

Article

Application of Radial Basis Functions for Partially-Parametric Modeling and Principal Component Analysis for Faster Hydrodynamic Optimization of a Catamaran

Stefan Harries ^{*,†} and Sebastian Uharek ^{*,†}

FRIENDSHIP SYSTEMS AG, 14482 Potsdam, Germany

* Correspondence: harries@friendship-systems.com (S.H.); uharek@friendship-systems.com (S.U.)

† These authors contributed equally to this work.

Abstract: The paper shows the application of a flexible approach of partially-parametric modelling on the basis of radial basis functions (RBF) for the modification of an existing hull form (baseline). Different to other partially-parametric modelling approaches, RBF functions allow defining sources which lie on the baseline and targets which define the intended new shape. Sources and targets can be corresponding sets of points, curves and surfaces. They are used to derive a transformation field that subsequently modifies those parts of the geometry which shall be subjected to variation, making the approach intuitive and quick to set up. Since the RBF approach may potentially introduce quite a few degrees-of-freedom (DoF) a principal component analysis (PCA) is utilized to reduce the dimensionality of the design space. PCA allows the deliberate sacrifice of variability in order to define variations of interest with fewer variables, then being called principal parameters (prinPar). The aim of combining RBFs and PCA is to make simulation-driven design (SDD) easier and faster to use. Ideally, the turn-around time within which to achieve noticeable improvements should be 24 h, including the time needed to set up both the CAD model and the CFD simulation as well as to run a first optimisation campaign. An electric catamaran was chosen to illustrate the combined approach for a meaningful application case. Both a potential and a viscous solver were utilized, namely, SHIPFLOW XSPAN (SHF) and Neptuno (NEP), respectively. Rather than to compare the two codes in any detail the purpose of this was to study the efficacy of the proposed approach of combining RBF and PCA for solvers of different fidelity. All investigations were realized within CAESES, a versatile process integration and design optimisation environment (CAESES). It is shown that meaningful reductions of total resistance and, hence, improvements of energy efficiency can be realized within very few simulation runs. If a one-stop steepest descent is applied as a deterministic search strategy, for instance, some 10 to 12 CFD runs are needed to already identify better hulls, rendering turn-around times of a day of work and a night of number crunching a realistic option.

Keywords: computer aided design (CAD); partially-parametric modeling; radial basis functions (RBF); principal component analysis (PCA); simulation-driven design (SDD); hull form optimisation; computational fluid dynamics (CFD); potential flow code; viscous flow code; computer aided engineering (CAE); electric catamaran



Citation: Harries, S.; Uharek, S. Application of Radial Basis Functions for Partially-Parametric Modeling and Principal Component Analysis for Faster Hydrodynamic Optimization of a Catamaran. *J. Mar. Sci. Eng.* **2021**, *9*, 1069. <https://doi.org/10.3390/jmse9101069>

Academic Editor: Gregory Grigoropoulos

Received: 30 August 2021
Accepted: 9 September 2021
Published: 29 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the maritime industry, simulation-driven design (SDD) has become a widely accepted approach of developing functional surfaces such as ship hulls and appendages as well as turbochargers and engine component [1]. In simulation-driven design variants of a functional surface are analysed by means of simulations, often resource-intensive simulations of computational fluid dynamics (CFD) and finite element analysis (FEA). The simulation results are also used to come up with new variants which are then analysed. Instead of running the process manually, formal explorations and exploitations such as Designs-of-Experiment (DoE) and local or global search strategies, respectively, are put to

use. As a consequence, the simulations—more precisely, the results from the simulations for each of the variants investigated—drive the process, the aim being to improve selected objectives while complying to a set of constraints, see [1].

In SDD, the role of the design team is that of formulating the design task, setting up an efficient variation approach via a suitable computer aided design (CAD) model, selecting and configuring a meaningful simulation approach as well as monitoring and adjusting the process. Ideally, the busy work of changing geometry, pre-processing the simulations, transferring, extracting, post-processing and aggregating data, as well as of managing variants and results, is done by a process integration and design optimisation environment (PIDO). This frees the design team from cumbersome, repetitive and error-prone work and, naturally, allows and encourages the generation and analysis of very many variants. Simulation-driven design generally lives off the abundance and often the number of variants considered is one to two and sometimes even three orders of magnitude higher than in the more laborious traditional approach of manually generating a new variant and afterwards analysing its performance in a separate step.

Not surprisingly, many variants can be afforded the chance of identifying an exceptionally good design increase. Still, there are several good reasons why SDD approaches are being investigated and proposed that need as little time and as few simulations as possible:

- More and more high-fidelity simulations are used which require considerable computational resources, sometimes several hours or even days per variant.
- A number of operational points are considered concurrently which calls for more simulations per variant.
- A wider range of objectives from different disciplines are taken into account in parallel, again intensifying the computational burden.
- Time pressure to improve a product is continuously rising, rendering faster optimisation campaigns more attractive.
- Less expensive products developed with smaller budgets could and should also benefit from SDD.

Hence, one of the challenges is to reduce the number of simulation runs meaningfully, possibly by sacrificing some of the improvement potential for the sake of conducting a faster campaign. The actual number of variants to be analysed quickly scales up with the system's degrees-of-freedom (DoF), i.e., the number of free variables defining the functional surface or its modifications. Two recent approaches of counteracting this and of reducing turn-around times in SDD were discussed in [2], namely, parametric-adjoint simulation and dimensionality reduction. Both approaches are meant to improve the fluid-dynamic performance of a functional surface with not too many simulation runs, i.e., within less than 100 variations.

Mathematically speaking, parametric-adjoint simulation is a very elegant approach but requires the solution of the adjoint equations which, so far, only a few CFD solvers provide. The gradient of the objective with respect to the free variables is numerically approximated by concatenating the CAD model's design velocities and the CFD code's adjoint sensitivities, the latter resulting from the solution of the so-called adjoint equations. The complementing effort of determining the adjoint sensitivities is similar to solving the primal equation system, for instance the Reynolds-averaged Navier–Stokes equations (RANS), and independent of the number of free variables. Intrinsically, the approach is confined to a local search unless systematically repeated in various regions of the design space.

Dimensionality reduction is also mathematically elegant. It builds on a principal component analysis (PCA) of the design space spanned by the CAD variables, i.e., the geometric parameters that define the functional surface itself or its modifications. The original set of CAD variables is replaced by a (considerably) smaller set of principal parameters (or modes) which then capture the variability of the possible shape variations up to a user specified level. It is independent of the chosen simulation code(s) but room for improvement is literally speaking a bit smaller due to deliberately forgoing variability.

Another option of reducing the number of free variables and, hence, the number of necessary simulations, is to conduct an initial DoE from which the most important CAD variables are identified and kept for subsequent optimisation runs while the less important CAD variables are held constant. Quite a few variants need to be investigated before being able to select the subset of variables with which to continue. The potential of tangibly reducing simulations is therefore somewhat limited. Importantly, this approach should not be confused with the dimensionality reduction based on PCA. The principal parameters of the PCA form a new and orthonormal coordinate system, avoiding many inherent geometric dependencies between CAD variables.

Another challenge that design teams encounter when commencing with an SDD campaign is the time and expertise needed to produce a suitable model of variable geometry. In general, two approaches can be distinguished as elaborated in [3]: partially-parametric and fully-parametric modelling. In short, partially-parametric modelling builds on an existing geometry, from wherever it may originate, and only defines the modifications parametrically while fully-parametric modelling brings about a geometry from scratch by means of a self-sufficient hierarchical CAD model. A fully-parametric model usually is dedicated to a specific application, say a specific ship type, and supports different levels of modifications, from changing main dimensions down to fine-tuning specific regions, but it needs time and expertise to be developed. A partially-parametric model is often quicker to set up but rarely supports a wide range of modifications or the same level of sophistication as a fully-parametric model does.

A promising combination for an SDD campaign which brings about good optimisation results within reasonable effort, i.e., a campaign of high efficacy, appears to be a combination of partially-parametric modelling and dimensionality reduction. Ideally, the design team would see some improvements of their design within 24 h. In order to shed light on this, a comprehensive investigation of improving the energy efficiency of a ship hull by means of modifying its geometry via radial basis functions (RBF)—an intuitive and flexible approach of partially-parametric—and by applying a range of optimisation strategies, including dimensionality reduction via PCA, was undertaken. A fast catamaran was chosen as an illustrating application case. A thorough comparison is given between number of simulation runs and improvements achieved.

The paper first describes the design task and its deliberately chosen simplifications. It then explains the RBF approach along with the partially-parametric model realized with it. This is followed by a discussion of dimensionality reduction based on PCA. Both the RBF approach and the PCA were implemented in CAESES which was also utilized as the PIDO environment. Two different CFD codes were used for the simulations, a potential flow code (SHIPFLOW XPAN from FLOWTECH) and a RANS code (Neptuno from the Technical University Berlin), both of which were coupled to CAESES. The potential flow code is a non-linear free surface code with free sinkage and trim. It only needs a few minutes of CPU time per variant and was employed to cover the bulk of the investigations, i.e., the systematic testing and comparison of different optimisation approaches. The viscous code is a high-fidelity code that solves the RANS equations, taking into account the free surface and the cat's sinkage and trim for given weight and centre of gravity. It needs several hours per variant and was therefore employed to test and showcase the initial findings for substantially more expensive simulations. An overview of the two CFD codes and for the selected optimisation strategies is given as needed to appreciate the campaigns. This is followed by an elaboration of the results. The paper ends with conclusions and recommendations for future work.

2. Design Task And Simplifications

2.1. Design Task

A small catamaran ferry of 20 m length between perpendiculars at a design speed of 13 kn and a displacement mass of 25 t at 0.85 m draft shall be optimised for energy efficiency, see Table 1. The cat is meant to be solely battery powered and should serve up

to 50 passengers. The energy density of available batteries being substantially lower in comparison to the energy density of fossil fuels, the battery mass becomes critical for a given operational range. Consequently, highest energy efficiency turns out to be even more important for an electric ferry as it would be for a conventionally powered vessel.

Table 1. Main particulars.

length between perpendiculars	L_{PP}	20 m
beam demi-hull	B_{dH}	1.58 m
beam over all	B_{OA}	5.38 m
clearance between demi-hull centre planes	c_{dH}	3.8 m
design draft	T_0	0.85 m
displacement	∇	24.39 m ³
design speed	v_0	13 kn
Froude number	Fn	0.477

An existing hull shape, from hereon referred to as baseline, see Figure 1, was provided which featured a slender round-bilge demi-hull with a slightly submerged transom stern. The term baseline is used here to refer to the initial design, i.e., the parent hull from which variations are derived. Figure 2 gives an impression for various views, omitting the deck and superstructure. The blue demi-hull illustrates the cat's port side up to the deck while the green demi-hull shows the underwater portion of the cat's starboard side cut at design draft. The baseline's demi-hulls were symmetric with regard to both their centre planes and, naturally, the cat's mid plane. They neither featured any knuckle lines nor spray rails. No topological changes ought to be introduced during the optimisation campaign, but an asymmetry of the demi-hulls would be acceptable.

