the displacement mode (Fr under 0.35) [49]. The heel angle is limited to [0, 26]° to keep the deck out of water, even with the heaviest sails in design space. The yaw angle is set to [0, 5]° [50]. The sail angle is set within the optimal value ±20° proposed in the literature [26].

4.2. Setup of CFD Simulation and Optimisation Parameters

The optimisation processes are entirely automated, as shown in Figure 9. The interactions between the software is performed using Matlab scripts, while the parametric 3D

model of the sail is generated by calling CAESES. CAESES supports the generation of highly accurate models with different design parameters that can be directly imported into CFD software. The meshes and CFD simulations are performed with StarCCM+.

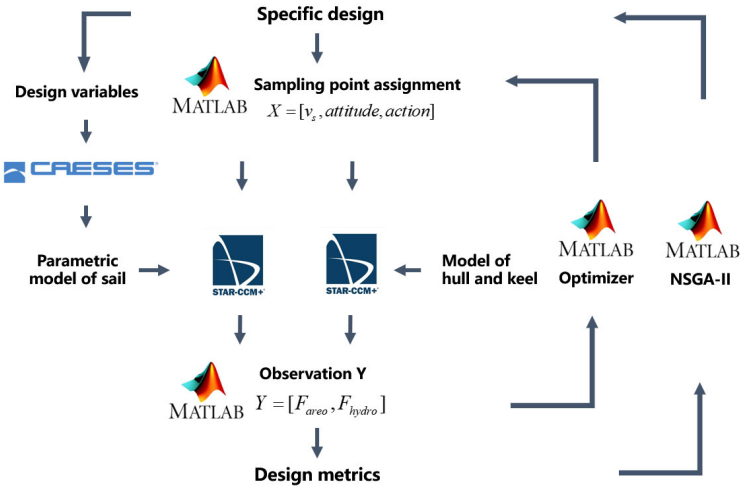

Figure 9. Software interaction for optimisation processes.

As illustrated in Figure 10, the computational domain for the hull and keel is block-shaped, with a size of 10 × 10 × 20 times L_{pp}. The inlet boundary is positioned 5 × L_{pp} upstream, and the pressure outlet condition is installed 15 × L_{pp} downstream. A no-slip condition is forced at the hull, and far-field free-slip wall conditions are applied in the surrounding area. The volume of fluid (VOF) technique is used to model the free surface. The initial position of the free surface is obtained by interpolating the weight of the design (hull +keel+ sail) in the hydrostatic curve, which is calculated before the simulation. The total number of grids is 3.7 million. Fine cells were focused on the free surface and in the stern-waves domain to accurately capture the characteristics (Figure 11).

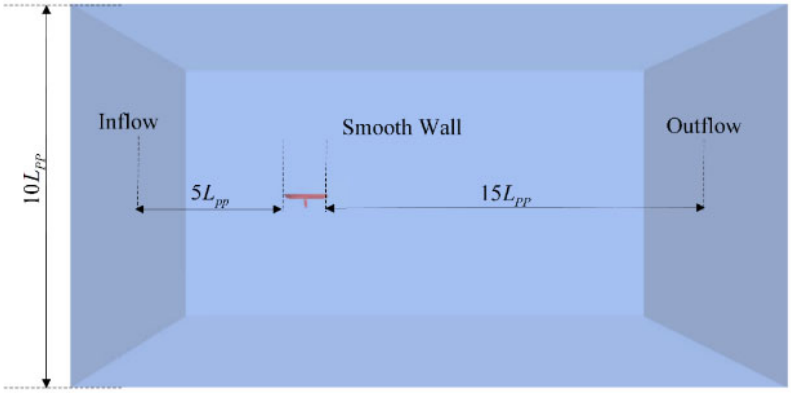

Figure 10. Setup of the computational domain of hull and keel.

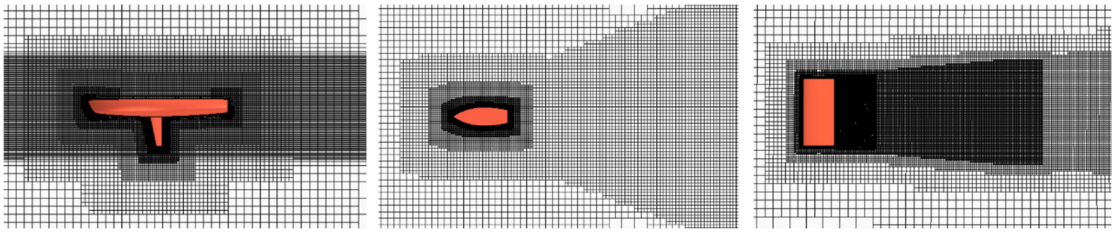

Figure 11. Mesh settings.

