Computationally aware and physically plausible hull form optimization incorporating the IMO intact stability code

Walker, Jake M.; Oneto, Luca; Coraddu, Andrea

DOI
10.1016/j.oceaneng.2025.122607

Publication date
2025

Document Version
Final published version

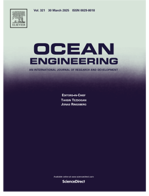
Published in
Ocean Engineering

Citation (APA)
Walker, J. M., Oneto, L., & Coraddu, A. (2025). Computationally aware and physically plausible hull form optimization incorporating the IMO intact stability code. *Ocean Engineering*, *341*, Article 122607. https://doi.org/10.1016/j.oceaneng.2025.122607

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng



Research paper

Computationally aware and physically plausible hull form optimization incorporating the IMO intact stability code

Jake M. Walker[a,*], Luca Oneto[b], Andrea Coraddu[a]

[a] *Delft University of Technology, Delft, Netherlands*
[b] *University of Genoa, Genoa, Italy*

ARTICLE INFO

Keywords:
Design optimization
Data-driven surrogate
Design-space dimensionality reduction
IMO intact stability code
Stability
Physical plausibility

ABSTRACT

Optimizing the hull form is crucial for enhancing vessel performance, requiring a careful exploration of design variations once an initial concept is chosen. To address this, a well-defined parameterization and search space must be established, followed by a numerical optimization process. This phase involves balancing three key requirements: improving performance indicators like resistance (and consequently emissions), ensuring physical plausibility (e.g., designs that meet stability standards), and maintaining computational efficiency. While existing methods partially address these challenges using surrogate models to optimize performance, conducting a-posteriori checks for physical plausibility, and reducing the search space a-priori, they fall short of fully integrating these aspects. In this work, we introduce a two-fold approach to address these challenges. First, we integrate a key physical plausibility criterion directly into the optimization loop: the International Maritime Organization (IMO) Intact Stability Code. For the first time, we use a surrogate model to include this constraint within the numerical optimization process. This enables the generation of designs that inherently meet stability requirements. Second, we reduce the computational demands by applying data-driven methods to limit the search space in a problem-specific manner. This targeted reduction reduces the computational load without compromising the overall optimization performance. We test our proposal by optimizing the KCS hull form. Our results demonstrate two main achievements. First, we find that current optimization pipelines fail to meet the IMO stability guidelines, whereas our method achieves compliance by design. Second, our approach reduces the computational burden by 30 % without sacrificing performance, representing a significant leap forward in practical hull design optimization.

1. Introduction

Within maritime engineering, optimizing the shape of hull forms is a critical task for both researchers (Diez et al., 2015; D'Agostino et al., 2020; Villa et al., 2020, 2021; Guan et al., 2021; Renaud et al., 2022; Zakerdoost and Ghassemi, 2023; Zhang et al., 2024b; Walker et al., 2024b) and practitioners (Bertram et al., 2024) to improve the economic and environmental performance of vessels.

Nowadays, various tools and techniques exist that enable the efficient design and optimization of hull forms (Papanikolaou et al., 2024). Nevertheless, there is currently no tool able to address the problem of hull form design optimization with an end-to-end approach, and the problem is divided into three stages: early-, mid-, and late-stages with specific supporting tools (La Rosa et al., 2024; Bagazinski and Ahmed, 2023; Khan et al., 2023; Diez et al., 2015; Guan et al., 2021; Zhang et al., 2021; Zakerdoost and Ghassemi, 2023; Zhang et al., 2024b; Walker et al., 2024b; Villa et al., 2020). During early-stage design (also called concept design (Papanikolaou et al., 2024)), mission criteria may still be loosely defined, and it is worth comparing different classes of vessels (La Rosa et al., 2024; Bagazinski and Ahmed, 2023; Khan et al., 2023). Novel generative tools, such as those found in (Bagazinski and Ahmed, 2023; Khan et al., 2023), serve as a useful way to explore different design concepts and select the most promising baseline design for further consideration. However, generative design tools often have limited precision and are not able to iteratively refine local design areas (e.g., the bulbous bow). Furthermore, these tools are still not mature enough, so they have not yet been widely adopted by practitioners (La Rosa et al., 2024). Once a baseline hull (or parent design) is chosen from the early-stage, the design needs to be further refined and optimized in the so called mid-stage (also referred to as contract design (Papanikolaou et al., 2024)). In this stage, the representation space is limited to variants around the parent design (Diez et al., 2015; Guan et al., 2021; Zhang et al., 2021; Zakerdoost and Ghassemi, 2023; Zhang et al., 2024b; Walker et al., 2024b), and a different set of tools is used to

[*] Corresponding author.
E-mail addresses: j.m.walker@tudelft.nl (J.M. Walker), luca.oneto@unige.it (L. Oneto), a.coraddu@tudelft.nl (A. Coraddu).

https://doi.org/10.1016/j.oceaneng.2025.122607
Received 25 February 2025; Received in revised form 18 August 2025; Accepted 25 August 2025
Available online 2 September 2025

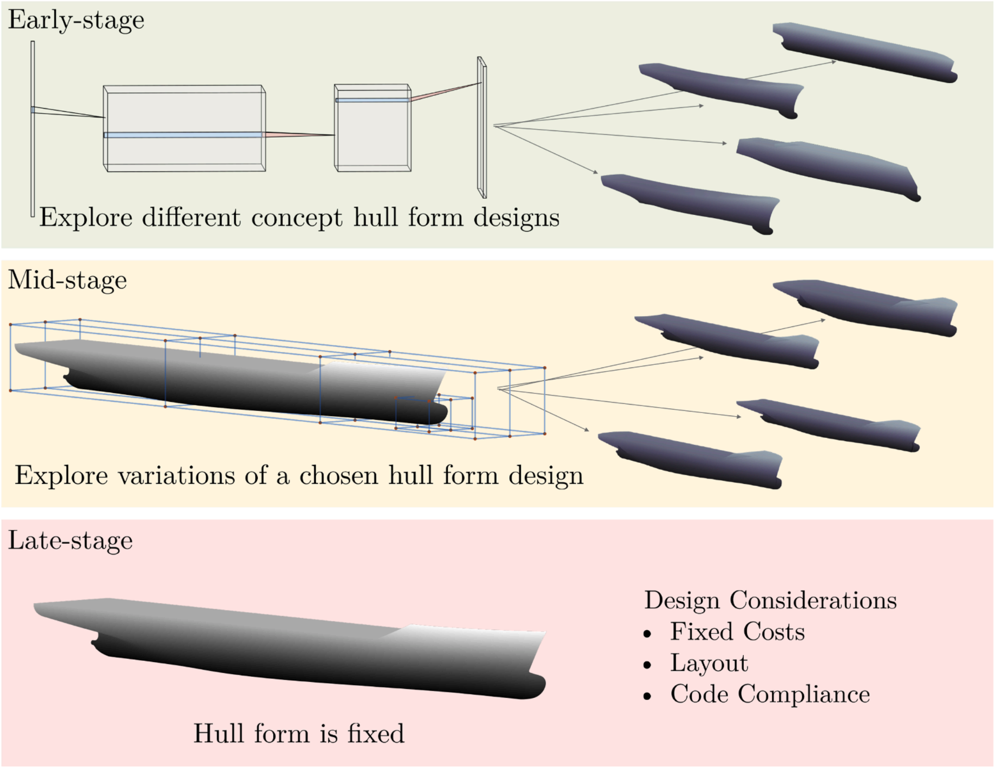


Fig. 1. Hull form design stages.

generate the candidate hull forms during optimization (Villa et al., 2020). The currently favoured approach to mid-stage design optimization is a simulation-based design pipeline (Diez and Serani, 2024). In particular, this stage leverages a Data-Driven Surrogate (DDS) of high-fidelity models to predict, less accurately but at a fraction of the computational costs, and then optimize different performance metrics of candidate designs (Walker et al., 2024a). Finally, during late-stage design, the hull form may be fixed due to mission criteria and costs, and designers focus their attention on other design considerations, e.g., the internal layout (Zhang et al., 2024a), safety (Papanikolaou, 2009), and code compliance (Huang et al., 2023). A summary of the aforementioned stages of hull form design is depicted in Fig. 1.

All stages of hull form design still have room for improvement both from the research and industrial point of view (Diez et al., 2015; D'Agostino et al., 2020; Villa et al., 2020; Guan et al., 2021; Renaud et al., 2022; Zakerdoost and Ghassemi, 2023; Zhang et al., 2024b; Walker et al., 2024b; Bertram et al., 2024). However, most of the attention nowadays is towards the mid-stage design (Diez et al., 2015; Villa et al., 2020; Guan et al., 2021; Zhang et al., 2021; Zakerdoost and Ghassemi, 2023; Diez and Serani, 2024; Walker et al., 2024b; Zhang et al., 2024b) because early-stage modifications can have a lot of impact on the production pipelines (Walker et al., 2024a) while late-stage designs often focus on aspects not directly connected with the hull geometry (e.g., internal layout) or are hard to encode in a mathematical framework that can be directly optimized (e.g., regulations) (Huang et al., 2022).

During the mid-stage, a DDS-based approach to hull optimization is preferred. This approach follows four-steps (for a detailed overview, please refer to Walker et al. (2024a)). The main idea of the four-step approach is to establish a well-defined parameterization and search space, perform a limited number of high-fidelity simulations that serve as the foundation for the DDS, perform the numerical optimization leveraging the DDS, and check the output using a high-fidelity simulation (e.g., Computational Fluid Dynamics - CFD). This output assumes the form of a series of optimal designs, the ones on the Pareto frontier, which are then validated in terms of physical plausibility providing feedback (sometimes also integrated with late-stage designs (Huang et al., 2023)) able to refine the mid-stage design (Liu et al., 2022a; Walker et al., 2024b; Wang et al., 2021; Zakerdoost and Ghassemi, 2023).

The concept of physical plausibility is crucial; however, it is not easy to formally define. For example, it is possible to check that the performance of the designs on the Pareto frontier are not actually numerical artifacts (e.g., too small resistance), induced by the use of the DSS, by performing high-fidelity simulation (Walker et al., 2024b). Some checks are more complex to perform, for example, the a-posterior verification of the performance of the geometry (e.g., seakeeping, stability, structural integrity) (Huang et al., 2022). Humans must also perform some other checks. For example, when the parameter space is defined too large, the optimizer can induce a geometry unsuitable for a specific application (e.g., too tight or too round to transport goods) (Bagazinski and Ahmed, 2023). Finally, some checks are not easy to code into a mathematical framework, such as regulatory aspects (Huang et al., 2022).

The two main lacks of current approaches to mid-stage hull form design are

Lack (1) the limited ability to take into account the stability-related properties of the design when focusing mainly on resistance (Huang et al., 2022);

Lack (2) the computational requirements of the tool even when a DDS is employed (Diez and Serani, 2024).

Lack (1) has already been addressed by incorporating the stability considerations a-posteriori (Nowacki, 2019; Yrjänäinen et al., 2019; Birk and Harries, 2003; Mesbahi and Atlar, 2000; Maissonneuve, 1993, 1994) or by including Response Amplitude Operator (RAO) in the objective of the optimization problem (Diez et al., 2015). Lack (2), instead, has already been addressed by applying Design Space Dimensionality Reduction (DSDR) (Diez and Serani, 2024) to the parameter search space to speed up the optimization (i.e., the smaller the number of parameters to be optimized, the faster the optimization is). See Section 2 for more details.

The problem with the current approach to Lack (1) is that it is not trivial to trade-off the existing metrics (e.g., RAO) with resistance, as there are no clear, predefined criteria to determine, a priori, what constitutes a good or bad RAO, nor how much resistance should be sacrificed for improved RAO performance. Instead, these trade-offs are often empirically evaluated by human experts. The problem with the current approach to addressing Lack (2) instead is that DSDR is often performed a priori to optimization. Although the DSDR approach preserves variability in the design space, it does so without incorporating the necessary knowledge to retain the best-performing designs (in terms of at least one of stability or resistance) while discarding the poorly performing designs. Moreover, to the best of the authors' knowledge, no work in the literature has suggested an approach to address both lacks simultaneously.

The purpose of our work is to take a step forward in finding the solution in both Lack (1) and Lack (2) and to blend the solutions to address them simultaneously. Specifically, to address Lack (1), we consider, the International Maritime Organization (IMO) Intact Stability Code (Organization, 2020; Petacco and Gualeni, 2020; Marlantes et al., 2021; Peters and Belenky, 2022), directly in the hull optimization problem. This constraint on stability allows us to easily distinguish between acceptable and unacceptable geometries. However, the problem with this constraint is that it is computationally challenging to assess, so we propose, for the first time, to surrogate it to include it in the optimization loop with minimal impact on the computational requirements. In order to address Lack (2), we compared different DSDR techniques able to reduce the number of parameters to optimize, i.e., Principal Component Analysis (PCA) and Neural Networks. Finally, in order to blend these solutions toward addressing Lack (1) and Lack (2) we incorporate the stability constraints in the DSDR and show that this further improves both the quality and computational requirements of the mid-stage hull form optimization schema. In particular, we build on our previous work (Walker et al., 2024b) which represents the state-of-the-art pipeline for mid-stage hull form optimization without considering the stability constraints. We test the quality of our proposal by optimizing the KCS hull form which has been well analyzed in the literature (Coppedè et al., 2019; Villa et al., 2020; Miao and Wan, 2020; Feng et al., 2021). Our results demonstrate two main achievements. First, we find that current optimization pipelines fail to meet the IMO Intact Stability Code, whereas our method achieves compliance by design. Second, our approach reduces the computational burden by 30% without sacrificing performance, representing a significant step forward in practical hull form design optimization.

The rest of this paper is as follows. Section 2 reviews the most relevant work to our proposal that supports the statements we made in the introduction; Section 3 describes the specific problem we examine in this work and the dataset we leverage to address it; Section 4 decribes the method we use to solve the problem using the data described in the very same section; Section 5 presents the results; finally, Section 6 concludes the work.

2. Related work

This section reviews the main works in the literature related to the mid-stage hull form design. In particular, we review the main works that deal with DSDR, to reduce computational requirements of the optimization, and the ones that incorporate stability into optimization as, to the best of the authors' knowledge, no work in the literature blends the two aspects.

Several authors have demonstrated how DSDR methods can be applied to reduce the computational burden of hull form optimization (Diez and Serani, 2024). The most popular approaches are linear dimensionality reduction methods (Villa et al., 2020; D'Agostino et al., 2020; Zhang et al., 2024b).