Each side of the demi-hull was modelled with just one single B-spline surface that featured a polyhedron of ten vertices in longitudinal direction (u-direction) and six vertices in vertical direction (v-direction). It should be noted that the B-spline surfaces were set up within CAESES which can also be used as system for free-form surface modelling. However, any other CAD system could have been used to establish the baseline. For the B-spline surfaces the degrees in u- and v-direction were five and three, respectively. Both the transom and the deck were modelled as ruled surfaces, following the B-spline surface's aft edge and upper edge, respectively.

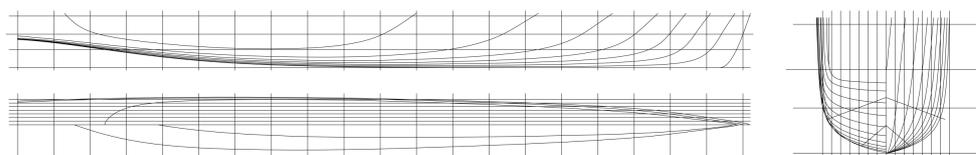


Figure 1. Linesplan baseline.

The Froude number at 20 m length and 13 kn is 0.4774 which, from a general point of view, falls into a rather unfavourable speed range. A resistance curve was computed for the baseline with both the potential flow code and with the RANS code checked for one speed, see Figure 3. It showed, not surprisingly, that the cat's total resistance steadily increases with speed. From the wave resistance coefficient it can be seen that the cat would indeed sail close to a resistance hump, the maximum wave resistance coefficient occurring at 13.5 kn. The bare hull's total resistance R_{T0} (both demi-hulls taken into account) as computed from SHIPFLOW amounted to 10.3 kN. All potential flow results were normalized with this value since any absolute values ought to be looked at with some caution. First of all, appendages and openings were not taken into account. Secondly, even though both potential flow and boundary layer theory was utilized, the form factor was simply set to zero. Thirdly, by definition the interactions for the free surface and viscous flow are not accounted for the potential flow analyses.

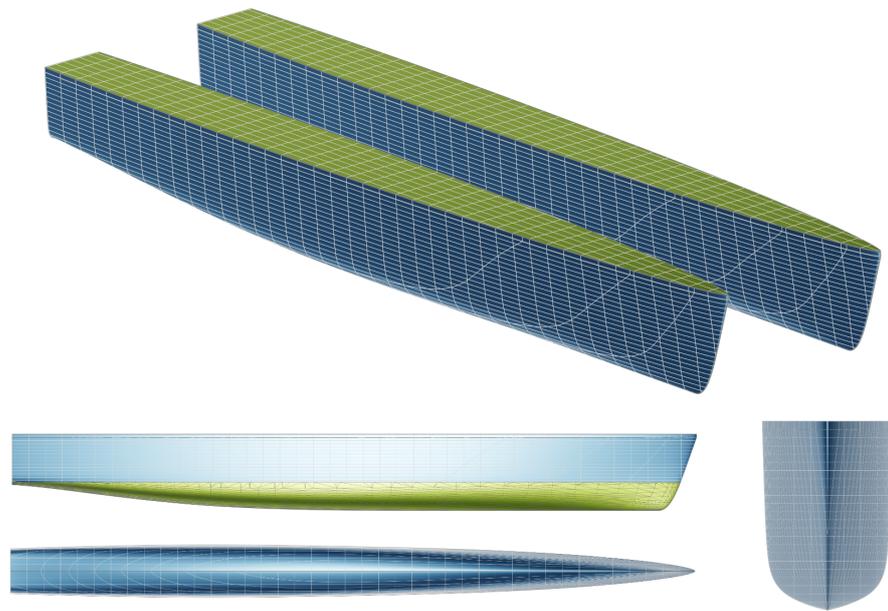


Figure 2. Baseline.

A pure variation of the distance between the demi-hulls revealed that the cat’s resistance would steadily fall with increasing clearance up to its upper bound, corresponding to a beam over all of 5.376 m. It was therefore decided to set the clearance to its maximum value and exclude it from the optimisation campaign afterwards. Apart from thus reducing the number of free CAD variables—which is always beneficial—the maximum clearance would also yield the highest stability.

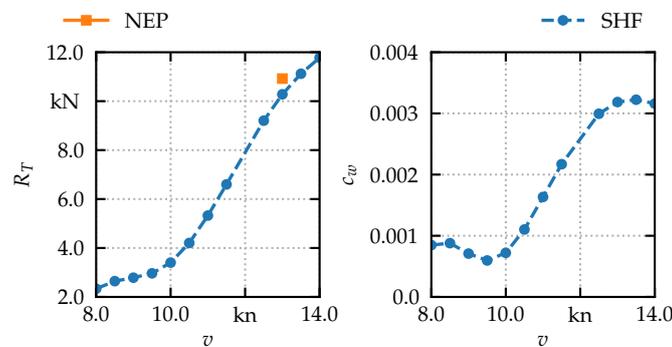


Figure 3. Resistance curve computed with SHIPFLOW XPAN (SHF) for the baseline and cross-checked with RANS solver Neptuno (NEP).

2.2. Simplifications

The distance from the keel K to the metacentre M, KM, was 7.58 m for the baseline and varied between 7.1 m and 7.8 m for the shapes investigated, keeping the clearance constant. This points towards a sufficiently stable vessel so that no constraints had to be monitored with regard to stability.

For a propulsion system a standard arrangement with shaft, bracket and propeller plus a spade rudder for each of the demi-hulls was chosen. As this was not subject to any changes during the optimisation campaign, it was assumed—for the sake of simplicity—that no unfavourable effects would be encountered for any of the anticipated variants. Consequently, the propulsive efficiency was assumed to stay constant so that the cat with least total resistance would also yield the design of highest energy efficiency. It

needs to be pointed out that this, naturally, neglects an important component of calm-water hydrodynamics which can be justified solely on the grounds of focusing on methodology and not on proposing a final design.

Additional hydrodynamic factors would also actually have to be considered, such as seakeeping and manoeuvring performance as well as possible resistance increase due to canal and shallow water effects, depending on the cat's final operational profile. It could be argued that the anticipated shape modifications would primarily influence calm-water performance and should only lead to minor effects in seakeeping etc. as main dimensions, including length, draft, beam and displacement, were all kept constant.

An upfront design-of-experiment (DoE) for two speeds (applying a Sobol sequence [4]), using the potential flow code, showed that an improvement of the hull at its design speed of 13 kn would not necessarily give an improvement also at a lower speed of interest, here 9 kn. The normalized resistance values for the different variants at both speeds are shown in Figure 4. The general tendency is that variants which perform well at 13 kn may come with slightly higher resistance values at 9 kn. For certain designs, however, this is not necessarily the case, indicating that there would be favourable compromises.

In addition, a similar DoE study (also applying a Sobol sequence) was run at 13 kn for the design draft of 0.85 m and at a higher draft of 0.95 m, see Figure 5. It shows that performance is correlated but that variants do not always yield improvements at both drafts. Both DoEs clearly indicate that a thorough optimisation for calm-water hydrodynamics would have to be multi-objective, including (at least) two speeds, possibly two drafts and the influence of the propulsion system. If a design team has to consider all these aspects, further complemented by non-hydrodynamic aspects, it becomes all the more clear that the number of variants to be investigated needs to be as small as possible when running expensive simulations.

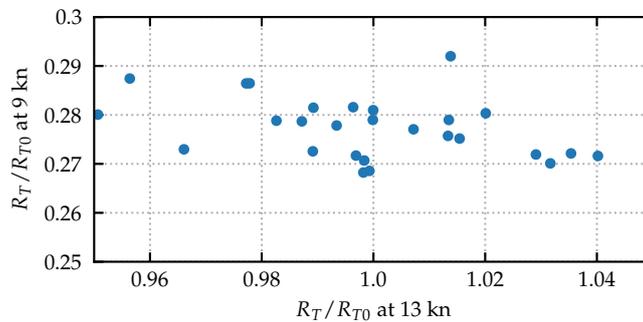


Figure 4. Non-dimensional resistance at design speed vs. non-dimensional resistance at lower speed.

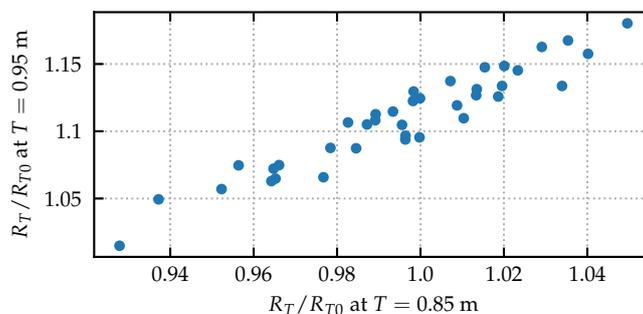


Figure 5. Non-dimensional resistance at design draft vs. non-dimensional resistance at higher draft.

3. Parametric Model

3.1. Partially-Parametric Model

With the aim of realizing first and meaningful improvements of a given hull form within a short turn-around time, ideally 24 h, a partially-parametric modelling approach is proposed. While fully-parametric modelling is more powerful and more precise, a partially-parametric model is faster and, generally speaking, a bit more intuitive to build, requiring less mathematical know-how, see [3].

A flexible partially-parametric modelling approach is based on radial basis functions (RBFs). The basic idea of the approach is to obtain a smooth transformation between a source and target geometry by using the RBFs to translate the control polygon of the surface, see Figure 6. While the present application features a rather simple hull shape, RBFs as implemented in CAESES are capable of handling complex geometries as well, e.g., ships with bulbous bows but also engine and turbo machinery components [5]. The implementation in CAESES is based on the work by [6] and explained in some detail in [7]. In general, radial basis functions (RBFs) are used in the context of scattered data interpolation. In particular, when utilizing RBFs for partially-parametric modeling a vector-valued space deformation

$$\vec{d}(\vec{x}) = \sum_j \vec{w}_j \cdot \phi_j(\vec{x}) + \vec{p}(\vec{x}) \tag{1}$$

is to be computed. It makes use of influence functions, here the triharmonic RBF

$$\phi_j(\vec{x}) = |(\vec{x} - \vec{c}_j)|^3 \tag{2}$$

featuring control points vector $\vec{c}(\vec{x})$ along with a triharmonic quadratic polynomial vector $\vec{p}(\vec{x})$. A set of weights vector $\vec{w}(\vec{x})$ has to be found to smoothly interpolate a number of user defined and, hence, known displacements. To this end, fixed points, so-called clamped supports and sources, as well as displaced points, so-called targets, are used, see Figure 7. The targets are made subject to variation, establishing a flexible parametric model. For each fixed point, lying either on a clamped support or on a source, a new position in space is known from the associated point on the topologically identical yet geometrically different target, determining the vector-valued space deformation.

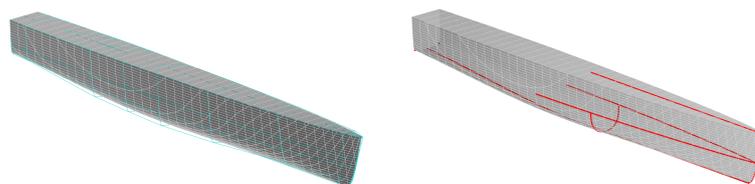


Figure 6. Polyhedron of the Brep and RBF source curves.