The wing sail is placed relative to the hull in the same inertial coordinate system. Considering that there are two different wind conditions (downwind and upwind), two models are established, with different velocity inlets and pressure outlets. As shown in Figure 12, the computational domain is 40 times the length, 30 times the height and 30 times the width of the root chord RC. There were coarse structured cells in the exterior subdomain and finely structured cells in the subdomain around the sails. In total, 2.8 million cells are employed (Figure 11). The simulation focuses on the effect of the wind gradient. The inlet wind speed is defined by field function, according to [51], as in Equation (23):

$$v_w(h) = v_{10} \cdot \left(\frac{h}{h_{10}}\right)^{\alpha} \tag{23}$$

where $v_w(h)$ is the velocity of the wind at height h, v_{10} is the wind velocity at 10 m and α is the Hellmann exponent.

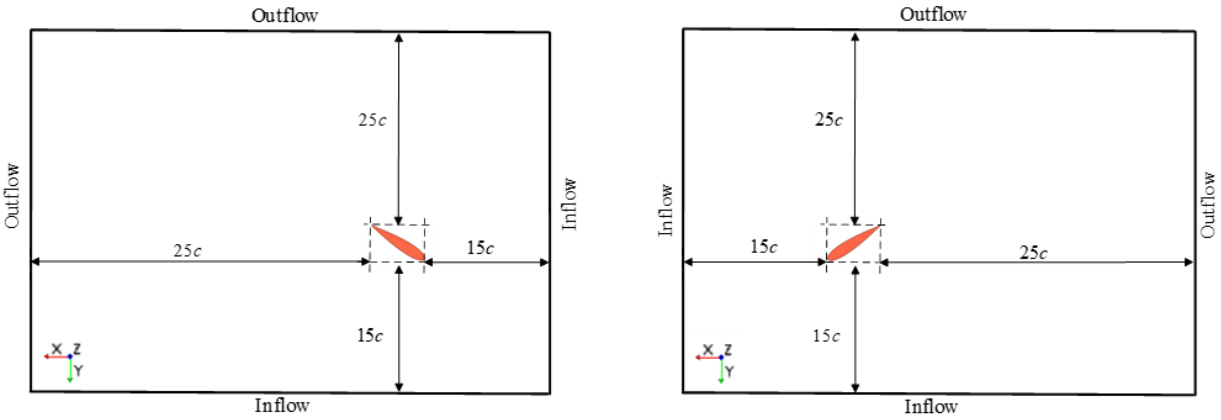

Figure 12. Setup of the computational domain of sail downwind (**left**) and upwind (**right**).

The optimisation considers the balance of three DOFs: surge, sway, and roll. The heave is ignored because autonomous sailing monohulls usually operate in displacement mode with very little hydrodynamic lift. The pitch motion is neglected for the significant longitudinal moment of inertia. Yaw motion is ignored because the rudder is not considered in the model, and the rudder can effectively ensure yaw balance. Considering that each combination of hydrodynamic and aerodynamic simulations takes about 2 h on an Intel Xeon Platinum 8160 Processor (48 cores), we validate the optimisation progress with only three generations of a population of size 10. The parameters in the optimisation process are set as Table 1.

Table 1. Parameters settings in the optimisation process.

FE_{ini}	20	k_2	5
FE_{up}	60	k_3	1
k	1	Δ	0.02
k_1	10	δ(each DOF)	10%

4.3. Algorithm Validation

In order to verify the validity of the BO method, we conducted a comparison experiment using the first six designs (without introducing knowledge transfer surrogate) with the offline surrogate method. Results are obtained by performing LHS sampling with the same number of samples when the BO converges (the maximum permitted samples are exhausted, the pre-determined accuracy is achieved or the minimum sampling interval is satisfied), and by performing CFD simulation to verify the solved equilibrium quality.

The results (Table 2) show that it is difficult to obtain an acceptable 'equilibrium state' with the offline method for the same number of samples that would allow the BO to converge without performance prior. At the same time, the quality of the equilibrium obtained by BO is much better. Therefore, we believe that the BO method effectively reduces the computational cost while obtaining better performance estimates.

