Reproducible industrial multifidelity optimization benchmark problems for air, space, and sea vehicles

Domenico Quagliarella
Italian Aerospace Research Centre
ITALY

Daniel Clark, Dean Bryson, Philip Beran
Air Force Research Laboratory
UNITED STATES

Andrew Thelen
University of Dayton
UNITED STATES

Laura Mainini
Politecnico di Torino
ITALY

Sihmehmet Yildiz, Melike Nikbay
Istanbul Technical University
TURKEY

Edmondo Minisci
University of Strathclyde
UNITED KINGDOM

Penelope Leyland
École Polytechnique Fédérale de Lausanne
SWITZERLAND

Andrea Serani, Matteo Diez
National Research Council
Institute of Marine Engineering
ITALY

Keywords: AVT-354 Workshop, multifidelity optimization, industrial benchmark problems

ABSTRACT

This article focuses on defining benchmarks of industrial interest for design problems in air, space, and sea vehicles. The intent is to create engineering benchmarks that are usable as a black box by the developers of multifidelity optimization methods to allow an analysis of problems as close as possible to the real world. The benchmarks and the chain of parametric generators of shapes and solvers developed in the framework of NATO STO-AVT-331 are presented in detail. NATO group members will use the benchmark problems and related computation chains to calibrate and test their multifidelity optimization procedures. A reference optimum is also reported for each benchmark, and its characterizing aspects are discussed in detail.

1 INTRODUCTION

One of the aspects that are often critical in the development of a new multifidelity design methodology is the lack of a set of test problems that can effectively be used for performance analysis and comparison. Indeed, the definition of a benchmark problem for a multidisciplinary design problem, even a very complex one, is fundamentally limited to the definition of a baseline configuration, a parameterization methodology and a set of operating conditions, eventually complemented by the definition of suitable physical/computational models for the multidisciplinary analysis. Conversely, a benchmark problem for a multifidelity methodology needs to

specify the hierarchy of methodologies and physical/computational models used when searching for an optimal solution. Consider, for example, how difficult can be the comparison of two different multifidelity optimization methodologies without considering the effect that different computational meshes or numerical solvers may have on the results.

Beran et al. [1] propose the use of three types of benchmark problems for the assessment of multifidelity methods for vehicle applications, characterized by different levels of complexity. The most simple problems (referred to as L1) are analytic functions with exact evaluations, negligible computational cost, and a little adherence to engineering problems. Despite their simplicity, these problems are useful in the initial development of multifidelity methodologies, for timely assessment and comparison, and as demonstrators of proof of concept. *Intermediate-complexity problems* (referred to as L2) use ordinary or partial differential equations solutions, have small to moderate numerical errors/uncertainties, a moderate computational cost, and are idealizations or real engineering problems. L2 benchmark problems represent a simpler/idealized version of more complex multifidelity problems and are useful to challenge and compare various multifidelity methods in a more realistic framework. The computational cost of L2 problems should be sufficiently low to address the algorithmic challenges to a level of detail not possible for more expensive problems. Finally, L3 benchmark problems are the most complex, use partial differential equations solutions, are affected by moderate to large numerical errors/uncertainties, have a high computational cost, and address vehicle-level engineering models. Their solution in a timely and affordable way is the final goal of multifidelity approaches.

The objective of this work is to present the set of candidate intermediate-complexity problem (L2) for air, space, and sea problems that have been proposed in the NATO STO-AVT-331 working group on "Goal-driven, multifidelity approaches for military vehicle system-level design". These benchmarks are based on a standard set of analysis modules at different levels of fidelity. These modules are open-source and easily connectable between them to build L2 benchmarks on which to test innovative multifidelity methodologies. The advantage, compared to the use of classic analytical benchmark functions (L1), is that new multifidelity methodologies can be tested and compared using a set of standard problems very close in nature and challenges to real-world applications (L3).

Some of the open-source modules presented here were previously assembled for benchmarking uncertainty quantification methods [2] within the activities of NATO STO-AVT-252 (Stochastic Design Optimization for Naval and Aero Military Vehicles). The current work is performed mostly within NATO STO-AVT-331 and presents extensions to multifidelity approaches, embedding different physical modeling and discretization levels.

The remainder of this paper is organized as follows. Sections 2, 3, and 4 introduce and discuss the air, space, and sea benchmarks, respectively. Each of these sections provides the problem formulation, along with the descriptions of the design space parameterization and dimensionality, numerical solvers, problem discretization and analysis, reference optimum, as well as software archive and documentation details. Finally, summary and conclusions are given in section 5.

2 AIR BENCHMARK

The open-source software modules that define the air L2 benchmark include (a) airfoil shape parameterization, (b) automatic meshing, and (c) multifidelity aerodynamic solvers. These building blocks allow the resolution of an aerodynamic shape design optimization problem using a standardized and common set of objective func-

tion evaluators among all the AVT-331 partners that want to compare the efficiency and effectiveness of their multifidelity approaches to design optimization.

The proposed common air L2 problem is the shape optimization problem of an RAE-2822 airfoil under transonic conditions. A preliminary version of this problem is reported in [3] along with further details on airfoil shape handling, mesh generation and flow field computation.

The baseline configuration is the classical airfoil working at in fully turbulent flow conditions modeled using the Spalart-Allmaras turbulence model [4].

2.1 Problem Formulation

The operating conditions of the airfoil are Mach $M = 0.734$ and Reynolds $R = 6.5 \times 10^6$, and the design optimization problem definition is the following:

$$\begin{array}{ll} \min\limits_{\boldsymbol{w} \in W \subseteq \mathbb{R}^n} & c_d \\ \text{subject to:} & c_l = 0.824 \\ & -0.11 \leq c_m \leq -0.01 \\ & t/c = 0.1211 \\ & r \geq 0.007c \\ & \tau \geq 5^\circ \\ & t_{85}/c \geq 0.02 \end{array} \tag{1}$$

where w is the design variable vector, and the minimization objective is the drag coefficient c_d. The constraint on lift coefficient (c_l) is satisfied using the angle of attack α as a free parameter and is automatically handled by the flow analysis driver. Constraint handling is part of the free parameters of the multifidelity optimization method and can hence be freely managed by each partner. In the proposed design problem, constraints are imposed on pitching moment c_m, maximum thickness to chord ratio t/c, leading edge radius r, trailing edge angle τ, and thickness at 85% of the chord t_{85}/c. However, in the case of penalty-based methods, we suggest the following formulation:

$$\begin{aligned} \min_{\boldsymbol{w} \in W \subseteq \mathbb{R}^n} c_d \quad + \quad & k_1 p^+(c_m, -0.1) + k_1 p^-(c_m, -0.01) + \\ & k_2 p^+(r/c, 0.007) + k_3 p^+(\tau, 5) + k_4 p^+(t_{85}/c, 0.02) \end{aligned} \tag{2}$$

where, according to the quadratic penalty definition:

$$p^+(x, y) = \begin{cases} 0 & \text{if} \quad x \geq y \\ (x - y)^2 & \text{if} \quad x < y \end{cases} \tag{3}$$

and

$$p^-(x, y) = \begin{cases} 0 & \text{if} \quad x \leq y \\ (x - y)^2 & \text{if} \quad x > y \end{cases} \tag{4}$$

with $k_1 = 1000$, $k_2 = 5000$, $k_3 = 10$ and $k_4 = 30$. The constraint on the maximum thickness does not appear because the shape handler automatically satisfies it by scaling the airfoil after its parametric modification.

2.2 Design Parameterization and Design Space Dimensionality

The working group partners can choose between three predefined sets of design variables characterized, respectively, by 6, 10 or 20 design variables, each linked to a different shape function that operates either on upper or lower side of the airfoil. Up to 32 shape functions can be selected through the test case driver, namely 6 polynomials, 12 Hicks-Henne bumps, 2 Wagner functions and 12 arbitrary defined function using a file-based interface.

2.3 Numerical Solvers

The numerical analysis procedure allows managing the entire computational chain that goes from the generation of the airfoil according to the design parameters to the generation of the computational mesh, when required, to the launch of the aerodynamic solver. This software module is written in the `bash` and `awk` languages and interfaces with codes of geometric manipulation, grid generation, and fluid dynamics analysis written in `C++`, `C`, and `FORTRAN`.

The heart of the procedure is a bash script (`run_solver.sh`) that acts as an automated driver to the `SU2` Reynolds-averaged Navier–Stokes equations (RANS) solver, the `GMSH` mesh generator, and the `WG2AER` shape handler and calculates the aerodynamic characteristics of a wing section defined by a number of parameters that modify a starting airfoil, which can be specified in the input. A lighter version of this computational chain allows the use of the `XFOIL` potential panel solver with boundary layer correction instead of `SU2` and `GMSH`. The scheme of the software module is reported in Figure 1.

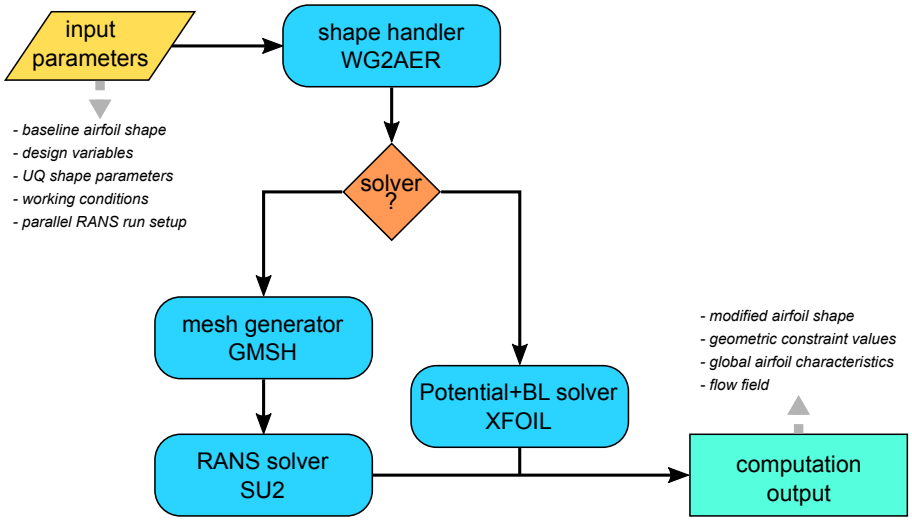

Figure 1: Air benchmark: software module scheme and computation data flow.