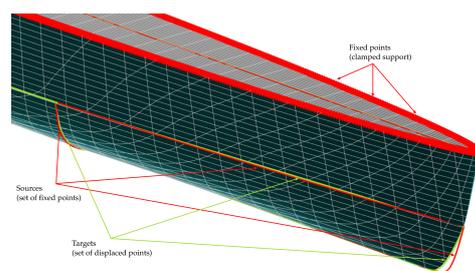


Figure 7. Fixed point (clamped support and sources) and displaced points (targets) for the forebody of the cat.

The advantage of this procedure is that it can be established within just a few hours of interactive work. Points, curves and, if of interest, surfaces are identified on the baseline which capture prominent features of the shape to be varied, keeping in mind a design team's background knowledge and intuition about which parts of a shape may positively influence the objective functions when modified. These entities, here points and curves, form the sources of the RBF model, see Figure 8.

Matching entities are then introduced, i.e., a point for a point and a curve for a curve etc., which form the targets. The RBFs are computed, here within CAESES, such that a vector-valued space deformation is established that gives the desired displacements for all points in space. For each point of interest it can thus be determined to which new position it shall be transferred, see Figure 9.

The space deformation is calculated from a set of control points, typically a few thousand, which comprise fixed points, i.e., boundary points not to be moved, and points for which the initial and the new positions are known from the sources and targets, respectively. For the cat a total of 14 CAD variables were used to control the target functions, 13 of which are related to RBF morphing and one from a subsequent Lackenby transformation (namely, ΔXCB), see Table 2. Their corresponding bounds are listed in Table 3. The Lackenby transformation, see [8], was utilized to maintain the displacement and actively control the longitudinal centre of buoyancy.

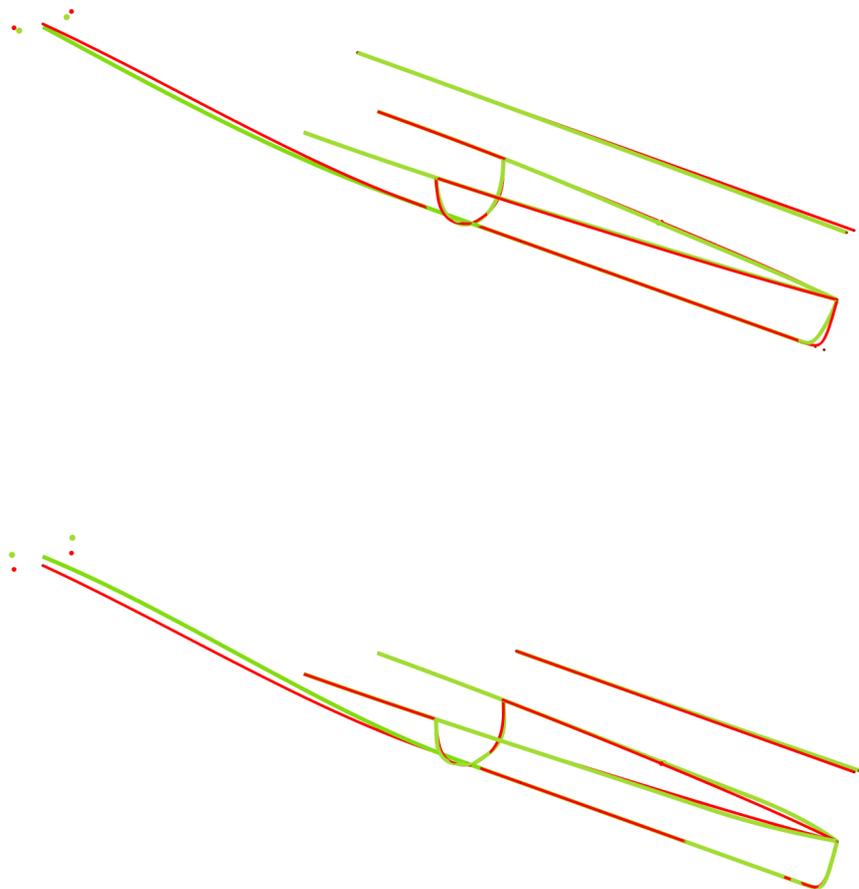


Figure 8. Matching source (red) and target (green) curves corresponding to minimum and maximum values of free variables in CAD space.

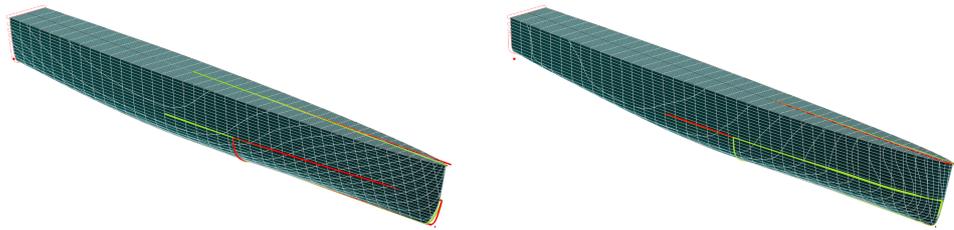


Figure 9. Resulting geometry of the cat's demi-hull for target curves with minimum and maximum values of free variables.

Table 2. Design variables in CAD space used for optimisation.

Name	Explanation
iE	half-angle of the design waterline at forward perpendicular (FP)
touchFor	change of the transverse position of the design waterline (DWL) midway between the FP and the maximum section
bowAsymmetry	change of the transverse position of the stem contour at the deck forward of the FP
stemContour	weight for one of the vertices of the NURBS curve defining the stem contour, influencing the stem's sharpness
keepCPCfor	position along the forward part of the center plane curve (CPC) at which an asymmetry of the demi-hull, if present, starts
modTranAngleCPC	modification of the angle of the CPC at the transom stern (within the demi-hull's center plane)
moveParallel	change of the aft position of the parallel part of CPC
startAsymmetry	longitudinal position at which an asymmetry of the demi-hull, if present, starts
deltaAreaMid	change of the sectional area at midship
deltaDeadrise	change of the deadrise angle of the section at midship
deltaY	transversal shift of the rounded part of the transom, causing changes in the transom's beam
deltaZ	vertical shift of the rounded part of the transom, causing changes of the deadrise
deltaTransom	change of the submergence of the transom stern (in the demi-hull's center plane)
deltaXCB	change of the longitudinal center of buoyancy

3.2. Dimensionality Reduction

A flexible parametric model may easily feature quite a few free variables, i.e., parameters that a design team feels could be advantageous to change. Since the effort of evaluating the objective function scales up quickly with the number of free variables, i.e., the systems' degree-of-freedom (DoF), it readily appears to be mandatory to keep the DoF as low as possible. Parametric modelling already is key to small numbers of free variables, often between 10 and 50 for hull shapes, propellers and appendages [1]. If a representative evaluation of the objective function takes just one hour per variant than a turn-around time of 24 h can only be achieved if very few variants need to be investigated, say 20 when also taking into account the set-up time for modelling and simulation. This is naturally based on the assumption that the people involved are familiar with the design task in general and with all tools in particular. N.B. One can always argue that more computational power can be brought in so as to compute the objective function more quickly. While this is true, it should be kept in mind that if computational resources are high more objective functions are usually taken into consideration, see Section 1, and that, more often than not, a human

desire quickly grows to utilize higher-fidelity tools, for instance, to increase accuracy as will also be addressed below.

Table 3. Bounds of design variables used for optimisation (CAD space).

	Name	Lower Bound	Baseline Value	Upper Bound
1	iE	5	7.5	10
2	touchFor	−0.05	0	0.05
3	bowAsymmetry	−0.15	0	0.1
4	stemContour	−0.05	1.75	2.5
5	keepCPCfor	0.55	0.7	0.85
6	modTranAngleCPC	−3	0	2
7	moveParallel	−0.75	0	1
8	startAsymmetry	8	10	12
9	deltaAreaMid	−0.008	0	0.008
10	deltaDeadrise	−10	0	10
11	deltaY	−0.1	0	0.03
12	deltaZ	−0.02	0	0.1
13	deltaTransom	−0.05	0	0.15
14	deltaXCB	−0.0025	0	0.00125

A powerful and rather new approach of reducing the dimensionality of the design space is to undertake a principal component analysis (PCA) as first introduced in naval architecture by Diez et al. [9]. The idea is to identify how the parameters of a CAD model selected as free variables, here the partially-parametric RBF model, are actually coupled and to replace these CAD variables by a new and smaller set of free variables that are independent of each other albeit less easy to interpret. These new and independent free variables are called principal parameters (or super parameters). They result from the eigendecomposition of the covariance matrix for point data derived by a large sample of geometric variants, as described in [2,9–11].

To this end a DoE, here a Sobol sequence, of purely geometric variants is produced from the CAD variables, say 1000 variants for a set of 10 to 15 free variables. On each of the variants, an equal number of evenly distributed points are generated whose geometric positions in cartesian space necessarily vary but which are topologically identical. Here some 6000 points per cat’s demi-hull were used. For a sample of 1000 variants this forms a collection of 60,000 points that serves as input to the PCA. The main benefit of employing a PCA lies in deliberately sacrificing some of the variability of the CAD model for the sake of subsequently working with a (considerably) smaller set of principal parameters [10]. Typically, only the first and most important principal parameters are put to use while the less contributing modes are cut off. For the cat’s demi-hull the variability achieved by accumulating the first principal parameters is given in Table 4

Table 4. Influence of the first seven principal parameters (prinPar) and the PCA model’s variability in comparison to the variability of the CAD model.

	Principal Parameter	Variability	Accumulation
1	prinPar1	37.93%	37.93%
2	prinPar2	27.66%	65.59%
3	prinPar3	14.20%	79.79%
4	prinPar4	6.06%	85.85%
5	prinPar5	4.80%	90.65%
6	prinPar6	3.34%	93.99%
7	prinPar7	2.25%	96.24%

It needs to be pointed out that the principal parameters from the PCA are less easy to be interpreted than any of the CAD variables. Mathematically speaking, they represent the Eigenvalues associated to the Eigenvectors that span the orthonormal PCA space. Practically, they bring together the free variables defined in CAD space in a new form.

For the investigation it was thus decided to run the optimisation campaigns with the first seven principal parameters which together yield a pretty high variability while cutting the number of free variables to half the number of CAD variables. Figure 10 illustrates the influence that the first six principal parameters have on the hull shape. From the colour distributions it can be nicely seen that indeed different regions are addressed by each of the principal parameters. For further details and elaborations of the PCA, see [2,11].