Table 2. Comparison of equilibrium states solved by offline methods and BO.

Case		Condition	Method	v_s(m/s)	θ (°)	λ (°)	β_s(°)	eFX	eFY	eMX	eSUM
RC	1.32 m	Downwind	BO	0.77	0.14	0.90	47.08	3%	6%	4%	13%
SPAN	1.32 m		Offline	0.78	0.24	0.66	45.00	10%	43%	36%	88%
TR	0.4	Upwind	BO	1.15	0.87	1.43	31.40	4%	4%	2%	10%
itr	70		Offline	0.43	0.1	7.31	20.43	99%	24%	37%	160%
RC	0.70 m	Downwind	BO	1.67	2.59	0.78	55.02	1%	3%	2%	6%
SPAN	4.92 m		Offline	1.32	3.02	1.43	56.05	61%	27%	32%	120%
TR	1	Upwind	BO	1.70	15.54	4.25	54.80	5%	9%	4%	17%
itr	59		Offline	1.47	16.59	4.63	32.36	23%	2%	11%	36%
RC	0.77 m	Downwind	BO	0.81	0.25	0.83	47.43	0%	1%	9%	10%
SPAN	1.65 m		Offline	0.42	0.19	2.59	62.54	167%	33%	187%	387%
TR	1	Upwind	BO	1.36	1.62	1.46	31.51	4%	8%	7%	19%
itr	72		Offline	0.75	0.95	4.16	24.11	15%	11%	19%	45%
RC	1.12 m	Downwind	BO	0.91	0.14	0.49	62.24	0%	8%	4%	12%
SPAN	1.30 m		Offline	1.00	0.10	0.41	45.00	47%	87%	131%	265%
TR	0.93	Upwind	BO	1.24	1.29	1.67	26.69	7%	3%	4%	14%
itr	78		Offline	1.19	0.74	1.69	27.96	11%	0%	74%	86%
RC	0.87 m	Downwind	BO	1.27	1.18	0.91	46.08	2%	1%	2%	5%
SPAN	3.46 m		Offline	0.73	1.72	3.11	61.28	107%	107%	200%	414%
TR	0.5	Upwind	BO	1.68	5.74	1.99	30.39	0%	4%	2%	6%
itr	65		Offline	0.98	4.35	4.92	15.58	3%	59%	33%	95%
RC	0.54 m	Downwind	BO	1.16	0.49	0.47	62.97	0%	5%	2%	7%
SPAN	3.64 m		Offline	0.94	4.42	0.52	62.65	108%	179%	122%	410%
TR	0.53	Upwind	BO	1.40	3.31	1.74	33.00	1%	1%	1%	3%
itr	66		Offline	1.59	10.29	1.12	29.42	74%	35%	219%	328%

Downwind = 5 m/s TWS @ 150° TWA, Upwind = 12 m/s TWS @ 50° TWA.

To verify the validity of the knowledge transfer surrogate, we give the respective statistics with and without the knowledge transfer surrogate, including the number of iterations needed to converge and the fitness of the first ten BO iterations, as shown in Figure 13. The results show that the knowledge transfer surrogate effectively guides the initial stage of the optimisation process and accelerates convergence.

4.4. Optimisation Results

The Pareto fronts of speed performance for all 30 designs are shown in Figure 14. The effects of the three design parameters Span, RC, and TR on upwind speed and downwind speed are shown in Figure 15. The upwind and downwind performance showed a consistent trend. In general, the effect of SPAN on speed performance is the most significant, which is in line with expectations, as higher sails can gain more power in response to the wind gradient. Due to the limitations of the maximum sail area, a larger SPAN usually corresponds to a smaller RC. Theoretically, a larger TR makes better use of the wind but simultaneously raises the centre of gravity and aerodynamic centre of the sail. The optimisation results show that the better performing designs are more concentrated around TR $\in$[0.6, 0.8]; however, their effect is limited.