In Villa et al. (2020), the authors considered Proper Orthogonal Decomposition (POD) basis functions for DSDR of a Free-Form Deformation design space. Two approaches were examined: first, authors performed DSDR using geometric-based criteria highlighting new problem-dependent criteria for ranking POD modes, significantly reducing the design space size. Second, the authors apply POD to Design Velocities, directly correlated to shape variations, which demonstrated reducing a 68-dimensional parametrization of the DTC hull form to 30 variables for shape reconstruction. This work showed that the POD-based approach was effective for generating hull shape variations for optimization processes.

In D'Agostino et al. (2020), DSDR is performed using the Karhunen-Loève Expansion (Diez and Serani, 2024), which is based on eigenfunction decomposition, for assessing the design space. Authors use a quantitative measure to show the design variability in the eigenvalue space and reduce the dimensionality of a parametrizaion of the DTMB5415 from 100 design variables to 25. In fact, authors generate 40000 candidate hulls but later reduce that to 2400 feasible samples and perform the hull form optimization. This method was able to generate a reparameterization of the shape modification vector for efficient optimization. However, this reparametrization does not allow the reduced-dimensionality representation to be projected in the original design space. This presents a challenge because well-established parametric models are commonly used by practitioners to preserve specific knowledge of existing designs (Serani and Diez, 2023).

In Zhang et al. (2024b), the authors parameterized the KCS hull form using 37 design variables controlling a T-spline geometry representation. They performed DSDR using PCA and showed that, for varying levels of data compression (e.g., preserving between 68 % and 100 % of the original design variance), it was possible to obtain optimized hull forms (in terms of resistance) compared to the parent design. The study demonstrated that 6-dimensional parametrizations could achieve optimized hull forms with minimal computational effort (e.g., within tens of iterations). However, lower-dimensional parametrizations were consistently outperformed according to the key performance indicators when more than hundreds of iterations were used.

Regarding the integration of stability constraints into the mid-stage hull form design, it is possible to find several milestone works (Birk and Harries, 2003; Mesbahi and Atlar, 2000; Maissonneuve, 1993, 1994; Diez et al., 2015; Zakerdoost and Ghassemi, 2023; Renaud et al., 2022; Guan et al., 2021).

In Diez et al. (2015), authors presented the multi-objective hull form optimization of the DTMB5415. In particular, the study focused on improving hydrodynamic performance, both in terms of resistance and the vertical acceleration of the bridge, using low fidelity solvers. A merit factor based on the vertical acceleration of the bridge (located 27[m] forward amidships and 24.75[m] above keel) at 30[kn] in head waves and is the roll angle at 18[kn] in stern long-crested wave was used as the objective function during optimization. Results showed resistance and vertical acceleration of the bridge improvements of 6÷23 % and high-fidelity simulations verified the findings.

In Zakerdoost and Ghassemi (2023), authors performed multi-level hull-propulsor optimization integrating medium- and low-fidelity solvers. A bi-fidelity Co-Kriging surrogate model and a multi-objective evolutionary algorithm were used to optimize the lifetime fuel consumption and energy efficiency design index of the S175 containership with KP505 propeller and MAN B&W engine. In particular, the authors imposed constraints on metacentric height (GM) so that the value would be greater than a given value of $GM_0 = 0.2$[m] based on the IMO Intact Stability Guidelines.

In Renaud et al. (2022), performed surrogate-based multi-objective optimization of a SWATH hull form, considering stability performance and ship resistance. A parametric model of the SWATH ship, with varying torpedo semi-axis and strut angles, is optimized using Gaussian process surrogate models. Three fidelity levels for ship resistance are considered, ranging from wetted surface calculations to high-fidelity CFD simulations with stabilizing fins. Results showed it was possible to optimize for stability, resistance, and trade-off in terms of the median comprimise of both objectives.

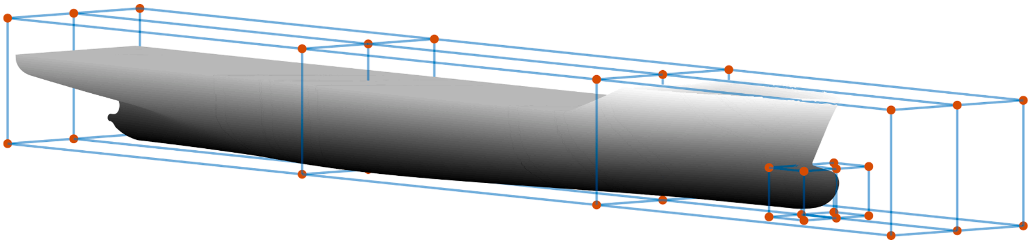


Fig. 2. The FFD control network shows the global and local control networks, which account for a total of 36 control points.

In Guan et al. (2021), authors presented the multi-objective optimization method for unmanned surface vehicle hull design. The optimization considers wave resistance, vertical acceleration, and pitch amplitude, using a parametric model to generate candidate designs. To reduce computation time, a second-order response surface method surrogated numerical simulations for the resistance, vertical acceleration, and pitch amplitude parameters, and the optimization was performed with the Particle Swarm and Sequential Quadratic Programming algorithms. Results on a 7[m] vessel demonstrated the effectiveness of the proposed method integrating the vertical acceleration and pitch amplitude into the objective function.

3. Problem and data description

In this work, we propose a novel computationally aware and physically plausible hull form optimization framework to address the two main gaps of the current literature Lack (1) and Lack (2).

For Lack (1), we propose to integrate the IMO Intact Stability Guidelines (Organization, 2020; Petacco and Gualeni, 2020; Marlantes et al., 2021; Peters and Belenky, 2022) directly into the optimization problem. For Lack (2), we propose using DSDR techniques (D'Agostino et al., 2020; Diez and Serani, 2024) to reduce the number of parameters we need to optimize. Moreover, for the first time in the literature, we simultaneously address both identified lacks by blending the two solutions. Specifically, we propose incorporating the stability constraints into the DSDR and show that this further improves both the quality and computational requirements of the mid-stage hull form optimization schema. We test our approach by optimizing the KCS hull form in a state-of-the-art mid-stage hull form optimization pipeline. For this purpose, we build on our previous work (Walker et al., 2024b) that leveraged the mid-stage hull form optimization pipeline without considering the stability constraints.

In this work, we optimize the KCS hull form which has a complex topology (i.e., a cargo ship with a bulbous bow).

The pipeline for Mid-Stage Hull Form Optimization (MSHFO) consists of four main steps (Walker et al., 2024a)

MSHFO (1) Shape parametrization, parameter ranges, and KPIs definition;
MSHFO (2) Sampling, data generation, and DDS;
MSHFO (3) Shape optimization;
MSHFO (4) Physical plausibility and feedback;

that we will improve to address Lack (1) and Lack (2).

Regarding MSHFO (1), the KCS requires a suitable parameterization and parameter ranges. To this end, we utilize the Free-Form Deformation (FFD) method to parameterize the KCS hull. FFD is an effective way to create a diverse design space while ensuring that the designs maintain geometric continuity and feasibility (Coppedé et al., 2018; Villa et al., 2020; D'Agostino et al., 2020). For the FFD parametrization, we overlay a control network, characterized by control points, over the parent design. We then induce various candidate hull form designs by displacing the control points. We utilized an in-house developed FFD tool that leverages subdivision surfaces, as referenced in Coppedé et al. (2018), along with an empirically defined control network. The control network consists of two overlapping subnetworks: a global network that covers the entire hull and includes 24 control points, and a local network focused on the bulbous bow comprising 12 control points. By structuring the networks hierarchically, the density of control points is increased around the bulbous bow due to its intricate geometry. Fig. 2 illustrates the FFD control network showing the global and local control networks, which account for a total of 36 control points.

A symmetry constraint is applied along the XZ plane to reduce complexity, bringing the number of independent control points down to 12. For the global control network, we restrict the Degrees of Freedom (DoF) of the control points to allow displacements only along the X and Y axes. However, we do not restrict the control points in the local control network. Hence, the independent control points were adjusted according to a vector $\boldsymbol{x} = [x_1, \cdots, x_{29}]$, where $[x_1, \cdots, x_{16}]$ govern the displacements of the global control points (e.g., 2 DoF for each of the 8 independent global control points) and $[x_{17}, \cdots, x_{29}]$ manage the displacements of the local ones (e.g., 3 DoF for each of the 4 independent local control points). The original placement of these control points establishes the parametrization rule, $\mathbb{R}$ (which is fixed during optimization). $\mathbb{R}$ allows for a specific candidate geometry, Ω, to be created from the parent geometry, Ω^{p}, according to the displacement vector $\boldsymbol{x}$ (which varies during optimization), such that $\Omega = \mathbb{R}(\Omega^{\mathrm{p}}, \boldsymbol{x})$. It is important to note that $\mathbb{R}$ is homomorphic, as it should be according to the literature on the topic (Walker et al., 2024a,b). For the parameter ranges, we defined the lower and upper bounds of each parameter of $\boldsymbol{x}$ in such a way as to ensure the solution is inside the range, i.e., a posteriori, we check that each solution we found was actually inside the range and none actually belonged to the borders. We set the lower ($\boldsymbol{x}^l$) and upper ($\boldsymbol{x}^u$) bounds of the parameters in the global control network, e.g., $\boldsymbol{x}^l_{1,\cdots,16}$ and $\boldsymbol{x}^u_{1,\cdots,16}$, so that they could move ±10 % along the length of the vessel and up to ±20 % the width of the vessel. Similarly, we set the lower and upper bounds of the parameters for the local control network, e.g., $\boldsymbol{x}^l_{17,\cdots,29}$ and $\boldsymbol{x}^u_{17,\cdots,29}$, so that they could move ±10 % in each direction of the bulbous bow. Fig. 3 shows

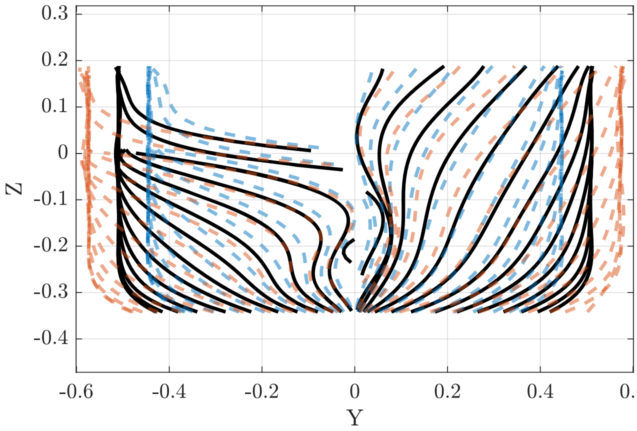


Fig. 3. Linesplans of the KCS hull in black and the geometries represented by $\boldsymbol{x}^u$ and $\boldsymbol{x}^l$ in dashed orange and blue, respectively.

the linesplans of the KCS hull and the geometries on the border of the representation space.

In MSHFO (1), apart from leveraging the symmetrical properties, we can further reduce the parameters to be optimized. This allows us to address Lack (2) and reduce the computational cost of the optimization phase. To this end, we employ DSDR algorithms such as the PCA and the Neural Networks to reduce the dimensionality of the parametrization. The first sanity check of DSDR is to ensure that the inverse operator of the PCA or the Neural Networks, which maps the reduced dimensional representation back to the original parameterization, performs well (e.g., it should reconstruct the original parameters with minimal error), see Section 4.3 for details. Note that this sanity check does not guarantee that the optimizer will still perform effectively (e.g., find the optimal hull in the original parameterization), as the impact of reconstruction error on the ability of the optimizer to evaluate the performance of candidate designs is unknown. A more informative check will be performed later in the process that is not standard in the literature, see explanation of MSHFO (3).

Regarding MSHFO (2), we need to generate the data from scratch by performing a limited number of CFD simulations. Selecting the right simulation to be performed, i.e., sampling a representative selection of hulls, is a critical step in the process (Walker et al., 2024a). In this work, we opt for a Full Factorial Design of Experiments (Antony, 2014). Due to computational constraints, we randomly sample 1000 geometries from the full factorial. The choice of 1000 samples, in line with the literature (Diez and Serani, 2021), has been made since it allows for adequate sampling of the design space defined by the parameterization, ensuring that the range of possible hull forms is sufficiently represented, covering a wide variety of designs that may meet the desired objectives and constraints. Subsequently, for each sampled hull, we performed

- a high-fidelity CFD simulation to estimate total Resistance (R_t) at three different Froude numbers (Fr) selected uniformly random in a range $0.108 \div 0.282$ [${}^{m}/_{s}$];
- a panel code simulation to assess the stability performance according to the IMO Intact Stability Code.

For what concerns the high-fidelity CFD, we employ the same approach and software as in Walker et al. (2024b) with some additional fine-tuning and validation (described in Section 4.1) for the particular problem under examination. Due to the high computational cost of the CFD model, we performed the CFD for 218 randomly selected geometries out of 1000. The choice of 218 samples exceeds the typical number of samples used in the literature to develop a DDS (Walker et al., 2024a). A description of the dataset to learn R_t and the number of examples for each feature can be found in Table 1. For what concerns the IMO Intact Stability Code, we focus on a single Stability Failure Mode (SFM): the Dead Ship Condition. We leverage an in-house developed panel code, as referenced in Coraddu et al. (2011, 2012), to assess the Dead Ship SFM considering the International Code on Intact Stability Section 2.3 Severe wind and rolling criterion (weather criterion) (Organization, 2008) using the righting-arm approach (Coraddu et al., 2011). The weather criterion (Organization, 2008) mandates that for a steady wind heeling lever (l_{w1}), angle of equilibrium (φ_0), wave action roll (φ_1), angle of heel (φ_2), and gust wind heeling lever (l_{w2}) evaluated as ${}^{3}/_{2}\, l_{w1}$, the area b (bounded between l_{w2} on the bottom and the GZ curve on the top) shall be equal to or greater than area a (bounded between GZ curve on the bottom l_{w2} on the top, and up to φ_1), as indicated in Fig. 4.

Table 1
Dataset Desription to learn R_t (218 samples for $R_t(\boldsymbol{x}, 0.108)$ and 218 samples for $R_t(\boldsymbol{x}, 0.282)$).