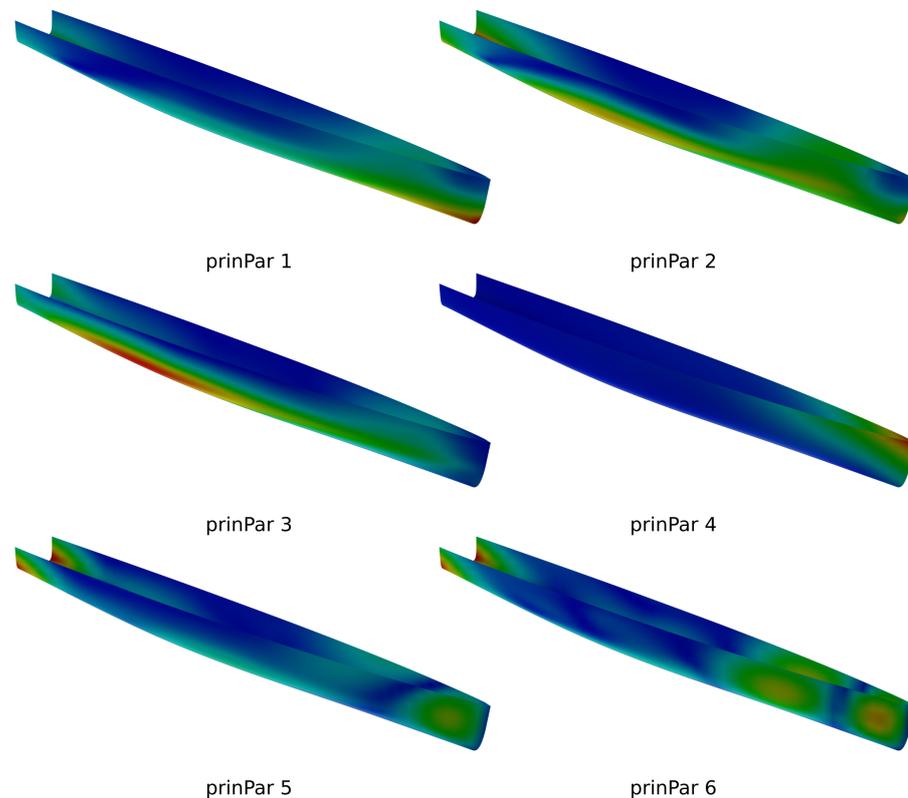


Figure 10. Influence of first six principal parameters (prinPar) on the hull shape.

Figure 11 shows the correlations between selected design variables in CAD space (bowAsymmetry, keepCPCfor, iE and deltaTransom) and the first four principal parameters obtained from the dimensionality reduction (prinPar1, prinPar2, prinPar3 and prinPar4). It is obvious that in some cases a strong coupling exist, such as, for example between the parameter bowAsymmetry and prinPar4. In other cases, the coupling between principal parameters and CAD variables is a bit less pronounced, e.g., keepCPCfor and prinPar1 or even less obvious, e.g., bowAsymmetry and prinPar2. In addition, one principal parameter usually influences several CAD parameters, such as, for example keepCPCfor and stemContour in relationship to prinPar1. This behaviour is expected and exactly this correlation between CAD variables is used by the dimensionality reduction to reduce the degrees-of-freedom of the system.

In order to check by how much the design space is reduced hydrodynamically when employing a dimensionality reduction in the process, a design space exploration using the Sobol algorithm is performed. The results are presented and discussed in Section 5.1.

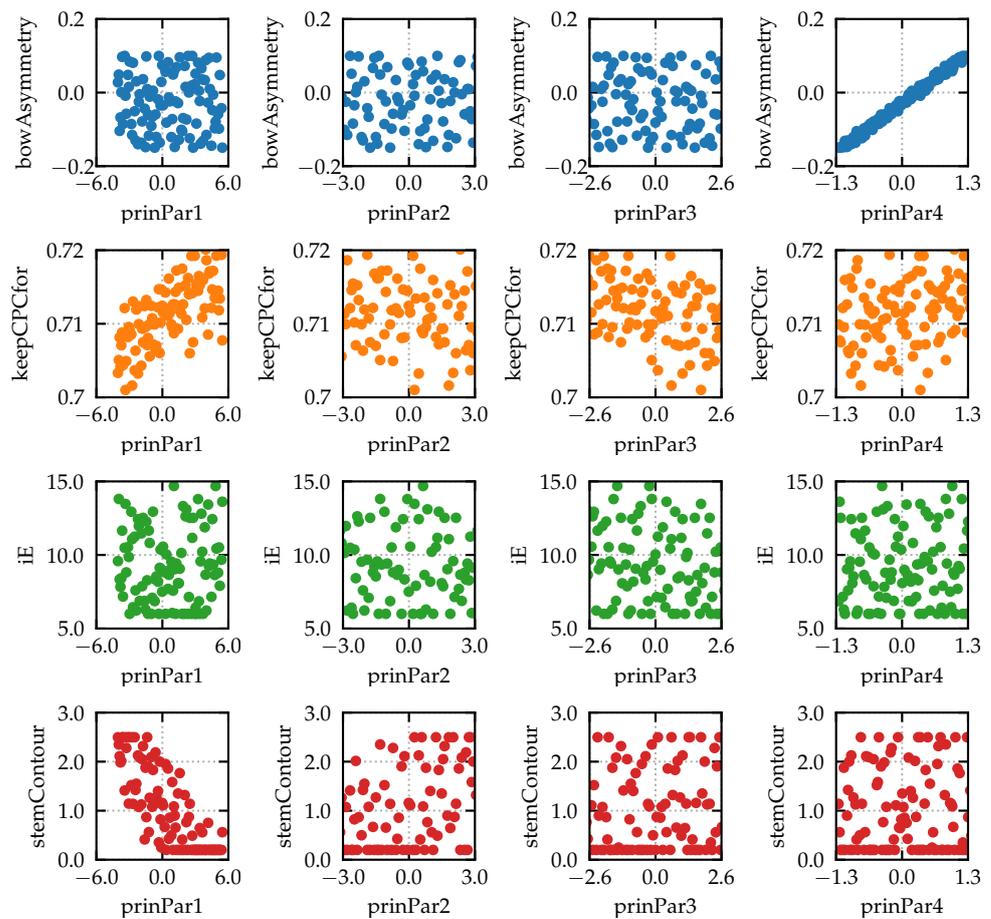


Figure 11. Correlations between several selected CAD variables and the first four principal parameters resulting from the PCA.

4. CFD Codes and Numerical Set-Ups

For the evaluation of the towing resistance of the cat, which serves as an objective function for the optimisation study, two numerical codes are used. The first code is SHIPFLOW XPAN/XBOUND, a non-linear potential flow code including sinkage and trim, complemented with a boundary layer simulation, developed by FLOWTECH. The advantage of using a potential flow code in numerical optimisation studies is the rather short computation time, which allows the evaluation of the objective function in just a few minutes per variant. Especially for studies where the geometry variation is limited to the bow area, potential flow results provide at least a good ranking of the designs.

For a few years, however, it has been rather common to use more sophisticated methods such as a RANS solver to compute the viscous flow and the free surface around the hull so as to predict the absolute values with higher accuracy for the price of a significantly higher computation time. In the present study, selected computations have been performed with the RANS code Neptuno, which has proven to yield very accurate results in diverse marine applications, see for example [12].

4.1. Potential Flow Computation

In the SHIPFLOW package (SHF), the non-linear potential flow code XPAN is shipped together with the panel generator XMESH and a module for thin turbulent boundary layer analysis, XBOUND, which solves the momentum equation along the streamlines traced from the potential flow solution. All these programs are tightly integrated into CAESSES and their set-up is rather straightforward. The geometry is exported as offsets at each

station. The free surface mesh extends $1L_{PP}$ upstream of the bow, $2L_{PP}$ downstream of the stern and $1.5L_{PP}$ to the side.

For the present application case, some fine-tuning was required in order to obtain converging solutions for all variants. For example, the panel mesh size between the two demi-hulls was adjusted, since some wave breaking between the demi-hulls would occur, leading to divergence unless numerically suppressed. Figure 12 shows the final panel mesh on the free surface and on the hull. During the course of the computation, sinkage and trim is taken into account and at the end, the results are imported back into CAESSES for visualization.

It should be noted that the cat's hydrodynamics approaches the limits of a potential flow analysis, with possible wave breaking and modifications of the transom. However, XPAN provides a very fast way to evaluate the objective function and since the main goal of the present work is to study the capabilities of the RBF and PCA approach, these restrictions were deliberately accepted.

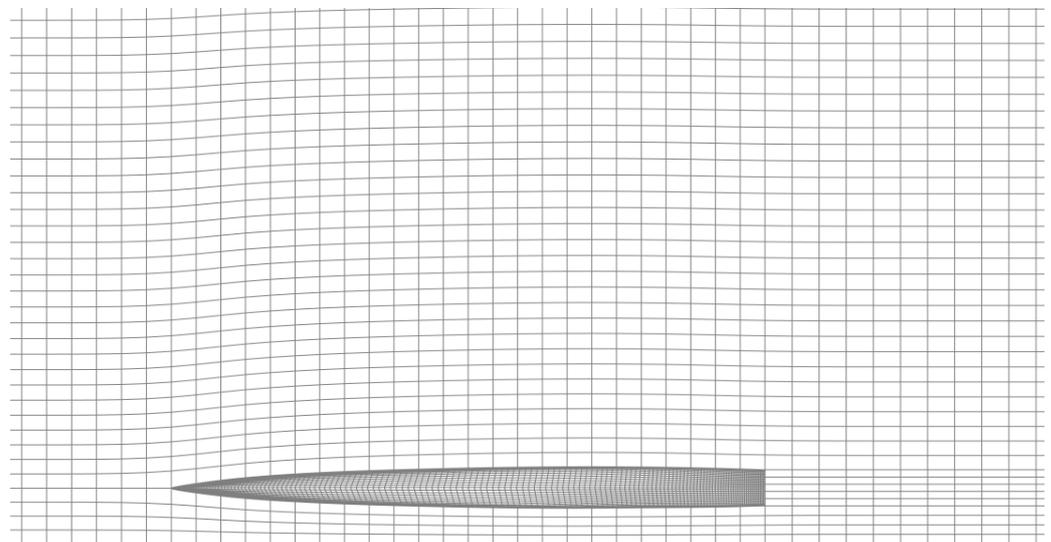


Figure 12. Close-up of the panel mesh on the hull and free surface in the vicinity of the hull.

4.2. Viscous Flow Computation

The RANS code used as a high-fidelity solver is Neptuno (NEP). Neptuno is an in-house CFD code developed by Prof. Cura [13,14]. It solves the RANS equation on a multi-block structured grid using the finite volume method. The pressure–velocity coupling is done using the SIMPLE algorithm from Patankar [15]. The turbulence is modelled using the standard two equation $k-\omega$ model from Wilcox [16]. The free surface is captured using a two phase level set method [17,18]. The code has been validated in several workshops, see [19–21].

The numerical grid required for the computations is built using the commercial meshing software GridPro. The main advantage of GridPro is the ability to separate topology and geometry. This allows setting up the initial topology (block structure) of the grid prior to the optimisation study and simply re-run the meshing process for each variant. The resulting grids are all of high quality and due to the identical block structure, the dependency of the objective function on the numerical grid is reduced. By using a symmetry boundary condition at the centre plane between the two demi-hulls, only one side of the domain needed to be meshed. The boundaries are located $1L_{PP}$ upstream, $2L_{PP}$ downstream, $1L_{PP}$ beside and $1L_{PP}$ below the ship. The selected boundary conditions were slip-wall (symmetry) at the centre plane and starboard boundary while at the inlet, top and bottom side of the domain the velocity is prescribed. At the outlet the hydrostatic pressure is given.