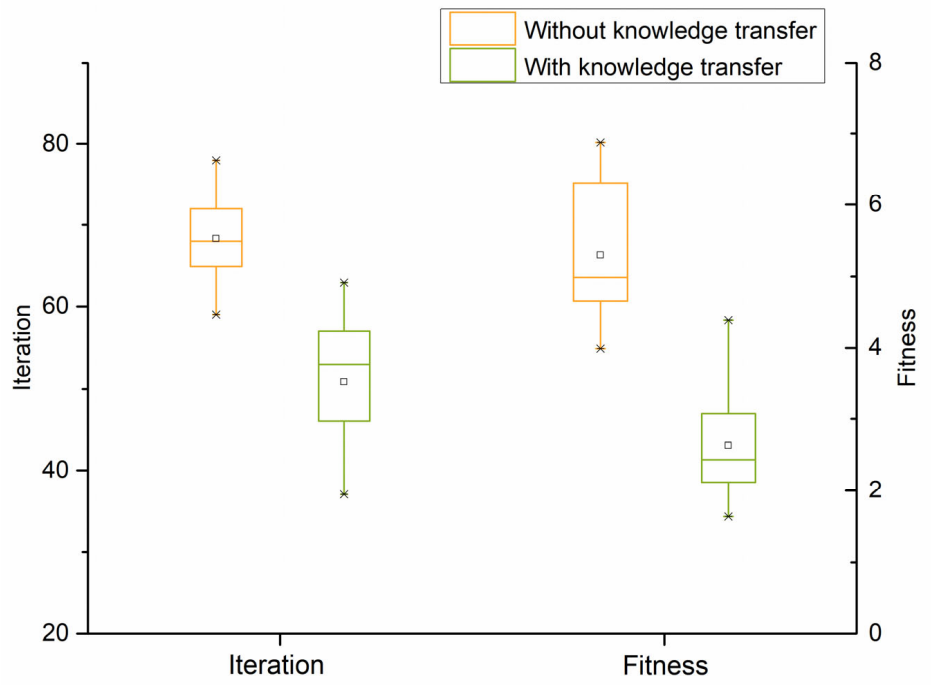

Figure 13. Comparison of statistics with and without introducing the knowledge transfer surrogate. Square represents mean.

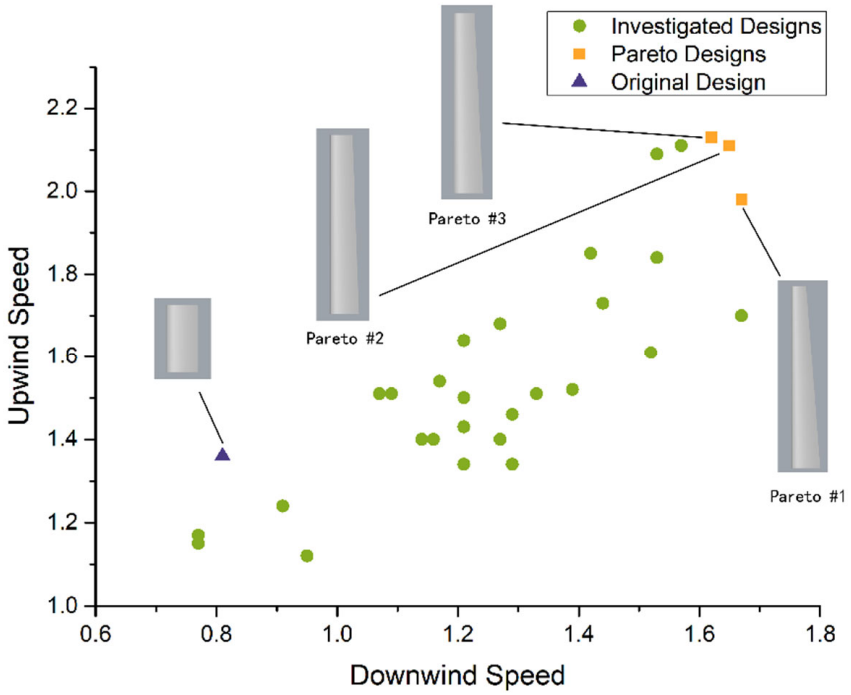

Figure 14. Pareto front of designs.

Table 3 compares the performance of the original design and Pareto-optimised designs. The results show that the optimisation frame effectively improves the speed performance of the original design within the design constraints, making more efficient use of the recovery moment of the sailboat while ensuring that the heeling angle does not exceed the permissible value.

