UNIONE EUROPEA
Fondo Europeo di Sviluppo Regionale

Ministero dell'Università e della Ricerca

PON
RICERCA
E INNOVAZIONE
2014 - 2020

Consiglio Nazionale delle Ricerche

Cluster Trasporti

CTN01_00176_163601

TRIM
Tecnologia e Ricerca industriale per la Mobilità Marina

TRIM

Tecnologia e Ricerca Industriale per la Mobilità Marina

Parametric analysis techniques for preliminary design based on parameter space reduction

Sotto-Progetto	Progettazione
Obiettivo Realizzativo	Sviluppo del tool di supporto decisionale
Descrizione attività	Analisi preliminare di progettazione e produzione di imbarcazioni Scelta degli strumenti di supporto decisionale – fasi di offering e pre-design
Tipo di documento	Rapporto Tecnico
Codice del documento	SP.3-OR.4-D.1
Data di emissione	06/12/2021
Redazione	Marco Tezzele, Andrea Mola, Gianluigi Rozza

SISSA
Scuola Internazionale Superiore di Studi Avanzati

NE PER SEGUIR VIRTUTE E CANOSCENZA · SISSA ·

Consiglio Nazionale delle Ricerche   FINCANTIERI   CETENA   SEASTEMA   AZIMUT BENETTI GROUP   PERINI NAVI

mare FVG   Navtec Distretto Tecnologico Trasporti Navali   LIBERTYlines COMPAGNIA DI NAVIGAZIONE VELOCE   OPUS automazione   NEXTWORKS ENGINEERING FORWARD   ib influencing business   SULTAN ALLESTIMENTI NAVALI

Titolo documento

Parametric analysis techniques for preliminary design based on parameter space reduction

Codice documento   SP.3-OR.4-D.1

Distribuzione   Pubblico

Rev.	Data	Pagine	Redazione	Responsabile
0	06/12/2021	2+28	Marco Tezzele, Andrea Mola, Gianluigi Rozza	Gianluigi Rozza

Contents

Summary

We propose parametric analysis techniques based on merit indexes, the goal is to develop parameter space reduction techniques by active subspaces. Test cases are provided with significant examples in naval applications, by considering geometric shape parametrisation, as well as physical parameters/quantities.

1 Introduction

Nowadays engineering simulations present a wide range of different parameters. When it comes to find an optimal solution with respect to the physical constraints it is easy to be affected by the curse of dimensionality, when the number of parameters makes the simulation unfeasible. This problem arises quite easily even with a small parameter space dimension (depending on the simulation, even ten parameters could take months to be optimized). In this framework reducing the dimension of this space becomes crucial and a priority. To tackle it we focus on the active subspaces property (see [1]) to carry out a technique applied on a naval engineering problem, that is the computation of the total wave resistance of a hull advancing in calm water. In the framework of simulation-based design and shape optimization we cite, among others, [2, 3, 4, 5]. The computational pipeline we are going to present is composed first by a geometrical parametrization and deformation of the hull through free form deformation (see [6]). Then the use of a high fidelity solver based on Boundary Element Method (BEM) to get the wave resistance with respect to the geometrical parameters. We consider also a structural parameter — the initial displacement of the hull — and a physical one — the velocity of the hull. Subsequently active subspaces are identified thanks to the data collected from the high fidelity solver, and finally a proper reduced response surface is constructed. The result allows the final user to have an estimate of the wave resistance under a certain threshold and within a time of one second with respect to the hours needed for a single classic full simulation. Moreover during the process is possible to identify the most important parameters and have insights on how they influence the output of interest. Figure 1 summarizes the proposed computational pipeline.

$$\boxed{\text{Problem settings}} \rangle \boxed{\text{FFD}} \rangle \boxed{\text{BEM}} \rangle \boxed{\text{AS}} \rangle \boxed{\text{RSM}} \rangle \boxed{\text{Fast CFD evaluations}}$$
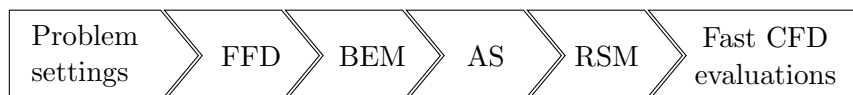
Figure 1: Scheme of the structure of the pipeline proposed. The geometrical deformation is performed via free form deformation (FFD), then a Boundary Element Method (BEM) solver computes the wave resistance, the active subspaces (AS) are detected and finally a response surface method (RSM) is employed.

The content of this contribution is organized as follows. Section 2.1 introduces the ship resistance prediction problem, its dependence on hull shape deformations, and equations of the fluid structure interaction model used for the simulations. In section 3 we recall the free form deformation technique and we show the main features of the developed tool to manage parametric shapes. Section 4 has the purpose of introducing some detail about the high fidelity solver implementation. In section 5 we present the active subspaces properties and its features, with a numerical recipe to identify them. Then section 6 shows the numerical results obtained by coupling the three methods in sequence. Finally conclusions and perspectives are drawn in section 8.

2   A model naval problem: wave resistance estimation of a hull advancing in calm water

In this section we introduce the problem of the estimation of the resistance of a hull advancing in calm water. The model hull shape considered in this work is the DTMB 5415, which was originally conceived for the preliminary design of a US Navy Combatant ship. Due to the wealth of experimental data available in the literature (see for example [7, 8]) such shape, which includes a sonar dome and a transom stern (see Figure 2), has become a common benchmark for naval hydrodynamics simulation tools.

Let $\Omega \subset \mathbb{R}^3$, be a domain (see Figure 2) associated with our DTMB 5415 model hull. We call Ω the *reference* domain; for practical reasons this domain happens to correspond to the undeformed hull, even though this assumption is not fundamental for the remainder of the paper. We here remark that the domain considered in the fluid dynamic simulations is in principle the volume surrounding the hull which is occupied by water, namely Ω_w. Further details about the fluid dynamic domain will be provided in Section 2.1.

Figure 2: Representation of the reference domain Ω, that is the DTMB 5415 hull.

Let $\mathcal{M}(\boldsymbol{x}; \boldsymbol{\mu}^{\text{GEOM}}) : \mathbb{R}^3 \to \mathbb{R}^3$ be a shape morphing that maps the reference domain Ω into the deformed domain $\Omega(\boldsymbol{\mu}^{\text{GEOM}})$ as follows:

$$\Omega(\boldsymbol{\mu}^{\text{GEOM}}) = \mathcal{M}(\Omega; \boldsymbol{\mu}^{\text{GEOM}}).$$

Quite naturally, the results of the fluid dynamic simulations will depend on the specific hull shape considered, which are in turn associated to the parameters defining the morphing $\mathcal{M}$ (which will be extensively defined in Section 3). It is worth pointing out here that the geometrical quantity having the most effect on the resistance is clearly the immersed volume of a hull shape, as higher volumes will generate higher drag values. Considering only the absolute resistance value might then lead designers to disregard hull shapes that have relatively good drag performances despite having higher immersed volume. This could be avoided adopting different strategies. A first possibility is considering a resistance value which is nondimensionalized using the displaced volume. Alternatively one can constrain the shape morphing parameters so as to obtain constant volume shapes. Yet, the former solution poses some problem in the identification of the nondimensionalization strategy which would result in resistance output indices truly independent of the hull immersed volume. As for the latter possibility, it would result in an undesired complication of the shape parametrization algorithm. In this work the effect of immersed volume variations on the hull resistance

is more naturally taken into account by considering the rigid motions of the hull into the fluid dynamic simulations. This allows for only imposing the ship displacement, letting any hull shape considered reach the equilibrium position under the action of gravity and hydrodynamic forces. If a shape morphing is characterized by a higher volume, the computed sink will reduce to obtain the same vertical component of the hydrodynamic force.

For this reason, along with the geometrical parameters associated with the hull morphing, the results of our simulations are also affected by the ship displacement and cruise speed, which are instead physical parameters determining the hydrodynamic equilibrium position and forces. Thus, considering both the geometric morphing and variations in the physical parameters, we have a set of $m \in \mathbb{N}$ parameters which affect the output of the fluid dynamic simulations. The parametric domain considered is then defined as $\mathbb{D} \subset \mathbb{R}^m$, and is assumed to be a box in $\mathbb{R}^m$.

By a practical standpoint, once a point in the parameter domain $\mathbb{D}$ is identified, the specific hull geometry as well as the desired ship displacement and cruise speed are provided to the fluid dynamic solver, which carries out a flow simulation to provide a resistance estimate. In this framework free form deformation has been employed for the generation of a very large number of hull geometries obtained from the morphing of the DTMB 5415 naval combatant hull. Each geometry generated has been used to set up a high fidelity hydrodynamic simulation with the desired ship displacement and hull speed. The output resistances for all the configurations tested have been finally analyzed by means of active subspaces in order to reduce the parameter space.

In the next subsections, we will provide a brief description of the unsteady fully non-linear potential fluid dynamic model used to carry out the high fidelity simulations. In addition, we will describe the rigid body equations based on hull quaternions used to compute the hull linear and angular displacements corresponding to the final hydrodynamic equilibrium position reached at the end of each simulation. We refer the interested reader to [9, 10, 11] for further information on the fully nonlinear potential free surface model, on its application to complex hull geometries, and on the treatment of the hull rigid motions, respectively.

