Available online at www.prace-ri.eu

Partnership for Advanced Computing in Europe

SHAPE Project Hydros Innovation: Automatic Optimal Hull Design by Means of VPP Applications on HPC Platforms

S. Dyen[a*], A. Lukowski[a], R. Ponzini[b]

[a]*Hydros Innovation, Switzerland*
[b]*CINECA, Italy*

Abstract

Hydros is an Engineering & Research Swiss company founded in 2007 with several patented designs in the field of marine and sailing yachting. In recent years Hydros is exploring new market segments such as yachts and super-yachts hull design. The usage of HPC resources and open-source softwares can be a valuable tool in the massive shape design optimizations usually performed by Hydros. The main scope of the proposed SHAPE pilot was therefore to evaluate the feasibility of automatic optimal hull design on HPC infrastructure and the impact of such a workflow on the day-by-day work of Hydros personnel. To accomplish this task the project was subdivided into a set of steps including a preliminary validation of a 2DoF CFD analysis of an industrial hull design using open-source code, the scalability test of commercial and open-source CFD code for hull 2DoF modelling, the coupling of the CFD result into an existing CAD modification and optimization loop, and finally the running of a complete optimization loop for an industrial hull design using open-source code on the HPC platform and usability evaluation of the solution provided.

1. Introduction

Hydros is an Engineering & Research Swiss company founded in 2007 with several patented designs in the field of marine and sailing yachting. Hydros most famous projects are:

- l'Hydroptère: two world speed records in 2009 and absolute sailing speed record (51.36kts);
- Hydros C-Class: first yacht built in TPT, 2nd place at Little Cup 2013;
- HY-X: first hybrid « Fly and Float » motorboat.

In recent years Hydros is exploring new market segments such as yachts and super-yachts hull design.

2. Project objectives and methods

The main scope of the proposed application is to evaluate the feasibility of automatic optimal hull design on HPC infrastructure and the impact of such a workflow on the day-by-day work of Hydros personnel with the final scope of reducing the cost and time required to obtain large CFD computational campaigns.
To do so the project will be subdivided into:

- Scalability test of commercial and open-source Computational Fluid Dynamics (CFD) code for hull two Degree of Freedom (2DoF) modelling;
- Validation of 2DoF CFD analysis of an industrial hull design using open-source code;
- Coupling of the CFD result into an existing Computer Aided Design (CAD) modification and optimization loop;
- Submit a complete optimization loop for an industrial hull design using open-source code on the High Performance Computing (HPC) platform and evaluate usability of the solution provided.

The hull selected for the overall project is an industrial hull, the baseline CAD is shown in Figure 1.

[*] Corresponding author: Stéphane Dyen, stephane@hydros.ch, Hydros Innovation (CH)

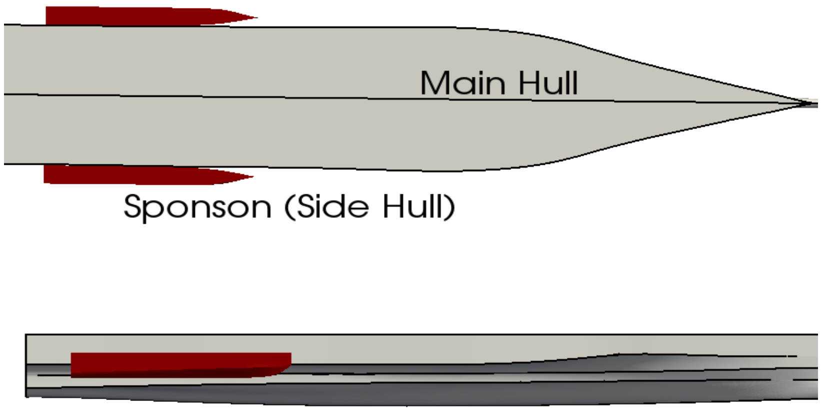

Figure 1. CAD model views (top and side) of the selected industrial hull

Notably the hull is made of two main parts: a so-called 'Main Hull' (in grey in Figure 1) and a couple of identical side hulls named Sponsons (in red in Figure 1). The main hull shape is considered fixed for this project while the side hull shape is parametric. The fluid dynamics is studied using a 2DoF, free sink and trim, CFD modelling. This kind of modelling is very standard in industry and allows to find out the attitude of the boat for a given cruise velocity in a very cost-effective way. Today Hydros performs this kind of study using an home-made Velocity Prediction Program (VPP) coupled with a commercial CFD software (FineMarine, NUMECA Inc.) on a small in-house computational resource. The CFD reference values used in this paper are related to the commercial software that is considered the state-of-the-art for Hydros personnel day by day work. According to the scopes of the pilot study, in the forthcoming, we will discuss in details the scalability test performed, the validation study and the coupling between the CFD solver and the parametric CAD engine.