Type	Feature	Symbol	Range
Input	Global parametrization [−]	$x_1, \cdots, x_{16}$	$\{-0.74, -0.71, \cdots, 0.74\}$
	Local parametrization [-]	$x_{17}, \cdots, x_{29}$	$\{-0.05, -0.02, \cdots, 0.05\}$
	Froude Number [-]	Fr	$[0.108 \div 0.282]$
Output	Total Resistance [N]	$R_t(\boldsymbol{x}, \mathrm{Fr})$	$[12.9 \div 221.2]$

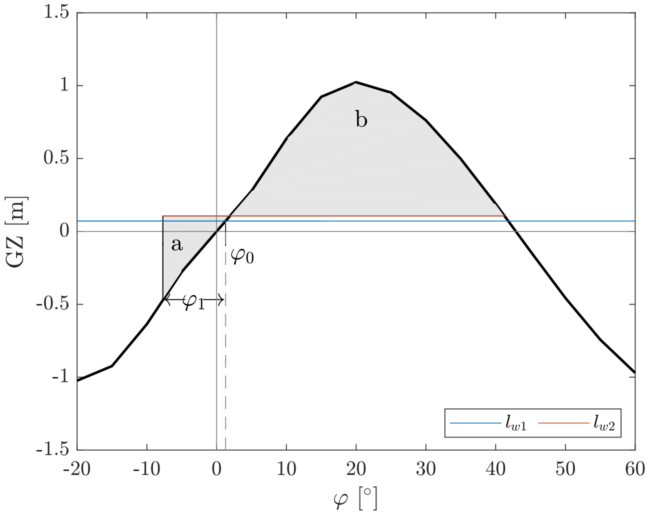


Fig. 4. Example Righting-arm.

We also paid attention to four additional criteria on top of the Dead Ship Condition coming from the same code section (Organization, 2008)

SFM (1) The Dead Ship Condition;
SFM (1.1) The area under the GZ curve from 0° to 30°;
SFM (1.2) Minimum GZ at 30°;
SFM (1.3) The angle at which the max righting arm occurs;
SFM (1.4) The minimum initial GM.

The SFM criteria for each specific geometry were determined in the following way. Initially, we computed the actual values for GZ (for 17 different transversal inclination angle φ sampled uniformly in the range $[-20° \div 60°]$) and GM, with the panel code because each of the criteria, i.e., SFM (1), SFM (1.1) $\div$ SFM (1.4), are related to these quantities. A description of the dataset to learn GZ and GM and the number of examples for each feature can be found in Table 2. Subsequently, we derived the actual values for each of the SFMs, i.e., the areas a and b, the area under the GZ curve from 0° to 30°, the GZ at 30°, the angle of the maximum righting arm, and GM. Finally, these values are compared against predefined thresholds from the IMO Intact Stability Code and the design is classified as either “pass” or “fail” indicating whether it meets the thresholds. In this way the SFM criteria are mapped from their actual values into a binary (“pass” 0 and “fail” 1) condition (or constraint). The last step is to condense the five binary conditions, i.e., SFM (1), SFM (1.1) $\div$ SFM (1.4), into a single function IMO($\boldsymbol{x}$) that defines if a particular hull, generated by the parametrization $\boldsymbol{x}$, passes all of the stability

Table 2
Dataset Desription to learn GZ and GM (1000 samples for GM and 17000 samples for GZ namely 1000 geometries for 17 transversal inclination angle $\varphi \in [-1.25 \div 1.25]$).

Type	Feature	Symbol	Range
Input	Global parametrization [−]	$x_1, \cdots, x_{16}$	$\{-0.74, -0.71, \cdots, 0.74\}$
	Local parametrization [-]	$x_{17}, \cdots, x_{29}$	$\{-0.05, -0.02, \cdots, 0.05\}$
	Transversal Inclination [°]	φ	$[-20 \div 60]$
Output	Righting Arm [m]	GZ($\boldsymbol{x}, \varphi$)	$[-1.5 \div 1.6]$
	Metacentric Height [m]	GM($\boldsymbol{x}$)	$[11.5 \div 16.8]$

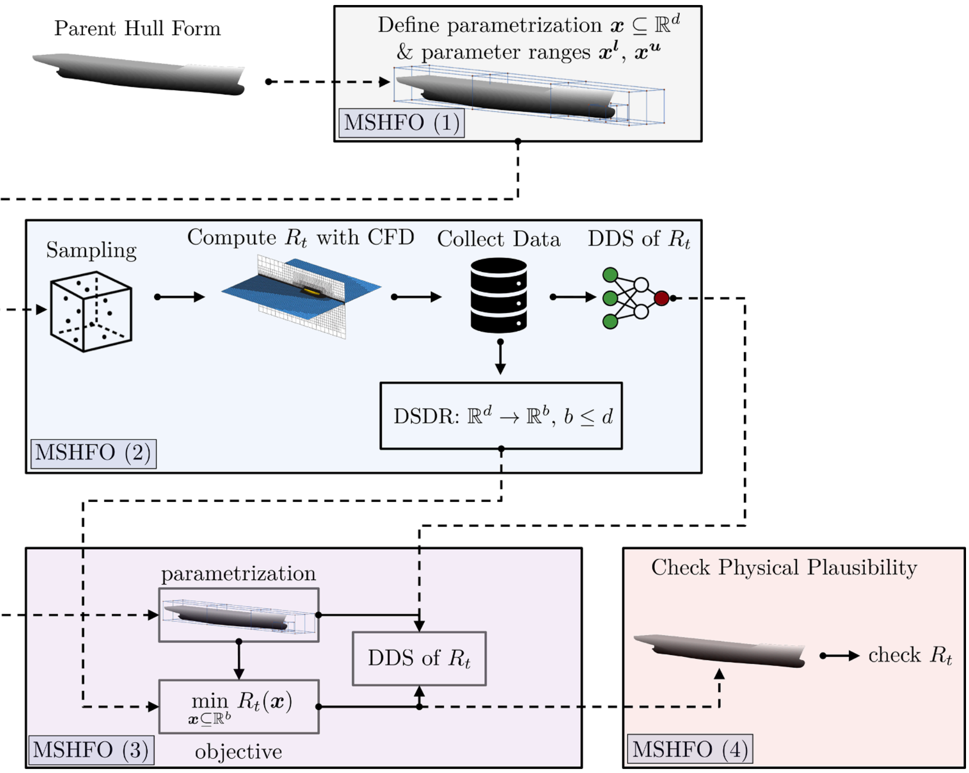

Parent Hull Form
Define parametrization $\boldsymbol{x} \subseteq \mathbb{R}^d$
& parameter ranges $\boldsymbol{x^l}$, $\boldsymbol{x^u}$
MSHFO (1)
Sampling
Compute R_t with CFD
Collect Data
DDS of R_t
DSDR: $\mathbb{R}^d \to \mathbb{R}^b$, $b \leq d$
MSHFO (2)
parametrization
DDS of R_t
$\min_{\boldsymbol{x} \subseteq \mathbb{R}^b} R_t(\boldsymbol{x})$
objective
MSHFO (3)
Check Physical Plausibility
check R_t
MSHFO (4)

Fig. 5. Classic MSHFO pipeline.

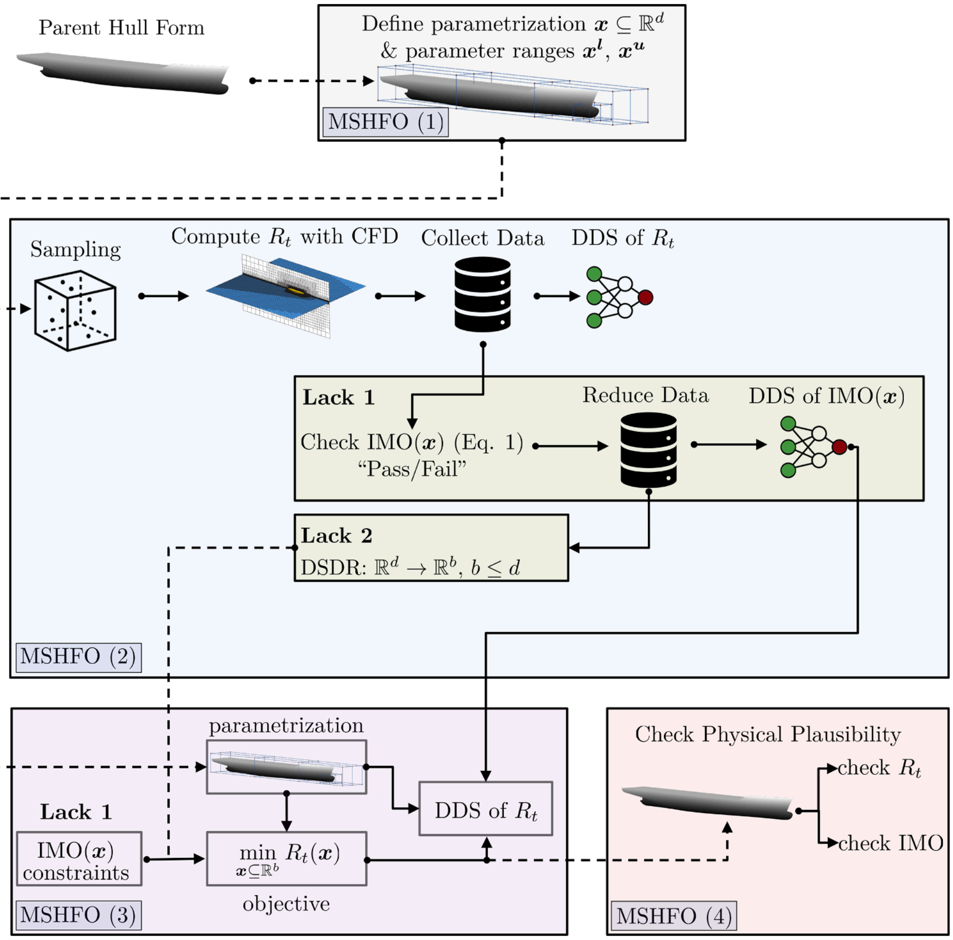

Parent Hull Form
Define parametrization $\boldsymbol{x} \subseteq \mathbb{R}^d$
& parameter ranges $\boldsymbol{x^l}$, $\boldsymbol{x^u}$
MSHFO (1)
Sampling
Compute R_t with CFD
Collect Data
DDS of R_t
Lack 1
Reduce Data
DDS of IMO($\boldsymbol{x}$)
Check IMO($\boldsymbol{x}$) (Eq. 1)
"Pass/Fail"
Lack 2
DSDR: $\mathbb{R}^d \to \mathbb{R}^b$, $b \leq d$
MSHFO (2)
parametrization
DDS of R_t
Lack 1
IMO($\boldsymbol{x}$)
constraints
$\min_{\boldsymbol{x} \subseteq \mathbb{R}^b} R_t(\boldsymbol{x})$
objective
MSHFO (3)
Check Physical Plausibility
check R_t
check IMO
MSHFO (4)

Fig. 6. MDHFO pipeline with the proposed modifications to address Lack (1) and Lack (2).

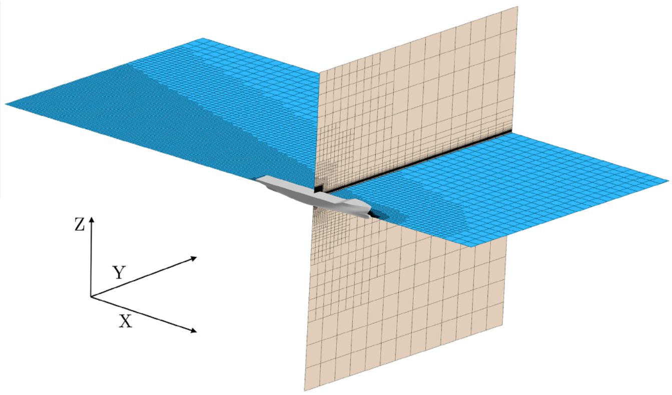


Fig. 7. The mesh exploited for the CFD simulations with the Star CCM+[1] package. The mesh included a surface mesh refinement on the vessel hull and on the boundaries of the domain, in addition to volume mesh refinements around the hull, wake, and free surface.

criteria

$$\mathrm{IMO}(\boldsymbol{x}) = \begin{cases} 1 \quad \text{if} & \begin{cases} a(\boldsymbol{x}) \le b(\boldsymbol{x}), \\ \displaystyle\int_{0^\circ}^{30^\circ} \mathrm{GZ}(\boldsymbol{x}, \varphi)\, d\varphi \ge 0.055, \\ \mathrm{GZ}(\boldsymbol{x}, 30^\circ) \ge 0.2\mathrm{m}, \\ \mathrm{GZ}(\boldsymbol{x}, \varphi_{\max}) \text{ occurs at } \varphi_{\max} \ge 25^\circ, \\ \mathrm{GM}(\boldsymbol{x}) \ge 0.15\mathrm{m} \end{cases} \\ 0 \quad \text{otherwise} \end{cases} \qquad (1)$$

Leveraging the datasets described in Tables 1 and 2, we build DDSs of R_t and $\mathrm{IMO}(\boldsymbol{x})$ and validate its performance (see Section 4.1).

After MSHFO (2), we can move to MSHFO (3) (see Section 4.2). Once the numerical optimization is performed, and the series of optimal designs, i.e., the Pareto frontier of solutions, have been obtained, we have to verify that the optimizer has performed effectively and found the optimal hulls in the original parameterization. When the optimization is performed without DSDR, we skip directly to validating the solutions in terms of physical plausibility, MSHFO (4). However, when the optimization had been performed in a reduced dimensionality representation, i.e., obtained through DSDR, we needed to assess the impact of the reconstruction error on the optimizer's ability to evaluate the performance of candidate designs. To perform this assessment, we proceed as follows

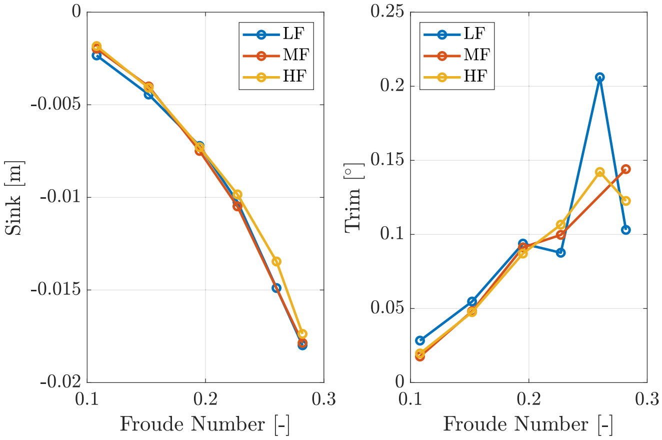


Fig. 9. Comparison of the Sink and Trim for the 3 levels of mesh refinement over varying Fr number.

- we obtained the Pareto frontier of solutions expressed in the reduced dimensionality representation;
- we mapped the reduced dimensionality representation back to the original parameterization using the inverse operator (i.e., PCA or Neural Networks);
- we evaluated the performance (in terms of resistance and stability) of the candidate solutions using the DDS built from the datasets described in Tables 1 and 2. This provides us with the estimated performance, including the reconstruction error.
- we check the performance without the reconstruction error. For this purpose, we estimated the performance based on the reduced dimensionality representation. Hence, we retrained the DDS using the same data as in Tables 1 and 2, but we performed DSDR on the input space, so the DDS accepted the reduced dimensionality representation as input. Using the DDS trained on the reduced dimensionality input, we evaluated the performance of the Pareto optimal solutions without the reconstruction error
- we compared the performance estimates obtained from the DDSs trained on the different input spaces to assess the impact of the reconstruction error on the optimizer's ability to identify optimal hulls in the original parameterization

This approach provided a more informative check than merely evaluating the reconstruction error, which had not been standard in the literature, and ensured that the optimization had been effective.