2.1 Fully nonlinear potential model

In the simulations we are only considering the motion of a ship advancing at constant speed in calm water. For such reason we solve the problem in a *global*, translating reference frame $\widehat{XYZ}$, which is moving with the constant horizontal velocity of the boat $\boldsymbol{V}_\infty = (V_\infty, 0, 0)$. Thus, the X axis of the reference frame is aligned with $\boldsymbol{V}_\infty$, the Z axis is directed vertically (positive upwards), while the Y axis is directed laterally (positive port side).

As aforementioned, the domain $\Omega_w(t)$ in which we are interested in computing the fluid velocity and pressure is represented by the portion of water surrounding the ship hull. The time varying shape of such domain — and in particular that of its boundary with the air above — is one of the unknowns of the fluid dynamic problem. By convention,

we place the origin of the vertical axis Z in correspondence with the undisturbed free surface level, and we start each simulation at time $t = 0$ from such undisturbed configuration. Thus, $\Omega_w(t = 0) = \mathbb{R}^3_{Z-} \ \Omega$, where $\mathbb{R}^3_{Z-}$ indicates the lower half-space of $\mathbb{R}^3$, for which $Z \leq 0$.

If overturning waves are not observed (which is typically the case for low cruise velocities typical of a ship), the domain $\Omega_w(t)$ is simply connected. So under the assumptions of irrotational flow and non viscous fluid the velocity field $\boldsymbol{v}(\boldsymbol{X}, t)$ admits a representation through a scalar potential function $\Phi(\boldsymbol{X}, t)$, namely

$$\boldsymbol{v} = \boldsymbol{\nabla}\Phi = \boldsymbol{\nabla}(\boldsymbol{V}_\infty \cdot \boldsymbol{X} + \phi) \qquad \forall \ \boldsymbol{X} \in \Omega_w(t), \tag{1}$$

in which $\phi(\boldsymbol{X}, t)$ is the *perturbation potential*. Under the present assumptions, the equations of motion simplify to the unsteady Bernoulli equation and to the Laplace equation for the perturbation potential:

$$\frac{\partial \phi}{\partial t} + \frac{1}{2}|\boldsymbol{\nabla}\phi + \boldsymbol{V}_\infty|^2 + \frac{p - p_a}{\rho} - \boldsymbol{g} \cdot \boldsymbol{X} = C(t) \qquad \text{in } \Omega_w(t), \tag{2a}$$

$$\Delta \phi = 0 \qquad \text{in } \Omega_w(t), \tag{2b}$$

where $C(t)$ is an arbitrary function of time, and $\boldsymbol{g} = (0, 0, -g)$, is the gravity acceleration vector, directed along the z axis. The unknowns of such mathematical problem ϕ and p are uncoupled. This means that the solution of the Poisson problem in Eq. (2b) can be obtained independently of the pressure field. Once such solution is obtained, the pressure can be obtained through a postprocessing step based on Bernoulli Eq. (2a). Thus, the Laplace equation is the governing equation of our model. Such equation is complemented by non penetration boundary conditions on the hull surface $\Gamma^b(t)$ and water basin bottom boundary $\Gamma^{bot}(t)$, and by homogeneous Neumann boundary conditions on the truncation boundaries $\Gamma^{far}(t)$ of the numerical domain. The bottom of the basin is located at a depth corresponding to 2 boat lengths, while the truncation boundaries are located approximatively at a distance from the boat of 6 boat lengths in the longitudinal direction X and of 2 boat lengths in the lateral direction Y). On the water free surface $\Gamma^w(t)$, we employ the kinematic and dynamic Semi-Lagrangian fully nonlinear boundary conditions, which respectively read

$$\frac{\delta \eta}{\delta t} = \frac{\partial \phi}{\partial z} + \boldsymbol{\nabla}\eta \cdot (\boldsymbol{w} - \boldsymbol{\nabla}\phi - \boldsymbol{V}_\infty) \qquad \text{in } \Gamma^w(t), \tag{3}$$

$$\frac{\delta \phi}{\delta t} = -g\eta + \frac{1}{2}|\boldsymbol{\nabla}\phi|^2 + \boldsymbol{\nabla}\phi \cdot (\boldsymbol{w} - \boldsymbol{\nabla}\phi - \boldsymbol{V}_\infty) \qquad \text{in } \Gamma^w(t). \tag{4}$$

The former equation expresses the fact that a material point moving on the free surface will stay on the free surface — here assumed to be a single valued function $\eta(X, Y, t)$ of the horizontal coordinates X and Y. The latter condition is derived from Bernoulli Eq. (2a), under the assumption of constant atmospheric pressure on the water surface. This peculiar form of the fully nonlinear boundary conditions was proposed by Beck [12]. Eq. (3) allows for the computation of the vertical velocity of markers which move on the water free surface with a prescribed horizontal speed (w_X, w_Y). Eq. (4) is used to obtain the velocity potential values in correspondence with such markers. The
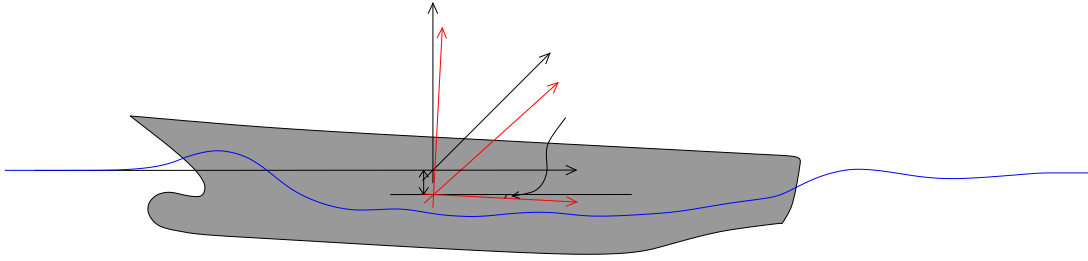
Figure 3: A sketch illutrating the hull-attached frame $\widehat{xyz}$ in red and the global reference frame $\widehat{XYZ}$ which is moving with the constant horizontal velocity of the boat. The ship here depicted is experiencing a vertical displacement s and an angular displacement characterized by the pitch angle ϕ.

resulting vector $\boldsymbol{w} = (w_X, w_Y, \frac{\delta\eta}{\delta t}) = \dot{\boldsymbol{X}}$ is the time derivative of the position of the free surface markers. In this work, such free surface markers are chosen as the free surface nodes of the computational grid. To avoid an undesirable mesh nodes drift along the water stream, the markers arbitrary horizontal velocity is set to 0 along the X direction. The Y component of the water nodes in contact with the ship — which is moved according with the computed linear and angular displacements — is chosen so as to keep such nodes on the hull surface. As for the remaining water nodes, the lateral velocity value is set to preserve mesh quality.

2.2   Three dimensional hull rigid motions

The ship hull is assumed to be a rigid body. A second, *hull-attached* reference frame $\widehat{xyz}$, which follows the hull in its translations and rotations is employed to study the ship motions. The center of such reference frame is located at the ship center of gravity, which in the global reference frame reads $\boldsymbol{X}^G(t) = X^G(t)\boldsymbol{e}_X + Y^G(t)\boldsymbol{e}_Y + Z^G(t)\boldsymbol{e}_Z$, where $\boldsymbol{e}_X$, $\boldsymbol{e}_Y$, $\boldsymbol{e}_Z$ are the unit vectors along the global system axes. See Figure 3 for a detailed sketch.

The rotation matrix $R(t)$ is used to convert the coordinates of a point $\boldsymbol{x}$ written in the hull-attached reference frame, to those in the global frame $\boldsymbol{X}$, namely

$$\boldsymbol{X}(t) = R(t)\boldsymbol{x} + \boldsymbol{X}^G(t). \tag{5}$$

The global frame velocity of a point having coordinates $\boldsymbol{x}$ in the hull-attached frame is obtained as

$$\boldsymbol{V}^{\mathrm{hp}}(t) = \boldsymbol{\omega}(t) \times \boldsymbol{x} + \dot{\boldsymbol{X}}^G(t), \tag{6}$$

in which $\boldsymbol{\omega}$ is the angular velocity vector.

Eqs. (5) and (6) imply that once $\boldsymbol{X}^G(t)$, $R(t)$, and $\boldsymbol{\omega}(t)$ are known at time t, the position and velocity of each point of the hull can be obtained. For this reason, writing the time evolution equations for $\boldsymbol{X}^G(t)$, $R(t)$, and $\boldsymbol{\omega}(t)$ is sufficient to fully determine the hull dynamics. In the next sections, we will present such evolution equations written in the global reference frame, as presented in [13] and [14].

The evolution equation for $\boldsymbol{X}^G(t)$ is obtained via the linear momentum conservation equation, which in the case of our hydrodynamics simulation framework reads

$$m_s \ddot{\boldsymbol{X}}^G(t) = m_s \boldsymbol{g} + \boldsymbol{F}^w(t). \tag{7}$$

In Eq. (7), m_s is the mass of the ship, while the hydrodynamic force vector $\boldsymbol{F}^w(t)$ is in the present work obtained as the sum of the pressure and viscous forces on the hull.