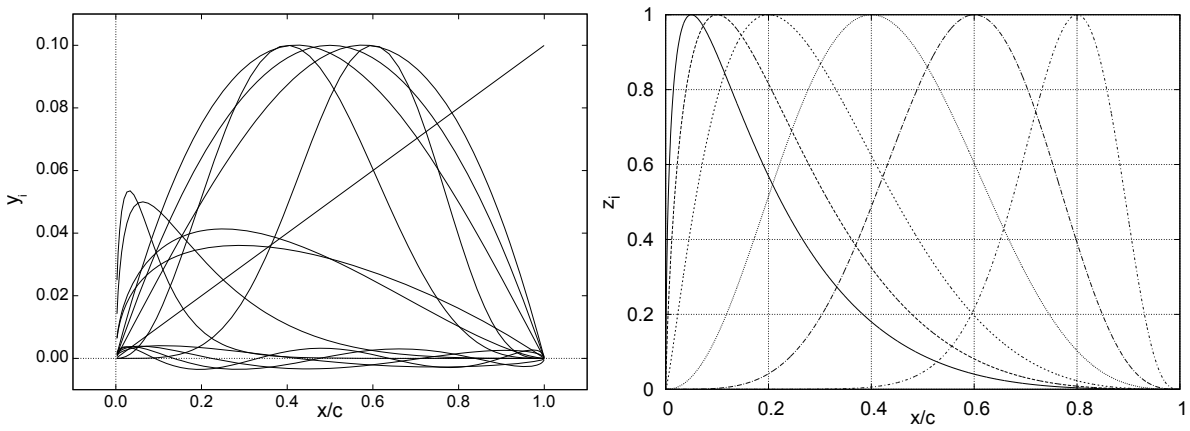

Figure 2: Air benchmark: modification functions to handle airfoil shape (left) and geometry uncertainty (right).

2.3.1 Shape Parameterization and Handling

The in-house developed `WG2AER` code parametrizes the airfoil as a linear combination of an initial geometry $(x_0(s), y_0(s))$, and some modification functions $y_i(s)$. Moreover, it is also possible to describe geometry uncertainties that are treated as further $z_i(s)$ modification functions (here Hicks-Henne bumps). So, the airfoil shape, including uncertainties, is described by

$$\begin{aligned} y(s) &= k\left(y_0(s) + \sum_{i=1}^{n} w_i y_i(s)\right) + \sum_{j=1}^{m} U_j z_j(s), \\ x(s) &= x_0(s), \\ z_j(s) &= \sin^3\left(\pi x^{\frac{\log 0.5}{\log s_{b_j}}}\right) \end{aligned} \tag{5}$$

where the airfoil shape is controlled by the design parameters w_i and by the scale factor k. The uncertainty on shape and thickness of the airfoil is described by the U_j random variables. The adopted modification functions are reported in Figure 2. Hence, the `run_solver.sh` script builds a parametric airfoil using Equation 5 and starting from the NACA 2412 baseline. Then, it computes the airfoil performance using an aerodynamic analysis code, namely either the `SU2` RANS code or the `XFOIL` potential flow with boundary layer correction code. The operating conditions (Mach number and Reynolds number) must be specified to calculate the aerodynamic performance. Besides, further parameters that have to be fixed are either the angle of attack, α, or, alternatively, the lift coefficient, c_l, and optionally, the maximum thickness of the airfoil, t/c.

2.3.2 Computational Tools for RANS Analysis

The numerical solution of the RANS equations using the finite volume method requires the generation of computational meshes with carefully specified characteristics. Indeed, in the context of optimization, it is not enough to generate a calculation grid for a single geometry. Furthermore, it is necessary to have an automatic procedure

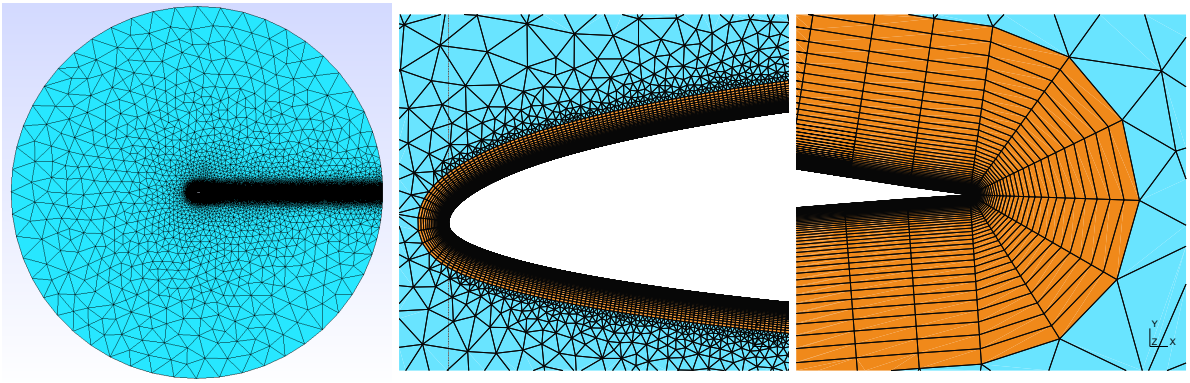

Figure 3: Air benchmark: mesh domain and close view of the airfoil front and rear grid areas.

able to generate a calculation grid for each new geometry that the optimizer needs to analyze. The implementation of this type of procedure is not trivial because it is necessary to find a compromise between calculation accuracy, fast Computational Fluid Dynamics (CFD) computation, and the robustness of the meshing procedure to minimize errors that could occur during the grid generation phase. The automated procedure presented here takes in input the airfoil points, some flow data, and some grid control parameters, and returns a hybrid grid composed of triangles and quadrangles in the format suitable for `SU2` solver. The grid generation engine is the `GMSH` v4 software [5] that is excellent in terms of efficiency and robustness and batch usage capabilities. The interface script to `GMSH` requires as input, basically, the coordinates of the airfoil, the dimensions of the fluid domain, the Reynolds number, and a reference length. According to input data, a hybrid grid with proper and accurate modeling of the boundary layer and the wake is obtained. An example of an automatically generated grid for RANS analysis (with about 160000 cells) is reported in Figure 3, in which the detail of the boundary layer zone and the treatment of the corner point at the trailing edge is evidenced in orange color.

The parametric mesh generation procedure uses the *Field* feature of `GMSH` to manage cell sizes and refinement zones, like the wake and the boundary layer.

The flow solver used to obtain RANS solutions is the open-source finite-volume code `SU2` v6.2.0 [6], conceived at Stanford University and steadily growing in diffusion in the aeronautical world thanks to a broad community of developers active all over the world. `SU2` is written in `C++` language and is characterized by a highly modular structure that is easily extensible for the solution of any set of Partial Differential Equations (`PDE`). The heart of the software is the parallel RANS solver capable of solving problems of interest to the mechanical and aerospace industry in the turbulent transonic regime.

In the procedure implemented in the present work, the "Spalart-Allmaras" turbulence model is used along with a II order *Monotone Upstream-Centered Scheme* for conservation law, and an adaptive Courant–Friedrichs–Lewy (`CFL`) number.

2.3.3 Computational Tool for Potential Plus Boundary Layer Analysis

In view of the construction of a system for benchmarking multiple fidelity algorithms, the wrapper interface allows the use of Prof. Drela's `XFOIL` code [7], which has a fidelity level and an application range not completely

overlapping with RANS. This code is based on a second order panel method interactively coupled to a boundary layer integral module. Laminar to turbulent flow transition is predicted using the e^N method. The input switches and variables are coincident for the parts related to the airfoil management and operating conditions setting, while the parameters that control mesh generation and RANS solver are, obviously, ignored. Two further parameters allow the introduction of constraints on laminar to turbulent flow transition: the constraints are defined using the variables `XTRUP` (boundary layer transition point on the upper airfoil surface) and `XTRLO` (boundary layer transition point on the airfoil lower surface). `XTRUP` set equal to 0.5, for example, means that the transition point cannot be set at a x/c greater than 0.5.

2.3.4 Input/Output Parameters

The input parameters of the procedure are specified in a single input file (`eval_obj.in`) with working conditions and shape controlling variables, including eventual uncertainties. An alternate baseline airfoil, if the user chooses to use an airfoil different from the hardwired NACA 2412 default may be used.

The results of the simulation are written in a single output file (`eval_obj.out`) containing:

1. ERROR flags that signal eventual problems of the mesh generator or the RANS solver.
2. Geometric features of the airfoil (leading-edge radius, trailing edge angle, maximum thickness, thickness at 85% chord)
3. Global aerodynamic characteristics (c_l, c_d, c_m).

Intermediate files can be optionally left in the run directory for debugging or visualizing purposes and examine the flow field and the computational mesh.

2.4 Problem Discretization and Analysis

The key feature of this software diver is that, for the type of RANS run, the fidelity level is controlled by varying the fineness of the computational mesh. This is controlled by a single input parameter, namely "ELEMENT_SCALE," and the effect of this parameter is illustrated in Figure 4. The larger the value of "ELEMENT_SCALE," the more coarse the mesh is. The values we have experimented with for the case in question vary from 2.5 (finest grid) to 40 (coarsest grid), and the results obtained with the RANS analysis differ significantly. Therefore a mesh convergence analysis was performed, which is reported below.