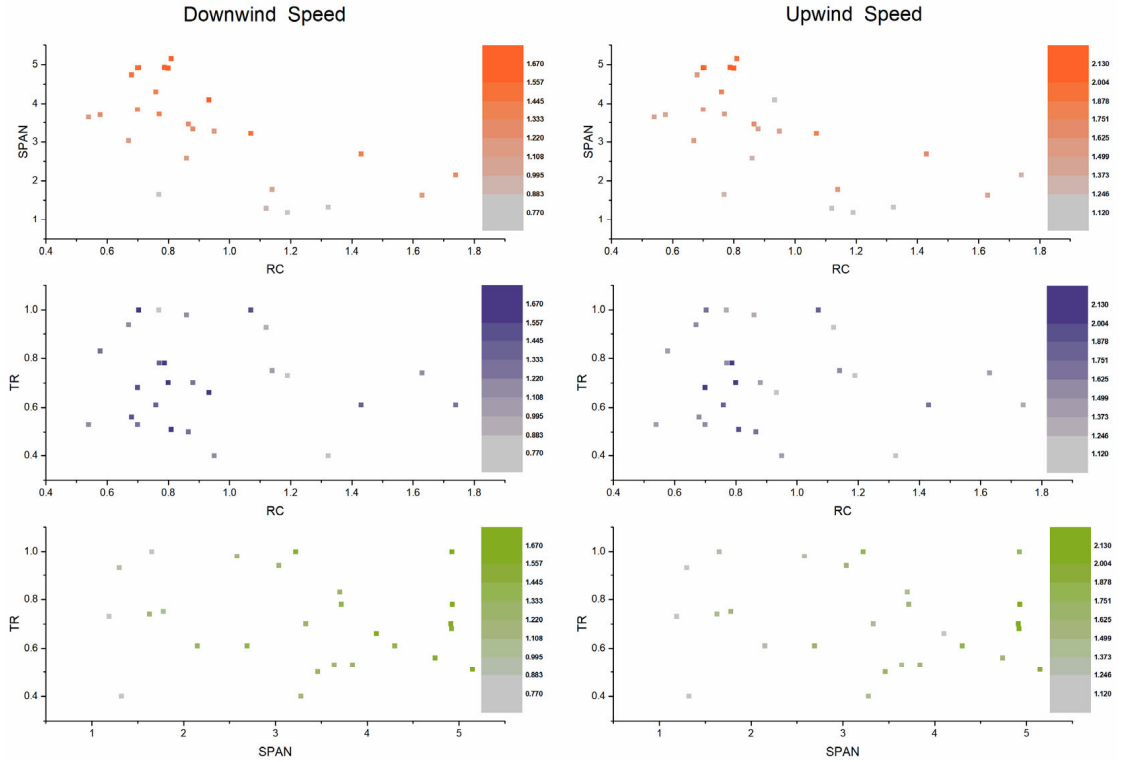

Figure 15. The influence of each of the two design variables on the optimisation objectives.

Table 3. Optimisation results.

Design	RC (m)	SPAN (m)	TR	Condition	v_s(m/s)	θ(°)	λ(°)	β_s(°)
Origin	0.77	1.60	1.00	Downwind	0.81	0.25	0.83	47.43
				Upwind	1.36	1.62	1.46	31.51
Pareto #1	0.81	5.15	0.51	Downwind	1.67	1.70	0.52	63.81
				Upwind	1.98	15.23	2.48	28.45
Pareto #2	0.79	4.93	0.78	Downwind	1.65	3.09	0.95	48.02
				Upwind	2.11	17.55	2.64	33.00
Pareto #3	0.80	4.91	0.7	Downwind	1.62	2.73	0.89	50.09
				Upwind	2.13	17.62	2.44	32.11

5. Closing Remarks and Future Work

Autonomous sailboat designs are often mission scenario-based, non-standard ad hoc tasks. The lack of design benchmarks and prior performance predictions has resulted in an unacceptable number of samples required to apply an offline surrogate-based simulation-driven design framework. In order to improve the speed performance of autonomous sailboats and hence their efficiency when conducting ocean observations, this paper provides a simulation-driven design framework that combines Bayesian optimisation and knowledge transfer. BO can effectively focus the sampling on the region of interest as much as possible without performance prior, and the introduction of knowledge transfer allows the data from the optimisation process to be reused to provide a priori information for subsequent designs. Simulations demonstrate the effectiveness of this method and its superiority over traditional offline surrogate methods. We have also used the proposed method to optimise our prototype "Seagull" sail with specified design parameters and heeling constraints, thereby significantly improving the speed performance (although "speed performance" in this paper can only be used as a qualitative description of the design). In this paper, we have only considered three DOFs and used StarCCM+ as the solver; however, if the designer has more design resources, more degrees of freedom and more delicate CFD

settings can be used to obtain more accurate results. Similarly, less accurate solvers can be used to obtain faster, trend-based conclusions.