After MSHFO (3) we can move to MSHFO (4). Following the approach of Walker et al. (2024b) we first check that the performance of

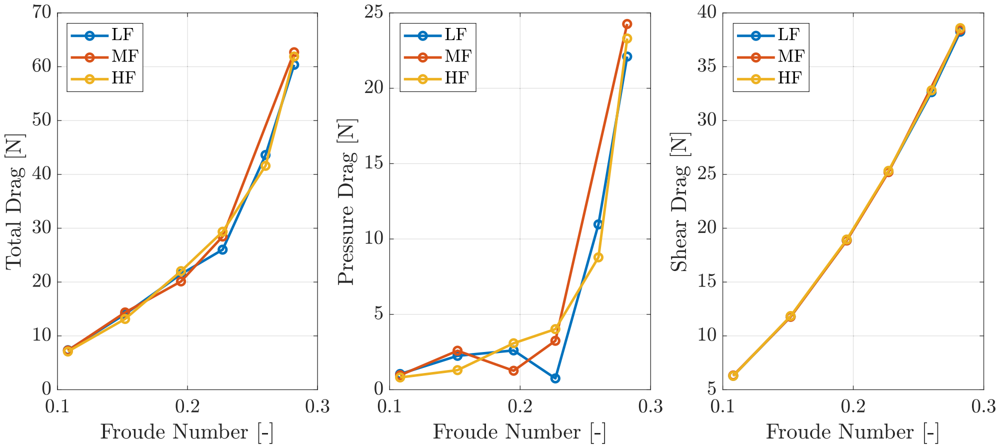


Fig. 8. Comparison of the Total, Pressure, and Shear Drag for the 3 levels of mesh refinement over varying Fr number.

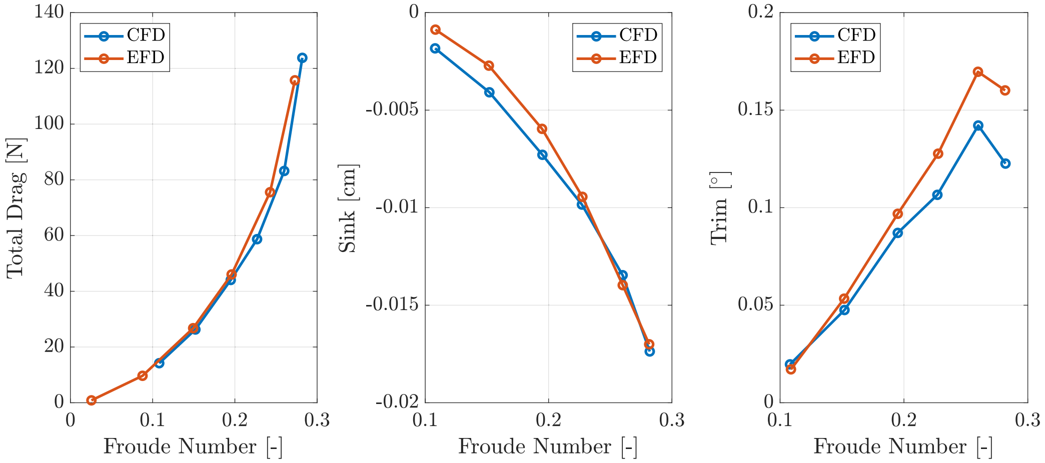


Fig. 10. Comparison of the Total Drag, Sink and Trim for the selected CFD model and EFD results over varying Fr number.

the designs on the Pareto frontier, in terms of R_t, are not actually numerical artefacts induced by the DSS and match the performance indicated by high-fidelity CFD. Furthermore, following the same principle as the check on R_t, we check that the performance of the DDS matches the panel code when assessing Eq. (1).

To summarize, we mofided the MSHFO pipeline to simultanaously address Lack (1) and Lack (2). For Lack (1), we propose to integrate the IMO Intact Stability Code directly into the optimization problem, i.e., modifying MSHFO (3). For Lack (2), we propose using DSDR techniques (D'Agostino et al., 2020; Diez and Serani, 2024) to reduce the number of parameters we need to optimize, i.e., modifying both MSHFO (2) and MSHFO (3).

More in detail, we can summarize our modification to the MSHFO pipeline as follows. The hard constraint ("pass" or "fail") derived from the IMO Intact Stability Code, i.e., according to Eq. (1), is inserted directly in the optimization problem. The same constraint is used to remove from the dataset the geometries that do not satisfy it, reducing the number of samples in the dataset of Table 2, to be used during the DSDR. This allows us to build a low dimensional search space that accounts for the IMO Intact Stability Code. At first sight, it may seem incomplete that we only reduced the dataset based on the IMO Intact Stability Code and did not account for the geometric constraints on ∇, T, and θ. However, this is a deliberate consideration, as the geometric constraints remain tunable by the user (depending on design preferences), while the IMO 2nd Generation Intact Stability constraints are hard ones.

Fig. 5 visually depicts the classic MSHFO pipeline and Fig. 6 depicts the proposed pipeline with modifications to address Lack (1) and Lack (2).

4. Method

In this section, we have to describe how to fully address the problem described in Section 3 with the dataset described in the very same section. In particular, to solve Lack (1) and Lack (2) of the current approach to hull form optimization and blend them together there are a number of steps we need to explain in detail. Section 4.1 describes how we build the DDSs of R_t, GZ, and GM so they can be included in the optimization and how we validate their performance; Section 4.2 describes how we address Lack (1) by directly incorporating the IMO Intact Stability Code into MSHFO pipeline to optimize the KCS hull form. Section 4.2 describes how we address Lack (2), in line with the approaches described in Section 2, by using DSDR to reduce the computational demand of optimization. Moreover, for the first time in the literature, we describe how to simultaneously address Lack (1) and Lack (2) by incorporating the stability constraints directly into DSDR.

4.1. Data-driven surrogates for hull form optimization

In this section we describe how we develop the DDSs of R_t, GZ, and GM.

For what concerns the data generation, namely, the CFD model to determine R_t, the mesh generation, computation of the solution, and post-processing of the results is carried out in the commercial CFD package Star CCM+.[1] The baseline High-Fidelity (HF) mesh consists of approximately 1.5 million cells for half of the hull. The mesh is clustered around the hull, with extra refinement around the bulb and stern, across the free surface, and along the Kelvin angle. Fig. 7 provides an overview of the mesh configuration used in this study.

To compute the solution, a wall function approach was used to compute near-wall velocities, along with a prism-layer mesh in the boundary layer zone to achieve an average wall distance of approximately 60 [-]. This baseline configuration also serves as the finest mesh used in the sensitivity analysis. The calculations were performed at a time step of 0.01 [s] to compute steady-state regimes for multi-phase flows using the Volume of Fluid technique. The k-ω shear stress transport model was applied for turbulence closure. For what concerns the degrees of freedom of the simulation, we leveraged the Dynamic Fluid Body Interaction module of Star CCM+ to allow for sink and trim in calm water across a range of velocities (v) and consequently, Fn. To validate the state-of-the-art CFD model for the KCS hull form, we perform two checks commonly found in the literature (Coppedè et al., 2019)

- First, we carry out a mesh sensitivity analysis to determine simulation convergence;
- Second, we compare the predictions from our baseline mesh with available Experimental Fluid Dynamics (EFD) measurements of the calm water resistance of the model-scale KCS hull.

For the first task, namely, the mesh coarsening analysis, we created two coarser meshes by reducing the number of cells by approximately 1/3 each time, i.e., Medium-Fidelity (MF) and Low-Fidelity (LF) respectively. We reported the resistance in Fig. 8 and the sink and trim in Fig. 9 for each mesh under analysis. Table 3 reports the cell count for each mesh and computational requirements as the core hours using an Intel XEON E5 − 6248R 24C 3.0GHz CPU.

For the second task, since we validated the mesh convergence, we compared our baseline CFD prediction with the available EFD measurements. Table 4 reports the total drag, sink, and trim metrics for both the CFD and EFD predictions. Fig. 10 shows a visual comparison of these same metrics.

[1] https://plm.sw.siemens.com/en-US/simcenter/fluids-thermal-simulation/star-ccm/

Table 3
Cell count and computational requirements for each Mesh.

Mesh	$\times 10^6$ cells	Core Hours
HF	1.5	140
MF	1.0	95
LF	0.7	84

Table 4
Comparison of EFD and CFD results.

Model test		EFD			CFD		
v $[m/s]$	Fn $[-]$	Total Drag $[N]$	Sink $[cm]$	Trim $[°]$	Total Drag $[N]$	Sink $[cm]$	Trim $[°]$
0.915	0.108	15.3	−0.09	−0.017	14.2	−0.20	−0.028
1.281	0.152	27.8	−0.27	−0.053	26.3	−0.40	−0.051
1.647	0.195	45.6	−0.60	−0.097	44.0	−0.72	−0.094
1.922	0.228	65.5	−0.95	−0.127	58.7	−1.00	−0.130
2.196	0.260	98.4	−1.40	−0.170	83.2	−1.39	−0.174
2.379	0.282	127.6	−1.70	−0.160	123.8	−1.74	−0.152

Table 5
Hyperparameters and hyperparameters search space for the learning from data algorithms.

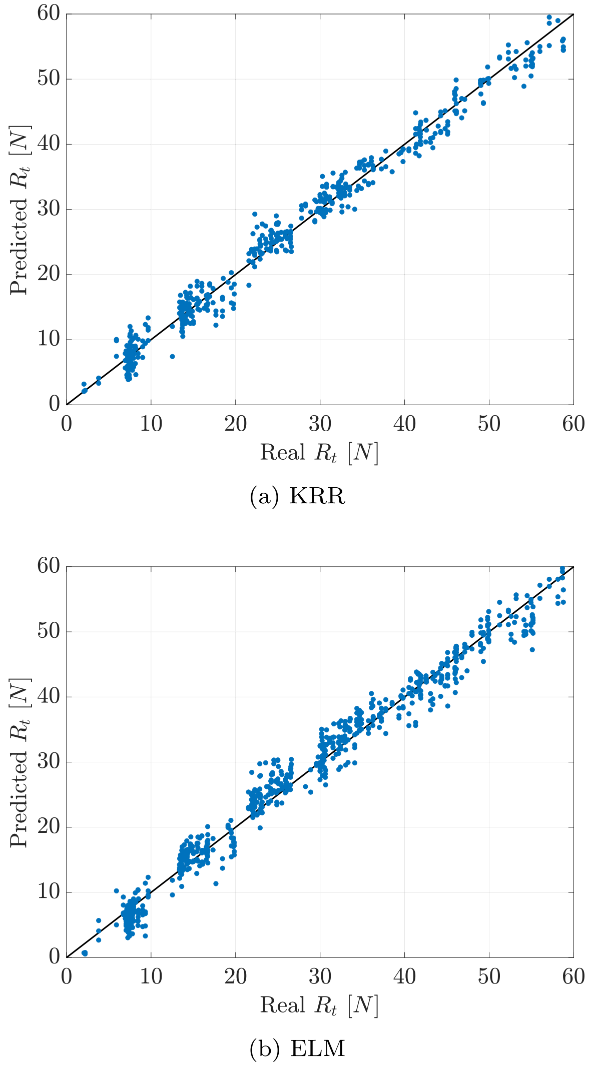
Algorithms	Hyperparameters
KRR	$\lambda : \{10^{-6}, 10^{-5.8}, \cdots, 10^{3}\}$ $\gamma : \{10^{-6}, 10^{-5.8}, \cdots, 10^{3}\}$
ELM	$h_l : \{2^5, 2^6, \cdots, 2^{16}\}$ $\lambda : \{10^{-6}, 10^{-5.8}, \cdots, 10^{3}\}$

Table 6
Hyperparameter settings for the different optimization algorithms.

Algorithm	Parameter	Value
IP	Maximum number of function evaluations allowed Maximum number of iterations allowed	10^6 10^6
AS	Maximum number of function evaluations allowed Maximum number of iterations allowed Maximum number of SQP iterations allowed	10^6 10^5 600

Table 7
DDS of R_t performance (in terms of accuracy - MAE, MSE, and MAPE - and time - Train Time and Test Time) for the different learning algorithms (KRR and ELM).

Learning	Accuracy			Time	
Algorithm	MAE $[N]$	MSE $[N^2]$	MAPE $[\%]$	Train $[s]$	Test $[\mu s]$
KRR	1.04 ± 0.07	1.32 ± 0.64	1.76 ± 0.09	322 ± 3	12.6 ± 5.6
ELM	1.59 ± 0.20	2.02 ± 0.83	4.09 ± 1.15	10095 ± 105	4.6 ± 2.7



(a) KRR

(b) ELM

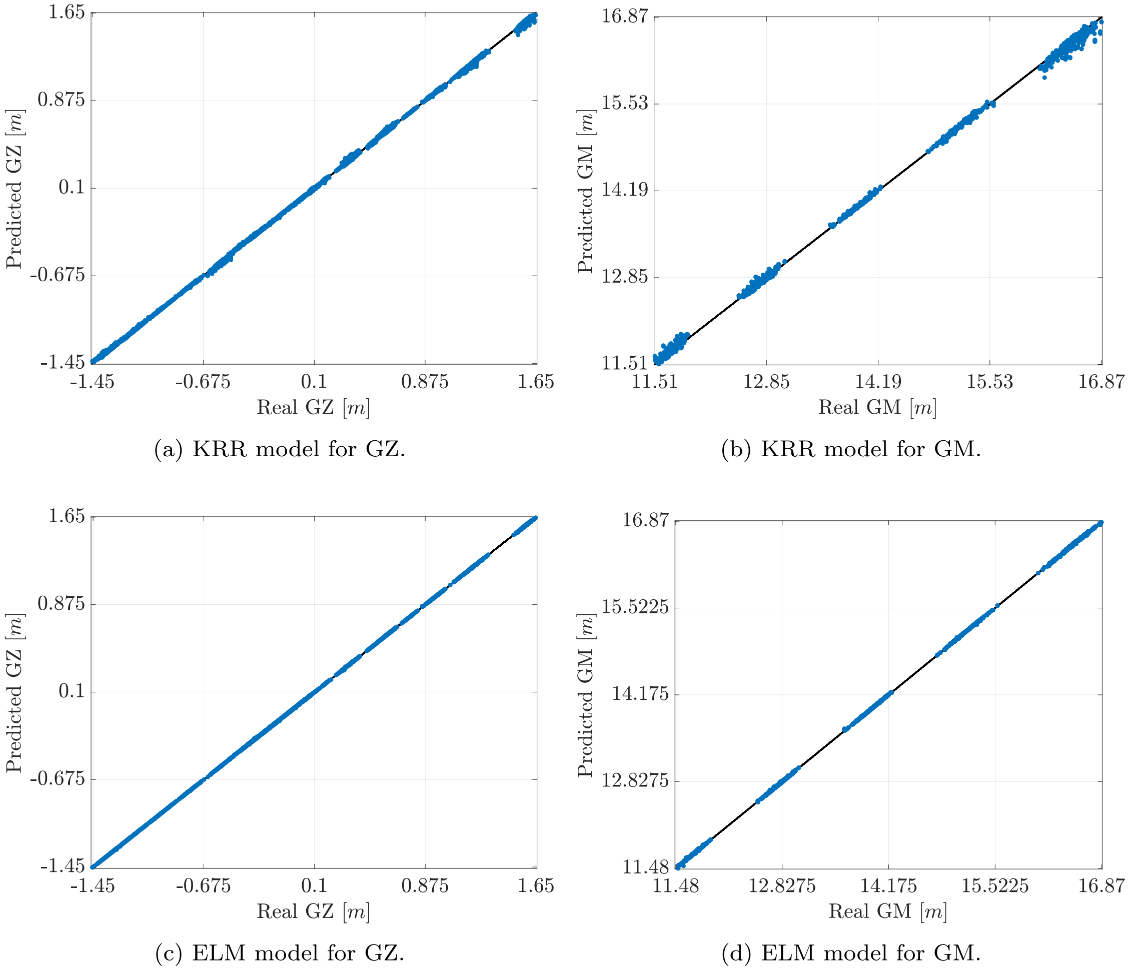
Fig. 11. Scatter plot of DDS of R_t for the models in Table 7.