Since the code has been widely and successfully validated at model scale, it was decided to carry out all computations at a fictive model scale of $\lambda = 4$, which would correspond to a model length of 5 m and a Reynolds number of $Re = 1.67 \cdot 10^7$. For the optimisation and comparison with XPAN, the values are extrapolated to full scale using the standard ITTC procedure, see [22].

Only for some geometric variants, e.g., the ones with an asymmetric bow, a slight modification of the topology is required. Figure 13 shows a slice of the resulting mesh in the bow region for two rather different geometries cut at the same longitudinal position. As can be seen, GridPro allows to change the grid in the vicinity of the hull, leaving the far-field grid untouched. In addition, the topology of the grid is practically identical, so it can be assumed, that changes in the grid have a rather small influence when evaluating the resistance.

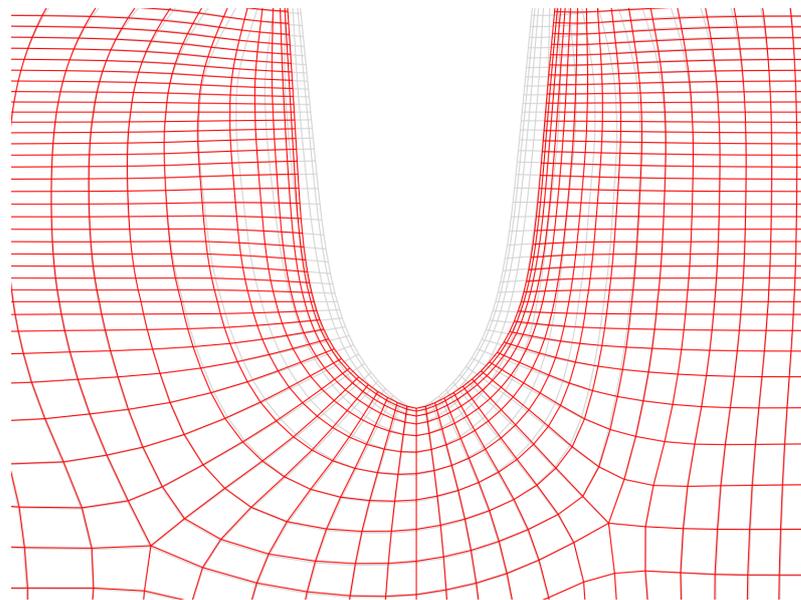


Figure 13. Mesh in the bow region for the baseline (grey) and a modified hull shape (red).

For the cat simulation with free surface, a constant number of 8000 time steps are computed. The non-dimensional time step is chosen at 0.008 and one SIMPLE iteration per timestep is performed. A numerical beach for the damping of waves in the far field is used, for details see [23]. Sinkage and trim is taken into account by a stepwise movement of the free surface relative to the ship. This is performed at the time steps 1500, 2500, 3500 and 4500. As an example, Figure 14 shows the time traces of the earth fixed longitudinal component of the hydrodynamic force (F_z) acting on the hull, the iterative history of sinkage (ζ_O) and trim angle (θ) along with the residuals. As can be seen, even after the force stabilized, the residuals keep dropping and thus the solution converges. Only very minor changes occur after 8000 steps, so in order to save time, only 8000 iterations are performed during the optimisation study.

In order to assess the dependence of the computed hydrodynamic forces on the grid resolution, a systematic variation has been performed with cell counts ranging from just 460,000 to 3.7 million cells. Figure 15 shows the force components resulting from the integration of pressure (F_P), and shear stresses (F_F) as well as the total force (F_T). In addition, Table 5 shows the results of the convergence study, with r_g being the convergence radius and u the uncertainty of the solution, as proposed by the ITTC [24]. As can be seen a monotonic convergence is achieved for each individual component and the quality of the prediction is quite satisfactory. It should be noted that the grid convergence study is carried out without considering sinkage and trim, since for each grid different trim angles would have resulted and thus it would have been impossible to run an uncertainty study for the resistance.

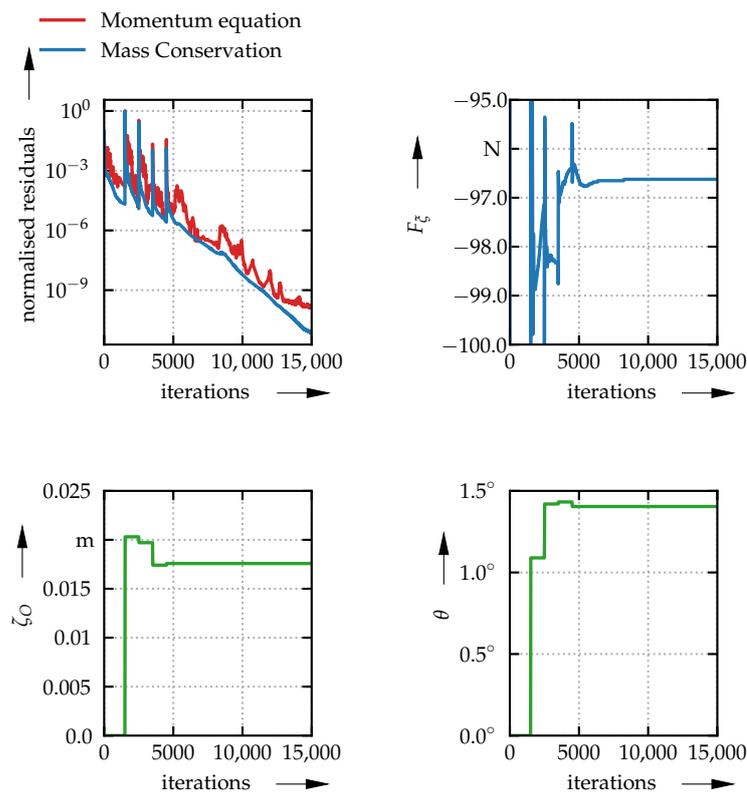


Figure 14. Convergence history, time history of earth fixed longitudinal force for one demi-hull, sinkage and trim for straight-ahead motion with stepwise adjustment of sinkage and trim.

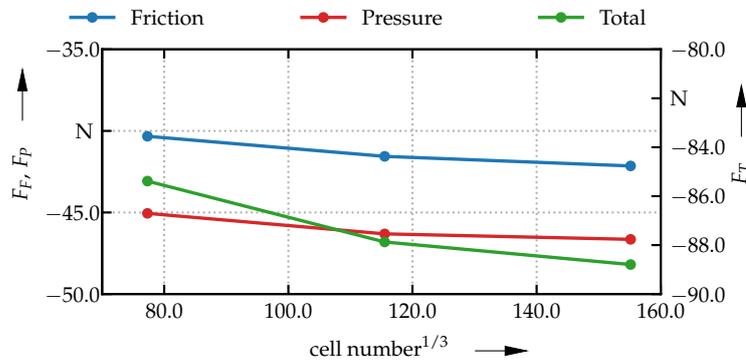


Figure 15. Grid dependency for RANS simulations.

Table 5. Results of grid convergence study for three grids with 3.7 million (S_1), 1.5 million (S_2) and 460,000 cells (S_3) for one demi-hull.

	S_3 Coarse	S_2 Medium	S_1 Fine	r_g	$u\%$
F_F	−40.331 N	−41.560 N	−42.147 N	0.477	1.57
F_P	−45.054 N	−46.312 N	−46.647 N	0.267	0.33
F_T	−85.385 N	−87.872 N	−88.794 N	0.371	0.76

Since the change in total resistance between the fine and coarse grid is only 3.85%, but the computation time differs by one order of magnitude, namely, 8 h (480 min) for

S_1 in comparison to 48 min for S_3 on a state-of-the-art workstation using eight cores, all computations for the optimisation study are carried out on the coarse grid. Figure 16 shows a comparison of the free surface elevations for the coarse and fine grid. One should keep in mind that parallel computations could be carried out on a cluster for the finer grid. However, that would accelerate the computations for all grids, naturally.

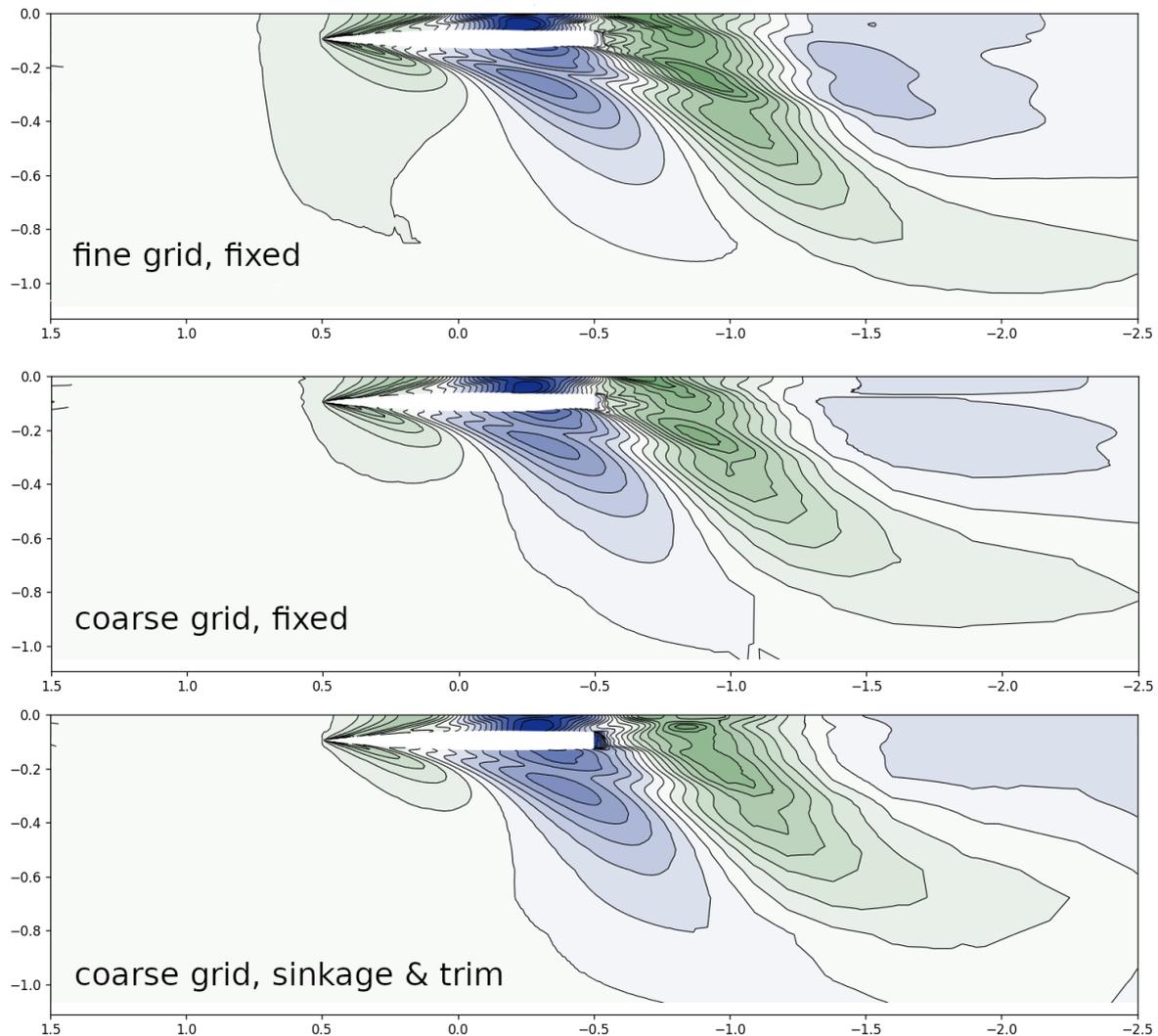


Figure 16. Contour plot of the free surface on the fine and coarse grid without sinkage and trim as well as on the coarse grid with sinkage and trim.

