Ref. Ares(2022)3380124 - 02/05/2022

Upscaling **P**roduct development **S**imulation **C**apabilities exploiting **A**rtificial inte**L**ligence for **E**lectrified vehicles

D2.5 Assessment of AI/ROM based optimization performances with respect to state-of-the-art methodologies

Authors

Kyriazis Nikolaos, Eugene De Villiers– Engys

Luca Miretti – CRF

Enric Aramburu – IDIADA

Guglielmo Minelli – Volvo Cars

Month Year

04/2022

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement Nº 824306.

Project Details

PROJECT TITLE	Upscaling product development simulation capabilities exploiting artificial intelligence for electrified vehicles
PROJECT ACRONYM	Upscale
GRANT AGREEMENT NUMBER	824306
INSTRUMENT	RIA
CALL	LC-GV-2018
STARTING DATE OF THE PROJECT	November, 1ST 2018
PROJECT DURATION	42 Months

The UpScale Project

The UPSCALE (Upscaling Product development Simulation Capabilities exploiting Artificial inteLligence for Electrified vehicles) goal is demonstrating the feasibility of using AI enhanced CAE methods in EV development processes, such as vehicle aerodynamics, battery thermal modelling and crash simulation and leading the deployment of AI tools for other CAE applications. UPSCALE is the first EU-project that has the specific goal to integrate artificial intelligence (AI) methods directly into traditional physics-based Computer Aided Engineering (CAE)-software and –methods. These CAE-tools are currently being used to develop road transportation not only in Europe but worldwide. The current focus of the project is to apply AI-methods to reduce the development time and increase the performance of electric vehicles (EVs) which are required by the automotive industry to reduce global emission levels. High performance computing (HPC) and CAE-software and –methods play a decisive role in vehicle development process. In order to make a significant impact on the development process, the two most HPC intensive CAE-applications have been chosen as use cases for the project: vehicle aero/thermal- and crash modelling. When considering total automotive HPC usage, approximately 20% is used for aero/thermal simulations and up to 50% of HPC resources are utilized for crash simulations. By improving the effectiveness of these two areas, great increases in efficiency will lead to a 20% reduction of product time to market. Other novel modelling approaches such as reduced order modelling will be coupled to the AI improved CAE-software and -methods to further reduce simulation time and ease the application of optimization tools needed to improve product quality. Through the combined effort of universities, research laboratories, European automotive OEMs, software companies and an AI-SME specialized in machine learning (ML), the UPSCALE project will provide a unique and effective environment to produce novel AI-based CAE-software solutions to improve European automotive competitiveness.

The UpScale Consortium

PARTICIPANT Nº	PARTICIPANT ORGANISATION NAME	COUNTRY
1 (Coordinator)	**IDIADA AUTOMOTIVE TECHNOLOGY SA (IDIADA),**	Spain
2	VOLVO PERSONVAGNAR AB (Volvo Cars)	Sweden
3	VOLKSWAGEN AG (VW)	Germany
4	CENTRO RICERCHE FIAT SCPA (CRF)	Italy
5	ESI GROUP (ESI GROUP)	France
6	ENGYS LTD (ENGYS LTD)	United Kingdom
7	Kompetenzzentrum - Das Virtuelle Fahrzeug, Forschungsgesellschaft mbH (VIF)	Austria
8	VRIJE UNIVERSITEIT BRUSSEL (VUB)	Belgium
9	ECOLE NATIONALE SUPERIEURE D'ARTS ET METIERS (ENSAM PARISTECH)	France
10	ALGORITHMICA TECHNOLOGIES GMBH (ALGORITHMICA)	Germany
11	F INICIATIVAS ESPAÑA I MAS D MAS I SLU (FI GROUP)	Spain

Document Details

DELIVERABLE TYPE	Report
DELIVERABLE Nº	2.5
DELIVERABLE TITLE	Assessment of AIROM based optimization performances with respect to state-of-the-art methodologies
NAME OF LEAD PARTNERS FOR THIS DELIVERABLE	CRF
VERSION	2
CONTRACTUAL DELIVERY DATE	M36
ACTUAL DELIVERY DATE	M42
DISSEMINATION LEVEL	Public