2.1 Scalability test of commercial and open-source codes

The hardware used for all the computations was provided by CINECA. We used a x86_64 architecture, IBM NeXtScale machine, made of 516 nodes with each node equipped with two 8-cores Intel Haswell 2.40 GHz and 128 GB RAM. The overall computational power available was therefore 8256 cores. The operating system installed was Linux CentOS version 7.0.
The scalability of the commercial code used by Hydros (FineMarine, NUMECA Inc.) was tested on the CINECA hardware. The FineMarine version used was 4.2, while the license manager version used was v11.13.0.2. Unfortunately due to unresolved licensing compatibility issues we were not able to run the case on more than 16 cores (single node). For this reason the scalability performance for the commercial software is not available.
In order to test the open-source code scalability we selected a 2DoF CFD model developed in OpenFOAM (OpenCFD) for a given configuration. The case has been run on CINECA hardware and tested with increasing numbers of computational cores from 16 to 80. Data are summarized in Table 1.

# of computational cores	# of computational nodes	Efficiency	# of mesh-cells/core
16	1	-	125.000
32	2	92%	62.000
48	3	86%	41.000
64	4	76%	31.000
80	5	66%	25.000

Table 1. Scalability test of a 2DoF reference setup using openFoam solver release 2.4.x

The results are coherent to the fact that for a given mesh size there is a sweet spot of computational resources that can be used efficiently to solve the case. So that, despite in general the total wall-time continues to decrease when increasing the number of computational cores, there is an optimal number of cores that ensures a cost-effective time-to-solution. For our setup this value is 80 and this corresponds to a density of mesh-cells per core of about 25.000 as stated in Table 1 and shown in Figure 2. This value can be considered as, for a given mesh size problem, a reference to an optimal time-to-solution in this field of application and on a similar hardware configuration.