2.4.1 Grid Convergence Analysis

We performed a convergence study on the baseline (airfoil RAE-2822) with a sweep of ELEMENT_SCALE from 2.5 to 40. The driver had the constraint on c_l active and set at the design value. As shown in Figure 5, the fixed c_l procedure does not converge to acceptable values of the angle of attack for ELEMENT_SCALE> 10. Consequently, for fixed c_l runs in this design case, it is suggested to remain in a range between 2.5 and 10 for

ELEMENT_SCALE. In this range, the constraint on c_l is respected, and also c_m and c_d vary in an acceptable range for a multifidelity optimization method.

However, it should be noted that this behavior is presumably due to the presence of an intense shock wave on the back of the RAE-2822 under the specified conditions of c_l and that, presumably, in less demanding conditions, it is possible to use even more coarse grids.

Finally, runs with a fixed angle of attack are less demanding from the point of view of the mesh fineness than those with fixed lift. In this case, we have observed that the solution tends to converge more easily and, above all, much more quickly. Consequently, an appropriate multifidelity optimization strategy may still be able to exploit higher ELEMENT_SCALE values.

The results of the mesh convergence study are also confirmed as the design parameters that control the shape of the airfoil vary. In Figure 6, we have considered a variation of the first design variable between -0.5 and 0.5, and we observe that, while it is identifiable a trend for c_m and c_d with ELEMENT_SCALE set to 2.5, this is no longer verified for ELEMENT_SCALE equal to 20 and 40.

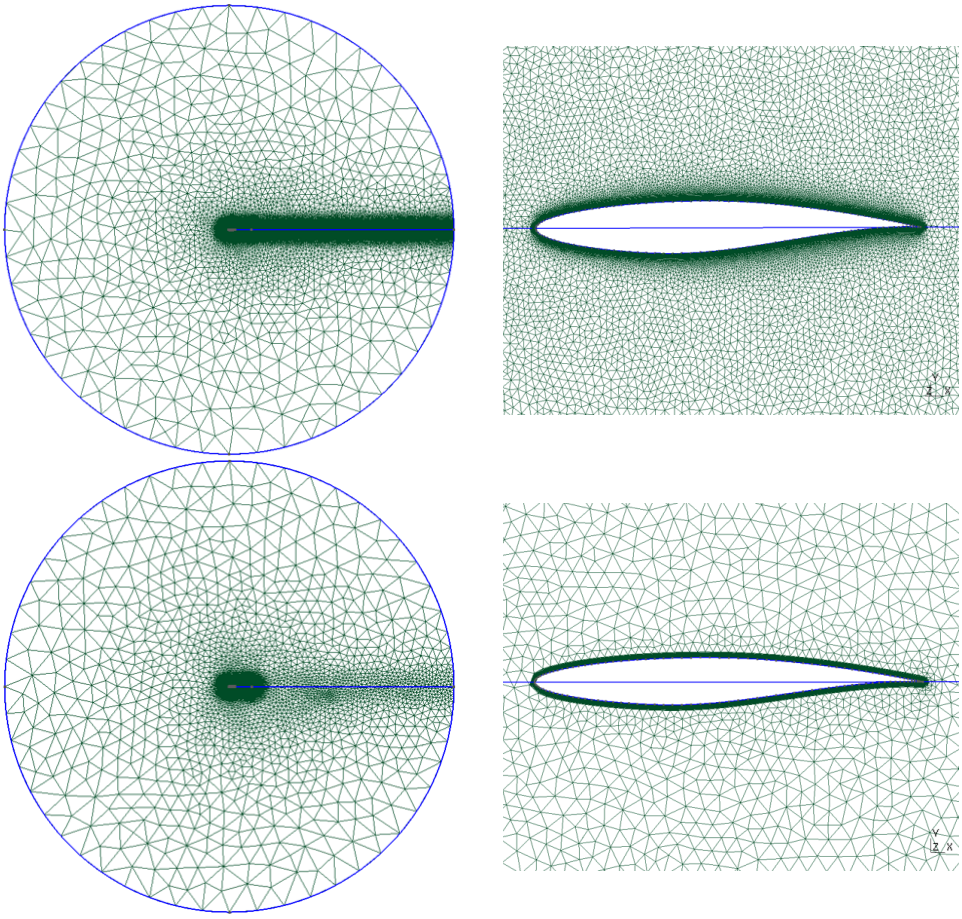

Figure 4: Air benchmark: mesh refinement levels.

2.4.2 High-Fidelity 2K Factorial Design

We performed a full-factorial experiment using the first three design variables in the range $[-0.5, 0.5]$ to study the effect of each design variable and their interactions on the aerodynamic objective and constraints. We set the solver at the highest fidelity level, and the full-factorial analysis required 2k aerodynamic evaluations.

This 2k Design of Experiments (DoE) captured c_d from 0.0171 to 0.0429 where the baseline was $c_{d_0} = 0.020527$. Figure 7 shows that fourteen DoE points have c_d less than c_{d_0}, and four of those are feasible $(\min c_d = 0.0178)$, achieving a drag reduction of 13%.

The analysis of this DoE suggests that a range between -0.5 and 0.5 for the design variables can lead to significant improvements in the objective function and can be safely used for response surface training as cases where the calculation fails are very few. However, as we will see below with the reference optimization run, some of these parameters can exceed the limits mentioned above in the optimum zone. This indication strongly suggests that when using response surfaces, it is appropriate to resort to adaptive training schemes that favor local refinement of the surrogate model.

2.5 Reference Optimum

The reference optimization run used the fluid dynamic solver only in high fidelity mode, with a grid characterized by approximately 150,000 cells. `SU2` v 6.2.0 ("Spalart-Allmaras" T. Mod.) [6] runs at c_l constant required 20,000 iterations (using the implicit solver) to achieve the accuracy needed and took an average elapsed time of about 140 minutes on 8 Intel (R) Xeon (R) cores E5-2697 v4 @ 2.30GHz. `SU2` CFL adaptation strategy is used with a lower limit of 2, and an upper one equal to 40.

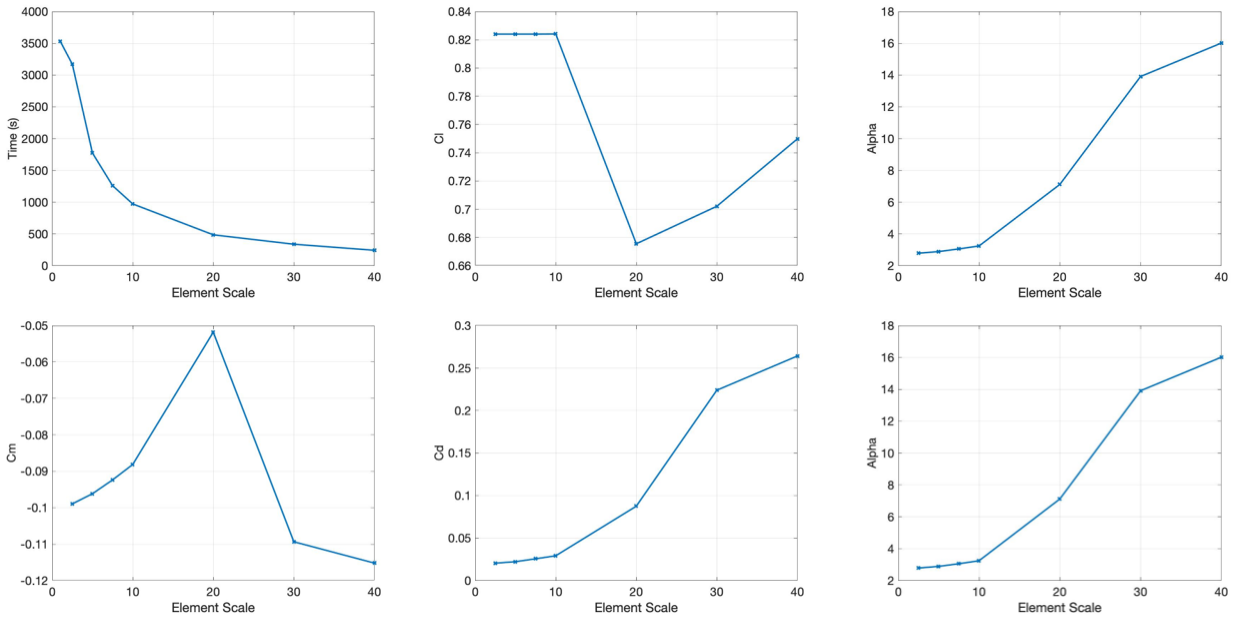

Figure 5: Air benchmark: mesh Convergence Study —- Element Scale 40 to 2.5.

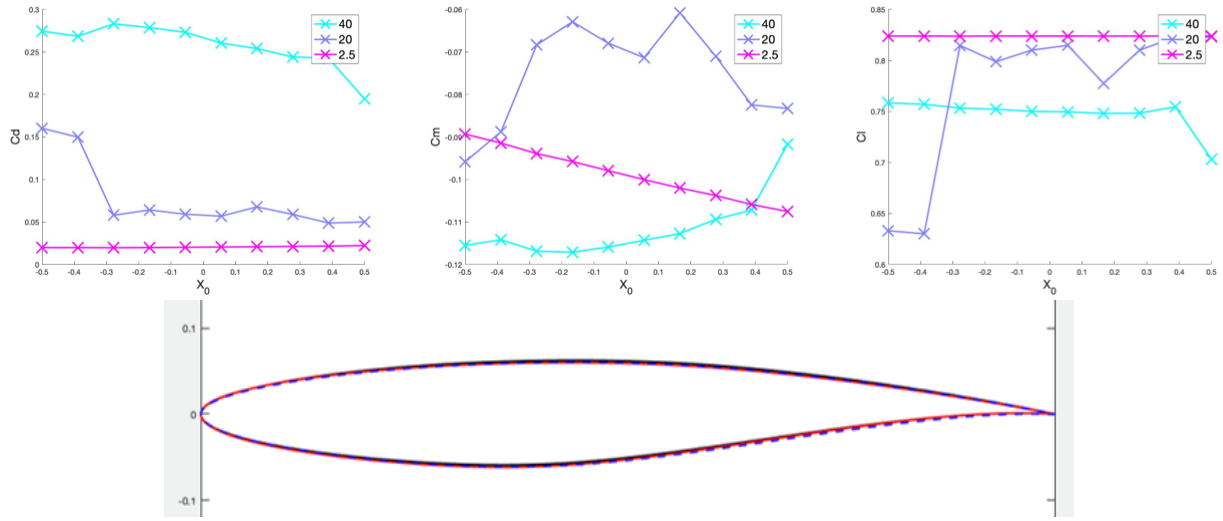

Figure 6: Air benchmark: parameter Sweep effect —- $x_0 \in [-0.5, 0.5]$ for different Element Scale values.