The evolution equation for $\boldsymbol{\omega}$ is obtained writing the angular momentum conservation, namely

$$R(t) I^G R(t)^T \dot{\boldsymbol{\omega}}(t) + \boldsymbol{\omega}(t) \times R(t) I^G R(t)^T \boldsymbol{\omega}(t) = \boldsymbol{M}^w(t), \tag{8}$$

where I^G is the matrix of inertia of the ship in the hull-attached reference frame, and hydrodynamic moment vector $\boldsymbol{M}^w(t)$ is the sum of the moment about the ship center of gravity of the pressure and viscous forces on hull, propeller and appendages.

To write an evolution equation for R, starting from the angular velocity vector $\boldsymbol{\omega}(t) = [\omega_X(t), \omega_Y(t), \omega_Z(t)]$, we first introduce the skew symmetric tensor

$$\omega(t) = \begin{bmatrix} 0 & -\omega_Z(t) & \omega_Y(t) \\ \omega_Z(t) & 0 & -\omega_X(t) \\ -\omega_Y(t) & \omega_X(t) & 0 \end{bmatrix}. \tag{9}$$

Note that tensor $\omega(t)$ will act on a vector $\boldsymbol{u} \in \mathbb{R}^3$ as if the vector product by $\boldsymbol{\omega}(t)$ were applied to $\boldsymbol{u}$:

$$\boldsymbol{\omega}(t) \times \boldsymbol{u} = \omega(t) \boldsymbol{u}. \tag{10}$$

Making use of such tensor, an evolution equation for the rotation matrix $R(t)$ reads

$$\dot{R}(t) = \omega(t) R(t), \tag{11}$$

which can be advanced in time to obtain the components of R and close the equations of motions of a rigid body in three dimension. Yet, in the common practice of rigid body simulations, direct numerical integration of Eq. (11) is avoided. The most important reason for this is related to numerical drift. If we in fact keep track of the orientation of a rigid body integrating Eq. (11), numerical error will build up in the entries of $R(t)$, so that it will no longer be a rotation matrix, losing its properties of orthogonality and of having determinant equal to 1. Physically, the effect would be that applying $R(t)$ to a body would cause a skewing effect.

A better way to represent the orientation of a rigid body in three dimensions (even with large rotations) is represented by the use of *unit quaternions* (see the work of [15] for details). For our purposes, quaternions can be considered as a particular type of four element vector, normalized to unit length. If we indicate the quaternion $\boldsymbol{q} = s + v_X \boldsymbol{e}_X + v_Y \boldsymbol{e}_Y + v_Z \boldsymbol{e}_Z$ as $[s, \boldsymbol{v}]$, the internal product of two quaternions $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ is defined as

$$\boldsymbol{q}_1 \boldsymbol{q}_2 = [s_1, \boldsymbol{v}_1] [s_2, \boldsymbol{v}_2] = [s_1 s_2 - \boldsymbol{v}_1 \cdot \boldsymbol{v}_2, \, s_1 v2 + s_2 v1 + \boldsymbol{v}_1 \times \boldsymbol{v}_2]. \tag{12}$$

The norm of a quaternion $\boldsymbol{q}$ is defined as $||\boldsymbol{q}|| = \sqrt{s^2 + v_X^2 + v_Y^2 + v_Z^2}$. Unit quaternions can be used to represent rotations in a three dimensional space. In fact, given a quaternion $\boldsymbol{q}(t) : ||\boldsymbol{q}(t)|| = 1 \quad \forall t$, we can obtain the corresponding rotation matrix as

$$R = \begin{bmatrix} 1 - 2v_Y^2 - 2v_Z^2 & 2v_X v_Y - 2sv_Z & 2v_X v_Z + 2sv_Y \\ 2v_X v_Y + 2sv_Z & 1 - 2v_Y^2 - 2v_Z^2 & 2v_Y v_Z - 2sv_X \\ 2v_X v_Z - 2sv_Y & 2v_Y v_Z + 2sv_X & 1 - 2v_Y^2 - 2v_Z^2 \end{bmatrix}, \tag{13}$$

in which to lighten the notation we omitted the time dependence of both $R(t)$ and the components of $\boldsymbol{q}(t)$.

Finally, the equation needed to describe the time evolution for the hull quaternion $\boldsymbol{q}(t)$ is

$$\dot{\boldsymbol{q}}(t) = \frac{1}{2}\boldsymbol{\omega}_q(t)\boldsymbol{q}(t), \tag{14}$$

where $\boldsymbol{\omega}_q(t) = [0, \boldsymbol{\omega}(t)]$ is the quaternion associated with the angular velocity vector $\boldsymbol{\omega}(t)$. As quaternions only have four entries, there only is one extra variable used to describe the three degrees freedoms of a three dimensional rotation. A rotation matrix instead employs nine parameters for the same three degrees of freedom; thus, the quaternions present far less redundancy than rotation matrices. Consequently, quaternions experience far less numerical drift than rotation matrices. The only possible source of drift in a quaternion occurs when the quaternion has lost its unit magnitude. This can be easily corrected by periodically renormalizing the quaternion to unit length [15].

3 Shape morphing based on Free Form Deformation

As already mentioned, we are interested in problems characterized by both physical and geometrical parameters. In such framework, the Free Form Deformation (FFD) approach is adopted to implement the hull deformations corresponding to each geometrical parameter set considered.

A very detailed description of FFD is beyond the scope of the present work. In the following we will give only a brief overview. For a further insight see [6] for the original formulation and [16, 17, 18, 19, 20] for more recent works.

We decided to adopt free form deformation among other possibilities (including, for instance, Radial Basis Functions or Inverse Distance Weighting) because it allows to have global deformations with a few parameters. For the complexity of the problem at hand, by trying to reduce the number of parameters starting from hundreds of them can be infeasible for the number of Monte Carlo simulations required. One of the possible drawbacks of FFD is generally that the parameters do not have a specific geometric meaning, like, for instance, a prescribed length or distance. In the case of application to AS this is not a problem since AS itself identifies new parameters, obtained by combination of the original ones, meaningless from the geometric and physical point of view.

FFD consists basically in three different step, as shown in Figure 4:

- Mapping the physical domain Ω to the reference one $\widehat{\Omega}$ with the map $\boldsymbol{\psi}$.

- Moving some control points $\boldsymbol{P}$ to deform the lattice with $\widehat{T}$. The movement of the control points is given by the weights of FFD, and represent our geometrical parameters $\boldsymbol{\mu}^{\mathrm{GEOM}}$.

- Mapping back to the physical domain $\Omega(\boldsymbol{\mu})$ with the map $\boldsymbol{\psi}^{-1}$.

So FFD map T is the composition of the three maps, i.e.

$$T(\cdot, \boldsymbol{\mu}^{\mathrm{GEOM}}) = (\boldsymbol{\psi}^{-1} \circ \widehat{T} \circ \boldsymbol{\psi})(\cdot, \boldsymbol{\mu}^{\mathrm{GEOM}}). \tag{15}$$

In particular, in the three dimensional case, for every point $\boldsymbol{X} \in \Omega$ inside the FFD box, its position changes according to

$$T(\boldsymbol{X}, \boldsymbol{\mu}^{\mathrm{GEOM}}) = \boldsymbol{\psi}^{-1} \left(\sum_{l=0}^{L} \sum_{m=0}^{M} \sum_{n=0}^{N} b_{lmn}(\boldsymbol{\psi}(\boldsymbol{X})) \boldsymbol{P}_{lmn}^{0} \left(\boldsymbol{\mu}_{lmn}^{\mathrm{GEOM}} \right) \right), \tag{16}$$

where b_{lmn} are Bernstein polynomials of degree l, m, n in each direction, respectively, and $\boldsymbol{P}_{lmn}^{0} \left(\boldsymbol{\mu}_{lmn}^{\mathrm{GEOM}} \right) = \boldsymbol{P}_{lmn} + \boldsymbol{\mu}_{lmn}^{\mathrm{GEOM}}$. In Figure 5, we show, for example, the application of the FFD morphing on a very simple geometry, that is a sphere.