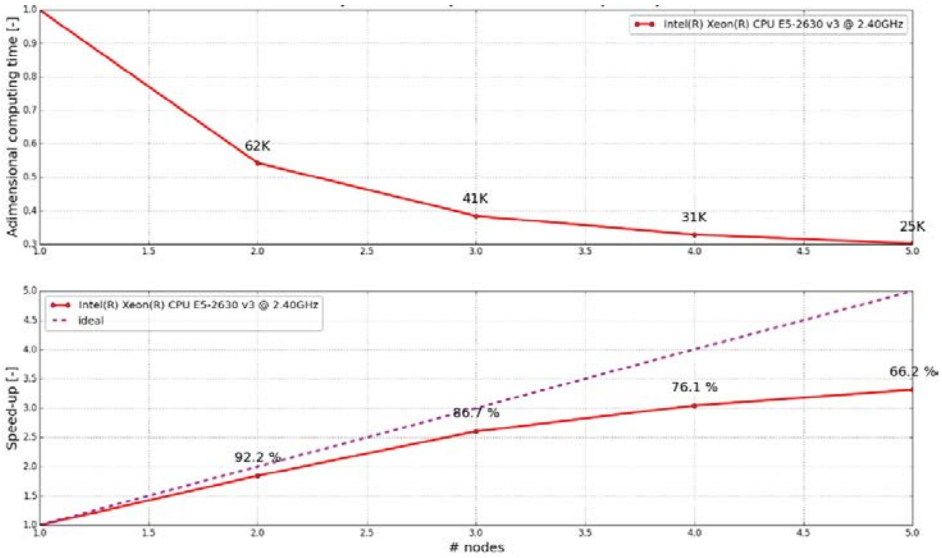

Figure 2. Scalability results for OpenFOAM 2.4.x 2DoF solver

2.2 Validation of 2DoF CFD analysis of an industrial hull design using open-source code

The open-source 2DoF CFD solver validation has been performed on a given hull design for a set of three velocity values (namely 15, 28 and 32 knots). The meshing strategy was designed in order to obtain a high quality hexa dominant mesh according to snappyHexMesh quality criteria and to ensure a proper number of layers added around the hull. The final mesh obtained was of about 8 million cells. The CFD solver selected is the interDyMFoam solver available in the OpenFOAM toolbox 2.4.x release. The turbulent behavior of the problem was solved using a Reynolds-averaged Navier–Stokes (RANS) strategy using the (SST) k-omega turbulent model and standard wall functions. The single runs were performed on CINECA hardware on 4 nodes (96 cores), the meshing time was about 1 hour while the solving time was ranging from 16 to 45 hours depending on the considered velocity of the hull. The obtained results were compared to the ones obtained with the commercial software owned by Hydros and they were considered acceptable in terms of accuracy. In Figure 3 a view of three outcomes of the wave elevation performed during the solver validation procedure is shown.

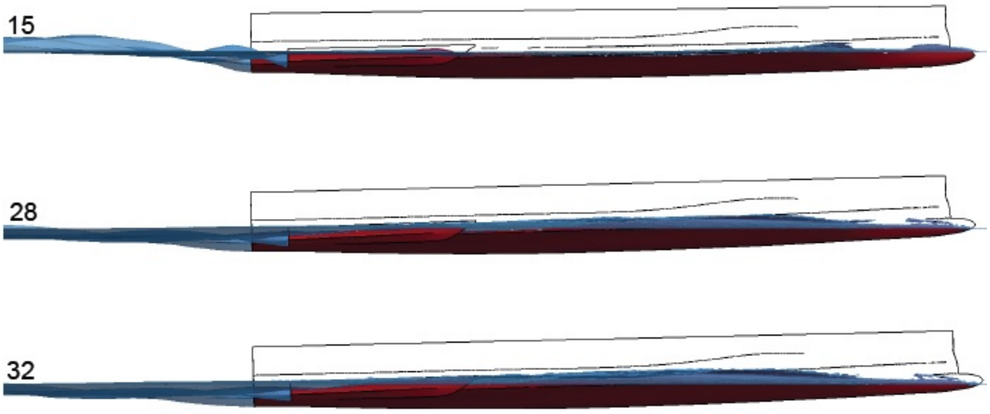

Figure 3. Validation procedure: wave elevation analysis at three hull velocities

2.3 Coupling of the CFD result into an existing CAD modification and optimization loop

The coupling of a CFD solver and of a parametric CAD design engine is today of capital interest in industrial Research and Development applications. Today Hydros is using CAESES® as the parametric CAD design tool. For this reason we decided to adopt it for the present project as parametric CAD engine and to couple it with OpenFOAM as CFD solver.

The CAD block is used as the main block and the CFD block is used as a callable function. Communication between the two blocks is made by text files exchange. To increase the robustness of the workflow, the sponson geometry is generated separately from the main hull. The Boolean operation to get the fluid domain is postponed to the CFD block.

In the CAD block, the main technical limitation to overcome was the parametrization of the hull with a reasonable number of parameters. The problem dimensions dramatically increase with the complexity of the geometry, that's why direct optimization is not used for now on hull geometry. For the CAD parametrization of the sponson, we decided to define a hard-chined asymmetric hull with a set of 20 independent parameters that controls the deadrise variation, beam and chine height from bow to transom. The strategy to have only 20 parameters in order to well define the hull is the following: parameters control the position in space of a list of points, tangent values or curvature values (bow inlet angle, transom height) then from these 20 constraints we draw the main lines of the hull. Center Plane Curve (CPC), Front Off-side (FOS) and Chine line. Finally surfaces and a watertight volume are reconstructed.

But even with this small but complete list of parameters, a direct optimization would have requested thousands of CFD evaluations to be properly converged. The geometry parametrization has to be reduced. We have defined 5 parameters that have a physical meaning (flow inlet fullness on port side, flow outlet fullness on port side, flow inlet fullness on starboard side, flow outlet fullness on starboard side, center plane curve fullness). These parameters range between 0 and 1 (0 for minimum fullness, 1 for maximum). The 20 geometrical parameters are then fed with a linear combination of the 5 parameters, meaning that we define a bijective transformation between the 5 physical parameters and the 20 geometrical parameters. The physical meaning of the 5 reduced parameters is useful to ensure fast convergence of the optimization process.