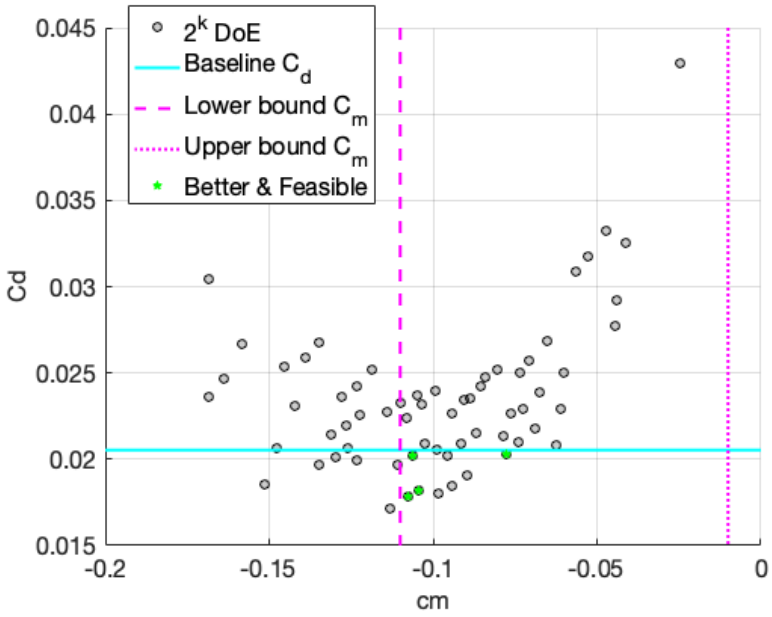

Figure 7: Air benchmark: high-Fidelity 2K factorial design.

The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [8] evolutionary strategy was chosen for the optimization. The parameters used for the CMA-ES optimization algorithm are the maximum number of allowed evaluations, the population size λ, and the initial standard deviation σ. The parameters set for this problem are reported in Table 1.

Twenty shape functions modify the airfoil shape (ten reserved to the upper side and ten to the lower side). In particular, they are 6 polynomials, 12 Hicks-Henne bumps and 2 Wagner functions. The allowed variation range

for each shape design variable is [-6.0, 6.0].

Thanks to the fact that the optimization algorithm evaluates the population elements fully in parallel, the evaluation of each generation also takes about 140 minutes using 128 cores.

Fig. 8 reports the baseline airfoil shape end the evolution history of the reference optimization run performed using the L2 driver in single, high-fidelity mode.

The design variable vector values for the initial and the optimal airfoil shapes are reported in table 2.

Table 3 shows a summary of the evaluation of the results of the reference optimization. The result obtained is particularly interesting in terms of single design point airfoil performance, although obtained at a relatively high computational cost.

Figure 9 show the comparison of the airfoils and the pressure coefficients between the baseline and the optimal solution found. Finally, Figure 10 shows the Mach number flow fields around the baseline RAE-2822 (left) and the optimal solution (right) that highlight a significant intensity reduction of the shock wave.

2.6 Software Archive and Documentation

The current benchmark problem is available at the GitLab repository of the AVT-331 working group. A public release is in preparation and will be available under the GPL license. The code is conceived to work under Linux operating system but should be easily adaptable to modern Windows releases using the Windows subsystem for Linux.

Table 1: Air benchmark: CMA-ES parameters for the reference high-fidelity optimization run.

Max. evaluations	λ	σ
3000	16	0.15

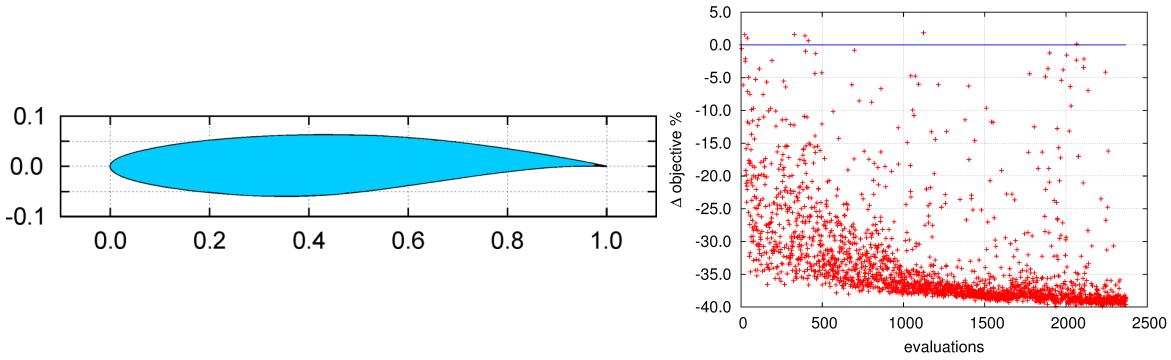

Figure 8: Air benchmark: baseline airfoil and evolution history.

Table 2: Air benchmark: design variable values for RAE-2822 and optimized airfoil shape.

DV	RAE-2822	Optimized
w_0	0.00000000	-0.79727763
w_1	0.00000000	0.29257670
w_2	0.00000000	0.29650784
w_3	0.00000000	0.09121300
w_4	0.00000000	1.11069288
w_5	0.00000000	-0.55537671
w_6	0.00000000	0.79334328
w_7	0.00000000	-0.04209589
w_8	0.00000000	-0.59810476
w_9	0.00000000	0.38285159
w_{10}	0.00000000	0.51927560
w_{11}	0.00000000	-0.17718765
w_{12}	0.00000000	-0.27731534
w_{13}	0.00000000	1.26317646
w_{14}	0.00000000	0.50190416
w_{15}	0.00000000	-0.08360092
w_{16}	0.00000000	-0.47016052
w_{17}	0.00000000	-0.73298168
w_{18}	0.00000000	-0.19085208
w_{19}	0.00000000	-0.36425107

Table 3: Air benchmark: summary of the optimization results.

airfoil	c_l	α	c_d	c_m	r	τ	t_{85}/c	objective
baseline RAE-2822	0.824	2.77	0.02048	-0.0997	0.0079	8.6808	0.0318	0.02048
best hi-fi opt.	0.824	1.29	0.01230	-0.1093	0.0085	8.6979	0.0223	0.01230

3 SPACE BENCHMARK

The proposed space L2 problem regards the shape optimisation of the IXV technology demonstrator vehicle [9]. The surface mesh of the vehicle is morphed via radial basis functions.

3.1 Problem Formulation

The design optimization problem is formulated as follows:

$$\begin{aligned} \min_{\boldsymbol{Lb} \leq \boldsymbol{x} \leq \boldsymbol{Ub}} \quad & -C_l/C_d \\ \text{subject to:} \quad & L \geq L_r \\ & \dot{Q}_{max} \leq \dot{Q}_{r,max} \\ & C_{V,1} = 0 \\ & C_{V,2} = 0 \end{aligned} \tag{6}$$

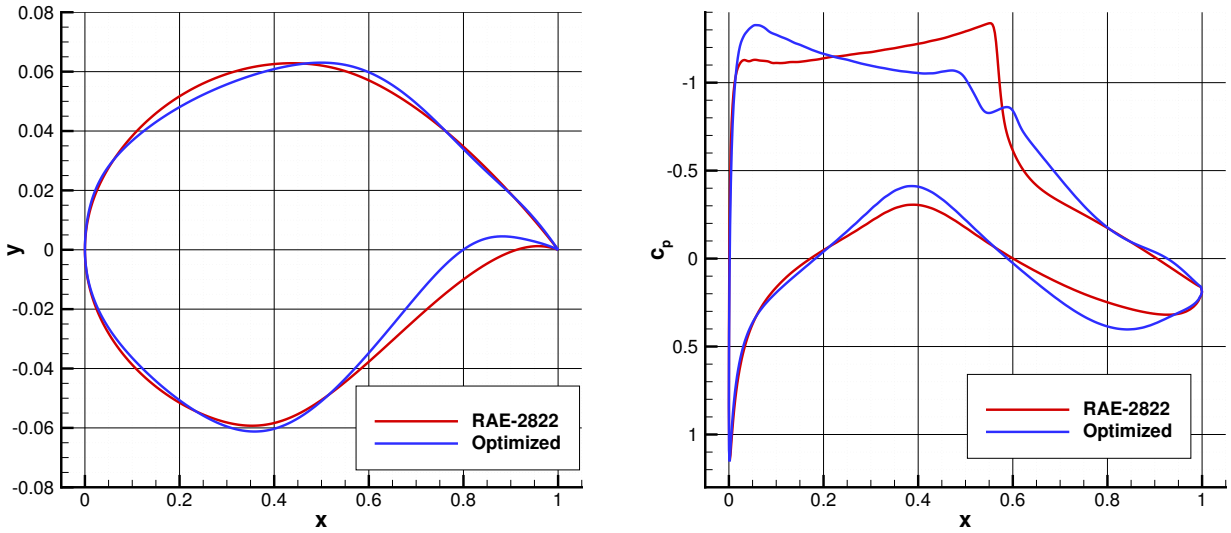

Figure 9: Air benchmark: comparison of baseline RAE-2822 and optimized airfoil shapes and pressure coefficients.