It is worth reiterating that while the computational demand of the CFD simulation is manageable for evaluating the performance of a limited number of hull designs, it becomes a significant obstacle in an optimization process where thousands of hull designs must be evaluated. It is also worth mentioning that one of the key limitations of the CFD model is that it evaluates R_t at a model scale (the approximate ratio for this simulation is 1:31.6 due to computational constraints) and scaling resistance prediction from model-scale to full-scale ships is still a challenging task (Terziev et al., 2022), but this approach remains an appropriate and effective approximation while running hundreds of full-scale CFD simulations is not computationally tractable.

In this work, to build the DDSs of R_t (see Table 1), GZ and GM (see Table 2), we test two Machine Learning algorithms which come from the family of kernel methods (Shawe-Taylor and Cristianini, 2004) and neural networks (Goodfellow et al., 2016). Specifically, we test Kernel Ridge Regression (KRR) (Shawe-Taylor and Cristianini, 2004) and Extreme Learning Machine (ELM) (Huang et al., 2006b,a). These two models share an effective training phase and a smooth functional form (Rosasco et al., 2004) which is simple to optimized (Bottou et al., 2018) with respect to other methods like XGBoost (Chen and Guestrin, 2016) and Random Forests (Breiman, 2001). To build the DDSs using the data in Tables 1 and 2, we still have to face the problem of how to tune the hyperparameters for each learning from data algorithm (namely, model selection) and how to estimate the performance of the final model (namely, error estimation) (Oneto, 2020). During model selection, the main consideration is how to tune the hyperparameters of each algorithm. For KRR, we rely on the Gaussian Kernel for theoretical and practical reasons in Keerthi and Lin (2003). Thus we have to tune the regularization hyperparameter (λ) and the kernel hyperparameter (γ). Lastly, for ELM, we chose to rely on the Sigmoid activation function and tune the number of neurons in the hidden layer (h_l) and the regularization hyperparameter (λ). The hyperparameters and search space for each algorithm are found in Table 5. Since we fixed the boundaries of the representation space sufficiently large to ensure that the solution was not on the border, we only need to validate the performance of the DDS when interpolating within the boundaries of the dataset. Therefore, we validate the performance of the DDS by performing a Leave One Out cross-validation (Oneto, 2020). For what concerns error estimation, the accuracy is measured by different metrics: three quantitative (the Mean Absolute Error - MAE, the Mean Square Error - MSE, and the Mean Ab-

Table 8
GZ and GM DDSs: performance (in terms of accuracy - MAE, MSE, and MAPE - and time - Train Time and Test Time) for the different learning algorithms (KRR and ELM).

DDS	Learning	Accuracy			Time	
Target	Algorithm	MAE $\times 10^{-3}$ $[m]$	MSE $\times 10^{-3}$ $[m^2]$	MAPE [%]	Train $[s]$	Test $[\mu s]$
GZ	KRR	4.5 ± 0.3	58.0 ± 5.6	1.00 ± 0.00	453 ± 5	9.2 ± 5.0
	ELM	2.7 ± 0.2	24.6 ± 3.8	1.00 ± 0.01	10245 ± 216	5.2 ± 2.9
GM	KRR	4.8 ± 0.2	26.1 ± 7.5	1.00 ± 0.00	429 ± 4	9.1 ± 4.6
	ELM	3.3 ± 0.4	18.5 ± 3.6	1.00 ± 0.01	9383 ± 921	16.4 ± 5.3



(a) KRR model for GZ.
(b) KRR model for GM.
(c) ELM model for GZ.
(d) ELM model for GM.

Fig. 12. Scatter plot of DDSs of GZ and GM for the models in Table 8.

solute Percentage Error - MAPE) (Naser and Alavi, 2023) and one qualitative (the scatter plot actual versus predicted value) (Sainani, 2016). In terms of computational requirements, the performance is measured using time to build the model (Train Time) and time to make a prediction (Test Time). Since our surrogate will be leveraged in an optimizer, the most important computational metric is the Test Time.

4.2. Hull form optimization and physical plausibility

This section describes the optimization problem under examination and how we integrate the IMO Intact Stability Code into the MSHFO schema to achieve compliance by design. We formulate a multi-objective hull form optimization problem at high and low Fr subject to (s.t.) constraints on the geometry generated by the parameters $\boldsymbol{x}$, and properties of the actual geometry, namely the draft (T), the trim (θ), and the displacement (∇). To trade off a fast cruise speed versus a slow steaming case we selected the high and low Fr as $\mathrm{Fr}^{\mathrm{High}} = 0.282$ and $\mathrm{Fr}^{\mathrm{Low}} = 0.108$ respectively. For the constraints on T, θ, and ∇, note that, unlike the hard constraints informed by the IMO Intact Stability Code, these ones are tunable according to design preference. For simplicity, we constrained $\mathrm{T}(\boldsymbol{x})$ and $\theta(\boldsymbol{x})$ to match the conditions of the parent design, i.e., $\mathrm{T}^p = 10.8$ $[m]$ for the KCS at full-scale and $\theta^p = 0$ [°] for the even keel position. Whereas, we constrained $\nabla(\boldsymbol{x})$ to vary within $\pm 10\,\%$ of the conditions of the parent design, e.g., a lower bound of $0.9 \cdot \nabla^p$ and upper bound of $1.1 \cdot \nabla^p$, where $\nabla^p = 1.08 \times 10^4$ $[m^3]$ is the displacement of the KCS at full-scale. Consequently, the optimization problem becomes

$$\begin{aligned} \min_{\boldsymbol{x}^l \le \boldsymbol{x} \le \boldsymbol{x}^u} \quad & (\lambda)\tilde{R}_t(\boldsymbol{x}, \mathrm{Fr}^{\mathrm{High}}) + (1-\lambda)\tilde{R}_t(\boldsymbol{x}, \mathrm{Fr}^{\mathrm{Low}}), \\ \text{s.t.} \quad & \mathrm{T}(\boldsymbol{x}) = \mathrm{T}^p, \\ & \theta(\boldsymbol{x}) = \theta^p, \\ & 0.9\nabla^p \le \nabla(\boldsymbol{x}) \le 1.1\nabla^p. \end{aligned} \tag{2}$$

Furthermore, for the reasons described in Section 1, and to solve Lack (1), we propose constraining the optimization problem with the IMO

Table 9
IMO (SFM (1), SFM (1.1), SFM (1.2), SFM (1.3), SFM (1.4), and the IMO($\boldsymbol{x}$) of Eq. (1)) DDS based on the GZ and GM DDDs: Confusion Matrices (values are in %).

(a) SFM (1).

		DDS	
		Pass	Fail
Real	Pass	100	0
	Fail	0	0

(b) SFM (1.1).

		DDS	
		Pass	Fail
Real	Pass	100	0
	Fail	0	0

(c) SFM (1.2).

		DDS	
		Pass	Fail
Real	Pass	80	1
	Fail	0	19

(d) SFM (1.3).

		DDS	
		Pass	Fail
Real	Pass	89	0
	Fail	0	11

(e) SFM (1.4).

		DDS	
		Pass	Fail
Real	Pass	49	2
	Fail	1	43

(f) IMO($\boldsymbol{x}$) - Eq. (1).

		DDS	
		Pass	Fail
Real	Pass	49	4
	Fail	2	45

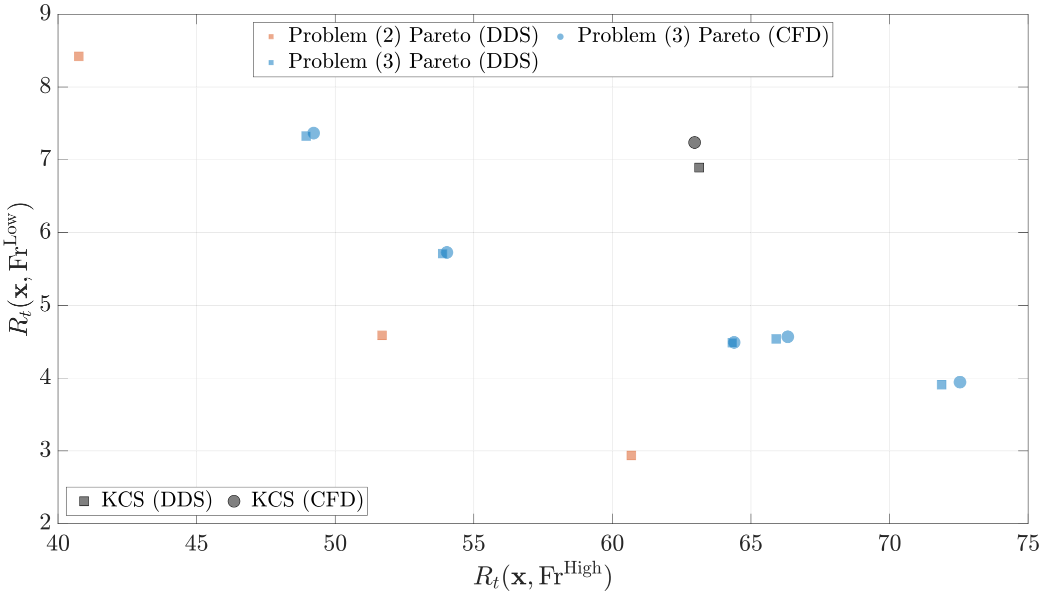


Fig. 13. Pareto Front for solving Problems (2) and (3) and the KCS (parent design).

Intact Stability Code. In particular, we add another constraint to Problem (2) to check that the geometry induced by $\boldsymbol{x}$ passes the IMO Intact Stability Code according to Eq. (1), namely

$$\begin{aligned} \min_{\boldsymbol{x}^l \le \boldsymbol{x} \le \boldsymbol{x}^u} \quad & (\lambda)\tilde{R}_t(\boldsymbol{x}, \mathrm{Fr}^{\mathrm{High}}) + (1-\lambda)\tilde{R}_t(\boldsymbol{x}, \mathrm{Fr}^{\mathrm{Low}}), \\ \text{s.t.} \quad & \mathrm{T}(\boldsymbol{x}) = \mathrm{T}^p, \\ & \theta(\boldsymbol{x}) = \theta^p, \\ & 0.9\nabla^p \le \nabla(\boldsymbol{x}) \le 1.1\nabla^p, \\ & \mathrm{IMO}(\boldsymbol{x}) = 1 \quad \text{see Eq. (1).} \end{aligned} \tag{3}$$

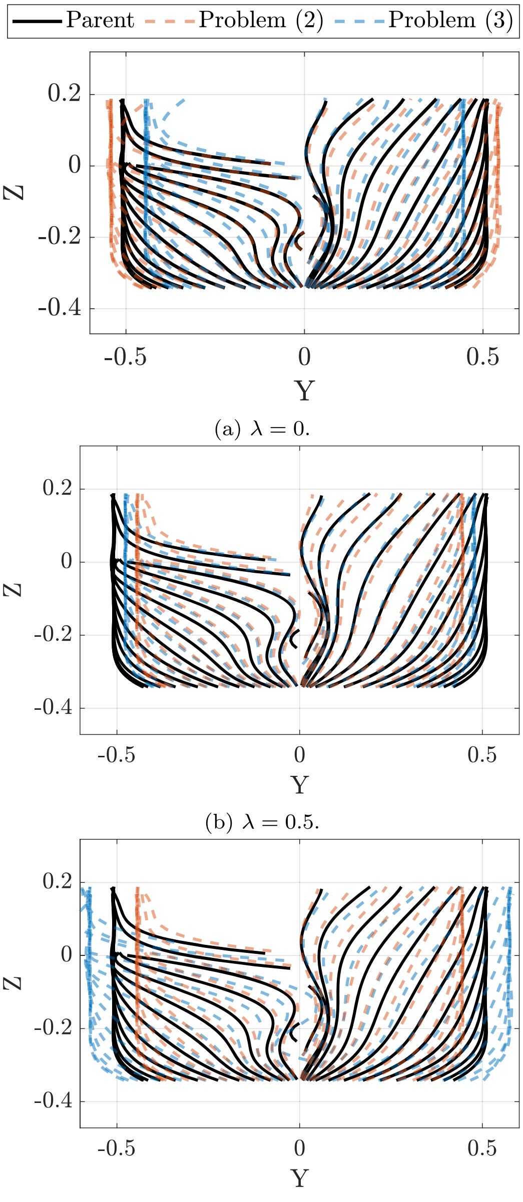
Problem (3) is now a non-linear and non-linearly constrained optimization problem and a number of approaches exist in literature to address it Wolpert and Macready (1997). We take advantage of some of the most popular gradient-based optimization algorithms (Hei et al., 2008). Table 6 outlines the optimization algorithms and their hyperparameter settings. Note that, for all of the approaches summarized in Table 6, we manually implemented a multi-start with 10 repetitions.

4.3. Computationally awareness and physically plausiblity for hull form optimization

This section addresses how we use DSDR to address Lack (2) of the current approach to MSHFO and, for the first time in the literature, how we blend the solutions to address both Lack (1) and Lack (2) simultaneously.