Once we are able to define any geometry with a small set of parameters, we have to limit the number of unfeasible designs provided by CAESES® to the CFD block. These unfeasible designs have been analyzed and it appears that they are mainly designs that are not usable for the CFD block and designs of sponson that do not give reasonable values of displacement force. On the watertight issues, the problem has been solved by an inner loop of optimization that tries different meshing set-ups until the watertight triangulation mesh is obtained and then passes the triangulation to the CFD block. Typically, CAESES® uses a Brent algorithm on the triangulation size for 20 times in order to find the optimum triangulation. The second issue is solved by a second inner optimization loop that positions the sponsons vertically to ensure the right static displacement by mean of an hydrostatic study carried out in CAESES® directly.
To test the ability to produce design variation and get results back from the CFD block, we had used a Sobol algorithm to generate 20 designs well covering the range of parameter values (from 0 to 1).

	CPCfulness	InletFullnessI	InletFullness!	OutletFlatPOI	OutletFlatSTE	Boyancy_kg	TAR_Z_after!	IsWatertight
Sobol_51_des0000	0.55	0.25	0.75	0.5	0.5	246.54646	1.5679206	11.604617
Sobol_51_des0001	0.775	0.125	0.875	0.25	0.75	262.1176	1.5967478	11.777396
Sobol_51_des0002	0.325	0.375	0.625	0.75	0.25	251.81495	1.5072672	11.410504
Sobol_51_des0003	0.4375	0.1875	0.8125	0.125	0.875	247.26993	1.5653737	11.45499
Sobol_51_des0004	0.8875	0.4375	0.5625	0.625	0.375	264.01693	1.5967478	11.964547
Sobol_51_des0005	0.6625	0.0625	0.6875	0.375	0.125	249.71015	1.4611027	10.55085
Sobol_51_des0006	0.2125	0.3125	0.9375	0.875	0.625	250.09902	1.5890392	12.042186
Sobol_51_des0007	0.26875	0.15625	0.65625	0.6875	0.5625	250.74212	1.5144378	11.606348
Sobol_51_des0008	0.71875	0.40625	0.90625	0.1875	0.0625	251.41725	1.4943648	10.64186
Sobol_51_des0009	0.94375	0.03125	0.78125	0.9375	0.3125	267.2412	1.5967478	11.88237
Sobol_51_des0010	0.49375	0.28125	0.53125	0.4375	0.8125	242.65075	1.5835907	11.78107
Sobol_51_des0011	0.38125	0.09375	0.96875	0.5625	0.4375	250.3784	1.532493	11.384007
Sobol_51_des0012	0.83125	0.34375	0.71875	0.0625	0.9375	272.36169	1.5967478	12.019953
Sobol_51_des0013	0.60625	0.21875	0.59375	0.8125	0.6875	263.14673	1.5967478	12.326503
Sobol_51_des0014	0.15625	0.46875	0.84375	0.3125	0.1875	255.42133	1.4002963	10.725883
Sobol_51_des0015	0.184375	0.234375	0.921875	0.40625	0.28125	249.59914	1.4382247	10.894656
Sobol_51_des0016	0.634375	0.484375	0.671875	0.90625	0.78125	307.93129	1.5967478	13.284624
Sobol_51_des0017	0.859375	0.109375	0.546875	0.15625	0.53125	251.84409	1.5278153	11.104171
Sobol_51_des0018	0.409375	0.359375	0.796875	0.65625	0.03125	252.37785	1.4837393	10.945345
Sobol_51_des0019	0.521875	0.046875	0.734375	0.28125	0.65625	252.1969	1.5234776	11.314291

Figure 4: Sobol variation, parameters values (yellow), inner loop of optimization results (blue) with target buoyancy and a watertight criteria on the mesh

On every HPC platform usage of a parametric design loop is a batch process where there is no interaction between the end-user and the different design processes. To ensure the correct evolution of the parametric design study, a robust workflow must be provided. The workflow has been designed as follows using a Python scripting procedure:

- Starting from a parametric CAD definition (main directory of the CAESES® project), the selected CAESES® Sobol engine defines the 'next' Sponson CAD to be studied in a CFD solver engine and the input parameters (velocity of the boat);
- For every 'next' design a so called 'CFDbloc' subdirectory is automatically created by CAESES® and within this subdirectory a CFD workflow is performed;
- This CFD workflow is very standard and consists of a coupling of the given 'next' Sponson CAD with the main hull CAD (fixed for every design), a meshing of the obtained complete hull plus a 2DoF solver run for a given cruise velocity value;
- At convergence the CFD workflow provides a post-processing of meaningful physical key parameters (in our case four main quantities were extracted from the CFD solution) into an output file;
- At the end of the CFD run CAESES® re-collects the output file for the given 'next' design for further analysis of optimality.

In Figure 5, a visual representation of the workflow is given.

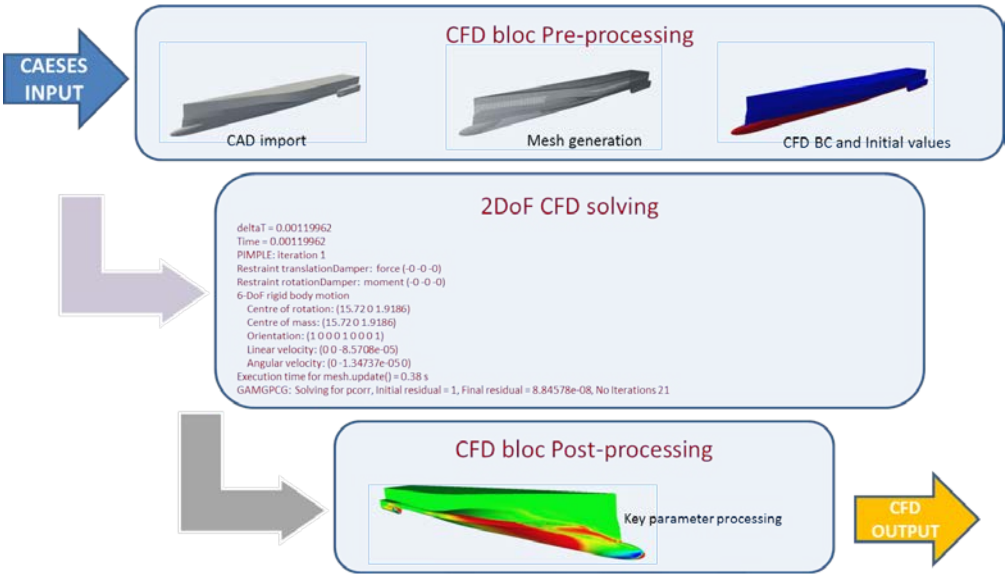

Figure 5: Sketch of the parametric workflow

The central issue in this kind of workflow is dependent on the ability of CFD methodology designer to ensure a proper meshing strategy able to catch, in an automated fashion, the possible changes that will occur due to the parametric CAD definition and, at the same time, guarantee a good mesh quality for each configuration. In the present project this has been a major issue since the CFD problem of the coupled hulls is per se complex (even for a single fixed design) and highly requiring in terms of meshing quality. Moreover the ranges of the changes in shape that the Sponson can take during the parametric study is wide, introducing a novel degree of complexity to be handled. To overcome this issue we played with relative distance between the main hull and the Sponson and with the meshing criteria. The final meshing strategy ensured an homogenous mesh quality and topology resulting in a size of about 3 million cells at every design allowing therefore to perform reliable CFD runs and compare the obtained results. In Figure 6 a taste of these changes is given selecting just four over the twenty possible designs.

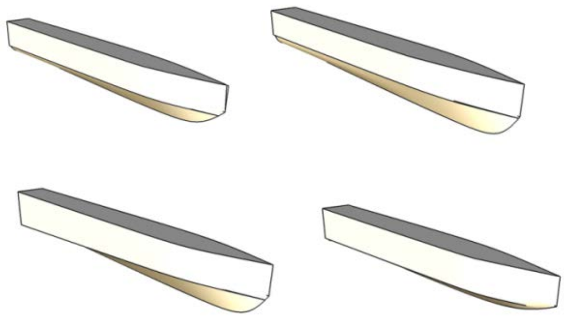

Figure 6. Selected design of the Sponson CAD.