We implemented this algorithm in a *python* package called *PyGeM* [21] in order to deal with the major industrial CAD file formats. It handles *iges*, *stl*, *step*, *vtk*, *unv*,
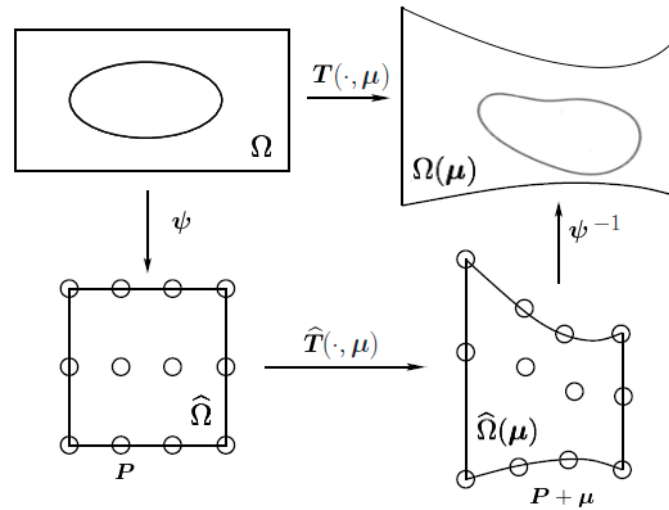
Figure 4: Sketch of the FFD map construction. For ease of readability we dropped the superscript from $\boldsymbol{\mu}^{\text{GEOM}}$.
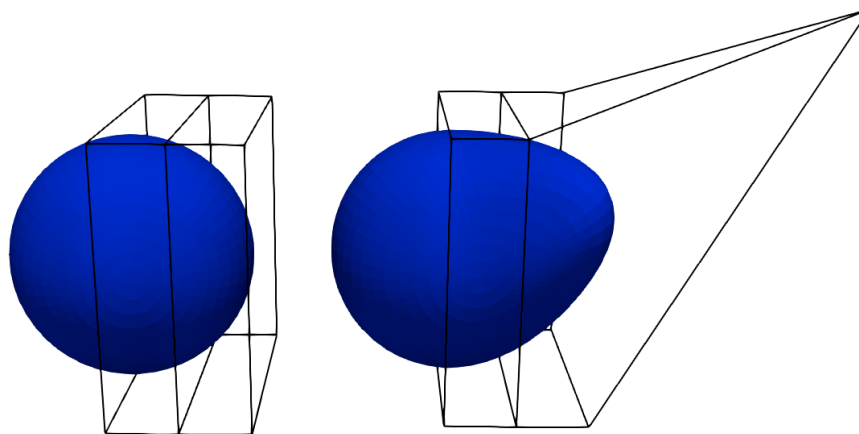

Figure 5: FFD morphing on a simple geometry: the sphere case. Here we move only one FFD control point in the lattice.

keyword, and *openfoam* files. It extracts the coordinates of the control points, deforms them according to the inputs given by the user and writes a new file with the morphed CAD. We improve the traditional version of the algorithm by allowing a rotation of the FFD lattice in order to give more flexibility to the tool. In general with our package it is possible to have a generic bounding box (not only a cube) as long as the ψ map is affine.

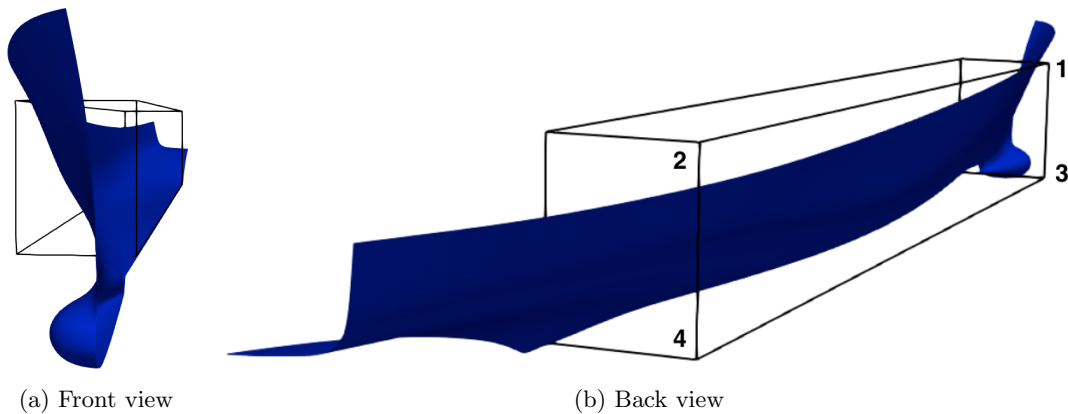(a) Front view   (b) Back view

Figure 6: Plot (a) shows the FFD lattice over one side wall of the hull from the front, while plot (b) depicts the hull and the lattice from the back together with the numbers that identify the FFD points.

In order to exemplify the equations above to our case, let us consider Figure 6, where the control points we are going to move are marked with numbers. As geometrical parameters we select six components of these four control points of the FFD lattice over one side wall of the hull. Then we apply the same deformation to the other side. This because one of our hypothesis is the symmetry of the deformed hull. In particular, Table 1 summarizes the set of design variables, the associated FFD-node coordinate modified (y is the span of the hull, x its length and z its depth) and the lower and upper bound of the modification. There are also two more parameters that do not affect the geometry, and are related to the physics of the problem, that is the displacement and the velocity of the hull. From now on we denote with $\boldsymbol{\mu} := \{\mu_i\}_{i\in[1,...,8]}$ the column vector of the parameters, where μ_i are defined in Table 1. To denote only the parameters affecting the geometrical deformation we use $\boldsymbol{\mu}^{\text{GEOM}} := \{\mu_i\}_{i\in[1,...,6]}$. For sake of clarity we underline that the undeformed original domain is obtained setting all the geometrical parameters to 0. All the upper and lower bounds are chosen in order to satisfy physical constraints.

To create the dataset with all the deformation, we started from the original *iges* file of the hull and deformed it with the *PyGeM* package. The deformations are generated randomly with an uniform distribution.

Table 1: Design space (FFD lattice $2 \times 2 \times 2$) with eight design parameters. Six geometrical parameters chosen among the FFD control points, one structural parameter that is the initial displacement of the hull and one physical parameter given by the velocity.

Parameter	Nature	Lower bound	Upper bound
μ_1	FFD Point 1 y	-0.2	0.3
μ_2	FFD Point 2 y	-0.2	0.3
μ_3	FFD Point 3 y	-0.2	0.3
μ_4	FFD Point 4 y	-0.2	0.3
μ_5	FFD Point 3 z	-0.2	0.5
μ_6	FFD Point 4 z	-0.2	0.5
μ_7	weight (kg)	500	800
μ_8	velocity (m/s)	1.87	2.70

4 Implementation of high fidelity potential solver based on the Boundary Element Method

The boundary value problem described in Section 2.1 is governed by the linear Laplace operator. Yet, it is nonlinear due to the presence of the boundary conditions in Eqs. (3) and (4). Further sources of nonlinearity are given by continuous change of the domain shape over time and by the arbitrary shape of the ship hull. Thus, for each time instant, we will look for the correct values of the unknown potential and node displacement fields by solving a specific nonlinear problem resulting from the spatial and time discretization of the original boundary value problem. The spatial discretization of the Laplace problem is based upon a boundary integral formulation described in [22]. In this framework, the domain boundary is subdivided into quadrilateral cells, on which bi-linear shape functions are used to approximate the surface, the flow potential values, and the normal component of its surface gradient. The resulting Boundary Element Method (BEM, see [23]) consists in collocating a Boundary Intergal Equation (BIE) in correpondence with each node of the numerical grid, and computing the integrals appearing in such equation by means of the described iso-parametric formulation. The linear algebraic equations obtained from such discretization method are combined with the Ordinary Differential Equations (ODE) derived from the Finite Element spatial discretization of the fully nonlinear free surface boundary conditions in Eqs. (3) and (4). The spatial discretization described is carried out making use of the deal.II open source library for C++ implementation of finite element discretizations ([24, 25]). To enforce a strong coupling between the fluid and structural problem, the aforementioned system of Differential Algebraic Equations (DAE) is complemented by the equations of the rigid hull dynamics (Eqs. (7)-(8) and (14)). The DAE system solution is time integrated by means of an arbitrary order and arbitrary time step implicit Backward Difference Formula (BDF) scheme implemented in the IDA package of the open source C++ library SUNDIALS ([26]). The potential flow model described has been implemented in a stand alone C++ software, the main features of which are described in [9].

The solver is complemented with a mesh module directly interfaced with CAD data structures [10]. Such feature allows for fully automated mesh generation once a hull shape is assigned at the start of each simulation by means of a — possibly non watertight — CAD geometry. Figure 7, on the left, displays the mesh generated on the surface of a DTMB 5415 Navy Combatant hull. At each time step of the simulation, the wave resistance is computed as

$$R^w = \int_{\Gamma^b} p\boldsymbol{n} \, d\Gamma \cdot \boldsymbol{e}_X, \tag{17}$$

where p is the pressure value obtained introducing the computed potential in Eq. (2a). The inviscid fluid dynamic model drag prediction is then corrected by adding a viscous drag contribution obtained by means of the ITTC-57 formula [27]. A full assessment of the accuracy of the high fidelity fluid structure interaction solver described is clearly beyond the scope of the present work (again, we refer the interested reader to [9, 10, 11]
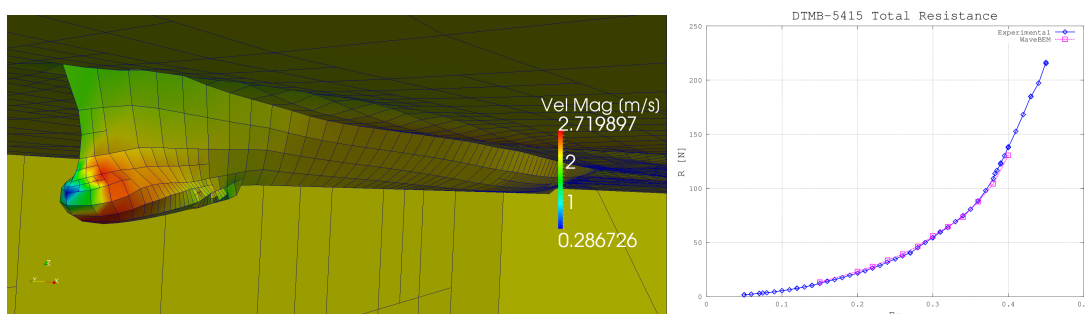
Figure 7: On the left, the mesh automatically generated on the surface of one of the deformed hulls obtained starting from the DTMB 5415 Navy Combatant Hull. On the right, the total resistance of the DTMB 5415 hull as a function of the Froude number associated with the surge velocity imposed in the simulations. The blue continuous line represents the experimental values presented in Olivieri et al. [7]. The values obtained in this work are represented by the dashed magenta line.

for more details). Yet, in Figure 7, on the right, we present a comparison between the computed total resistance curve and the corresponding one measured by Olivieri et al. [7]. As it can be appreciated in the plot, for all the Froude numbers tested the computed total drag difference with respect to the measurements is less then 6%. Given the fact that all the geometries tested are deformations of the present hull, and that all the velocities imposed fall in the range reported in the plot, it is reasonable to infer that for each simulation carried out the accuracy of the high fidelity model prediction will be similar to that of the results presented.