This framework can facilitate the comparison and optimisation of designs considering the coupling of autonomous sailboat design with external conditions. In the future, we will use this framework to investigate the combination of hydrofoils and autonomous sailing boats. The hydrofoil can provide more lift (and recovery moment) at high speeds, which is expected to alleviate the conflict between the "speed–overturning resistance" of autonomous sailing boats.

Author Contributions: Y.A.: Conceptualisation, Methodology, Software, Visualisation, Writing (Original Draft); F.H.: Software (CFD), Validation and Writing (Review and Editing); K.C.: Software, Validation; J.Y.: Supervision, Project Administration and Writing (Review and Editing). All authors have read and agreed to the published version of the manuscript.

Funding: This research and the APC were funded in part by the National Natural Science Foundation of China (Grant No. 51909257), by Natural Science Foundation of Liaoning Province (Grant No. 2021-MS-031).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: We are grateful to Jin Zhang for critically reading our manuscript. We thank the anonymous reviewers whose comments helped improve and clarify this manuscript.

Conflicts of Interest: The authors declare no known competing financial interest or personal relationships that could appear to influence the work reported in this paper.

References

1. Silva, M.F.; Friebe, A.; Malheiro, B.; Guedes, P.; Ferreira, P.; Waller, M. Rigid Wing Sailboats: A State of the Art Survey. *Ocean Eng.* **2019**, *187*, 106150. [CrossRef]
2. Saildrone Surveyor. Available online: https://www.saildrone.com/news/surveyor-completes-first-trans-pacific-ocean-mapping-mission (accessed on 14 July 2021).
3. Mordy, C.; Cokelet, E.; De Robertis, A.; Jenkins, R.; Kuhn, C.; Lawrence-Slavas, N.; Berchok, C.; Crance, J.; Sterling, J.; Cross, J.; et al. Advances in Ecosystem Research: Saildrone Surveys of Oceanography, Fish, and Marine Mammals in the Bering Sea. *Oceanography* **2017**, *30*, 113–115. [CrossRef]
4. Cokelet, E.D.; Meinig, C.; Lawrence-Slavas, N.; Stabeno, P.J.; Mordy, C.W.; Tabisola, H.M.; Jenkins, R.; Cross, J.N. The Use of Saildrones to Examine Spring Conditions in the Bering Sea: Instrument Comparisons, Sea Ice Meltwater and Yukon River Plume Studies. In Proceedings of the OCEANS 2015-MTS/IEEE Washington, Washington, DC, USA, 19–22 October 2015; pp. 1–7.
5. Ghani, M.H.; Hole, L.R.; Fer, I.; Kourafalou, V.H.; Wienders, N.; Kang, H.; Drushka, K.; Peddie, D. The SailBuoy Remotely-Controlled Unmanned Vessel: Measurements of near Surface Temperature, Salinity and Oxygen Concentration in the Northern Gulf of Mexico. *Methods Oceanogr.* **2014**, *10*, 104–121. [CrossRef]
6. Kilpin, G. Modelling and Design of an Autonomous Sailboat for Ocean Observation. Ph.D. Thesis, University of Cape Town, Cape Town, South Africa, 2014.
7. Domínguez-Brito, A.C.; Valle-Fernández, B.; Cabrera-Gámez, J.; Ramos-de-Miguel, A.; García, J.C. A-TIRMA G2: An Oceanic Autonomous Sailboat. In *Robotic Sailing 2015*; Friebe, A., Haug, F., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 3–13, ISBN 978-3-319-23334-5.
8. Clemens, K. First Autonomous Circumnavigation of Antarctica. Available online: https://www.designnews.com/batteryenergy-storage/first-autonomous-circumnavigation-antarctica (accessed on 7 June 2021).
9. CUSail | Fleet. Available online: https://cusail.engineering.cornell.edu/app/fleet.html (accessed on 8 March 2021).
10. Miller, P.H.; Hamlet, M.; Rossman, J. Continuous Improvements to USNA Sailbots for Inshore Racing and Offshore Voyaging. In *Robotic Sailing 2012*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 49–60.
11. An, Y.; Yu, J.; Zhang, J. Autonomous Sailboat Design: A Review from the Performance Perspective. *Ocean Eng.* **2021**, *238*, 109753. [CrossRef]
12. Jung, C.; Schindler, D. Wind Speed Distribution Selection—A Review of Recent Development and Progress. *Renew. Sustain. Energy Rev.* **2019**, *114*, 109290. [CrossRef]
13. Microtransat-History. Available online: https://www.microtransat.org/history.php (accessed on 7 June 2021).
14. Rathour, S.S. Design and Devlopment of an Autonomous Robotic Platform for Detecting, Monitoring and Tracking of Oil Spill on the Sea Surface. Ph.D. Thesis, Osaka University, Osaka, Japan, 2016.