4.3. Exploration and Optimisation Algorithms

For the exploration of the design space, a Sobol algorithm is used which is based on a quasi-random yet deterministic sequence [4]. The values of the free variables are set such that the sampling points iteratively fill up the design space as evenly as possible, an additional sampling point always reducing the region of the design space that has been least populated so far. There are other exploration algorithms available in CAESES, for example the Latin-Hypercube sampling (LHS), which is made available through the DAKOTA optimisation package. However, an advantage of using a Sobol sequence is, that contrary to the LHS, the number of samples in the design space can be readily extended, whereas with the LHS a complete new design set would have to be generated. Furthermore, a Sobol sequence can be easily repeated with the same sampling points, provided that the bounds of the free variables have not been modified.

The exploitations, i.e., various optimisation runs, have been undertaken on the basis of three purely deterministic strategies, namely, a T-Search, a Nelder–Mead simplex and a one-stop steepest descent. This was done with the aim of fast turn-around times, i.e., improvements within 24 h. All three strategies perform local searches and are typically employed when quickly trying to understand the potential for improvement of a baseline without introducing major modifications and/or when fine-tuning a design.

The T-Search, i.e., tangent search by Hillary [25], first evaluates the baseline and then starts with a small positive perturbation of the first design variable. If this readily gives an improvement the second design variable undergoes a small perturbation. Otherwise a small negative perturbation is introduced to the first design variable. Setting out from the best design so far the second design variable is slightly changed, first in positive direction than, if no improvement was found, in negative direction. This is repeated until all free variables have been modified at least once, establishing a local search pattern. Using both the best design found during such a local search and its current starting point (e.g., the baseline at the onset) a global search direction can be determined and a global step is then undertaken, hoping to bring home a more substantial improvement. If that was successful further global steps can follow until a suitable starting point for a new local search pattern has been found. Without going into the strategy's details any further, it can be appreciated that the algorithm scales up linearly with the number of free variables. In the best case, all positive increments would readily give improvements during the local search. In the worst case, all free variables would have to be perturbed twice, once into positive and once into negative direction. Therefore, in practical situations an objective function needs to be computed once for the baseline and then anything between the number of free variables (best case) and twice that number (worst case) before a global step can be taken.

The Nelder–Mead simplex [26] also is a local search method which flips through the design space with a simplex, i.e., the simplest possible polytope in any given space. In one-dimensional space a simplex is a line segment while in two-dimensional space it is a triangle. In three-dimensional space it is a tetrahedron while, by abstraction, a simplex represents a cell with $n+1$ corners in an n -dimensional space. The search, in short, always takes the corner with the least favourable objective value and flips it through the centroid of the n remaining corners that offered better objective values. This step is called a reflection. Depending on the changes found for the objective an extension may take place, trying to benefit further. In general, these flips are repeated (with some additional tweaks such as contracting and shrinking the simplex), slowly advancing the simplex through the design space. As can be readily appreciated the first simplex requires $n+1$ evaluations of the objective, one evaluation for the baseline and n evaluations for a small perturbation of each free variable (or any independent set of n combinations). Each additional evaluation than brings about an improvement (or information how to contract or shrink the current simplex).

The one-stop steepest descent takes a similar approach as the Nelder–Mead simplex for the first $n+1$ evaluations. Basically, a gradient of the objective function is numerically determined by adding a small delta to each free variable while keeping all other free variables constant, establishing an approximation through forward differencing. As soon as the gradient is known, a line search is undertaken into the direction of anticipated improvement. In case the objective of the new variant is better than the baseline's a second step is taken into gradient direction, e.g., by doubling the step size. Otherwise, the first step apparently has been too far and a bisection (or any other subdivision) of the distance between the baseline and the first new variant takes place. This is repeated a few times, say three to five times, until the local search is stopped, establishing a "one-stop" strategy. Alternatively, a conjugate gradient can be utilized to advance further through the design space. In principle, the one-stop steepest descent allows setting the maximum number of evaluations of the objective function upfront, namely, one evaluation for the baseline, n evaluations for the gradient approximation plus three to five evaluations during the one-dimensional line search. This means that as soon as the time needed to simulate a single variant, say the baseline, and the time available before improvements have to be

realized are known, it is possible to select the maximum number of free variables—either CAD variables or principal parameters—to take into consideration. For instance, if a single variant needs one hour to simulate for hydrodynamics, a one-stop steepest descent with seven principal parameters would take eight hours before the gradient is known (one hour for the baseline and seven hours for each perturbation). With another four variants evaluated during the line search that sums up to 12 h. That would then leave another 12 h to do both the pre- and post-processing. With some eight to ten hours of pre-processing, including parametric modelling (three to four hours), undertaking the PCA (five to six hours) and coming up with a good CFD set-up (several hours while the PCA is running), that would leave about four hours to complete the job within a single day of turn-around time.

5. Results

The following section presents the results obtained from exploration and optimisation runs performed with SHIPFLOW and Neptuno, respectively, and analyses the possible gains from the PCA.

5.1. Optimization Studies Using Potential Flow Code

In order to check the process improvement, which can be achieved by using a PCA compared to the traditional approach using design variables in CAD space, a high number of simulations were performed using XPAN.

Figure 17 shows the total resistance made non-dimensional by the resistance of the baseline from two Sobol runs. The first one was performed in CAD space with 200 samples (green) and the second one using the PCA with 100 samples (orange). As can be seen, the design space covered by both runs is fairly similar. There are three design variables which are not really correlating to any of the chosen principal parameters and which are not covered when running the optimisation with a PCA, namely, keepCPCfor, startAsymmetry and deltaXCB. Apparently, the same geometry variation is also achievable with a suitable combination of other design variables, making these CAD variables rather superfluous. The longitudinal shift of the centre of buoyancy, which was achieved using a Lackenby transformation, could for example also be achieved by moving the parallel midship section (moveParallel). Most importantly, however, it can be seen that the range of the objective function is very similar for both the CAD space and the PCA space, the maximum being approximately 1.05 and the minimum being around 0.9. This means, that the optimisation potential was not reduced by reducing the dimensionality of the problem by a factor of two and evaluating 100 instead of 200 variants.

In the next step, six optimisation runs are performed starting from the baseline design. Figure 18 shows the various histories and associated improvement in non-dimensional resistance over the iteration count. All three deterministic optimisation algorithms are applied for both the traditional CAD approach and the PCA. The first thing to observe is, that the T-Search finds the highest improvement, namely, a reduction in total resistance by 10.5% for the CAD based approach and 9.9% when using the PCA. However, the first significant improvement when using the T-Search with the PCA is achieved after 16 steps compared to 24 steps when using the traditional approach, which leads to a reduction in computational time of 33%. The best result in terms of computational time was achieved by the one-stop steepest descent, which lead to a reduction of 8.5% in 11 steps when using the PCA compared to 6.8% in 16 steps for the optimisation in CAD space. The performance of the Nelder–Mead simplex was worse under both aspects for the present application; however, the PCA still performed better than the optimisation in CAD space.

5.2. Validation of the Results Using the Viscous Flow Code

In order to validate the results obtained by the potential flow optimisation, computations with the viscous flow code Neptuno are run and the results are compared to the ones obtained with XPAN. It is a popular practice when performing optimisations with a lot of

variants to perform the optimisation with a potential flow solver, which is cheap by means of computational costs, and only check the final result using the viscous flow solver.

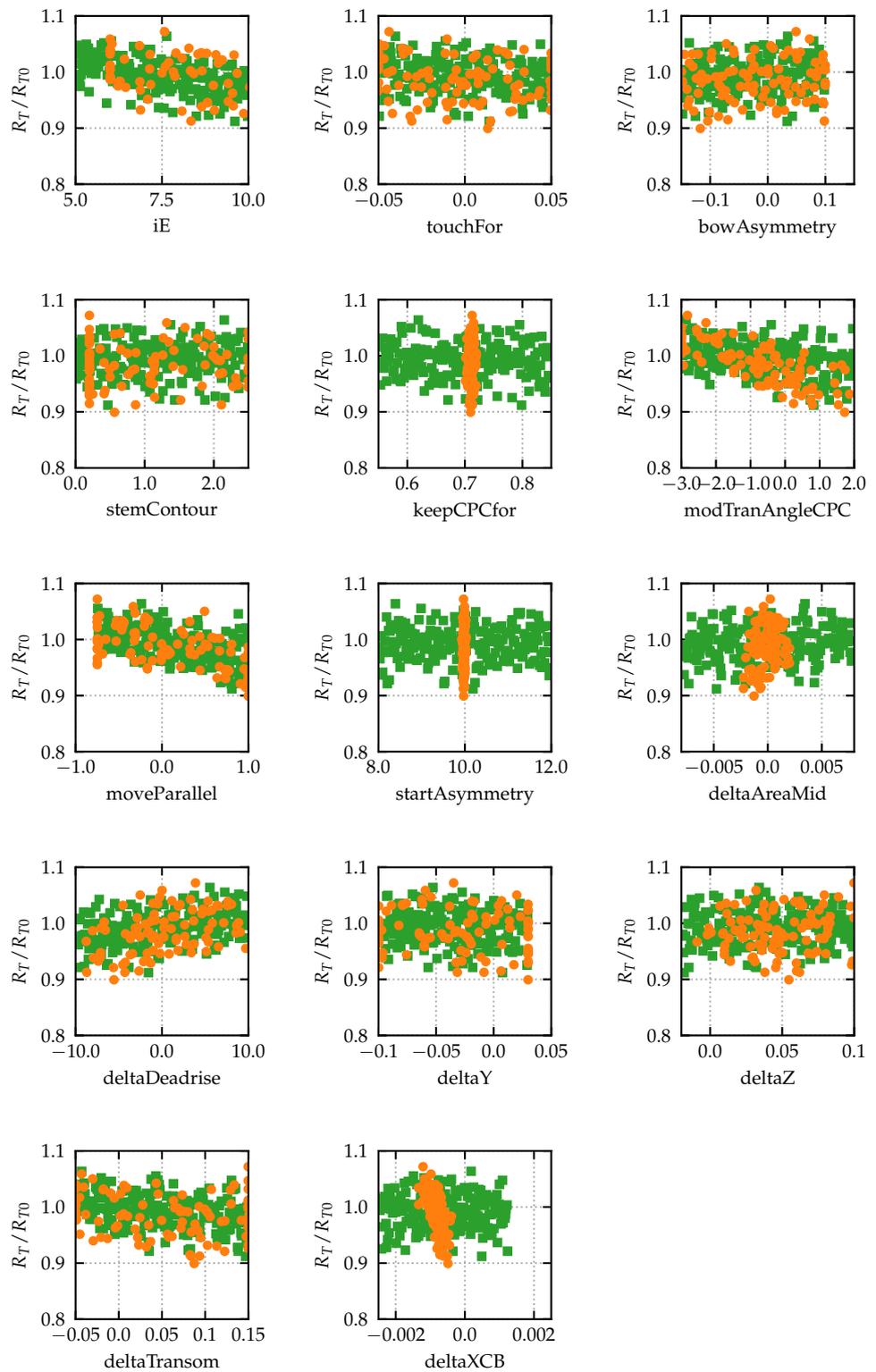


Figure 17. Dependency of the objective function evaluated with XPAN on CAD parameters (green) and on PCA parameters (orange).