5 Parameter space reduction by Active Subspaces

The active subspaces (AS) approach represents one of the emerging ideas for dimension reduction in the parameter studies and it is based on the homonymous properties. The concept was introduced by Paul Constantine in [1], for example, and employed in different real world problems. We mention, among others, aerodynamic shape optimization [28], the parameter reduction for the HyShot II scramjet model [29], active subspaces for integrated hydrologic model [30], and in combination with POD-Galerkin method in cardiovascular problems [31].

A characteristic of the active subspaces is that they identify a set of important directions in the space of all inputs, instead of identifying a subset of the inputs as important. If the simulation output does not change as the inputs move along a particular direction, then we can safely ignore that direction in the parameter study. In Figure 8 it is possible to capture the main idea behind the active subspaces approach: we try to rotate the inputs domain in such a way lower dimension behavior of the output function is revealed. When an active subspace is identified for the problem of interest, then it is possible to perform different parameter studies such as response surfaces [32], integration, optimization and statistical inversion [33].

(a) Original output function   (b) First rotation   (c) Second rotation   (d) Output function with respect to the active variable
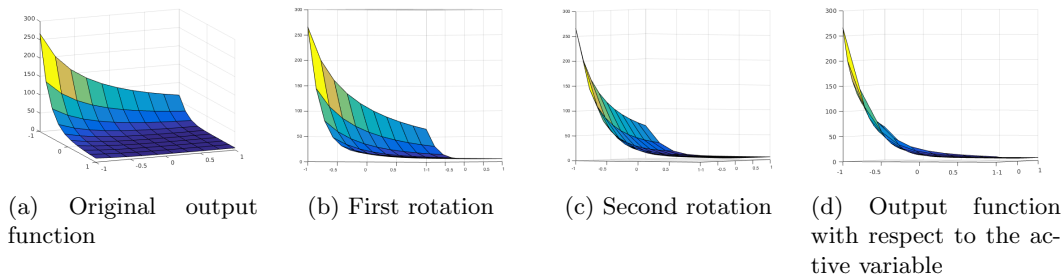
Figure 8: Example of a bivariate output function (a), intermediate rotations of the domain (b) and (c), and the final state (d), where we can see the variation of the function along the active variable.

There are some main ingredients in order to employ active subspaces. The first is a scalar function $f : \mathbb{R}^m \to \mathbb{R}$ smooth enough depending on the inputs $\boldsymbol{\mu}$ that represents the quantity of interest. Moreover we need the gradients of this map with respect to the inputs $\nabla_{\boldsymbol{\mu}} f(\boldsymbol{\mu})$ or an approximation of them, a sampling density ρ, and a gap between eigenvalues of the covariance matrix associated to the gradients. That is a symmetric, positive semidefinite matrix whose elements are the average products of partial derivatives of the simulations' input/output map, that reads

$$\boldsymbol{\Sigma} = \mathbb{E}\left[\nabla_{\boldsymbol{\mu}} f \, \nabla_{\boldsymbol{\mu}} f^T\right] = \int_{\mathbb{D}} (\nabla_{\boldsymbol{\mu}} f)(\nabla_{\boldsymbol{\mu}} f)^T \rho \, d\boldsymbol{\mu}, \tag{18}$$

where $\mathbb{E}$ is the expected value, $\rho : \mathbb{R}^m \to \mathbb{R}^+$ a probability density function — usually a uniform one, and $\boldsymbol{\Sigma}$ the so-called uncentered covariance matrix of the gradients of

f (for a more deep understanding of these operators see for example [34]). Usually a Monte Carlo method (see [35]) is used in order to approximate the eigenpairs of this matrix (see [36]) as follows

$$\boldsymbol{\Sigma} \approx \frac{1}{N_{\text{train}}^{\text{AS}}} \sum_{i=1}^{N_{\text{train}}^{\text{AS}}} \nabla_{\boldsymbol{\mu}} f_i \, \nabla_{\boldsymbol{\mu}} f_i^T, \tag{19}$$

where we draw $N_{\text{train}}^{\text{AS}}$ samples $\boldsymbol{\mu}^{(i)}$ independently from the measure ρ and where $\nabla_{\boldsymbol{\mu}} f_i = \nabla_{\boldsymbol{\mu}} f(\boldsymbol{\mu}^{(i)})$. This matrix is symmetric and positive semidefinite, so it admits a real eigenvalue decomposition

$$\boldsymbol{\Sigma} = \mathbf{W} \boldsymbol{\Lambda} \mathbf{W}^T, \tag{20}$$

where $\mathbf{W}$ is a $m \times m$ column matrix of eigenvectors, and $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues. Then we order the eigenpairs in descending order. We will select the first M eigenvectors to form a reduced-order basis. This is where we reduce the dimensionality of the design problem. We will attempt to describe the behaviour of the objective function by projecting the full-space design variables onto this active subspace. On average, perturbations in the first set of coordinates change f more than perturbations in the second set of coordinates. Low eigenvalues suggest that the corresponding vector is in the nullspace of the covariance matrix, and to form an approximation we can discard these vectors. We select the basis as follow

$$\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\Lambda}_1 & \\ & \boldsymbol{\Lambda}_2 \end{bmatrix}, \qquad \mathbf{W} = \begin{bmatrix} \mathbf{W}_1 & \mathbf{W}_2 \end{bmatrix}, \tag{21}$$

where $\boldsymbol{\Lambda}_1 = diag(\lambda_1, \ldots, \lambda_M)$ with $M < m$, and $\mathbf{W}_1$ contains the first M eigenvectors. The active subspace is the span of the first few eigenvectors of the covariance matrix. We define the active variables to be the linear combinations of the input parameters with weights from the important eigenvectors. In particular we define the active subspace to be the range of $\mathbf{W}_1$. The inactive subspace is the range of the remaining eigenvectors in $\mathbf{W}_2$. With the basis identified, we can map forward to the active subspace. So $\boldsymbol{\mu}_M$ is the active variable and $\boldsymbol{\nu}$ the inactive one, respectively:

$$\boldsymbol{\mu}_M = \mathbf{W}_1^T \boldsymbol{\mu} \in \mathbb{R}^M, \qquad \boldsymbol{\nu} = \mathbf{W}_2^T \boldsymbol{\mu} \in \mathbb{R}^{m-M}. \tag{22}$$

In particular any point in the parameter space $\boldsymbol{\mu} \in \mathbb{R}^m$ can be expressed in terms of $\boldsymbol{\mu}_M$ and $\boldsymbol{\nu}$:

$$\boldsymbol{\mu} = \mathbf{W} \mathbf{W}^T \boldsymbol{\mu} = \mathbf{W}_1 \mathbf{W}_1^T \boldsymbol{\mu} + \mathbf{W}_2 \mathbf{W}_2^T \boldsymbol{\mu} = \mathbf{W}_1 \boldsymbol{\mu}_M + \mathbf{W}_2 \boldsymbol{\nu}.$$

Having this decomposition in mind we can rewrite f

$$f(\boldsymbol{\mu}) = f(\mathbf{W}_1 \boldsymbol{\mu}_M + \mathbf{W}_2 \boldsymbol{\nu}),$$

and construct a surrogate model g discarding the inactive variables

$$f(\boldsymbol{\mu}) \approx g(\mathbf{W}_1^T \boldsymbol{\mu}) = g(\boldsymbol{\mu}_M).$$

We underline that the size of the eigenvalue problem is the limiting factor. We need to compute eigenvalue decompositions with $m \times m$ matrices, where m is the dimension of the simulation, that is the number of inputs.

Active subspaces can be seen in the more general context of ridge approximation (see [37, 38]). In particular it can be proved that, under certain conditions, the active subspace is nearly stationary and it is a good starting point in optimal ridge approximation as shown in [39, 40].

6 Numerical results

In this section we present the results of the complete pipeline, presented in the previous sections and in Figure 1, applied to the DTMB 5415 hull.