Regarding Lack (2), we still need to describe how to validate our approach to using PCA or Neural Networks to reduce the complexity of the design space (namely, DSDR) and how to validate the reconstruction. The most straightforward approach for DSDR is to use PCA (Diez and Serani, 2024).

PCA assumes that the data resides in a low-dimensional informative space that has been transformed into a higher-dimensional space. Basically, PCA fits an d-dimensional ellipsoid to the data, with each ellipsoid axis representing a new component. The longer the axis, the greater the data variance along that dimension, making that component more significant due to its higher variability. Conversely, components with lower variance are less informative. However, it is worth mentioning that this method is simplistic considering linear combinations of the original design variables. On the one hand, this means that PCA has a very low



(a) $\lambda = 0$.

(b) $\lambda = 0.5$.

(c) $\lambda = 1.0$.

Fig. 14. Comparison between KCS (parent) design in black and optimized (Problems (2) and (3)) designs.

computational cost, which is ideal for the problem at hand. On the other hand, PCA has been shown to be unsuitable for complex data in other applications (Salem and Hussein, 2019).

Another approach to DSDR is to use Neural Networks (Liu et al., 2022b). This approach tries to map the input data into a lower dimensional space, which can maintain the information to reconstruct the original input data (auto-encoder) (Schmidhuber, 2015). The advantage of Neural Networks is that they can also find non-linear relations that are not findable with PCA. By learning a compressed data representation, the Neural Network effectively reduces dimensionality while preserving essential features. However, despite its efficacy, this approach requires careful tuning of a number of hyperparameters, e.g., the number of hidden layers, the number of nodes in each layer, and the choice of activation function, to ensure optimal performance. This can result in a large computational burden during both the training phase and a more significant computational overhead than PCA during the test time (Serani and Diez, 2024).

Table 10
Average Number of Function Calls and Running Time to solve Problems (2) and (3).

Problem	Optimization Algorithm	Calls $[-]$	Time $[s]$
Problem (2)	IP AS	3006 ± 786 1218 ± 198	164 ± 47 153 ± 24
Problem (3)	IP AS	4471 ± 434 1893 ± 1108	223 ± 16 224 ± 128

Table 11
DSDR reconstruction error (in terms of MAE, MSE, and MAPE) and the resulting dimension (DIM) of the new low dimensional subspace using PCA and NN algorithms. We perform the dimensionality reduction using two datasets: (i) the entire dataset (i.e., DSDR_{STD}) and (ii) just the subset of the dataset that satisfies the IMO Intact Stability Code (i.e., DSDR_{IMO}).

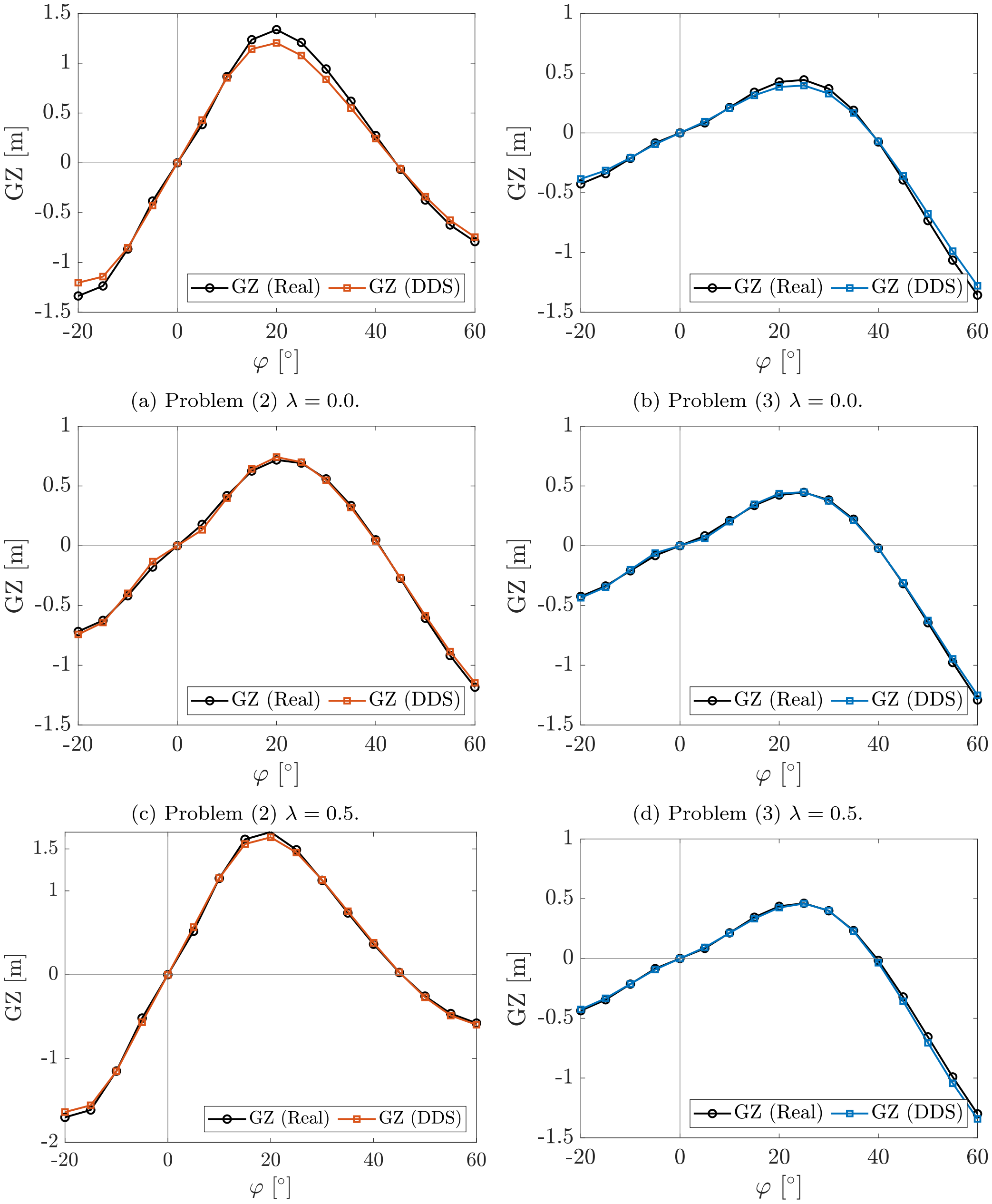
Dataset	DIM	DSDR	Reconstruction Error		
		Algorithm	MAE $[\times 10^{-2} N]$	MSE $[\times 10^{-2} N^2]$	MAPE [%]
DSDR_{STD}	7	PCA NN	22.15 ± 0.03 21.56 ± 0.04	10.50 ± 0.01 10.48 ± 0.00	0.70 ± 0.03 0.67 ± 0.04
DSDR_{IMO}	4	PCA NN	20.45 ± 0.04 20.32 ± 0.05	10.90 ± 0.02 10.88 ± 0.01	0.50 ± 0.04 0.46 ± 0.05

To perform the DSDR, using PCA or Neural Networks, we first randomly sample 1000 geometries from the design space and generate a realistic sample set of the design space $\boldsymbol{x}$ (Diez and Serani, 2021). We then leverage the approach of PCA or Neural Networks on $\boldsymbol{x}$, to reduce the dimensionality of the parametrization from $\boldsymbol{x} = \mathbb{R}^d \rightarrow \boldsymbol{x}' = \mathbb{R}^b$. Where $\boldsymbol{x}'$ is the projection of $\boldsymbol{x}$ into a lower-dimensional representation, i.e., $b \leq d$. When we perform the DSDR, we have to tune the hyperparameters of PCA or Neural Networks to perform the projection from $\boldsymbol{x} \rightarrow \boldsymbol{x}'$. For PCA, the hyperparameter is simply the percentage of variance in the $\boldsymbol{x}$ we want to preserve in $\boldsymbol{x}'$ (%var) which we set to 99 %. For Neural Networks, we use a single hidden layer with the sigmoid activation function and have to tune the number of neurons in the hidden layer (h_l) which we set to match the dimensionality of the data projected by the PCA. Note that, for Neural Networks, we also have to select the optimizer, in our case ADAM (Kingma and Ba, 2014), the learning rate l_r, in our case 0.001, and the number of epochs, in our case 1000. Note that the DSDR is fully unsupervised, so there is no risk of overfitting.

Once we have performed the DSDR, for the reasons described in Section 3, we have to validate that the representation in $\mathbb{R}^b$ can be retrieved and well-represented in $\mathbb{R}^d$. To perform this check is quite simple. For PCA, we reconstruct the data using the covariance matrix's eigenvectors to project the lower-dimensional data back into the original design space. For Neural Networks, the reconstruction comes for free from the auto-encoders. Then we check the reconstruction error, which is the difference between the original data in $\mathbb{R}^d$ and the data projected first into $\mathbb{R}^b$ and then back to $\mathbb{R}^d$. We measure this error using the same three quantitative metrics as before: the MAE, MSE, and MAPE.

5. Results

In this section, we will present the result of applying the methods described in Section 4 to the problem described in Section 3 using the data described in this very same section. The results are organized as follows. First, we report the performance of the developed DDSs to predict R_t, GZ, and GM. Subsequently, to address Lack (1), we report the results of optimization problem (3), with and without the stability constraints informed by IMO($\boldsymbol{x}$) - Eq. (1). Lastly, to address Lack (2), we report the results of our optimization problem with DSDR and show that we can simultaneously address Lack (1) and Lack (2) to further improve the computational complexity of optimization.



(a) Problem (2) $\lambda = 0.0$.

(b) Problem (3) $\lambda = 0.0$.

(c) Problem (2) $\lambda = 0.5$.

(d) Problem (3) $\lambda = 0.5$.

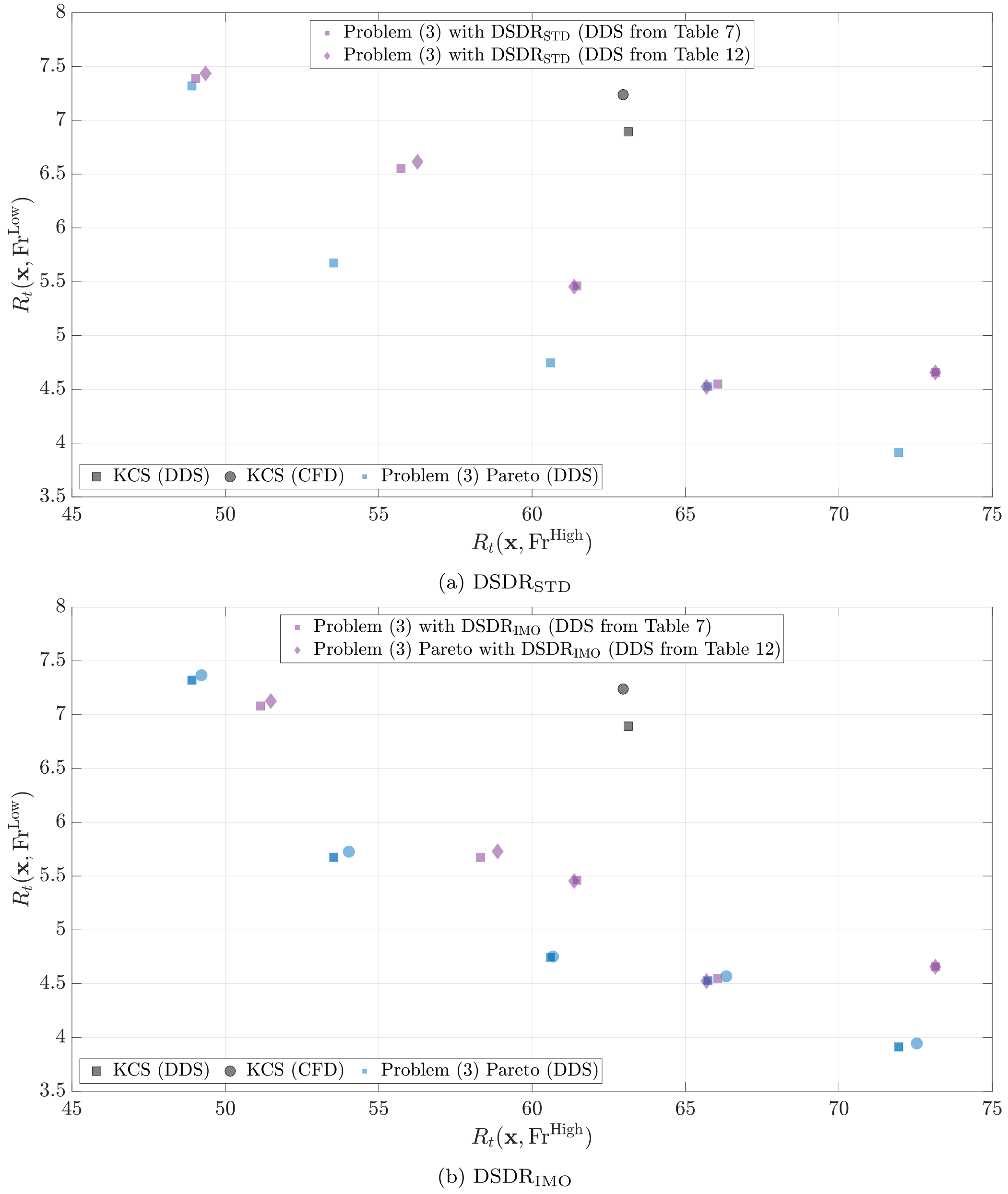
(e) Problem (2) $\lambda = 1.0$.

(f) Problem (3) $\lambda = 1.0$.

Fig. 15. Comparison between DDS and Panel Code Stability for Problems (2) and (3) designs.

Table 12
Surrogate performance (in terms of accuracy - MAE, MSE, and MAPE - and time - Train Time and Test Time) for the different learning algorithms to build DDS for R_t when we use all the inputs, or a subset of the input induced by the DSDR (with NN - i.e., the best performing DSDR method according to Table 11) for both the DSDR$_{\text{STD}}$ and DSDR$_{\text{IMO}}$ cases.