Revision History

The following table describes the main changes done in the document since it was created

REVISION	DATE	DESCRIPTION	AUTHOR (ORGANIZATION)
0	26/04/2022	First draft version of deliverable	Luca Miretti (CRF)
1	28/04/2022	Second draft version of deliverable	Nikolaos Kyriazis (ENG)
2	29/04/2022	First complete version of the deliverable	Luca Miretti (CRF)

Disclaimer

Table of contents

1. Executive Summary

This report is a part of work package 2, focusing on AI based design for Aerodynamics, and describes the work performed in Task 2.5, including subtasks 2.5.1 and 2.5.2 and 2.5.3. The results are used to drive the activity in work package 4.

The main goal of Task 2.5 is to integrate methods and tools developed in previous tasks into an optimization tool to be used for the aerodynamic development of new vehicles. A full workflow, starting from the geometric parametrization and aiming to the best aerodynamic shape identification is presented, exploiting machine learning tools for aerodynamic coefficients prediction.

Capabilities of the developed workflow are compared to state-of-the-art technologies generally used among OEMs for the aerodynamic design optimization. Several non ML based optimization approaches, from more conservative to more aggressive, are taken into account to be considered as reference condition.

Applications on simplified shapes following these approaches are presented, to verify the capability of the developed process to correctly predict shape variation effect on aerodynamic performance. A comparative analysis between ML method and standard methods is presented, in order to try to assess potential benefits of the produced output.

This deliverable doesn’t deviate from the plan in regard to its content.
This deliverable deviates from original planning in terms of delivery time (from M36 to M42). The delay is mainly related to reporting activity, while technical activity was completed in time and all the required outputs were delivered to WP4 in order to allow the application of the developed technologies on industrially relevant cases, without significant consequences on the final outcomes of the project. The main reason of the delay was an internal re-organization of the company of the author of the deliverable (CRF), causing an unplanned reduction of the activity on the work package. The discrepancy between planned/performed activities is reflected also on claimed costs from the partner.

2. Optimization workflow overview

This task of the project consists in the coupling of the developed workflow for the prediction of the aerodynamic coefficients with an optimizer, in order to be able to drive the design of new vehicles from an aerodynamic point of view.

One of the main bottlenecks of current state-of-the-art CFD optimization processes is the large computational times required for the design space exploration with high-fidelity models. Reduced order models, by definition characterized by more affordable cost, will be exploited in order to try to speed-up the whole process. Machine learning based methods developed in the previous tasks are here summarized, in conjunction with an optimization workflow having aerodynamic coefficients (Cd) as target.

In this chapter a general overview of the whole process, from geometry parametrization to optimum identification, is presented. This methodology is then delivered to Work Package 4, to be used for the optimization of actual vehicle shapes (city car and SUV).

2.1. Geometry parametrization

General approach used for aerodynamic optimization starts from a parametrization of the shape, that can be obtained directly with a parametric CAD or within specific CAE pre-processors and CFD software. The most common process in use among OEMs for production activities is generally the latter due to the easier integrations into CFD workflows.

An automated tool for geometry morphing setup and execution has been in developed in BETA CAE ANSA® environment, since the pre-processor is available among all the partners involved in the project and is generally the most used tool in the worldwide aerodynamic community.

The developed tool is able to create about twenty different “standard” geometrical parameters, requiring as input only a naming convention. In the list of the available parameters the most common geometrical modifications that are of interest in the early development phase, when generally significant style modifications are allowed, have been included. The current available parameters that can be generated are reported in Table 1:

Table 1: List of parameters generated by the automatic tool

ID	Parameter Name	Parameter Description
1	SpoilerAngle	Rotation of the rear spoiler
2	TailLightSpan	Y-position of the tail-light
3	CpillarNolderSpan	Y-position of the C-pillar nolder
4	RearWindowSlide	Extension of the roof
5	RearWindowZtranslation	Z-position of the roof trailing edge
6	RearEndTaperRatio	Boat-tailing of the upper body
7	RockersYtranslation	Y-position of the rocker/door-sill
8	WindscreenXtranslation	X-position of the windscreen trailing edge
9	WindscreenZtranslation	Z-position of the windscreen trailing edge
10	HoodZtranslation	Z-position of the hood trailing edge

11	HoodXtranslation	X-position of the hood trailing edge
12	FrontWheelCovering	Modification of the front bumper in front of wheel region
13	AirIntakeLowerYtranslation	Lower air intake Y extension
14	AirIntakeLowerZtranslation	Lower air intake Z extension
15	AirIntakeMiddleYtranslation	Middle air intake Y extension
16	AirIntakeMiddleZtranslation	Middle air intake Z extension
17	AirIntakeUpperYtranslation	Upper air intake Y extension
18	AirIntakeUpperZtranslation	Upper air intake Z extension
19	FrontWheelSpatsHeight	Front wheel spats Z-extension
20	RearDiffuserZtranslation	Rear diffuser angle

The generation of the parameters is driven by a python script that can be launched directly in ANSA environment. Current list of parameters could be easily extended to similar geometry modifications (translations, rotations, …), thanks to the modular setup of the script.

```
#---------------------------------------------------------------------------------------------------
#function to create DFM translate paramter for single deformations
def _createDFMtranslateParameter(paramName, translateElements, translateAxis, morphedEntities):

    translateArgs = []
    morphingBounds = None
    autoBounds = True
    translateArgs.append(morph.MorphCreateDFMTranslateArgs(translateElements, translateAxis[0], translateAxis[1], translateAxis[2]))
    paramDFMtranslate = morph.MorphParamCreateDFMTranslate(paramName, translateArgs, morphedEntities, morphingBounds, autoBounds)
```

Figure 1: Extract of the script to generate parameters

An example of PID definition required by the tool is shown in Figure 2:

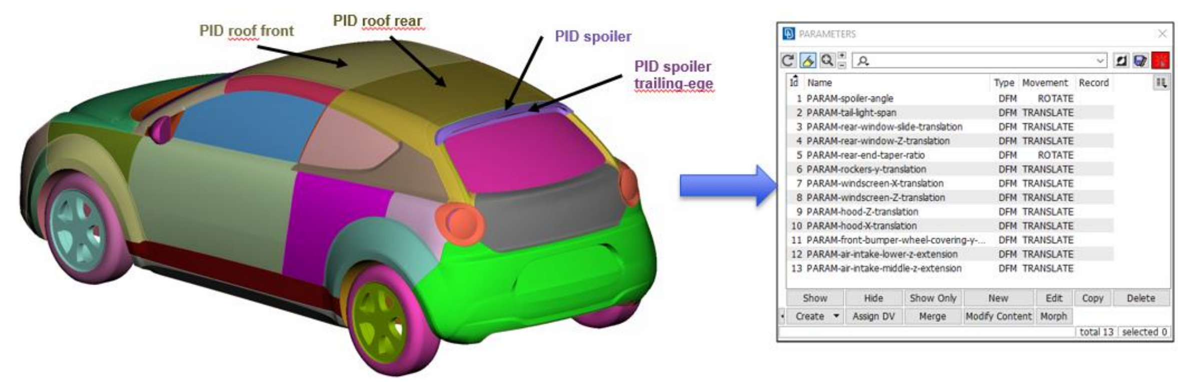

Figure 2: Example of PID definition on the ANSA model

An example of morphed shape obtainable with the abovementioned parametrization (spoiler angle) is visible in Figure 3:

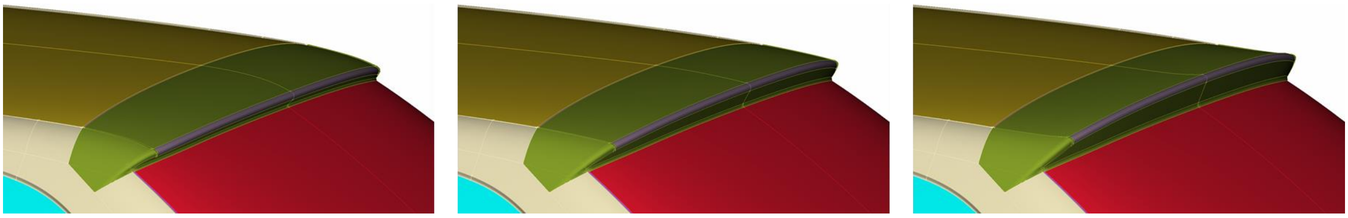

Figure 3: Example of deformation obtainable with the automatically generated “SpoilerAngle” parameter

The parametrization based on this approach is able to reduce the timing for this phase of about one order of magnitude, considering the same number of parameters, with respect to a typical manual approach generally used in the automotive industry.

The tool has been delivered to all partners in Work Package 4 to be used for the geometrical parametrization of the industrial vehicles involved in the application phase of the project.

2.2. DOE shape generation

In order to train surrogate models an initial dataset of CFD configurations needs to be generated: in the proposed workflow these shapes are generated by shape morphing. The initial dataset, however, can be populated following different approaches (parametric CAD, available geometries from previous programs, style proposals, …).

Considering the shape morphing approach and a number of geometrical parameters between 10 and 15, according to experience developed in Work Package 1 and previous tasks of Work Package 2, a number of different shapes of the order of one thousand is required to achieve a reasonable accuracy in the prediction of aerodynamic coefficients and flow fields. This can be considered a limitation of the proposed approach, similarly to all AI techniques: if data are not already available, the “offline” phase could be a bottleneck in terms of time consumption.

Proposed approach relies on SOBOL sequences generation, quasi-random techniques able to fill the sampling space in a uniform way, without recognizable patterns.

2.3. Offline data generation

In this section, the workflow prior to the training procedure and the framework for automating it is described. This framework is robust and capable of handling different types of datasets, and either 2D or 3D geometries.

The workflow in brief includes:

1) geometry generation (see Section 2.2)

2) meshing (helyxHexMesh for 3D cases)

3) setup CFD configuration (caseSetup)

4) running the CFD solver (helyxCoupled)

5) resampling and additional post-processing of the results (helyxSample)

6) transformation of the data obtained by CFD to a format compatible with the ML library

The whole workflow, as it is proposed in this work, is performed using ENGYS® OpenFOAM (1) and it is performed automatically for each configuration of the DOE. Additional details are available in D2.2.

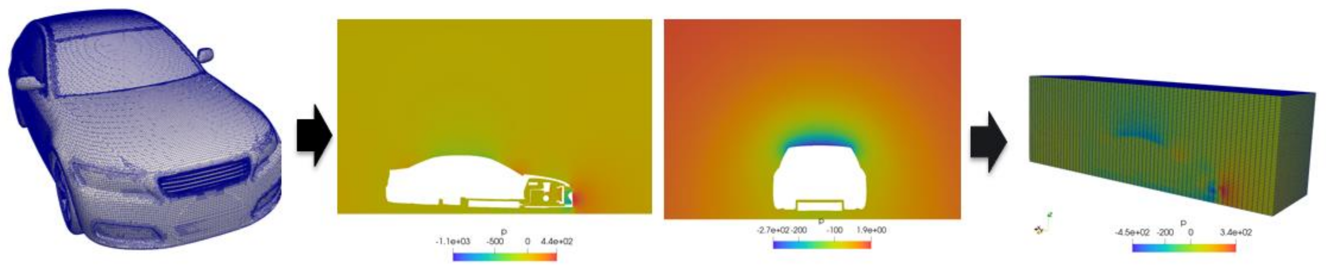

Figure 4: CFD workflow from volume mesh to data extraction

In the application example exposed in this deliverable (DrivAer 2D closed AGS, see D2.2 for test case description and (2)) the whole initial set of data was available from previous activity in the work package, and the generation of these “offline” data was not necessary for the developed optimization workflow.