15. Harries, S.; Abt, C.; Hochkirch, K. Hydrodynamic Modeling of Sailing Yachts. In Proceedings of the 15th Chesapeake Sailing Yacht Symposium, Annapolis, MD, USA, 26–27 January 2001.
16. Zanella, M.R. Improved Sailboat Design Process and Tools Using Systems Engineering Approach. Ph.D. Thesis, Virginia Tech, Blacksburg, VA, USA, 2020.
17. Borba Labi, G. Velocity Prediction Program Development for Hydrofoil-Assisted Sailing Monohulls. Master's Thesis, Rostock University, Rostock, Germany, 2019.
18. Horel, B. Review of Existing Benchmarks and Databases for Sailing Vessels. *J. Sail. Technol.* **2022**, *7*, 52–87. [CrossRef]
19. Oliver, C.; Gauvain, E. Practical Performance Prediction of Foil-Configured Monohull Yachts. In Proceedings of the SNAME 24th Chesapeake Sailing Yacht Symposium, Annapolis, MD, USA, 10 June 2022; p. D021S003R002.
20. Byrne, C.; Dickson, T.; Lauber, M.; Cairoli, C.; Weymouth, G. Using Machine Learning to Model Yacht Performance. *J. Sail. Technol.* **2022**, *7*, 104–119. [CrossRef]
21. Graf, K.; Boehm, C.; Renzsch, H. CFD-and VPP-Challenges in the Design of the New AC90 Americas Cup Yacht. In Proceedings of the 19th Chesapeake Sailing Yacht Symposium, Annapolis, MD, USA, 15–16 March 2009.
22. De Elvira, M.R. Design of a Generalized Tool for the Performance Assessment under Sail Based on Analytical, Numerical and Empirical Results: A Global Sailing Yacht Meta-Model. Ph.D. Thesis, Polytechnic University of Madrid, Madrid, Spain, 2015.
23. Eggert, F. Flight Dynamics and Stability of a Hydrofoiling International Moth with a Dynamic Velocity Prediction Program (DVPP). Ph.D. Thesis, Technische Universität Berlin, Berlin, Germany, 2018.
24. Eliasson, R.; Larsson, L.; Orych, M. *Principles of Yacht Design*; A&C Black: Lymington, UK, 2014.
25. Maskew, B.; DeBord, F. Upwind Sail Performance Prediction for a VPP Including "Flying Shape" Analysis. In Proceedings of the 19th Chesapeake Sailing Yacht Symposium, Annapolis, MD, USA, 15–16 March 2009.
26. An, Y.; Yu, J.; Hu, F.; Wang, Z. Towards a General Design Evaluation Tool: The Development and Validation of a VPP for Autonomous Sailing Monohulls. *Appl. Ocean Res.* **2022**, *120*, 103053. [CrossRef]
27. Lasher, W.C.; Sonnenmeier, J.R.; Forsman, D.R.; Zhang, C.; White, K. Experimental Force Coefficients for a Parametric Series of Spinnakers. In Proceedings of the SNAME 16th Chesapeake Sailing Yacht Symposium, Annapolis, MD, USA, 15–16 March 2003.
28. Biancolini, M.E.; Cella, U.; Clarich, A.; Franchini, F. Multi-Objective Optimization of A-Class Catamaran Foils Adopting a Geometric Parameterization Based on RBF Mesh Morphing. In *Evolutionary and Deterministic Methods for Design Optimization and Control with Applications to Industrial and Societal Problems*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 467–482.
29. Pelikan, M.; Goldberg, D.E.; Cantú-Paz, E. BOA: The Bayesian Optimization Algorithm. In Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99, Orlando, FL, USA, 13–17 July 1999; Volume 1, pp. 525–532.
30. Tran, A.; Sun, J.; Furlan, J.M.; Pagalthivarthi, K.V.; Visintainer, R.J.; Wang, Y. PBO-2GP-3B: A Batch Parallel Known/Unknown Constrained Bayesian Optimization with Feasibility Classification and Its Applications in Computational Fluid Dynamics. *Comput. Methods Appl. Mech. Eng.* **2019**, *347*, 827–852. [CrossRef]
31. Priem, R. Upper Trust Bound Feasibility Criterion for Mixed Constrained Bayesian Optimization with Application to Aircraft Design. *Aerosp. Sci. Technol.* **2020**, *24*, 105980. [CrossRef]
32. Park, S.; Atwair, M.; Kim, K.; Lee, U.; Na, J.; Zahid, U.; Lee, C.-J. Bayesian Optimization of Industrial-Scale Toluene Diisocyanate Liquid-Phase Jet Reactor with 3-D Computational Fluid Dynamics Model. *J. Ind. Eng. Chem.* **2021**, *98*, 327–339. [CrossRef]
33. Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.P.; de Freitas, N. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* **2016**, *104*, 148–175. [CrossRef]
34. Greenhill, S.; Rana, S.; Gupta, S.; Vellanki, P.; Venkatesh, S. Bayesian Optimization for Adaptive Experimental Design: A Review. *IEEE Access* **2020**, *8*, 13937–13948. [CrossRef]
35. Lam, R.; Poloczek, M.; Frazier, P.; Willcox, K.E. Advances in Bayesian Optimization with Applications in Aerospace Engineering. In Proceedings of the 2018 AIAA Non-Deterministic Approaches Conference, San Diego, CA, USA, 4–8 January 2016; American Institute of Aeronautics and Astronautics: Kissimmee, FL, USA, 2018.
36. Frazier, P.I. A Tutorial on Bayesian Optimization. *arXiv* **2018**, arXiv:1807.02811.
37. Qin, S.; Sun, C.; Jin, Y.; Zhang, G. Bayesian Approaches to Surrogate-Assisted Evolutionary Multi-Objective Optimization: A Comparative Study. In Proceedings of the 2019 IEEE Symposium Series on Computational Intelligence (SSCI), Xiamen, China, 6–9 December 2019; pp. 2074–2080.
38. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian Optimization of Machine Learning Algorithms. *arXiv* **2012**, arXiv:1206.2944. [CrossRef]
39. Torrey, L.; Shavlik, J. Transfer Learning. In *Handbook of Research on Machine Learning Applications and Trends*; Olivas, E.S., Guerrero, J.D.M., Martinez-Sober, M., Magdalena-Benedito, J.R., Serrano López, A.J., Eds.; IGI Global: Hershey, PA, USA, 2010; pp. 242–264, ISBN 978-1-60566-766-9.
40. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *Proc. IEEE* **2021**, *109*, 43–76. [CrossRef]
41. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
42. Niu, S.; Liu, Y.; Wang, J.; Song, H. A Decade Survey of Transfer Learning (2010–2020). *IEEE Trans. Artif. Intell.* **2020**, *1*, 151–166. [CrossRef]
43. Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Computat.* **2007**, *11*, 712–731. [CrossRef]

44. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. Evol. Computat.* **2002**, *6*, 182–197. [CrossRef]
45. McKay, M.D.; Beckman, R.J.; Conover, W.J. Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics* **1979**, *21*, 239–245. [CrossRef]
46. Jones, D.R. A Taxonomy of Global Optimization Methods Based on Response Surfaces. *J. Glob. Optim.* **2001**, *21*, 345–383. [CrossRef]
47. Jones, D.R.; Schonlau, M.; Welch, W.J. Efficient Global Optimization of Expensive Black-Box Functions. *J. Glob. Optim.* **1998**, *13*, 455–492. [CrossRef]
48. Dennis, J.; Torczon, V. Managing Approximation Models in Optimization. *Multidiscip. Des. Optim. State Art* **1997**, *5*, 330–347.
49. Dewavrin, J.; Souppez, J.-B.R. Experimental Investigation into Modern Hydofoils-Assisted Monohulls: How Hydrodynamically Efficient Are They? *Trans. R. Inst. Nav. Archit. Part B Int. J. Small Craft Technol.* **2018**, *160*, 111–120. [CrossRef]
50. Souppez, J.-B.R.G. Hydrofoil Configurations for Sailing Superyachts: Hydrodynamics, Stability and Performance. In Proceedings of the Design & Construction of Super and Mega Yachts, Genoa, Italy, 14–15 May 2019. [CrossRef]
51. Heier, S. *Grid Integration of Wind Energy Conversion Systems*; John Wiley & Sons: Chichester, UK, 2005; p. 45, ISBN 978-0-470-86899-7.