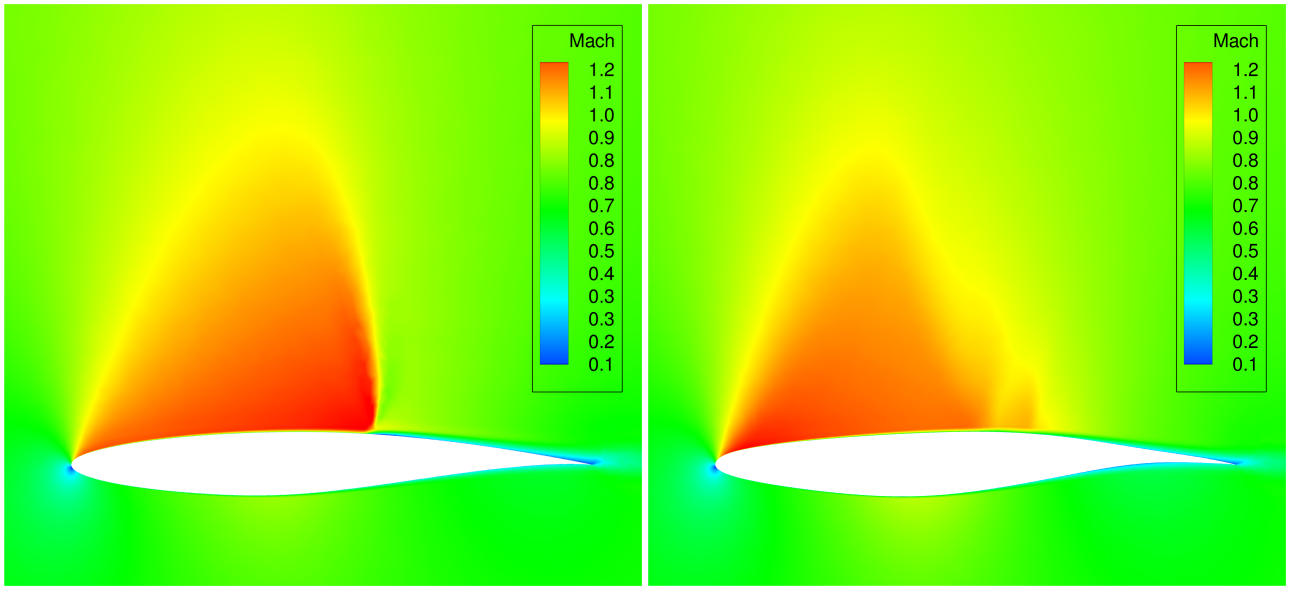

Figure 10: Air benchmark: comparison of baseline RAE-2822 (left) and optimized airfoil (right) mach fields.

where $\boldsymbol{x}$ are the applied morphing displacements in x, y, and z directions, considering only symmetric deformations, L is the lift aerodynamic force, L_r is the reference minimum lift ($1500N$), $DotQ_{max}$ is the maximum heat flux, $DotQ_{r,max}$ is the reference maximum heat flux ($550kW/m^2$), and $C_{V,1}$ and $C_{V,2}$ are two volume coefficients introduced to consider the presence of an internal bay area (a cylinder for this benchmark, see Fig. 11).

The volume indexes are computed as follows:

$$C_{V,1} = w_1 \frac{n_{exp}}{n_{tot}}$$
$$C_{V,2} = w_2 \frac{v_{exp}}{v_{tot}} \tag{7}$$

where where n_{exp} is the number of voxels center points of the bay outside the vehicle surface mesh, n_{tot} is the total number of voxels center points of the bay, v_{exp} is the number of bay/payload surface vertices outside the vehicle mesh, v_{tot} is the total number of bay/payload vertices, and w_1 and w_2 are weighting coefficients. To compute the voxels center points, the area surrounding the bay shape is voxelised and then all the voxels outside the bay volume are culled. Both the voxels based and the vertex based coefficients are used to have a more robust approach: due to nature of the voxelised representation of the bay, a step error introduced by the aliasing of surfaces could bring to an underestimation of the volume violation; on the other side, the surface mesh criterion depends on the bay mesh quality, and if the vertex density of the bay mesh differs significantly across its surface and low-density areas can be under-penalized.

The operating conditions of the vehicle are altitude $H = 70km$, velocity $V = 6500m/s$, angle of attach $\alpha = 10deg$.

3.2 Design Parameterization and Design Space Dimensionality

The morpher implements a radial-basis function approach which can have an arbitrary number of radial basis kernels to modify the surface mesh of the vehicle. The provided case considers 9 kernels with 15 parameters. Fig. 11 shows the baseline configuration and the kernels positioned at zero displacements.

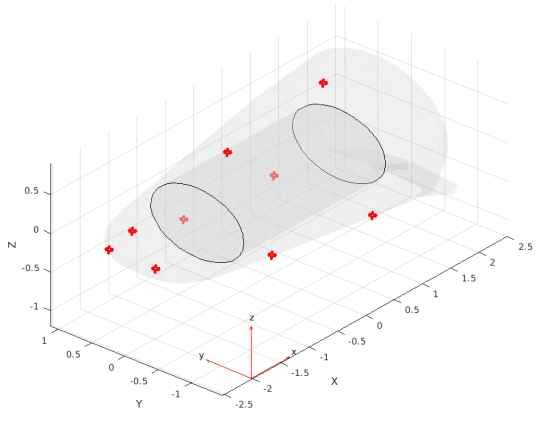

Figure 11: Space benchmark: baseline shape with control points at zero displacement, and the visualization of the internal cylindrical bay.

The position of the morphing kernels, M_p, is:

$$\begin{bmatrix} -2.402 & 0. & 0.364 \\ -2.402 & 0.349 & 0.004 \\ -0.717 & 0. & 0.696 \\ -0.717 & 0.659 & -0.400 \\ 0.969 & 0. & 0.909 \\ 0.969 & 0.740 & -0.548 \\ -2.402 & -0.349 & 0.004 \\ -0.717 & -0.659 & -0.400 \\ 0.969 & -0.740 & -0.548 \end{bmatrix} \tag{8}$$

and the displacement matrix, M_d, is:

$$\begin{bmatrix} x_1 & 0 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & 0 & x_7 \\ x_8 & x_9 & x_{10} \\ x_{11} & 0 & x_{12} \\ x_{13} & x_{14} & x_{15} \\ x_3 & -x_4 & x_5 \\ x_8 & -x_9 & x_{10} \\ x_{13} & -x_{14} & x_{15} \end{bmatrix} \tag{9}$$

For this implementation, lower and upper bounds are set at $Lb_i = -0.25$ and $Ub_i = 0.25$ for $i = 1, ..., 15$. Positions and displacements are in meters.

3.3 Numerical Solvers

The space L2 benchmark is built up by combining (a) an open source mesh morpher [10], and (b) a MATLAB module that implements a multifidelity, panel based, aerothermodynamic solver for re-entry analyses [11, 12]. These two modules allow the treatment of aerodynamic and aerothermodynamic shape optimization problems. The aerodynamic module implements the Modified Newton Theory for the continuum regime and the Schaaf and Chambre inclined flat-plate model in free molecular flow regime, and a bridging function allows to approximate the aerothermodynamic characteristics in the transition regime. The aerothermal module implements several semi-empirical models, and for this test case the Kemp-Rose-Detra method has been used for the continuum regime, while the aerothermal Schaaf and Chambre has been used for the rarefied r egime. A gain a bridging function allows the generalization within the transition regime.

3.4 Problem Discretization and Analysis

The case is implemented by considering three multifidelity levels (from the finest L_1 to the coarsest L_3) that are defined by varying the number of pixels for the visibility detection module of the panel methods. It must be noted that the computational time for all the fidelity levels is practically the same and of the order of $15 - 20s$

(MATLAB 2021b running on a machine with i7-8550U 1.80GHz processor). For this reason, the implementation can be very flexibly used for testing multifidelity techniques, for example, by assigning a cost to the different levels with the task to minimize the total cost or maintain the cost under a predefined threshold. The suggested computational cost index (CCI) for the three fidelity levels are $ICC_{L_1} = 1$; $ICC_{L_2} = 0.46$; $ICC_{L_3} = 0.2$, in line with the normalized computational costs of the sea benchmark.

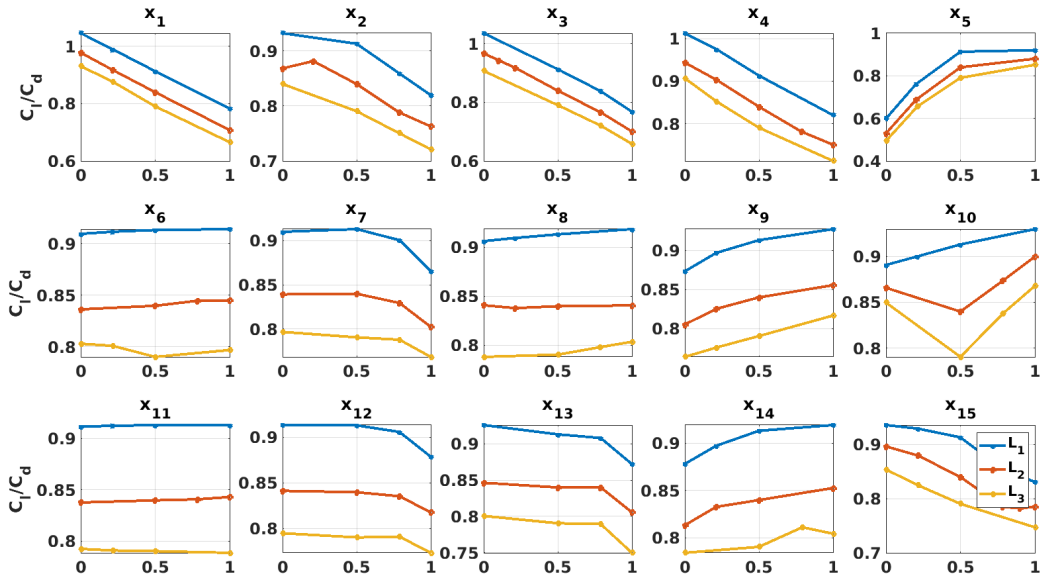

Figure 12: Space benchmark: multifidelity order 1 sensitivity analysis.