Case	DIM	Learning	Accuracy			Time	
		Algorithm	MAE [N]	MSE [N^2]	MAPE [%]	Train [s]	Test [μs]
No DSDR	Original	KRR ELM	1.04 ± 0.07 1.59 ± 0.20	1.32 ± 0.64 2.02 ± 0.83	1.76 ± 0.09 4.09 ± 1.15	322 ± 3 10095 ± 105	12.6 ± 5.6 4.6 ± 2.7
DSDR$_{\text{STD}}$	7	KRR ELM	0.99 ± 0.14 1.45 ± 0.27	0.99 ± 0.53 0.99 ± 0.23	1.36 ± 0.11 3.38 ± 0.08	205 ± 7 99987 ± 113	11.3 ± 3.8 4.5 ± 3.0
DSDR$_{\text{IMO}}$	4	KRR ELM	0.98 ± 0.14 1.45 ± 0.36	0.99 ± 0.36 0.99 ± 0.21	1.38 ± 0.09 3.41 ± 0.13	207 ± 6 99827 ± 103	11.1 ± 1.8 4.2 ± 3.1



(a) DSDR$_{\text{STD}}$

(b) DSDR$_{\text{IMO}}$

Fig. 16. Results of solving Problem (3) with DSDR$_{\text{STD}}$ and DSDR$_{\text{IMO}}$. Gray marks: the KCS (parent design) both using the DDS (i.e., Table 7) with a square and the CFD with a circle. Blue square marks: the results of Fig. 13 for Problem (3) (i.e., no DSDR). Purple square marks: solutions of Problem (3) with DSDR and DDS trained on the original input dimension but fed with the input reconstructed from the smaller dimensional space. Purple diamonds marks: solutions of Problem (3) with DSDR and DDS trained on the smaller input dimension induced by the DSDR.

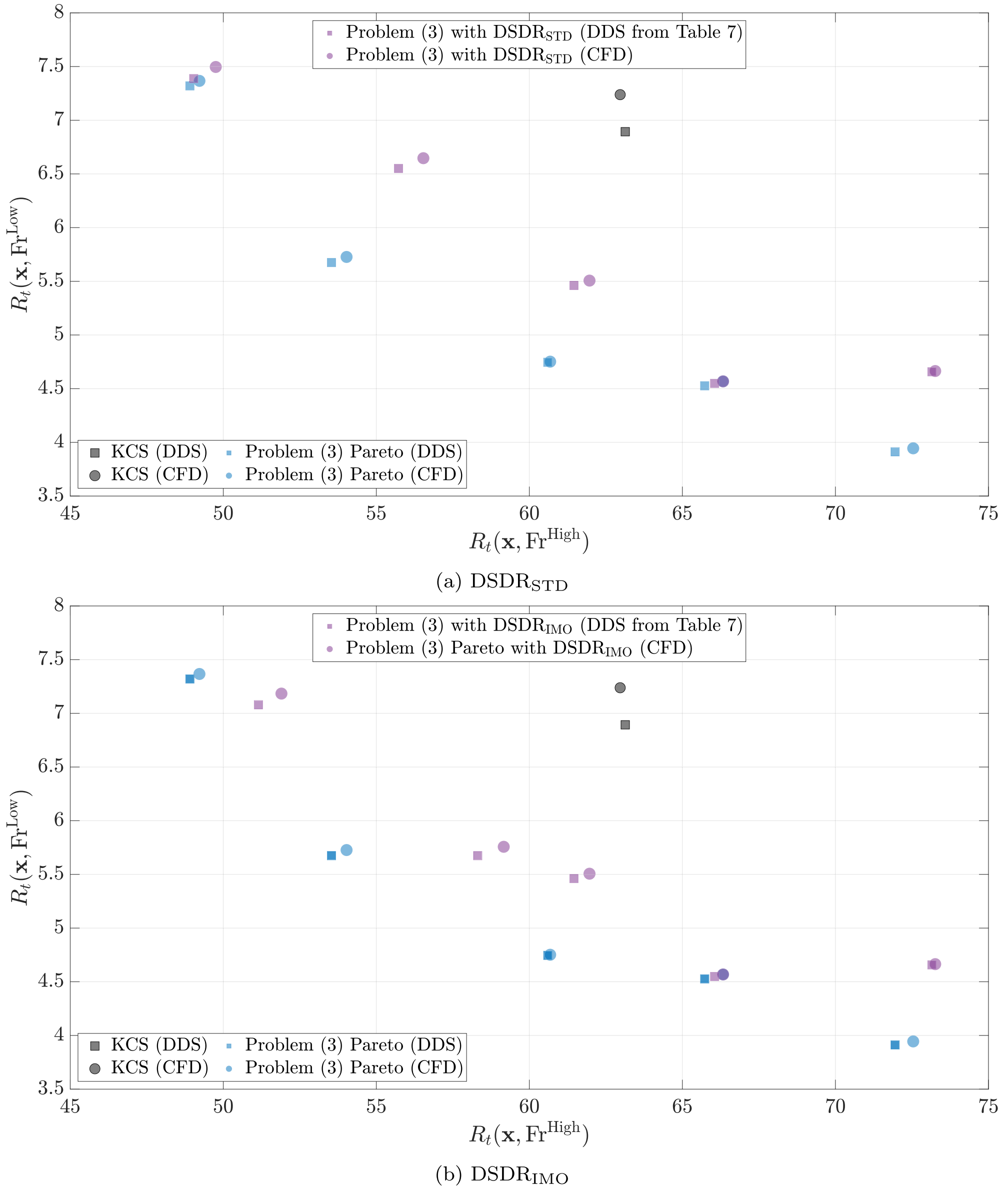
Table 7 reports the performance of the DDS of R_t (in terms of accuracy - MAE, MSE, and MAPE - and time - Train Time and Test Time) using the learning algorithms described in Section 4.1 (KRR and ELM). Fig. 11 reports the associated scatter plots.

From Table 7 it can be observed that KRR outperforms ELM in terms of MAE and MSE accuracy (visually represented in Fig. 11). Moreover, both KRR and ELM have a test time in fractions of milliseconds, which makes them well suited for use in MSHFO (3).

For what concerns the IMO constraint, we developed DDSs (according to Section 4.1) for GZ and GM as each of the constraint criteria - SFM (1), SFM (1.1), SFM (1.2), SFM (1.3), and SFM (1.4) - can be directly related to these quantities. Table 8 reports the results for the GZ and GM DDSs analogously to the ones of Table 7 for the DDS of R_t. Fig. 12, instead, is the equivalent of Fig. 11. From Table 8 it can be observed that ELM outperform KRR in terms of MAE and MSE accuracy (visually represented in Fig. 12) for both GM and GZ and both KRR and ELM have a test time in fractions of milliseconds which makes them well suited for use in MSHFO (3).

Since the constraint criteria (SFM (1), SFM (1.1), SFM (1.2), SFM (1.3), SFM (1.4), and IMO($\boldsymbol{x}$) - Eq. (1)) are used to determine feasible geometries (e.g., “pass”/“fail” a particular hull depending on the criteria), we test the efficacy of the best DDSs in this “pass”/“fail” binary classification scenario (Naser and Alavi, 2023). In Table 9 we compare the “pass”/“fail” criteria informed by the Real method (panel code) and the best DDS using Confusion Matricies (Naser and Alavi, 2023). From Table 9 it can be observed that

- for SFM (1) and SFM (1.1) all of the hulls pass the criteria;
- for SFM (1.2), SFM (1.3), SFM (1.4), and the IMO($\boldsymbol{x}$) - Eq. (1) the DDS performs well (low false positives and false negatives).



(a) $\mathrm{DSDR}_{\mathrm{STD}}$

(b) $\mathrm{DSDR}_{\mathrm{IMO}}$

Fig. 17. Physical plausibility for solving Problem (3) with $\mathrm{DSDR}_{\mathrm{STD}}$ and $\mathrm{DSDR}_{\mathrm{IMO}}$. We report the same results of Fig. 16 but (i) we removed purple diamonds marks (i.e., solutions of Problem (3) with DSDR and DDS trained on the smaller input dimension induced by the DSDR) as there is no need to define different surrogates for different dimensionalities and (ii) we added purple circle marks for the CFD physical plausibility check for each of the purple square marks (i.e., solutions of Problem (3) with DSDR and DDS trained on the original input dimension but fed with the input reconstructed from the smaller dimensional space).

At this point, we have developed and statistically validated the different DDSs. The next step is to highlight the results of the multi-objective hull form optimization of the KCS vessel (parent design) according to Section 4.2. We first solved the multi-objective optimization excluding the stability constraints, e.g., solving Problem (2) and show that the current approach fails to meet the IMO stability guidelines. Subsequently, we solve the multi-objective optimization, including the stability constraints, e.g., solving Problem (3) and show that our approach achieves compliance by design. Fig. 13 shows the Pareto front for solving Problems (2) and (3) and the KCS (parent design). Table 10 shows the average number of function calls and running time for Problems (2) and (3). Fig. 14 reports the comparison between the KCS (parent) design and the optimized designs. Fig. 15 shows a comparison between DDS and Panel Code informed GZ curves for Problems (2) and (3) designs. Note that, since the geometries found by solving Problem (2) failed IMO($\boldsymbol{x}$) - Eq. (1), there is no need to validate them with CFD; however, the cost informed by the DDS for Problem (3) is checked with the CFD model (see Section 4.1) for physical plausibility.

From Fig. 13 it can be observed that without including the stability constraints, i.e., solving Problem (2), the DDS induces the optimizer to find geometries that significantly outperform (according to the DDS) the KCS (parent design). Moreover, when validating the geometries found by solving Problem (2) with IMO($\boldsymbol{x}$) - Eq. (1), we find that these geometries are in fact not feasible (see Figs. 14 and 15 for more details). Finally, by including the stability constraints, i.e., solving Problem (3), the DDS induces the optimizer to find geometries that perform worse than the ones found by solving Problem (2) but still outperform the KCS (parent design).

From Fig. 14 it can be observed that for $\lambda = 1$ the design found from solving Problem (2) extended further in the transverse direction than the parent design and the one found by solving Problem (3) (see Fig. 14(c)). We validate the stability of the designs in Fig. 14 by comparing the DDS and Panel Code Stability in Fig. 15. For $\lambda = 0$, $\lambda = 0.5$, and $\lambda = 1.0$ it can be observed that the GZ curves found by solving Problem (2) fail Eq. (1) because the angle of the maximum GZ, namely, $\varphi_{\max}$, fails the condition to occur at $\varphi_{\max} \geq 25°$ (see Fig. 15(a), (c), and (e)). The GZ curves found by solving Problem (3) are in line with the expectation of IMO($\boldsymbol{x}$) - Eq. (1) (see Fig. 15(b), (d), and (f)).

From Table 10 it can be observed that including IMO($\boldsymbol{x}$) - Eq. (1) in the optimization problem increases the computational complexity of optimization by approximately 30 %.

After addressing Lack (1), by showing the importance of including the IMO($\boldsymbol{x}$) of Eq. (1) in the optimization problem, we need to address Lack (2), namely, the computational cost of optimization. For the reasons described in Section 3 and according to the method described in Section 4.3 we test different DSDR algorithms (PCA and Neural Networks - NN) to reduce the computational complexity of solving Problem (3).

Table 11 reports the reconstruction error (in terms of MAE, MSE, and MAPE) and the resulting dimension (DIM) of the new low dimensional subspace using PCA and NN algorithms. We perform the dimensionality reduction using two datasets: (i) the standard one that uses the entire dataset (i.e., DSDR_{STD}) and (ii) our new proposal that used just the subset of the dataset that satisfies the IMO Intact Stability Code (i.e., DSDR_{IMO}). The case DSDR_{IMO} allows simultaneously addressing Lack (1) and Lack (2).

From Table 11 we can observe that

- both the DSDR_{STD} and DSDR_{IMO} cases lead to a reduced dimensionality search space that can be well reconstructed back to the original dimensionality (small reconstruction errors);
- the DSDR_{IMO} case leads to a smaller search space than the DSDR_{STD} case because there is less variance among the examples preserved in the dataset.

Now that we have established we can project the data into a reduced complexity space (e.g., $\mathbb{R}^7$ in the DSDR_{STD} case or $\mathbb{R}^4$ in the DSDR_{IMO} case) we want to test the impact of the dimensionality reduction on the ability to predict R_t. Table 12 reports the surrogate performance (in terms of accuracy - MAE, MSE, and MAPE - and time - Train Time and Test Time) for the different learning algorithms to build DDS for R_t when we use all the inputs, or a subset of the input induced by the DSDR (with NN - i.e., the best performing DSDR method according to Table 11) for both the DSDR_{STD} and DSDR_{IMO} cases. From Table 12 we can observe that the accuracy remains high with low MAE, MSE, and MAPE errors for the two different low dimensional search spaces. Notably, performance slightly improves, meaning that some dimensions in the original space are actually not useful.

The next step is to perform the actual numerical optimization, i.e., solve Problem (3), with the two different low dimensional search spaces (i.e., the DSDR_{STD} and the DSDR_{IMO}), to address Lack (2). Fig. 16 shows the results of solving Problem (3) with DSDR_{STD} and DSDR_{IMO}. In particular, Fig. 16 reports

- gray marks: the KCS (parent design) both using the DDS (with no DSDR, i.e., Table 7) with a square and the CFD with a circle;
- blue square marks: the results of Fig. 13 for Problem (3) (i.e., no DSDR);
- purple square marks: solutions of Problem (3) with DSDR and DDS trained on the original input dimension but fed with the input reconstructed from the smaller dimensional space;
- purple diamonds marks: solutions of Problem (3) with DSDR and DDS trained on the smaller input dimension induced by the DSDR.

From Fig. 16 we can observe the following

- using the original dimension of the lower dimension based DDS are approximately equal, showing that the reconstruction does not negatively impact the optimization;
- the two different low dimensional search spaces (DSDR_{STD} and DSDR_{IMO}) both find geometries that outperform the KCS (parent) design;
- the two different low dimensional search spaces (DSDR_{STD} and DSDR_{IMO}) do not outperform solving Problem (3) in the original dimensionality.

Table 13
Average number of Function Calls and Running Time for solving Problem (3) for the two different low dimensional search spaces (DSDR and DSDR + IMO) and Dimensionality (Dim.) of the problem and Optimization Algorithm. The difference from solving the problem in the original dimensionality (Table 10) is also reported.

Case	DIM	Optimization	Calls	Time [s]	Savings	
		Algorithm			Calls	Time [s]
No DSDR	Original	IP	4471 ± 434	223 ± 16	-	-
		AS	1893 ± 1108	224 ± 128		
DSDR_{STD}	7	IP	2808 ± 331	194 ± 24	−37.2 %	−13.0 %
		AS	875 ± 413	186 ± 82	−53.8 %	−17.0 %
DSDR_{IMO}	4	IP	2137 ± 332	172 ± 22	−52.2 %	−22.8 %
		AS	621 ± 221	134 ± 76	−66.2 %	−39.9 %

The results of solving Problem (3) with the two different low dimensional search spaces (DSDR_{STD} and DSDR_{IMO}) still need to be checked for physical plausibility (i.e., using the CFD model in Section 4.1). Fig. 17 shows the same results as Fig. 16 but

- we removed purple diamonds marks (i.e., solutions of Problem (3) with DSDR and DDS trained on the smaller input dimension induced by the DSDR) as there is no need to define different surrogates for different dimensionalities;
- we added purple circle marks for the CFD physical plausibility check for each of the purple square marks (i.e., solutions of Problem (3) with DSDR and DDS trained on the original input dimension but fed with the input reconstructed from the smaller dimensional space).