The mesh is discretized with quadrilateral cells. The BEM uses bi-linear quadrilateral elements. This results in roughly 4000 degrees of freedom for each simulation realized. The high fidelity solver described in section 4 is implemented in WaveBEM [11] using the deal.II library [25].

The solver, after reading the CAD file of the deformed geometry, simulates the behaviour of the hull for 30 seconds. To further speedup the computations, the total resistance computed as in Eq. (17) is extrapolated to obtain the total resistance at the final, steady state regime, with an error in the order of 0.1%. In Figure 9 we can see the original simulation of the total resistance for the first 30 seconds and then the extrapolation we have done after a proper filtering of the data. We fit the maximums using the following function: $ae^{-bx} + c$. For the minimums we use $-ae^{-bx} + c$. Then we set the approximated wave resistance to the average of the two at infinity.

(a) Original wave resistance
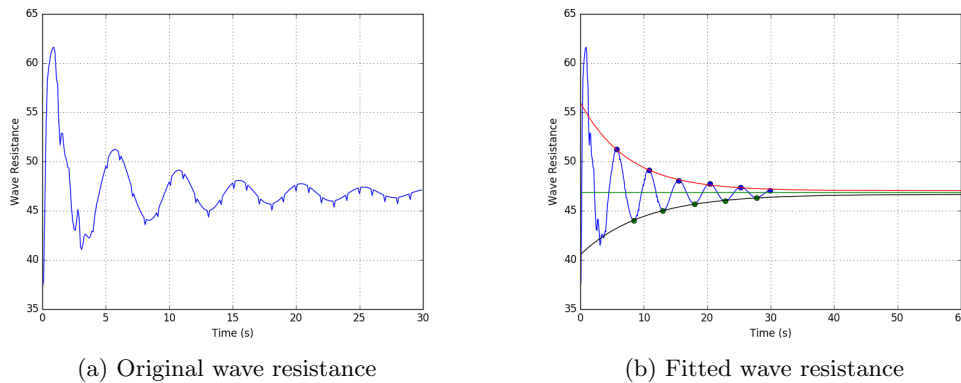
(b) Fitted wave resistance

Figure 9: Plot (a) shows the original wave resistance simulated for 30 seconds. Then plot (b) depicts, after a filter has been applied, the exponential fitting of the maximums and minimums and the average at regime for 60 seconds.

Let us recall that the parameter space is a $m = 8$ dimensional space. The parameters are showed in Table 1. We remark that the first six are geometrical parameters that produce the deformation of the original domain, while the last two are structural and physical parameters — the displacement and the velocity of the hull. The PyGeM open source package is used to perform the free form deformation [21].

We create a dataset with 130 different couples of input/output data. We split the dataset in two, creating a train dataset with 80% of the data, and a test dataset with the remaining 20%. That means that $N_{\text{train}}^{\text{AS}} = 104$ in Eq. (19). Even though it may be challenging to explore a 8 dimensional space, as reported in [1], heuristics suggest that this choice of $N_{\text{train}}^{\text{AS}}$ is enough for the purposes of the active subspaces identification described in section 5.

In order to construct the uncentered covariance matrix $\mathbf{\Sigma}$, defined in Eq. (18), we use a Monte Carlo method as shown in Eq. (19), employing the software in [41]. Since we have only pairs of input/output data we need to approximate the gradients of the total wave resistance with respect to the parameters, that is $\nabla_{\boldsymbol{\mu}} f$. We use a local linear model that approximates the gradients with the best linear approximation using 14 nearest neighbors. After constructing the matrix we calculate its real eigenvalue decomposition. Recalling section 5,since $m = 8$, we have that $\mathbf{\Sigma} \in \mathbb{M}(8, \mathbb{R})$.

In Figure 10a we see the eigenvalues of $\mathbf{\Sigma}$ and the bootstrap intervals. Bootstrapping is the practice of estimating properties of a quantity (such as its variance) by measuring those properties when sampling from an approximating distribution. It relies on random sampling with replacement. The bootstrap intervals in grey are computed using 1000 bootstrap replicates randomly extracted from the original gradient samples. We underline the presence of a major gap between the first and the second eigenvalue and a minor one between the second and the third. This suggests the existence of a one dimensional subspace and possibly the presence of a two dimensional one. To better investigate the first case, in Figure 10b we present the components of the eigenvector with index 1 that corresponds to the greatest eigenvalue, that is the matrix — in this case it is a vector — $\mathbf{W}_1 \in \mathbb{R}^8$ of Eq. (21). Since they are the weights of the linear combination used to construct the active direction they provide useful information. We can see that the major contributions are due to the velocity, the weight, and the depth of the hull. We underline that a weight that is almost zero means that the output function, on average, is almost flat along the direction identified by the corresponding parameter. This is a very useful information for a designer because in such a way he can deform the shape along prescribed directions without affecting the total wave resistance, allowing for example to store more goods inside the hull preserving the performances.

(a) Eigenvalues estimates.
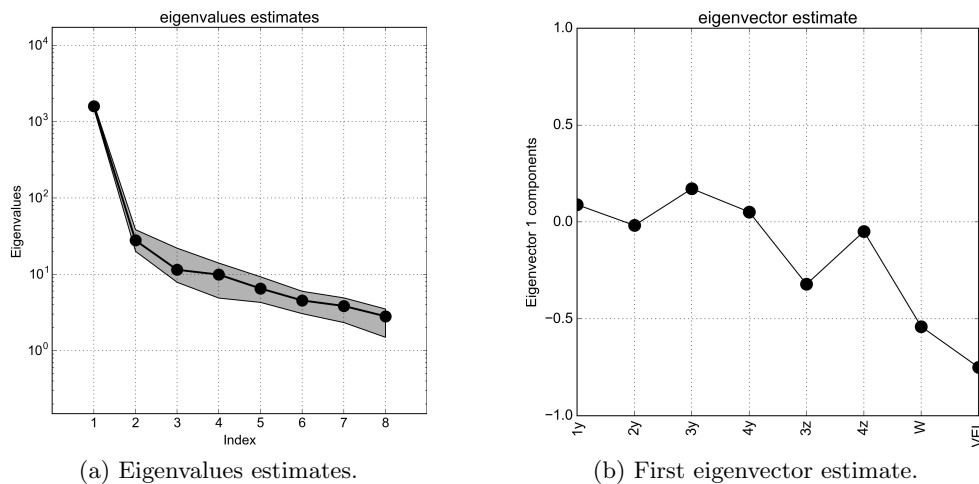
(b) First eigenvector estimate.

Figure 10: Plot (a) shows the eigenvalue estimates in block circles with the bootstrap intervals (grey region). The order-of-magnitude gaps between the eigenvalues suggest confidence in the dominance of the active subspace. Plot (b) shows the components of the eigenvector correspondent to the greatest eigenvalue.

In Figure 11 we explore the training dataset, plotting the sufficient summary plot (see [42]) for one and two active variables. Sufficient summary plots are powerful visualization tools to identifying low-dimensional structure in a quantity that depends on several input variables. A scatter plot that contains all available regression information is called a sufficient summary plot. Recalling Eq. (22), Figure 11 shows $f(\boldsymbol{\mu})$ against $\boldsymbol{\mu}_M = \mathbf{W}_1^T \boldsymbol{\mu}$, where $\mathbf{W}_1$ contains the first one and two eigenvectors respectively. In particular each point represents the value of the output function for a particular choice of the parameters, mapped in the active subspace. The two plots confirm the presence of an active subspace of dimension one and two. The latter seems to capture the output function in a much finer way, but as we are going to show the gain in terms of error committed is not so big.

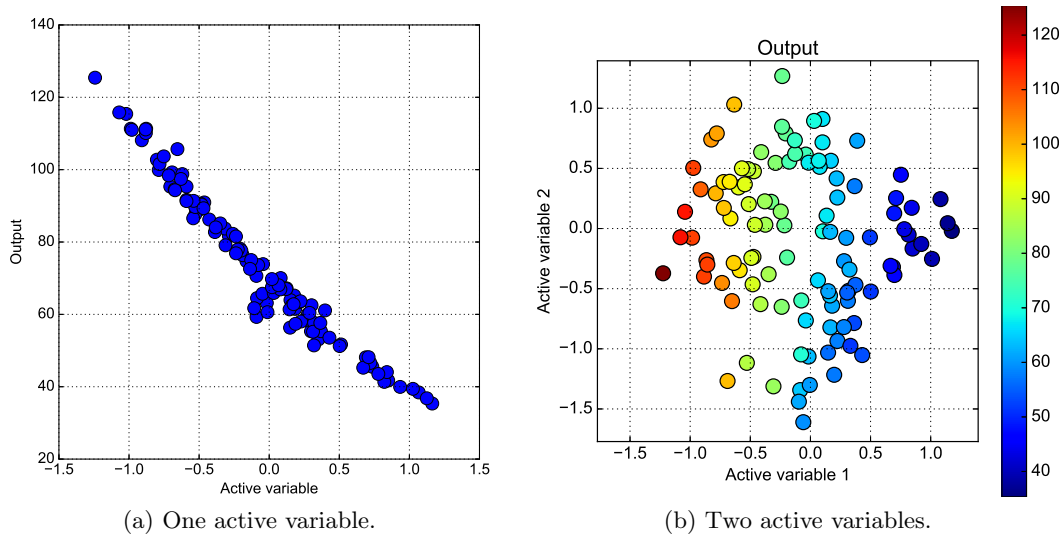(a) One active variable. (b) Two active variables.