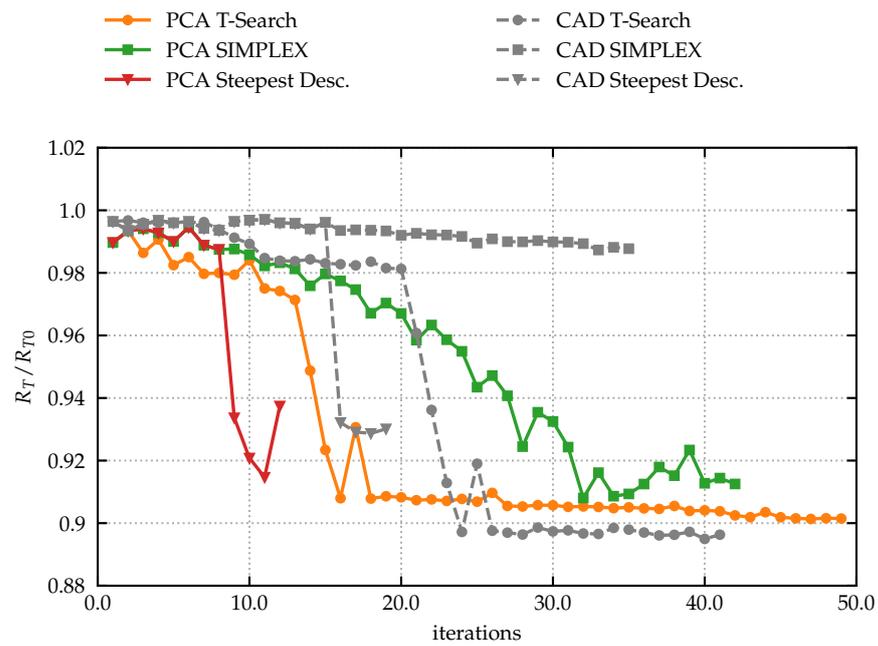


Figure 18. Evaluation of objective function with XPAN for three optimisation algorithms applied in CAD space and PCA space.

At first, the results for the baseline are compared. The results for the total resistance R_T , the trim angle θ and the change in sinkage ζ_O are summarised in Table 6. As can be seen, the agreement for the baseline is rather good, with a difference of 5.9% in the absolute value. It should be noted, however, that the coarse grid underestimates the resistance, so for the fine grid the discrepancy would be slightly larger. Interestingly, for the optimised hull form the differences significantly increase. The viscous flow solver did not confirm the optimisation potential found by the potential flow code. A possible explanation for this behaviour can be found by looking at the free surface elevation predicted by both codes. For the baseline, see Figure 19, a similar wave pattern can be observed for SHIPFLOW (upper half of the cat’s wave pattern) and Neptuno (lower half of the cat’s wave pattern), although the wave at the bow seems to be a little bit underestimated. The optimised design, see Figure 20, features a bow that is a little bit more blunt, increasing the bow wave which turns out to be problematic for the potential flow code.

Table 6. Comparison of total resistance, trim angle and sinkage computed with SHIPFLOW (SHF) and Neptuno (NEP).

		NEP	SHF	(SHF-NEP)/NEP
Baseline	R_T	10.92 kN	10.28 kN	−5.9%
	θ	1.4°	1.5°	+6.4%
	ζ_O	0.072 m	0.065 m	−8.9%
SHF-OPT	R_T	10.91 kN	9.62 kN	−11.8%
	R_T/R_{T0}	0.999	0.936	−11.8%
	θ	1.12°	1.18°	+5.3%
	ζ_O	0.085 m	0.083 m	−2.4%

5.3. Final Optimisation Using a Viscous Flow Code

Although the optimisation potential seems overestimated by the potential flow code, the reduction of the degrees-of-freedom worked well. Thus, it is possible to perform the optimisation directly with Neptuno instead of using it only to confirm the results of the potential flow solver. Figure 21 displays the results of the optimisation run carried out

with Neptuneo. In this case, due to the increased numerical effort required to compute the resistance, the optimisation was only performed in the PCA space and only for two of the three considered optimisation algorithms, namely, the steepest descend and the SIMPLEX algorithm. The behaviour of the RANS code is similar to the potential flow code optimisation. The performance of the steepest descend is rather good, whereas the SIMPLEX takes a little bit longer to achieve the same improvement.

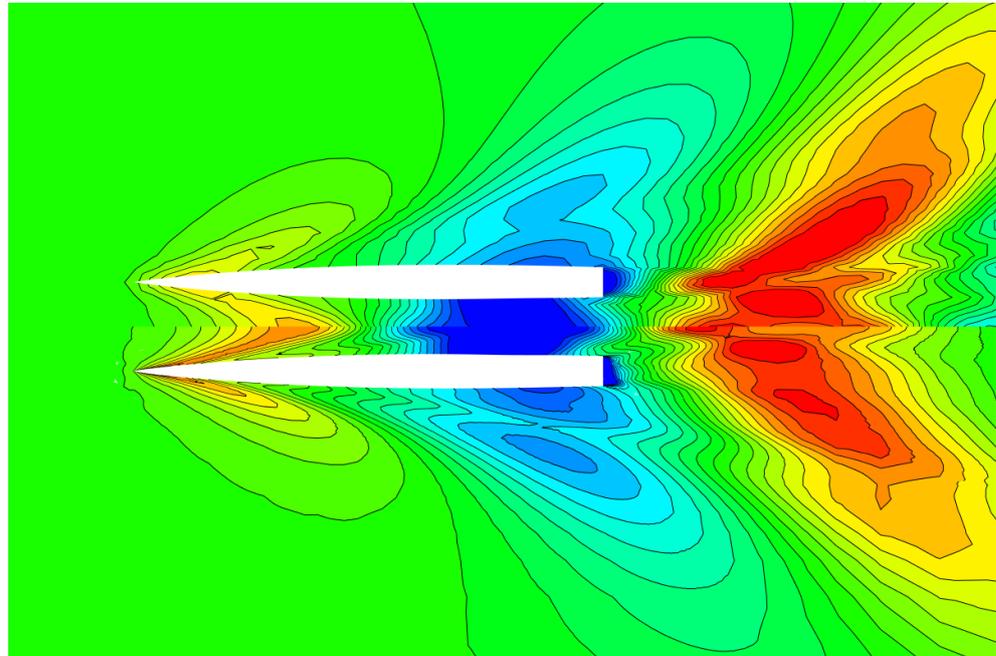


Figure 19. Free surface elevation computed for the baseline computed by SHIPFLOW (upper half of the cat's wave pattern) and Neptuneo (lower half of the cat's wave pattern).

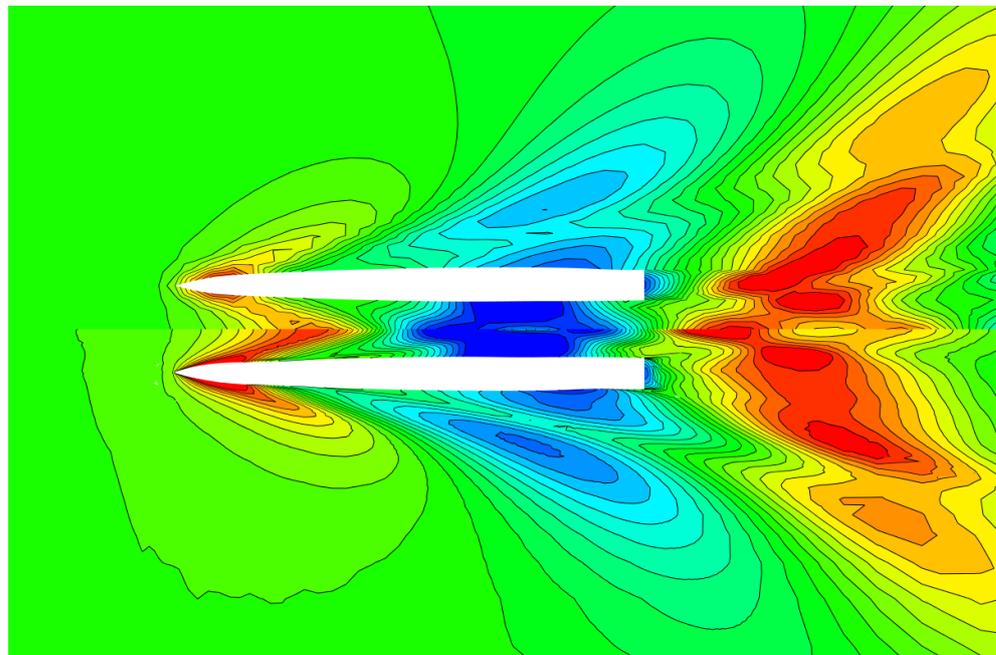


Figure 20. Free surface elevation computed for the design optimised with Shipflow (SHF-OPT) computed by SHIPFLOW (upper half of the cat's wave pattern) and Neptuneo (lower half of the cat's wave pattern).

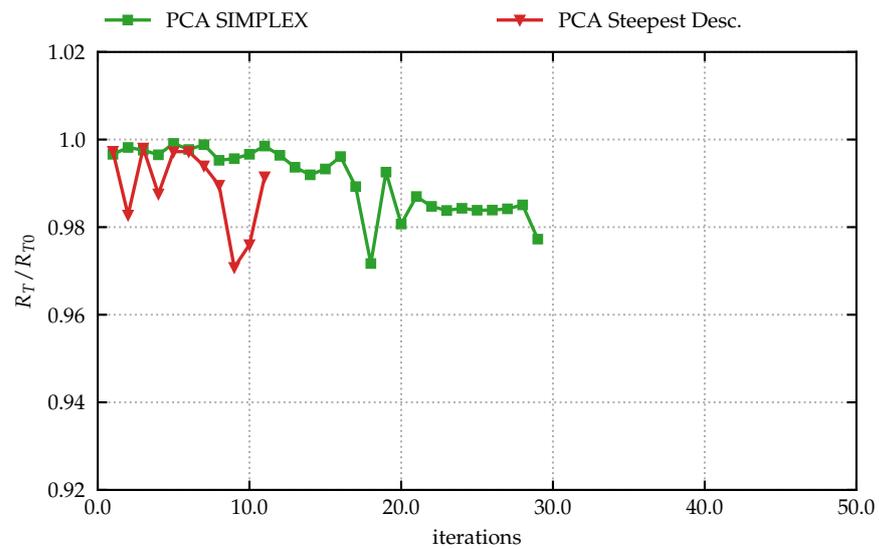


Figure 21. Evaluation of objective function with Neptune for both optimisation algorithms using PCA.