3.4.1 Analysis of Sample (Baseline) Results

The results of a preliminary sensitivity analysis over C_l/C_d are shown in Fig. 12. The analysis, performed considering the variables normalized in the intervals $[0, 1]$, show that some variables do not have much influence on the objective function, nonetheless they can be useful to manage the constraints of the problem. It can be seen how the fidelity levels provide different values of the objective function (for the original configuration the values are $C_l/C_d = 0.9130$ for L_1, $C_l/C_d = 0.8398$ for L_2, and $C_l/C_d = 0.7904$ for L_3) and almost similar trends. Some variable have different and more non-linear influence on the objective functions at the lowest fidelity levels. It can also be expected that the lowest fidelity levels are noisier.

3.5 Reference Optimum

The available reference optimum shows an improvement in terms of objective function of 54%, with an optimum value of $C_l/C_d = 1.41$, obtained by increasing the Lift (from ~1500 to ~2000N) and decreasing the Drag (from ~1660 to ~1450N, with $\mathbf{x} = \{-0.25, -0.25, -0.25, 0.004904, 0.215920, 0.113685, -0.202793, 0.249894,$

$0.25, 0.040783, 0.018520, -0.014108, -0.155942, 0.114036, -0.018369\}$. Figure 13 shows a comparison between the original and the reference optimal shape. It can be seen that the optimal shape is longer and has later part that is shadowed.

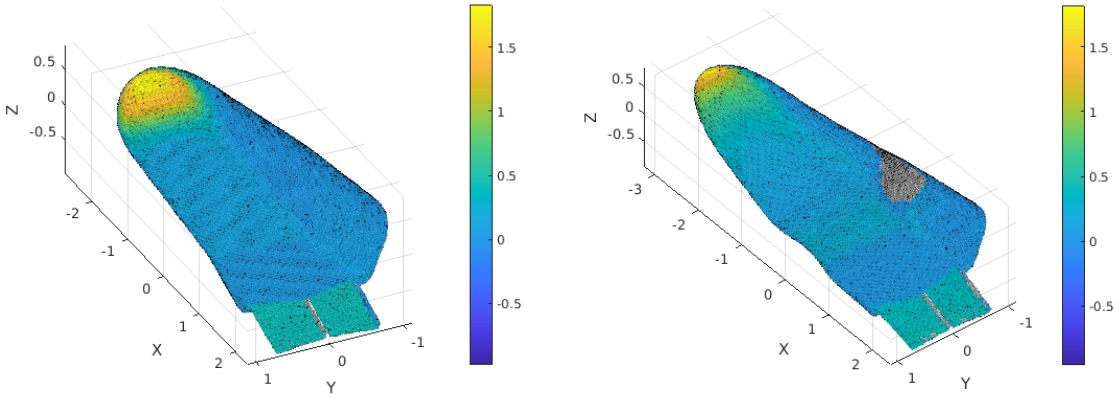

Figure 13: Space benchmark: baseline (left), and optimal reference (right).

3.6 Software Archive and Documentation

The current benchmark problem - including the aerodynamic and the aerothemodynamic modules and the interface with the morpher - is available at the GitLab repository of the AVT-331 working group. A public release will be available under the GPL license. The code works under Linux operating system.

The morpher is available at *https://optimad.github.io/mimmo/*.

4 SEA BENCHMARK

The sea benchmark pertains to the total resistance (R) reduction of a destroyer-type vessel in calm water at fixed speed, corresponding to a Froude number (Fr) equal to 0.28. Specifically, the hull under investigation is the DTMB 5415 model, a vessel extensively used as a benchmark for experimental [13] and computational fluid dynamics [14], as well as for shape optimization problems [15]. It is also used also as a test case for NATO AVT-204 (deterministic design optimization methods for military vehicles) [15], AVT-252 (stochastic design optimization methods for military vehicles) [16] and AVT-280 (prediction capabilities of ships large amplitude motions in heavy weather) [17].

4.1 Problem Formulation

The optimization problem reads

$$\begin{array}{rll} \text{minimize} & \Delta R(\mathbf{x}) = \frac{R(\mathbf{x})}{R_0} - 1 \qquad \text{with} & \mathbf{x} \in \mathbb{R}^N \\ \text{subject to} & L_{\text{pp}}(\mathbf{x}) = L_{\text{pp}_0} & \\ \text{and to} & \nabla(\mathbf{x}) = \nabla_0 & \\ & |\Delta B(\mathbf{x})| \leq 0.05 B_0 & \\ & |\Delta T(\mathbf{x})| \leq 0.05 T_0 & \\ & V(\mathbf{x}) \geq V_0 & \\ & -1 \leq x_i \leq 1 \qquad \text{with} & \forall i = 1, \ldots, N \end{array} \tag{10}$$

where $\mathbf{x}$ are the design variables, L_{pp} is the length between perpendiculars, B is the overall beam, T is the drought, and V is the volume reserved for the sonar in the bow dome. Subscript "0" indicates parent (original) hull values. Equality and inequality constraints for the geometry deformations are taken from [15]. Table 4 summarizes the main characteristics of the hull and simulation conditions for the hydrodynamic solver.

4.2 Design Parameterization and Design Space Dimensionality

The shape modifications $\tilde{\boldsymbol{\gamma}}(\boldsymbol{\xi}, \mathbf{x})$ are produced directly on the Cartesian coordinates $\boldsymbol{\xi}$ of the computational body surface grid $\mathbf{g}$, as per

$$\mathbf{g}(\boldsymbol{\xi}, \mathbf{x}) = \mathbf{g}_0(\boldsymbol{\xi}) + \boldsymbol{\gamma}(\boldsymbol{\xi}, \mathbf{x}) \tag{11}$$

where $\mathbf{g}_0$ is the original geometry and $\boldsymbol{\gamma}$ is a shape modification vector obtained by a physics-informed design-space dimensionality reduction [18, 19]

$$\boldsymbol{\gamma}(\boldsymbol{\xi}, \mathbf{x}) = \sum_{k=1}^{N} x_k \boldsymbol{\psi}_k(\boldsymbol{\xi}) \tag{12}$$

with $\boldsymbol{\psi}$ a set of orthonormal functions, with $N = 14$ the number of design variables ($\mathbf{x}$). It may be noted that the design variables and the associated shape modifications are organized in a hierarchical order, meaning that the first variables produce larger design modifications than the last ones [16].

Table 4: Sea benchmark: DTMB 5415 original hull main particulars and simulation conditions.

Quantity	Symbol	Unit	Value
Displacement	∇	m^3	0.549
Length between perpendiculars	L_{pp}	m	5.720
Beam	B	m	0.760
Draft	T	m	0.248
Water density	ρ	kg/m^3	998.5
Kinematic viscosity	ν	m^2/s	1.09E-6
Gravity acceleration	g	m/s^2	9.803
Froude number	Fr	–	0.280

4.3 Numerical Solver

The hydrodynamic solver, developed at the CNR-INM [20], is based on the Dawson linearization [21] of the potential flow equations. The total resistance is estimated as the sum of the wave and the frictional resistance. The wave component is evaluated using the pressure integral over the hull surface, whereas the frictional component is estimated using a flat-plate approximation based on the local Reynolds number [22].

4.4 Problem Discretization and Analysis

The multifidelity levels are defined by the computational grid size. Specifically, the benchmark is defined with seven grid (fidelity) levels with a refinement ratio of $2^{0.25}$, e.g., see the finest (G1) and coarsest (G7) grids and associated numerical solutions in Fig. 14.

Table 5 summarizes the grid sizes, along with the associated nodes number (M), the resistance computed for the original hull (R_0), an estimate of the grid error (E_G, see below), and the normalized computation cost (NCC). The NCC is evaluated as the ratio between the CPU time needed for the generic j-th fidelity and the highest-fidelity.

4.4.1 Analysis of Sample Results

Figure 15 shows the convergence of the resistance estimate as a function of the grid/fidelity level and the associated computational cost. Pressure on the hull and wave elevation patterns are shown in Fig. 16. Moving from the highest (G1) to the lowest (G7) fidelity, a lower pressure is predicted on the hull aft and, overall, smaller pressure gradients along the hull. Furthermore, the wave elevation decreases and the diverging bow waves are less evident. The linear regression based on G1, G3, and G5 triplet (see Fig. 15, left) converges to the solution given by the Richardson extrapolation [23], which provides an order of convergence of the numerical solver equal to $p_{\mathrm{RE}} \approx 2.8$. All the fidelities lie close to the regression and the associated grid errors E_G are evaluated with respect to the Richardson extrapolation value. For the sake of simplicity, this error is considered constant within the design space. Figure 15 (right) shows how the NCC scales nearly linearly with the square of the grid size. Finally, a preliminary sensitivity analysis is performed for each fidelity level and the results a shown in

Table 5: Sea benchmark, fidelity levels details: grid size, resistance, and associated error.