Figure 11: Sufficient summary plots for (a) one and (b) two active variables using the training dataset.

We can compare the decay of the eigenvalues with the decay of surrogate model error on the test dataset shown in Figure 12a. We project the data onto active subspaces of varying dimension, and construct a surrogate model with a least-squares-fit, global, multivariate polynomial approximation of different orders. Then we calculate the root-mean-square error of the test data against the surrogate. This error is scaled with respect to the range of the function evaluations, making it a relative error. We repeat this procedure 20 times constructing every time the uncentered covariance matrix of Eq. (19), since a Monte Carlo approximation is involved. Finally we take the average of the errors computed. Because we have a large amount of training data, we can expect the surrogate model constructed in a low dimension to be accurate if the data collapses into a manifold. Thus the test error is an indication of how well the active subspace has collapsed the data. In Figure 12a are depicted the errors with respect to the active subspace dimension and the order of the response surface. The subspace dimension varies from 1 to 3, while the order of the response surface from 1 to 4.

The minimum error is achieved using a two dimensional active variable and a response surface of degree 4 and it is $\approx 2.5\%$. Further investigations show that increasing the dimension of the active variable does not decrease significantly the error committed while the time to construct the corresponding response surface increases. This is confirmed by the marginal gains in the decay of the eigenvalues for active variables of dimension greater then three as shown in Figure 10a. We can affirm that the active subspace of dimension one is sufficient to model the wave resistance of the DTMB 5415 if we can afford an error of approximately 4.5%. In particular in Figure 12b we can see the predictions made with the surrogate model of dimension one and the actual observations. Otherwise, we can achieve a $\approx 2.5\%$ error if we take advantage of two active dimensions and a response surface of order four, preserving a fast evaluation of the surrogate model.
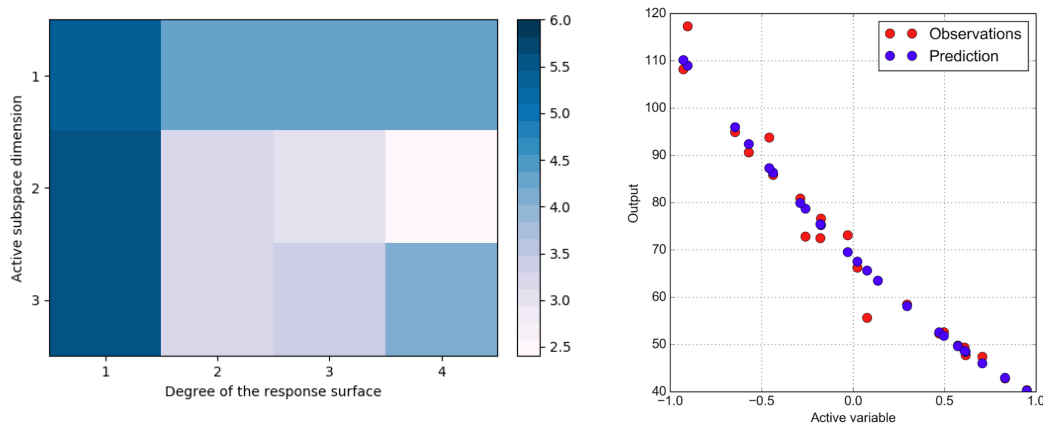
(a) Surrogate model error with respect to the active subspace dimension and the order of the response surface.

(b) Observations and the corresponding predictions using a polynomial response surface of order three and dimension one.

Figure 12: Plot (a) shows the relative error on the test dataset with respect to the active subspace dimension and the order of the response surface; plot (b) shows the observations of the test dataset and the corresponding predictions using a polynomial response surface of order three.

We want to stress the fact that the result is remarkable if we consider the heterogeneous nature of all the parameters involved. In the case of only geometrical parameters one can easily expect such a behaviour but considering also physical and structural ones make the result not straightforward at all. Moreover the evaluation of the response surface takes less than one second compared to the 12 hours of a full simulation per single set of parameters on the same computing machine. This opens new potential approaches to optimization problems.

7   A graphical decision support system

We incorporated all the present work in an interactive webapp based on the DTMB 5415. In Figure 13 there is the GUI of the app, where the user can move the FFD control points, change the weight and the Froude number, that is all the parameters defined in Table 1. The app deforms the hull in real-time and predicts the wave resistance using the methodologies presented in the previous sections. The user can see a preview of the deformed hull from the side, front, and back. In Figure 14 we can see the different views. Then it is also possible to download the deformed CAD. Using this tool allows for a quick and accurate prediction of the wave resistance while assessing the different geometrical configurations.
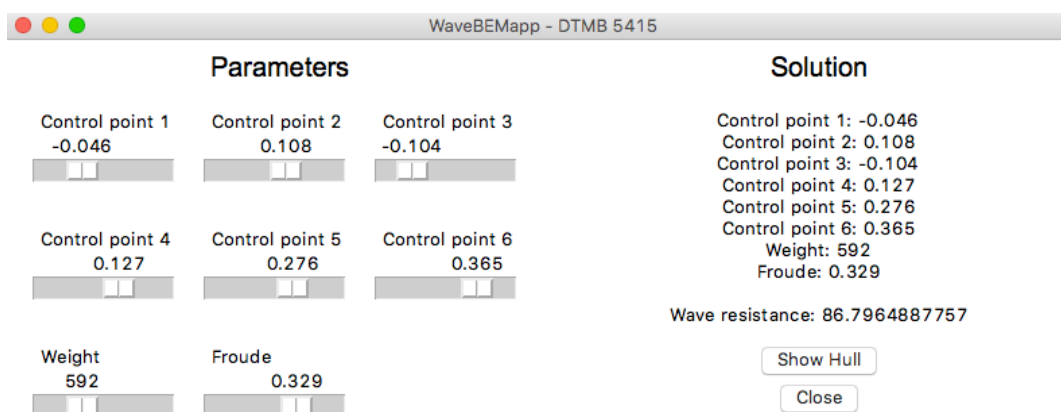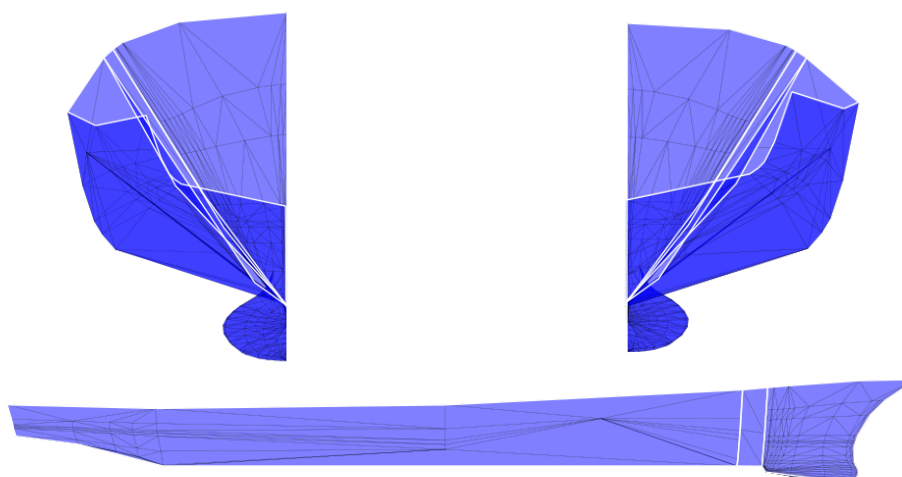

Figure 13: WaveBEM app GUI.

Figure 14: WaveBEM app views of the deformed hull. From left to right we have front, back, and side view.

8 Conclusions and perspectives

In this work we presented a numerical framework for the reduction of the parameter space and the construction of an optimized response surface to calculate the total wave resistance of the DTMB 5415 advancing in calm water. We integrate heterogeneous parameters in order to have insights on the more important parameters. The reduction both in terms of cost and time, remaining below the 4.2% of error on new unprocessed data, is very remarkable and promising as a new design interpreted tool. The methodological and computational pipeline is also easily extensible to different hulls and/or different parameters. This allows a fast preprocessing and a very good starting point to minimize the quantities of interest in the field of optimal shape design.

This work is directed in the development of reduced order methods (ROMs) and efficient parametric studies. Among others we would like to cite [43, 44, 45, 20] for a comprehensive overview on ROM and geometrical deformation. Future developments involve the application of the POD after the reduction of the parameter space through the active subspaces approach.

References

[1] Constantine P. G. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*, volume 2 of *SIAM Spotlights*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2015.

[2] Diez M., Campana E. F., and Stern F. Design-space dimensionality reduction in shape optimization by Karhunen–Loève expansion. *Computer Methods in Applied Mechanics and Engineering*, 283:1525–1544, 2015.