The steepest descend shows an improvement of 3% in total resistance at model scale (and 10.54 kN total resistance, corresponding to 3.6% improvement at full scale) relative to the baseline after only 11 steps, whereas the SIMPLEX algorithm yields a slightly lower value after 18 iterations. The best design (NEP-OPT) is cross-checked with XPAN as well, which yields an improvement of 7.2%. Figure 22 shows the corresponding free surface elevation for the optimised design as computed with XPAN.

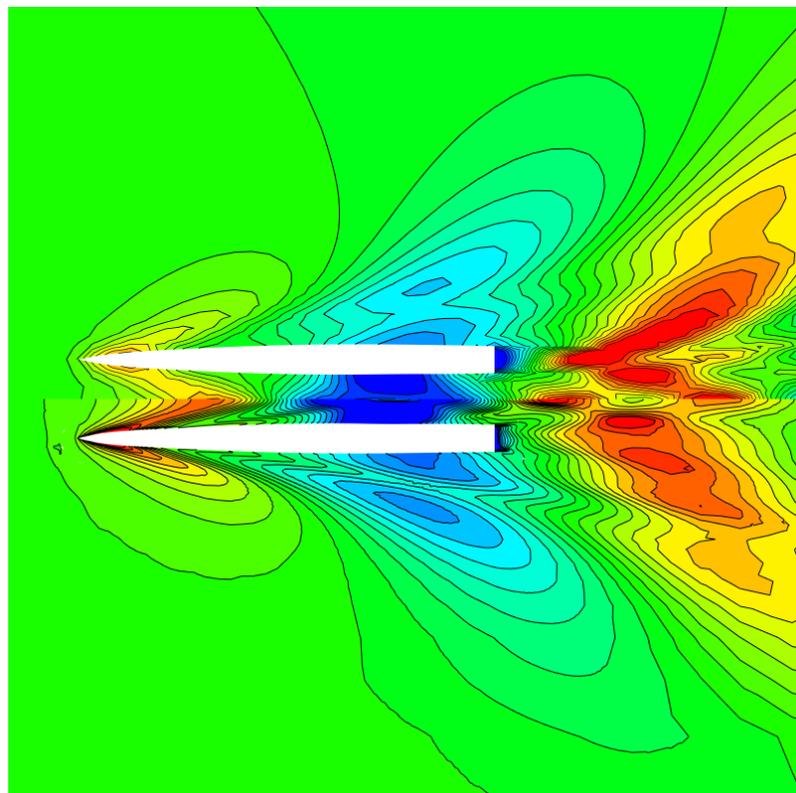


Figure 22. Free surface elevation computed for the design optimised with Neptune (NEP-OPT) by SHIPFLOW (upper half of the cat’s wave pattern) and Neptune (lower half of the cat’s wave pattern).

6. Conclusions and Future Work

With an approximate decrease of 3.6% in resistance at design draft and design speed when compared to the total resistance of the baseline, the improvements found for the catamaran are relatively small. The baseline's performance can therefore be considered pretty good already, at least when looking at the proposed and acceptable geometric variations. This is representative of practical design work where baselines are often quite mature already.

The performed optimisation shows that dimensionality reduction by using a PCA offers high potential in removing complexity from a model without giving away too much optimisation potential. In the present study, the number of free variables is reduced from 14 to 7, which lead to a reduction in computational time of 33%, without sacrificing much of the improvements.

It turns out that the limits of the potential code are reached during the course of the optimisation, requiring higher fidelity tools to be used. Due to the reduction in the degrees-of-freedom by performing a PCA, it is still feasible to run the whole optimisation using a viscous flow solver for practical application cases. This is an even bigger advantage for projects with even more free variables, where gradient-based algorithms take a very long time until they find the direction in which to improve the objective.

More research is needed, naturally. The one-stop steepest descent would be worthwhile to test within other optimisation campaigns. In addition, the PCA and its influence on both speed-up and optimisation gains requires more application cases to gather evidence. In addition, further fine-tuning should be undertaken with regard to questions of sample size for the PCA and the number of points collected. While a large set of samples and many points to consider certainly increase the accuracy of the dimensionality reduction, complex parametric models require a few seconds to a minute of time to update. The fewer samples are acceptable, the less time spent on the PCA.

Author Contributions: This work was jointly planned and carried out by both authors. Both authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the German Ministry of Economic Affairs through the research project AutoPLOP under grant number 03SX523A.

Data Availability Statement: No data are made available in public. Interested parties can contact the authors.

Acknowledgments: The authors would like to express their gratitude towards Andrés Cura Hochbaum for providing the RANS code Neptuno, FLOWTECH International AB—especially Michal Orych—for the support with the XPAN computations, Samuel James from GridPro for the support regarding the automated meshing, as well as Erik Bergmann for the implementation of the RBF morphing and dimensionality reduction in CAESSES.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Harries, S. Practical Shape Optimization Using CFD: State-Of-The-Art in Industry and Selected Trends. In Proceedings of the Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2020), Pontignano, Italy, 17–19 August 2020.
2. Harries, S.; Abt, C. Faster Turn-around Times for the Design and Optimization of Functional Surfaces. *Ocean Eng.* **2019**, *193*, 106470. [[CrossRef](#)]
3. Harries, S.; Abt, C.; Brenner, M. *Upfront CAD—Parametric Modeling Techniques for Shape Optimization*; Springer: Heidelberg, Germany, 2018.
4. Sobol, I.M.; Levitan, Y.L. *The Production of Points Uniformly Distributed in a Multidimensional Cube*; Technical Report; Institute of Applied Mathematics, USSR Academy of Sciences: Moscow, Russia, 1976. (In Russian)
5. Harries, S.; Abt, C. Integration of Tools for Application Case Studies. In *A Holistic Approach to Ship Design. Volume 2: Application Case Studies*; Springer: Heidelberg, Germany, 2021.
6. Botsch, M.; Kobbelt, L. *Real-Time Shape Editing Using Radial Basis Functions*; Blackwell Publishing, Inc.: Oxford, UK; Boston, MA, USA, 2005; Volume 24.

7. Albert, S.; Hildebrandt, T.; Harries, S.; Bergmann, E.; Kovacic, M. Parametric Modeling and Hydrodynamic Optimization of an Electric Catamaran Ferry based on Radial Basis Function for an Intuitive Set-Up. In Proceedings of the Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2021), Muehlheim an der Ruhr, Germany, 2021.
8. Abt, C.; Harries, S. Hull Variation and Improvement using the Generalised Lackenby Method of the FRIENDSHIP-Framework. *Nav. Archit.* **2007**, 166–167.
9. Diez, M.; Campagna, E.F.; Stern, F. Design-space dimensionality reduction in shape optimization by Karhunen–Loève expansion. *Comput. Methods Appl. Mech. Engrg.* **2015**, 283, 1525–1544. [[CrossRef](#)]
10. Pellegrini, R.; Serani, A.; Harries, S.; Diez, M. Multi-Objective Hull-Form Optimization of a SWATH Configuration via Design-Space Dimensionality Reduction, Multi-Fidelity Metamodels, and Swarm Intelligence. In Proceedings of the VII International Conference on Computational Methods in Marine Engineering (MARINE 2017), Nantes, France, 15–17 May 2017.
11. Bergmann, E.; Fütterer, C.; Harries, S.; Palluch, J. Massive Parameter Reduction for faster Fluid-dynamic Optimization of Shapes. In Proceedings of the International CAE Conference and Exhibition, Vicenza, Italy, 8–9 October 2018.
12. Uharek, S.; Cura Hochbaum, A. The influence of inertial effects on the mean forces and moments on a ship sailing in oblique waves Part B: Numerical prediction using a RANS code. *Ocean Eng.* **2018**, 165, 264–276. [[CrossRef](#)]
13. Cura Hochbaum, A. Ein Finite-Volumen-Verfahren zur Berechnung Turbulenter Schiffsumströmungen. Ph.D. Thesis, Technische Universität Hamburg, Hamburg, Germany, 1993. (In German)
14. Cura Hochbaum, A.; Vogt, M. Towards the simulation of seakeeping and maneuvering based on the computation of the free surface viscous ship flow. In Proceedings of the 24th ONR Symposium on Naval Hydrodynamics, Fukuoka, Japan, 8–13 July 2002.
15. Patankar, S.V.; Spalding, D.B. A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows. *Int. J. Heat Mass Transf.* **1972**, 15. [[CrossRef](#)]
16. Wilcox, D.C. *Turbulence Modeling for CFD*, 1st ed.; DCW Industries: Hong Kong, China, 1993.
17. Osher, S.; Sethian, J.A. Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations. *J. Comput. Phys.* **1988**, 79, 12–49. [[CrossRef](#)]
18. Sussman, M.; Smereka, P.; Osher, S. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.* **1994**, 114, 146–159. [[CrossRef](#)]
19. Cura Hochbaum, A.; Pierzynski, M. Flow Simulation for a Combatant in Head Waves. In Proceedings of the CFD Workshop Tokyo, Tokyo, Japan, 13–16 November 2005; pp. 1–5.
20. Cura Hochbaum, A.; Vogt, M.; Gatchell, S. Maneuvering prediction for two tankers based on RANS simulations. In Proceedings of the SIMMAN 2008 Workshop, Lyngby, Denmark, 14–16 April 2008; p. F22.
21. Cura Hochbaum, A.; Uharek, S. Prediction of the manoeuvring behaviour of the KCS based on virtual captive tests. In Proceedings of the SIMMAN 2014 Workshop, Copenhagen, Denmark, 2014.
22. ITTC. 1978 ITTC Performance Prediction Method. Recommended Procedures and Guidelines, Number 7.5-02-03-01.4. 2017.
23. Uharek, S. Numerical Prediction of Ship Manoeuvring Performance in Waves. Ph.D. Thesis, Technical University Berlin, Berlin, Germany, 2019.
24. ITTC. Uncertainty Analysis in CFD. Recommended Procedures and Guidelines. 2002, pp. 7.5–03.
25. Hilleary, R. *The Tangent Search Method of Constrained Minimization*; Technical Report, Research Paper 59; United States Naval Postgraduate School: Monterey, CA, USA, 1966.
26. Nelder, J.A.; Mead, R. A Simplex method for function minimization. *Comput. J.* **1965**, 7, 308–313. [[CrossRef](#)]