From Fig. 17 it is possible to observe that the CFD model validates the results of solving Problem (3) with the two different low dimensional search spaces (DSDR_{STD}, and with DSDR_{IMO}).

Finally, Table 13 reports the average number of function calls and running time for solving Problem (3) for the two different low dimensional search spaces (DSDR_{STD}, and with DSDR_{IMO}) and optimization algorithms. The time and call savings with respect to solving the problem without DSDR are also reported.

Fig. 18 provides a graphical representation of Table 13. From Table 13 and Fig. 18 it can be observed that the DSDR_{IMO} approach leads

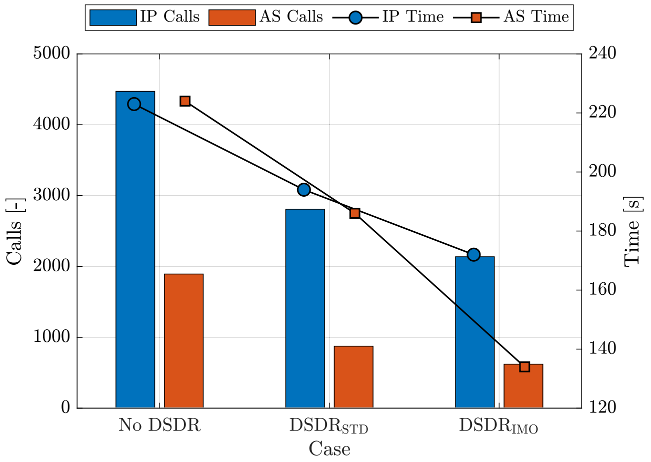


Fig. 18. Graphical representation of Table 13.

to significant computational time savings (20÷40 %) compared to solving Problem (3) in its original dimensionality. $\mathrm{DSDR}_{\mathrm{IMO}}$ also outperforms $\mathrm{DSDR}_{\mathrm{STD}}$ by approximately 10÷20 % in terms of computational time.

6. Conclusion

This study has presented a novel approach to hull form optimization that directly incorporates the IMO Second Generation Intact Stability Guidelines into the numerical optimization loop. By developing a data-driven surrogate model tailored to the stability constraint, we have ensured that all resulting designs inherently meet critical stability requirements without needing a-posteriori checks. Our results, obtained through the optimization of the KCS hull form, highlight two principal advantages of this method. The first one is that, unlike traditional optimization pipelines-where stability and other physical plausibility criteria are often verified after the fact, our integrated approach systematically excludes non-compliant designs. This marks a pivotal step toward more reliable, real-world-ready solutions. The second one is that, through a data-driven reduction in the search space, we successfully cut the computational cost by approximately 30 % while maintaining the quality of the optimized solutions. This reduction not only accelerates design workflows but also opens the door for more frequent or larger-scale optimization campaigns within the same computational budget.

Taken together, these contributions start to address the longstanding challenge of reconciling performance goals with practical, real-world constraints in a computationally efficient manner. By embedding a key regulatory requirement directly into the design process, our work paves the way for a new generation of design tools that balance hydrodynamic performance, physical plausibility, and computational efficiency.

Future work will focus on three main directions. First, extending the framework to incorporate additional regulatory and structural constraints can yield more robust designs. Second, integrating physics-informed, data-driven models may improve surrogate accuracy and generalizability. Third, expanding the methodology to diverse vessel types and operating conditions will broaden its applicability. Finally, we plan to integrate seakeeping considerations, such as comfort and operability in waves, into the optimization loop to capture a wider spectrum of performance metrics.

CRediT authorship contribution statement

Jake M. Walker: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Luca Oneto:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Andrea Coraddu:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Data availability

The data underlying this publication is available at the 4TU Data Repository.[2]

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The data generated in this study was created with the DelftBlue Supercomputer at Delft University of Technology (Technology, 2022).

References

Antony, J., 2014. Design of Experiments for Engineers and Scientists (Second Edition). Elsevier.

Bagazinski, N.J., Ahmed, F., 2023. ShipGen: a diffusion model for parametric ship hull generation with multiple objectives and constraints. J. Mar. Sci. Eng. 11 (12), 2215.

Bertram, V., Heimann, J., Hochkirch, K., 2024. Ship hull optimization - past, present, prospects. Ship Technol. Res. 71 (2), 82–91.

Birk, L., Harries, S., 2003. Optimistic: Optimization in Marine Design. Mensch & Buch Verlag.

Bottou, L., Curtis, F.E., Nocedal, J., 2018. Optimization methods for large-scale machine learning. SIAM Rev. 60 (2), 223–311.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Chen, T., Guestrin, C., 2016. XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Coppedè, A., Gaggero, S., Vernengo, G., Villa, D., 2019. Hydrodynamic shape optimization by high fidelity CFD solver and Gaussian process based response surface method. Appl. Ocean Res. 90, 101841.

Coppedé, A., Vernengo, G., Villa, D., 2018. A combined approach based on subdivision surface and free form deformation for smart ship hull form design and variation. Ship. Offsh. Struct. 13 (7), 769–778.

Coraddu, A., Gualeni, P., Villa, D., 2011. Investigation about wave profile effects on ship stability. In: Sustainable Maritime Transportation and Exploitation of Sea Resources: Proceedings of the 14th International Congress of the International Maritime Association of the Mediterranean.

Coraddu, A., Villa, D., Gualeni, P., 2012. Vulnerability assessment for the loss of stability in waves: some application cases for a further insight into the problem. In: International Conference on the Stability of Ships and Ocean Vehicles.

D'Agostino, D., Serani, A., Diez, A., 2020. Design-space assessment and dimensionality reduction: an off-line method for shape reparameterization in simulation-based optimization. Ocean Eng. 197, 106852.

Diez, M., Serani, A., 2021. From uncertainty quantification to shape optimization: cross-fertilization of methods for dimensionality reduction. In: Advances in Uncertainty Quantification and Optimization Under Uncertainty with Aerospace Applications.

Diez, M., Serani, A., 2024. Design-space dimensionality reduction in global optimization of functional surfaces: recent developments and way forward. Ship Technol. Res. 71 (2), 141–152.

Diez, M., Serani, A., Campana, E.F., Goren, O., Sarioz, K., Danisman, D.B., Grigoropoulos, G., Aloniati, E., Visonneau, M., Queutey, P., Stern, F., 2015. Multi-objective hydrodynamic optimization of the DTMB 5415 for resistance and seakeeping. In: International Conference on Fast Sea Transportation.

Feng, Y., el Moctar, O., Schellin, T.E., 2021. Parametric hull form optimization of containerships for minimum resistance in calm water and in waves. J. Mar. Sci. Appl. 20 (4), 670–693.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Guan, G., Wang, L., Geng, J., Zhuang, Z., Yang, Q., 2021. Parametric automatic optimal design of USV hull form with respect to wave resistance and seakeeping. Ocean Eng. 235, 109462.

Hei, L., Nocedal, J., Waltz, R.A., 2008. A numerical study of active-set and interior-point methods for bound constrained optimization. In: Modeling, Simulation and Optimization of Complex Processes.

Huang, G.B., Chen, L., Siew, C.K., 2006a. Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Transact. Neural Netw. 17 (4), 879–892.

Huang, G.B., Zhu, Q.Y., Siew, C.K., 2006b. Extreme learning machine: theory and applications. Neurocomputing 70 (1–3), 489–501.

Huang, J., Wei, Q., Liu, Y., 2023. Ship design optimization considering probabilistic compliance of decarbonization regulations. In: Advances in the Analysis and Design of Marine Structures.

Huang, L., Pena, B., Liu, Y., Anderlini, E., 2022. Machine learning in sustainable ship design and operation: a review. Ocean Eng. 266, 112907.

Keerthi, S.S., Lin, C.J., 2003. Asymptotic behaviors of support vector machines with gaussian kernel. Neural Comput. 15 (7), 1667–1689.

Khan, S., Goucher-Lambert, K., Kostas, K., Kaklis, P., 2023. ShipHullGAN: a generic parametric modeller for ship hull design using deep convolutional generative model. Comput. Method. Appl. Mech. Eng. 411, 116051.

Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Optimization. arXiv: 1412.6980.

La Rosa, A.G., Simpson, P., Zammit, R., 2024. Exploring the opportunities of generative artificial intelligence in concept ship design. In: International Marine Design Conference. Amsterdam, Netherlands.

Liu, X., Zhao, W., Wan, D., 2022a. Multi-fidelity Co-Kriging surrogate model for ship hull form optimization. Ocean Eng. 243, 110239.

Liu, Z., Wang, J., Wan, D., 2022b. Nonlinear design-space dimensionality reduction in shape optimization. In: International Ocean and Polar Engineering Conference.

Maissonneuve, J., 1993. Optimization tools for ship resistance and seakeeping problems. In: FAST '93.

[2] https://data.4tu.nl/datasets/83ddf541-3a59-4061-a54a-be3c035e6243

Maissonneuve, J., 1994. Software for the optimization of ships and marine structures resistance and seakeeping characteristics'. In: International Shipbuilding Conference, St. Petersburg, Russia.

Marlantes, K.E., Kim, S., Hurt, L.A., 2021. Implementation of the IMO second generation intact stability guidelines. J. Mar. Sci. Eng. 10 (1), 41.

Mesbahi, E., Atlar, M., 2000. Artificial neural networks: applications in marine design and modelling. In: Proceedings of the 1st International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT).

Miao, A., Wan, D., 2020. Hull form optimization based on an NM + CFD integrated method for KCS. Int. J. Comput. Method. 17 (10), 2050008.

Naser, M.Z., Alavi, A.H., 2023. Error metrics and performance fitness indicators for artificial intelligence and machine learning in engineering and sciences. Architect. Struct. Construct. 3 (4), 499–517.

Nowacki, H., 2019. On the history of ship design for the life cycle. In: A Holistic Approach to Ship Design: Volume 1: Optimisation of Ship Design and Operation for Life Cycle.

Oneto, L., 2020. Model Selection and Error Estimation in a Nutshell. Springer.

Organization, I.M., 2008. 2008 IS code - international code on intact stability, 2008. In: Statutory Documents - IMO Publications and Documents - International Codes - Part A - Mandatory Criteria - Chapter 2 - General Criteria - 2.3 Severe Wind and Rolling Criterion (Weather Criterion).

Organization, I.M., 2020. Finalization of second generation intact stability criteria-report of the drafting group on intact stability. In: International Maritime Organization Sub-Committee on Ship Design and Construction.

Papanikolaou, A., 2009. Risk-based ship design - methods, tools and applications. Springer Publishers.

Papanikolaou, A., Boulougouris, E., Erikstad, S., Harries, S., Kana, A.A., 2024. Ship design in the era of digital transition: a state-of-the-art report. In: International Marine Design Conference. Amsterdam, Netherlands.

Petacco, N., Gualeni, P., 2020. IMO Second generation intact stability criteria: general overview and focus on operational measures. J. Mar. Sci. Eng. 8 (7), 494.

Peters, W.S., Belenky, V.I., 2022. Second generation intact stability criteria: an overview. In: SNAME Maritime Convention.

Renaud, P., Sacher, M., Scolan, Y.M., 2022. Multi-objective hull form optimization of a SWATH configuration using surrogate models. Ocean Eng. 256, 111209.

Rosasco, L., De Vito, E., Caponnetto, A., Piana, M., Verri, A., 2004. Are loss functions all the same? Neural Comput. 16 (5), 1063–1076.

Sainani, K.L., 2016. The value of scatter plots. PM&R 8 (12), 1213–1217.

Salem, N., Hussein, S., 2019. Data dimensional reduction and principal components analysis. Proced. Comput. Sci. 163, 292–299.

Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117.

Serani, A., Diez, M., 2023. Parametric model embedding. Comput. Method. Appl. Mech. Eng. 404, 115776.

Serani, A., Diez, M., 2024. A survey on design-space dimensionality reduction methods for shape optimization. arXiv: 2405.13944.

Shawe-Taylor, J., Cristianini, N., 2004. Kernel Methods for Pattern Analysis. Cambridge university press.

Technology, D. U.o., 2022. DelftBlue Supercomputer (Phase 1). https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1.

Terziev, M., Tezdogan, T., Incecik, A., 2022. Scale effects and full-scale ship hydrodynamics: a review. Ocean Eng. 245, 110496.

Villa, D., Furcas, F., Pralits, J.O., Vernengo, G., Gaggero, S., 2021. An effective mesh deformation approach for hull shape design by optimization. J. Mar. Sci. Eng. 9 (10), 1107.

Villa, D., Gaggero, S., Coppede, A., Vernengo, G., 2020. Parametric hull shape variations by reduced order model based geometric transformation. Ocean Eng. 216, 107826.

Walker, J.M., Coraddu, A., Oneto, L., 2024a. A review on shape optimization of hulls and airfoils leveraging computational fluid dynamics data-Driven surrogate models. Ocean Eng. 312, 119263.

Walker, J.M., Coraddu, A., Oneto, L., 2024b. Data-driven models for Yacht hull resistance optimization: exploring geometric parameters beyond the boundaries of the delft systematic Yacht hull series. IEEE Access 12, 76102–76120.

Wang, P., Chen, Z., Feng, Y., 2021. Many-objective optimization for a deep-sea aquaculture vessel based on an improved RBF neural network surrogate model. J. Mar. Sci. Technol. 26 (2), 582–605.

Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. IEEE Trans. Evol. Comput. 1 (1), 67–82.

Yrjänäinen, A., Johnsen, T., Dæhlen, J.S., Kramer, H., Monden, R., 2019. Market conditions, mission requirements and operational profiles. In: A Holistic Approach to Ship Design: Volume 1: Optimisation of Ship Design and Operation for Life Cycle.

Zakerdoost, H., Ghassemi, H., 2023. Probabilistic surrogate-based optimization of ship hull-propulsor design with bi-level infill sampling technique. Ocean Eng. 286, 115614.

Zhang, H., Yu, Y., Zhang, Q., Yang, Y., Liu, H., Lin, Y., 2024a. A bidirectional collaborative method based on an improved artificial fish swarm algorithm for ship pipe and equipment layout design. Ocean Eng. 296, 117045.

Zhang, S., Tezdogan, T., Zhang, B., Lin, L., 2021. Research on the hull form optimization using the surrogate models. Eng. Applic. Comput. Fluid. Mech. 15 (1), 747–761.

Zhang, Y., Ma, N., Gu, X., Shi, Q., 2024b. Geometric space construction method combined of a spline-skinning based geometric variation method and PCA dimensionality reduction for ship hull form optimization. Ocean Eng. 302, 117604.