[3] Tahara Y., Kobayashi H., Kandasamy M., He W., Peri D., Diez M., Campana E. F., and Stern F. CFD-based multiobjective stochastic optimization of a waterjet propelled high speed ship. In *29th Symposium on Naval Hydrodynamics. Gothenburg, Sweden, 26-31 August, 2012*. Office of Naval Research, 2012.

[4] Volpi S., Diez M., Gaul N. J., Song H., Iemma U., Choi K. K., Campana E. F., and Stern F. Development and validation of a dynamic metamodel based on stochastic radial basis functions and uncertainty quantification. *Structural and Multidisciplinary Optimization*, 51(2):347–368, 2014.

[5] Dambrine J., Pierre M., and Rousseaux G. A theoretical and numerical determination of optimal ship forms based on Michell's wave resistance. *ESAIM: Control, Optimisation and Calculus of Variations*, 22(1):88–111, 2016.

[6] Sederberg T. W. and Parry S. R. Free-form deformation of solid geometric models. *ACM SIGGRAPH Computer Graphics*, 20(4):151–160, 1986.

[7] Olivieri A., Pistani F., Avanzini A., Stern F., and Penna R. Towing tank experiments of resistance, sinkage and trim, boundary layer, wake, and free surface flow around a naval combatant INSEAN 2340 model. Technical Report 421, Iowa Institute of Hydraulic Research (IIHR), 2001.

[8] Stern F., Longo J., Penna R., Olivieri A., Ratcliffe T., and Coleman H. International collaboration on benchmark CFD validation data for surface combatant DTMB model 5415. In *Twenty-Third Symposium on Naval Hydrodynamics. Val de Reuil, France, 17-20 September, 2000*. Office of Naval Research, 2001.

[9] Mola A., Heltai L., and De Simone A. A stable and adaptive semi-lagrangian potential model for unsteady and nonlinear ship-wave interactions. *Engineering Analysis with Boundary Elements*, 37(1), 2013.

[10] Mola A., Heltai L., and A. De Simone. Ship sinkage and trim predictions based on a CAD interfaced fully nonlinear potential model. In *The Proceedings of the 26th International Ocean and Polar Engineering Conference. Rhodes, Greece, 26 June â€" 2 July, 2016*, volume 3, pages 511–518. International Society of Offshore and Polar Engineers (ISOPE), 2016.

[11] Mola A., Heltai L., and De Simone A. Wet and dry transom stern treatment for unsteady and nonlinear potential flow model for naval hydrodynamics simulations. *Journal of Ship Research*, 61(1):1–14, 2017.

[12] Beck R. F. Time-domain computations for floating bodies. *Applied Ocean Research*, 16:267–282, 1994.

[13] Azcueta R. Computation of turbulent free-surface flows around ships and floating bodies. Schriftenreihe Schiffbau Bericht Nr. 612, Technische Universität Hamburg-Harburg, June 2001. ISBN: 3-89220-612-0.

[14] Formaggia L., Mola A., Parolini N., and Pischiutta M. A three-dimensional model for the dynamics and hydrodynamics of rowing boats. *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology*, 224(1):51–61, 2010.

[15] Shoemake K. Animating rotation with quaternion curves. *ACM SIGGRAPH Computer Graphics*, 19(3):245–254, 1985.

[16] Lombardi M., Parolini N., Quarteroni A., and Rozza G. Numerical simulation of sailing boats: Dynamics, FSI, and shape optimization. In Buttazzo G. and Frediani A., editors, *Variational Analysis and Aerospace Engineering: Mathematical Challenges for Aerospace Design*, pages 339–377. Springer, 2012.

[17] Rozza G., Koshakji A., and Quarteroni A. Free Form Deformation techniques applied to 3D shape optimization problems. *Communications in Applied and Industrial Mathematics*, 4, 2013.

[18] Salmoiraghi F., Ballarin F., Heltai L., and Rozza G. Isogeometric analysis-based reduced order modelling for incompressible linear viscous flows in parametrized shapes. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1):21, 2016.

[19] Forti D. and Rozza G. Efficient geometrical parametrisation techniques of interfaces for reduced-order modelling: application to fluid-structure interaction coupling problems. *International Journal of Computational Fluid Dynamics*, 28(3–4):158–169, 2014.

[20] Salmoiraghi F., Scardigli A., Telib H., and Rozza G. Free-form deformation, mesh morphing and reduced-order methods: enablers for efficient aerodynamic shape optimisation. *International Journal of Computational Fluid Dynamics*, 32(4–5):233–247, 2018.

[21] PyGeM: Python Geometrical Morphing. https://github.com/mathLab/PyGeM.

[22] Giuliani N., Mola A., Heltai L., and Formaggia L. FEM SUPG stabilisation of mixed isoparametric BEMs: application to linearised free surface flows. *Engineering Analysis with Boundary Elements*, 8–22:59, 2015.

[23] Brebbia C. A. *The Boundary Element Method for Engineers*. Pentech Press, London, 1978.

[24] Bangerth W., Hartmann R., and Kanschat G. deal.II – A general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software*, 33(4):24, 2007.

[25] Bangerth W., Davydov D., Heister T., Heltai L., Kanschat G., Kronbichler M., Maier M., Turcksin B., and Wells D. The deal.II library, version 8.4. *Journal of Numerical Mathematics*, 24(3):135–141, 2016.

[26] Hindmarsh A. C., Brown P. N., Grant K. E., Lee S. L., Serban R., Shumaker D. E., and Woodward C. S. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software*, 31(3):363–396, 2005.

[27] Morrall A. 1957 ittc model-ship correlation line values of frictional resistance coefficient. Technical Report 142, National Physical Laboratory, Ship Division, 1970.

[28] Lukaczyk T. W., Constantine P. G., Palacios F., and Alonso J. J. Active subspaces for shape optimization. In *10th AIAA Multidisciplinary Design Optimization Conference*, National Harbor, MD, USA, 2014. AIAA Paper 2014-1171.

[29] Constantine P. G., Emory M., Larsson J., and Iaccarino G. Exploiting active subspaces to quantify uncertainty in the numerical simulation of the HyShot II scramjet. *Journal of Computational Physics*, 302:1–20, 2015.

[30] Jefferson J. L., Gilbert J. M., Constantine P. G., and Maxwell R. M. Reprint of: Active subspaces for sensitivity analysis and dimension reduction of an integrated hydrologic model. *Computers & Geosciences*, 90:78–89, 2016.

[31] Tezzele M., Ballarin F., and Rozza G. Combined parameter and model reduction of cardiovascular problems by means of active subspaces and POD-Galerkin methods. In Boffi D., Pavarino L. F., Rozza G., Scacchi S., and Vergara C., editors, *Mathematical and Numerical Modeling of the Cardiovascular System and Applications*, volume 16 of *SEMA-SIMAI Series*, pages 185–207. Springer International Publishing, 2018.

[32] Box G. E. P. and Draper N. R. *Empirical Model-Building and Response Surfaces*. Wiley, New York, 1987.

[33] Kaipio J. P. and Somersalo E. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer Science & Business Media, 2006.

[34] Devore J. L. *Probability and Statistics for Engineering and the Sciences*. Cengage Learning, Andover, UK, 2015.

[35] Metropolis N. and Ulam S. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

[36] Constantine P. G. and Gleich D. Computing active subspaces with Monte Carlo. arXiv:1408.0545 [math.NA], 2015.

[37] Pinkus A. *Ridge functions*, volume 205 of *Cambridge Tracts in Mathematics*. Cambridge University Press, 2015.

[38] Keiper S. Analysis of generalized ridge functions in high dimensions. In *2015 International Conference on Sampling Theory and Applications (SampTA)*, pages 259–263. IEEE, 2015.

[39] Constantine P. G., Eftekhari A., Hokanson J., and Ward R. A near-stationary subspace for ridge approximation. arXiv:1606.01929 [math.NA], 2016.

[40] Hokanson J. M. and Constantine P. G. Data-driven polynomial ridge approximation using variable projection. arXiv:1702.05859 [math.NA], 2017.

[41] Constantine P. G., Howard R., Glaws A., Grey Z., Diaz P., and Fletcher L. Python active-subspaces utility library. *Journal of Open Source Software*, 1(5):79, 2016.

[42] Cook R. D. *Regression Graphics: Ideas for Studying Regressions through Graphics*, volume 482 of *Probability and Statistics*. Wiley, New York, 2009.

[43] Chinesta F., Huerta A., Rozza G., and Willcox K. Model reduction methods. In Stein E., de Borst R., and Highes T. J. R., editors, *Encyclopedia of Computational Mechanics*. Wiley, New York, second edition, 2017.

[44] Hesthaven J. S., Rozza G., and Stamm B. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. SpringerBriefs in Mathematics. Springer, Cham, Switzerland, 2016.

[45] Salmoiraghi F., Ballarin F., Corsi G., Mola A., Tezzele M., and Rozza G. Advances in geometrical parametrization and reduced order models and methods for computational fluid dynamics problems in applied sciences and engineering: overview and perspectives. In Papadrakakis M., Papadopoulos V., Stefanou G., and Plevris V., editors, *Proceedings of the VII European Congress on Computational Methods in Applied Sciences and Engineering,* Crete, Greece, 5-10 June 2016, volume 1, pages 1013–1031. ECCOMAS, 2016.