All the twenty different designs simulations run together allowing to obtain the full design matrix outcomes at the same time. Each CFD run used 20 computational cores for a total of 400 computational cores. The end time of each CFD bloc was fixed at 4 seconds of physical time. The total wall-time was of about 3 hours while the total computational cost of the design analysis was 1200 core hours for the 20 designs in total.
In Table 2 the synthetic key parameter outcome is given for every design considered.

DesignID	TotalDrag[N]	Pitch[deg]	Sink[m]	DragSponson[N]
des0000	45413	-1.23	-0.004	4396
des0001	45917	-1.26	0.000	4938
des0002	45430	-1.22	0.010	4651
des0003	45741	-1.25	-0.003	4397
des0004	45434	-1.18	0.003	4843
des0005	45598	-1.23	-0.002	4830
des0006	45320	-1.23	0.000	4621
des0007	45517	-1.20	0.004	4431
des0008	45403	-1.24	-0.001	5181
des0009	45520	-1.21	0.004	5777
des0010	45854	-1.20	0.002	4409
des0011	45249	-1.21	-0.008	4418
des0012	45713	-1.21	0.000	5163
des0013	45555	-1.22	0.004	4800
des0014	45745	-1.22	0.007	4830
des0015	45655	-1.22	0.003	4586
des0016	44806	-1.45	0.001	7103
des0017	45711	-1.22	-0.004	4805
des0018	45619	-1.24	-0.004	5001
des0019	45483	-1.23	0.011	4752

Table 2. Meaningful physical key parameters values for the 20 considered designs.

3. Results

The main goal of a Sobol variation is to identify promising directions of optimization with a small number of computations. Then the area of interest can be limited and a direct optimization process can be carried out in the reduced area. One can sometimes also get some correlations between the parameters and the results which helps a lot to define a direction of optimization. Below (Figure 7 and Table 3) the results are printed as an increasing list with associated parameters. For example the trend between parameter 1 and results is close to a quadratic law, meaning that for the next optimization we will be able to restrict to [0.3;0.6] the area of interest of the CPC fullness.