Grid	Hull nodes	Free-surface nodes	Total nodes (M)	Resistance [N]	Grid error (E_G%)	NCC
G1	150×50	180×50	16.5k	41.5	1.16	1.00
G2	126×42	151×42	11.6k	41.8	1.85	0.46
G3	106×35	127×35	8.2k	42.3	3.10	0.20
G4	89×29	107×29	5.7k	43.5	5.97	0.11
G5	76×25	90×25	4.2k	44.4	8.26	0.07
G6	64×21	76×21	2.9k	45.3	10.5	0.04
G7	54×18	64×18	2.1k	48.9	19.3	0.03

Note: refinement ratio equal to $2^{0.25}$

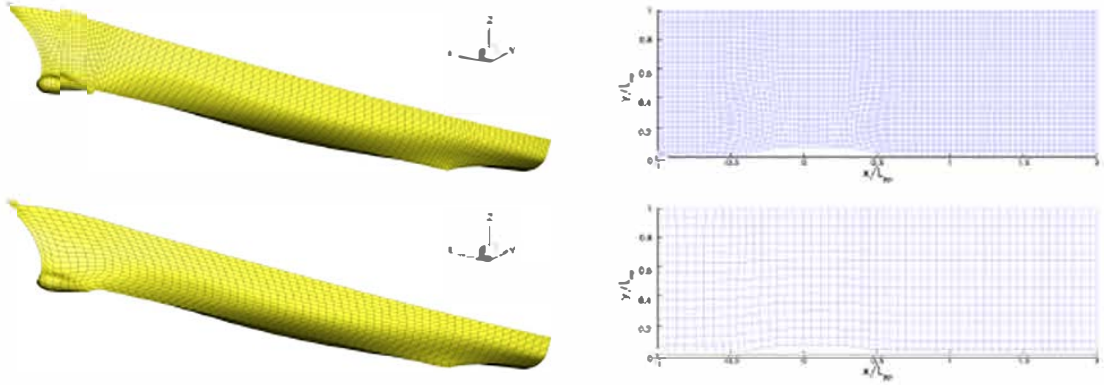

Figure 14: Sea benchmark: (left) hull and (right) free-surface discretizations for (top) highest- and (bottom) lowest fidelity levels.

Fig. 17. Each design variable provides potential resistance improvement. Furthermore, it can be seen how the fidelity levels provide almost similar trends, except for the lowest. This shows how the benchmark is of particular interest because, on the one hand, similar trends should favor multifidelity methods, on the other hand, the irregularity of the lowest-fidelity represents a challenge for the modeling of bridge functions.

4.5 Reference Optimum

A reference optimum has been found in [24], where an objective improvement of 12.5% is achieved with an NCC = 628. Specifically, the resistance is equal to 36.3132 N at $\mathbf{x} = \{-1.000000, -0.906250, -0.781250, -0.242188, 0.533203, -0.906250, 0.128418, 0.812500, -0.070313, -0.878296, 0.422363, 0.031250, 0.365234, -0.175659\}$. Figure 18 shows a comparison between the original and the reference optimal hull. The optimal hull shows a better recovery of the pressure towards the stern and more prominent cavities in the bow and aft regions of the hull, resulting in an overall lower wave elevation of both bow and stern diverging waves.

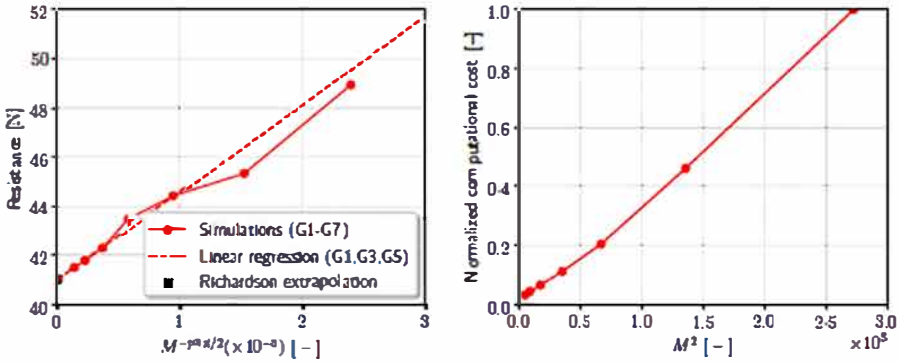

Figure 15: Sea benchmark: total resistance convergence (left) and grids error with respect to Richardson extrapolation (right) for the potential flow solver.

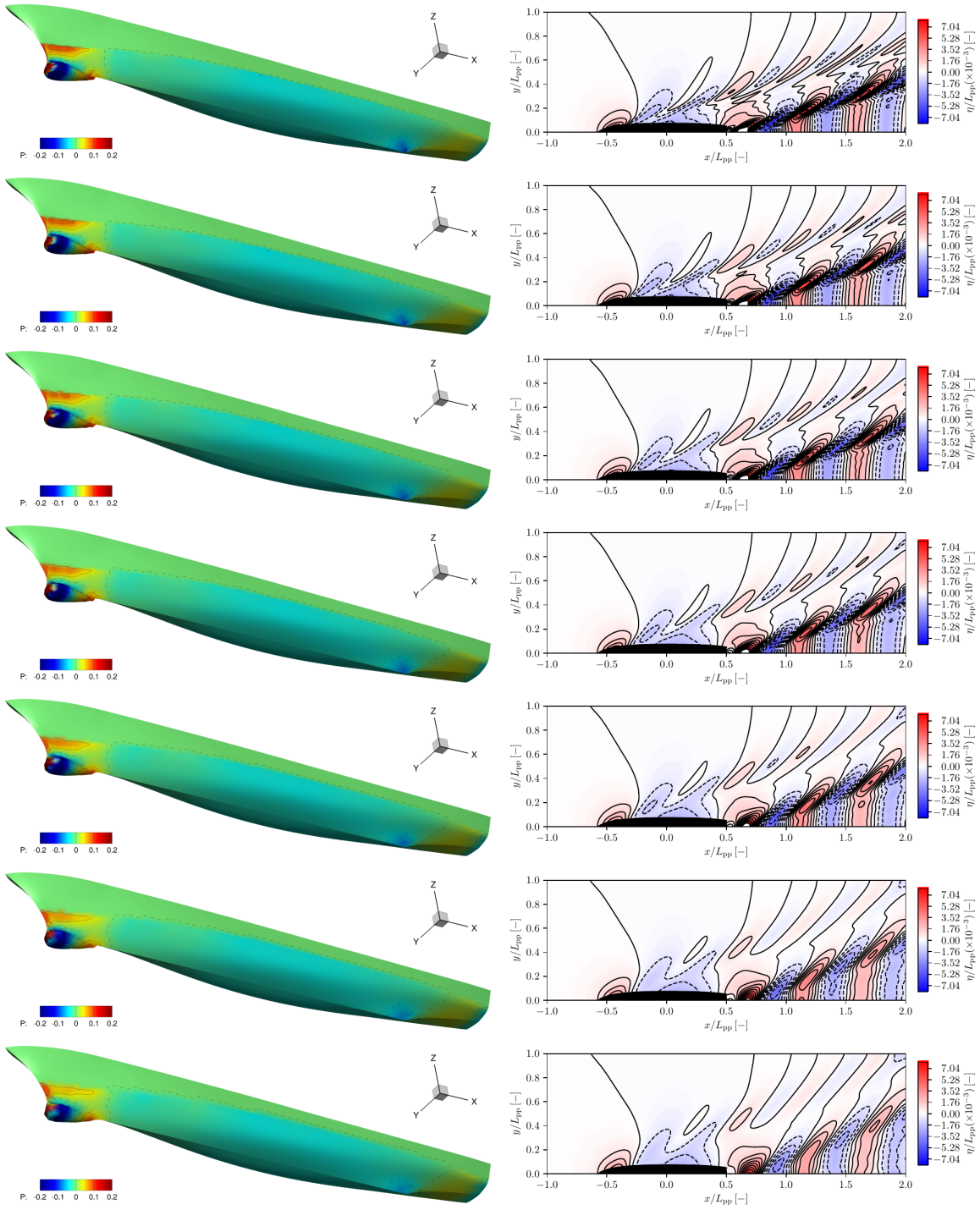

Figure 16: Sea benchmark: pressure on the hull (left) and free-surface elevation (right) of the original hull varying the grid refinement: from top to bottom highest- (G1) to lowest-fidelity (G7).

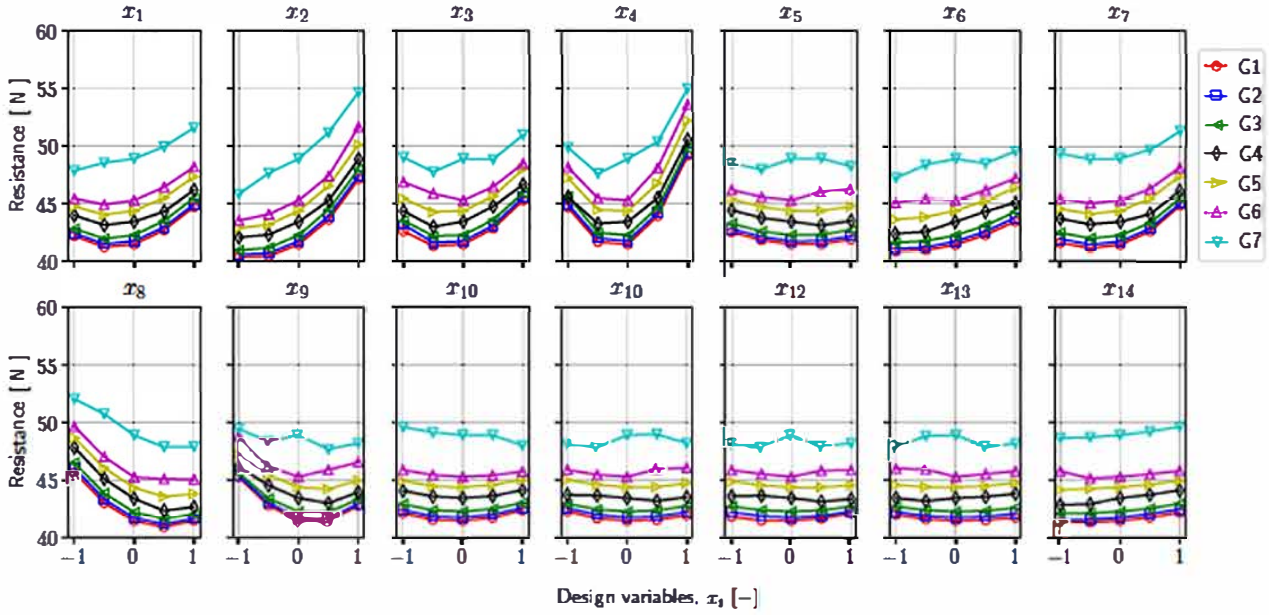

Figure 17: Sea benchmark: sensitivity analysis.