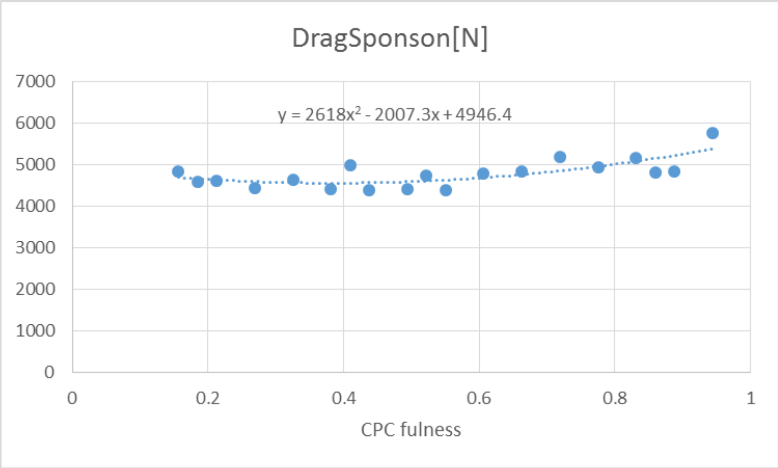

Figure 7. Sponson drag values and extrapolated law

name	DragSponson [N]	CPCfulness	InletFullnessPORT	InletFullnessSTB	OutletFlatPORT	OutletFlatSTB
Sobol_51_des0000	4396	0.55	0.25	0.75	0.5	0.5
Sobol_51_des0003	4397	0.4375	0.1875	0.8125	0.125	0.875
Sobol_51_des0010	4409	0.49375	0.28125	0.53125	0.4375	0.8125
Sobol_51_des0011	4418	0.38125	0.09375	0.96875	0.5625	0.4375
Sobol_51_des0007	4431	0.26875	0.15625	0.65625	0.6875	0.5625
Sobol_51_des0015	4586	0.184375	0.234375	0.921875	0.40625	0.28125
Sobol_51_des0006	4621	0.2125	0.3125	0.9375	0.875	0.625
Sobol_51_des0002	4651	0.325	0.375	0.625	0.75	0.25
Sobol_51_des0019	4752	0.521875	0.046875	0.734375	0.28125	0.65625
Sobol_51_des0013	4800	0.60625	0.21875	0.59375	0.8125	0.6875
Sobol_51_des0017	4805	0.859375	0.109375	0.546875	0.15625	0.53125
Sobol_51_des0005	4830	0.6625	0.0625	0.6875	0.375	0.125
Sobol_51_des0014	4830	0.15625	0.46875	0.84375	0.3125	0.1875
Sobol_51_des0004	4843	0.8875	0.4375	0.5625	0.625	0.375
Sobol_51_des0001	4938	0.775	0.125	0.875	0.25	0.75
Sobol_51_des0018	5001	0.409375	0.359375	0.796875	0.65625	0.03125
Sobol_51_des0012	5163	0.83125	0.34375	0.71875	0.0625	0.9375
Sobol_51_des0008	5181	0.71875	0.40625	0.90625	0.1875	0.0625
Sobol_51_des0009	5777	0.94375	0.03125	0.78125	0.9375	0.3125
Sobol_51_des0016	7103	0.634375	0.484375	0.671875	0.90625	0.78125

Table 3. Key parameters influence for the 20 designs.

Comparing the flow on the different designs is also of interest to understand the trends between parameters and the results, but only if the parameters have a physical meaning. For example the amount of spray on the bow of the sponson is characteristic of its drag. Another characteristic to check is if the bow wave is well broken by the hard chine. Design 16 presents the worst efficiency concerning the sponson drag but a low overall resistance of the ship, with values far from all the other values. The check of the visualization and history of the loads during computation allowed us to determine that it was due to a problem of convergence of the calculation and not a really overall efficient design. At this stage the pictures of the wetted area, the pressure, and the free surface shape have been exported automatically and can be consulted easily during the posttreatment process, to allow a verification of the quality of the calculation, as well as a good understanding of the physics behind.

4. Cooperation and benefit for the SME

The main technical limitation of the direct optimization of complex geometries and flow as a hull shape with multi-fluids physics is the CPU time required and the associated cost to evaluate a big amount of designs. The solution provided in this paper is to work with open-source software to evaluate the performance of each design, in our case OpenFOAM.

Then the reliability of the open-source CFD tool has to be proven. Here we have done a validation case to compare the results of the commercial software FineMarine with the results using OpenFOAM.

Assuming that OpenFOAM results are reliable we then have compared the computation time which stays reasonable for industrial purposes. HPC solutions are of interest since we are able to launch a great amount of design evaluations simultaneously, thanks to a robust and automatized workflow. This would not have been possible with usual license schemes of commercial software.

HPC makes complex shape optimization affordable for industrial purposes thanks to an automatized, reliable and robust workflow using open-source software.

5. Conclusions and future actions

HPC solutions allow us to carry out more complex studies, reducing the cost and time required to obtain large CFD computational campaigns. The methods and processes we have put in place belong to a growing technology. Before using it for industrial purposes, we have to make the market confident in the method, and convinced that the cost of automated optimization is in accordance with the gain in performances.

The ability to provide a robust black box for CFD evaluation is a big gain for the complete workflow. The parametrization of the hull still needs to be improved, taking the physical phenomenon involved in the optimization problem better into account. Scalability performance of these kinds of computations with small meshes remains a limit that slows down the global process of optimization, but it also allows to launch many calculations on the same time on several nodes.

References

[1] Site OpenFOAM: http://www.openfoam.com/
[2] Site Hydros Innovations: http:// www.hydros-innovation.com/
[3] Site CINECA SCAI: http://www.hpc.cineca.it/

Acknowledgements

This work was financially supported by the PRACE project funded in part by the EU's Horizon 2020 research and innovation programme (2014-2020) under grant agreement 653838.

The authors are grateful to Roberto Pieri, SCS Italy, for supporting the project with the baseline CFD modelling and solver validation in OpenFOAM and with technical hints and fruitful suggestions.

We would like also to acknowledge the support that CAESES® and FineMarine offered. They provided great amount of support time and offered licenses that allowed this project.