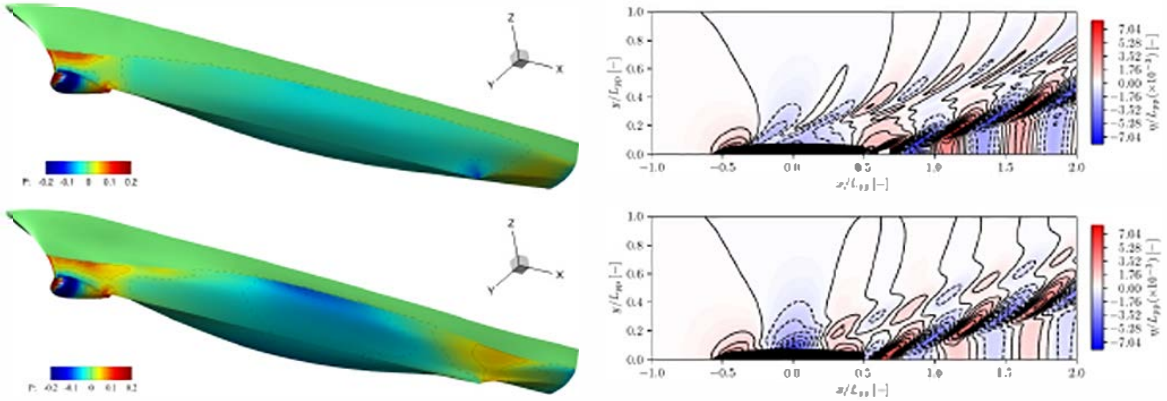

Figure 18: Sea benchmark: pressure on hull (left) and free-surface elevation (right) of the original (top) and reference optimum (bottom).

4.6 Software Archive and Documentation

The current benchmark problem is available as open-source Fortran code at the following `GitHub` repository *github.com/MAORG-CNR-INM/NATO-AVT-331-L2-Sea-Benchmark*. Installation instruction and documentation are available in the repository. The code can be compiled with both `GNU` and `Intel` compilers and, as major dependecy, needs `lapack` and `blas` libraries. Shared-memory (`openMP`) capability is also available and the code can be executed under `Linux` and `Windows subsystem for Linux`.

5 CONCLUSION

The three L2 benchmarks presented represent problems of industrial interest, respectively, for air, space, and sea vehicles and can be used with a relatively low effort as black-boxes to develop multifidelity optimization algorithms aimed at real-world problems.

All three benchmarks are characterized by good flexibility and set-up easiness of the calculation methods at different fidelity levels. The benchmarks are or will soon be freely available under the GPL license to allow widespread use within the community of developers of multifidelity optimization methods and aim to become a valuable and increasingly effective tool for the development of optimization methodologies.

Extensions of these benchmarks to more complex and even more interesting problems from the engineering point of view are being studied. In the case of the air benchmark, for example, we plan extensions to multi-component airfoils and simple 3D configurations together with the use of the ability of `SU2` to efficiently calculate gradients with the adjoint field method. The space benchmark can be extended to multi-point design optimization, as well as multidisciplinary problems, including the re-entry trajectory optimization. The sea benchmark tool is quite flexible and allows to be used for robust design optimization, uncertainty quantification, and multidisciplinary problem, which formulations are under development.

ACKNOWLEDGMENTS

A. Thelen and P. Beran acknowledge support of the US Air Force Office of Scientific Research (grant 20RQ-COR055, Dr. Fariba Fahroo, Computational Mathematics Program Officer). Distribution Statement A: Approved for Public Release; Distribution is Unlimited (AFRL-2022-2582).

REFERENCES

1. Beran, P. S., Bryson, D. E., Thelen, A. S., Diez, M. & Serani, A. *Comparison of Multi-Fidelity Approaches for Military Vehicle Design* in *21th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference (MA&O), AVIATION 2020* (Virtual Event, June 15-19, 2020).
2. Quagliarella, D. *et al. Benchmarking Uncertainty Quantification Methods Using the NACA 2412 Airfoil with Geometrical and Operational Uncertainties* in *AIAA Aviation 2019 Forum, Multidisciplinary Design Optimization* (2019), AIAA 2019–3555.
3. Quagliarella, D. & Diez, M. *An Open-Source Aerodynamic Framework for Benchmarking Multi-Fidelity Methods* in *AIAA AVIATION 2020 FORUM* (June 2020). eprint: `https://arc.aiaa.org/doi/pdf/10.2514/6.2020-3179`. `https://arc.aiaa.org/doi/abs/10.2514/6.2020-3179`.
4. Spalart, P. & Allmaras, S. *A one-equation turbulence model for aerodynamic flows* in *30th Aerospace Sciences Meeting and Exhibit* (AIAA, 1999). eprint: `https://arc.aiaa.org/doi/pdf/10.2514/6.1992-439`. `https://arc.aiaa.org/doi/abs/10.2514/6.1992-439`.

5. Geuzaine, C. & Remacle, J.-F. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering* **79,** 1309–1331. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2579. https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2579 (2009).
6. Economon, T. D., Palacios, F., Copeland, S. R., Lukaczyk, T. W. & Alonso, J. J. SU2: An Open-Source Suite for Multiphysics Simulation and Design. *AIAA Journal* **54,** 828–846. eprint: https://doi.org/10.2514/1.J053813. https://doi.org/10.2514/1.J053813 (2016).
7. Drela, M. *XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils* in *Low Reynolds Number Aerodynamics: Proceedings of the Conference Notre Dame, Indiana, USA, 5–7 June 1989* (ed Mueller, T. J.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 1989), 1–12. ISBN: 978-3-642-84010-4. http://dx.doi.org/10.1007/978-3-642-84010-4_1.
8. Hansen, N. & Ostermeier, A. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation* **9,** 159–195 (2001).
9. Tumino, G. *et al. The IXV project: the ESA re-entry system and technologies demonstrator paving the way to European autonomous space transportation and exploration endeavours* in *IAC 2008* (2008).
10. OPTIMAD engineering srl. *MIMMO: Surface manipulation and mesh morphing library* https://github.com/optimad/mimmo (2015).
11. Mehta, P., Minisci, E., Vasile, M., Walker, A. C. & Brown, M. *An open source hypersonic aerodynamic and aerothermodynamic modelling tool* in *8th European Symposium on Aerothermodynamics for Space Vehicles* (2015).
12. Falchi, A., Renato, V., Minisci, E. & Vasile, M. *FOSTRAD: An advanced open source tool for re-entry analysis* in *15th Reinventing Space Conference* (2017).
13. Irvine Jr., M., Longo, J. & Stern, F. Pitch and Heave Tests and Uncertainty Assessment for a Surface Combatant in Regular Head Waves. *Journal of Ship Research* **52,** 146–163 (2008).
14. Larsson, L. *et al. Proceedings, Tokyo 2015 Workshop on CFD in Ship Hydrodynamics* in *Tokyo CFD Workshop* (2015).
15. Grigoropoulos, G. *et al. Mission-based hull-form and propeller optimization of a transom stern destroyer for best performance in the sea environment* in *VII International Conference on Computational Methods in Marine Engineering MARINE 2017* (2017).
16. Serani, A., Stern, F., Campana, E. F. & Diez, M. Hull-form stochastic optimization via computational-cost reduction methods. *Engineering with Computers,* 1–25 (2021).
17. Serani, A., Diez, M., van Walree, F. & Stern, F. URANS analysis of a free-running destroyer sailing in irregular stern-quartering waves at sea state 7. *Ocean Engineering* **237,** 109600 (2021).
18. Serani, A., Campana, E. F., Diez, M. & Stern, F. *Towards Augmented Design-Space Exploration via Combined Geometry and Physics Based Karhunen-Loève Expansion* in *AIAA-AVIATION 2017 conference, Denver, USA, June 5-9* (2017).
19. Serani, A., Diez, M., Wackers, J., Visonneau, M. & Stern, F. *Stochastic shape optimization via design-space augmented dimensionality reduction and RANS computations* in *57th AIAA Aerospace Sciences Meeting, SciTech 2019* (2019), 2218.
20. Bassanini, P., Bulgarelli, U., Campana, E. F. & Lalli, F. The wave resistance problem in a boundary integral formulation. *Surveys on Mathematics for Industry* **4,** 151–194 (1994).

21. Dawson, C. W. *A practical computer method for solving ship-wave problems* in *Proceedings of the 2nd International Conference on Numerical Ship Hydrodynamics* (Berkeley, 1977), 30–38.

22. Schlichting, H. & Gersten, K. *Boundary-Layer Theory* (Springer-Verlag, Berlin, 2000).

23. Xing, T. & Stern, F. Factors of safety for Richardson extrapolation. *Journal of Fluids Engineering* **132,** 061403 (2010).

24. Pellegrini, R. *et al.* A Derivative-Free Line-Search Algorithm for Simulation-Driven Design Optimization Using Multi-Fidelity Computations. *Mathematics* **10,** 481 (2022).