2.4. Predictive model

The aerodynamic coefficients are predicted here by a regression tool, having as input the physical (CAD) parameters of each sample. More details about the regressor can be found in D1.3, section 3.4. The regression tool was preferred over the ANN classification architecture, which was described in detail in D2.2, due to its robustness and its lower computational cost. In addition, following the former methodology there is the capability to use as inputs the parameters from the latent space of the auto-encoder and thus, making the process more generic in the event of not having the physical CAD parameters of the samples.

Here, in order to make the regression tool as accurate as possible, the whole dataset consisting of 1050 samples has been used as the training dataset. The predictions which will be performed, will be for the geometries generated by the genetic algorithm (see next section) and therefore, CFD data won’t be available. CFD verification will be performed for the optimum geometry which will be found in the end of the optimization loop. Different regression tools have been employed, such as Linear regression of 1st and 2nd order, Gradient Boosting, KNN, ANN and Kriging. Based on previous numerical experiments (see Table 3 in D1.3), Kriging was selected here as the most accurate tool for predicting drag coefficient.

2.5. Optimization process

The above-mentioned predictive tool has been combined with a genetic algorithm optimization library to find new more aerodynamic shapes, i.e. to minimize the drag coefficient. For that purpose, PyGAD library (3), which is an open-source Python toolkit for evolutionary algorithms and compatible with PyTorch, has been employed and incorporated in the workflow. The drag coefficient of the new species is predicted by the regression tool instead of the traditional CFD

calculation and thus, the whole process is less computationally intensive and can be performed in real time.
As a preliminary investigation, a single objective optimization experiment was performed (drag coefficient minimization), while the CAD parameters were restricted to their acceptable range, in accordance with the parameter range introduced in the original closedAGS database.
PyGAD works with a maximization function, i.e. the solution evolves towards the direction that the fitness function is maximized. The fitness function introduced here is:

$$f = \frac{1}{|C_{d,pred} - C_{d,target}|}, \; C_{d,target} = 0, \qquad \text{(Eqn 1)}$$

since the objective is to minimize the drag coefficient.
The above methodology will be expanded in WP4 to multi-objective experiments for 3-D real car geometries, where the mass flow rate and the drag coefficient will be predicted by the regression tool and the second objective (mass flow rate) will act as a restriction i.e. mass flow rate should be higher than the 90% of the mass flow rate of the baseline case.

3. Reference optimization methods

In this chapter an overview of the chosen optimization approaches to be used as reference is presented. Most popular optimization workflows in use among car makers for aerodynamic optimization can be divided into two different categories: standard optimization methods (4) and adjoint-based method (5) (6). In this activity only the first category is considered, since more similar with respect to what is implemented here.

One of the main bottlenecks of these processes is the large computational times required for the design space exploration with high-fidelity models, reason for which they can be coupled with Response Surface Method techniques. Two different optimization setups are considered here: one more conservative, where all configurations requested by the optimizer are evaluated with full CFD simulations, and one more aggressive by coupling different stages of “virtual” space exploration and CFD verifications. More details on the adopted workflows are reported in the following paragraphs.

Optimizations are driven by ESTECO® modeFRONTIER commercial software, using a Non-dominated Sorting Genetic Algorithm II (NSGA-II)

3.1. Full CFD-based approach

In this approach the optimization driver is directly coupled with the high-fidelity data: each configuration requested from the algorithm is evaluated by a full CFD simulation.

Size of the population is equal to the number of the cases in the initial DOE: for the optimization of the 2D DrivAer shape with this approach 50 individuals are selected with a random algorithm. A maximum number of 50 generations is fixed, all parameters of the optimizer are setup according to Figure 5:

Parameters		
Number of Generations	[1,5000]	50
Maximum Number of Evaluations	[1,500000]	1000
Algorithm type		Original NSGA-II algorithm
GA Operators		
Crossover Probability	[0.0,1.0]	0.9
Mutation Probability for Real-Coded Vectors	[0.0,1.0]	1.0
Mutation Probability for Binary Strings	[0.0,1.0]	1.0
Advanced Parameters		
Automatic Scaling for Mutation Probability		☑
Distribution Index for Real-Coded Crossover	[5.0E-4,100.0]	20.0
Distribution Index for Real-Coded Mutation	[5.0E-4,500.0]	20.0
Crossover Type for Binary-Coded Variables		Simple
Random Generator Seed	[0,999]	435
Category Parameters		
Categorize Generations		☑

Figure 5: NSGA-II setup for full-CFD optimization approach

3.2. Mixed CFD-RSM-based approach

In this approach the optimization driver should guarantee faster convergence. This FAST optimizer uses Response Surface Models (Meta-Models) to speed up the optimization process.

FAST is an optimization algorithm combining real and RSM-based (virtual) optimization strategies. Both real and virtual optimization are performed by one of the evolutionary or heuristic algorithms for solving single and multi-objective problems.

FAST search scheme:

- 3 RSMs are trained for each objective and constraint using a database of designs: Polynomial SVD, Radial Basis Functions and Kriging

- The chosen optimization algorithm (NSGA-II) performs in parallel both the real and the virtual optimization.

- Candidate designs coming from the virtual optimization/virtual exploration and the real optimization phases are sent to the CFD solver for evaluation

The fitness of the RSM previously used for virtual optimization is evaluated using a performance metrics (mean normalized error). More details on the approach can be found in software user guide (7).

Parameters		
Number of Iterations	[1,500]	20
Select the Space Filling Algorithm		Incremental Space Filler
Select the Algorithm		NSGA-II
Training Set Policy		Use all points
Maximum Size of Training Set	[50,5000]	500
Fix Performance Threshold for RSM Competition		☐
Normalized Error Threshold Value	[1.0E-15,0.1]	1.0E-10
Advanced Parameters		
Use Design Table as Initial Database		☑
Real/Virtual Optimization Ratio	[0.0,1.0]	0.0
Exploration/Optimization Relative Ratio	[0.0,1.0]	0.8
Run all the requested evaluations		☐
Random Generator Seed	[0,999]	641

Figure 6: Setup of FAST-NSGA-II algorithm

As in the previous case an initial population of 50 individuals is generated with a random algorithm, then for this approach a maximum number of 20 generations is fixed for the optimization of the thirteen geometrical parameters on the 2D DrivAer model.

4. Application on DrivAer 2D model

DrivAer 2D shape with 13 different geometrical parameters is here optimized following procedure developed in UPSCALE project and with the two identified reference methods. Target is the optimization of the drag coefficient.

4.1. UPSCALE workflow

The same workflow as in the previous deliverables (D2.2, D2.3) of WP2 has been employed in the present work. The only difference is that the workflow has been enhanced by the optimization tool, which has been added in the loop.

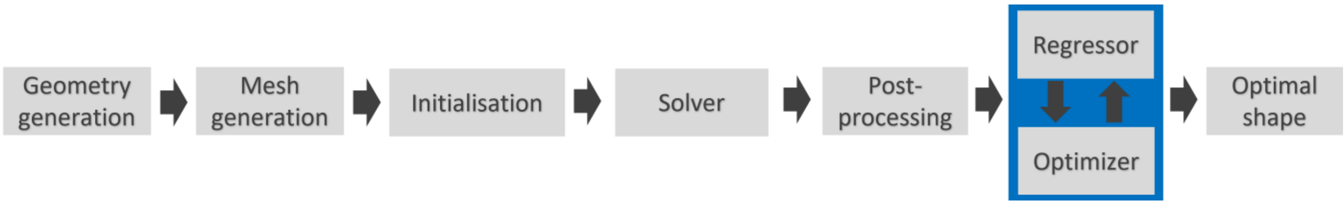

Figure 7: Workflow including CAD design, CFD analysis and regression in conjunction with optimization.

Regarding the ML-based method, the CFD-related workflow is only needed to create the DOE and calculate the aerodynamic coefficients for each sample. This is necessary as these values will be used for training the regressor. Apart from the aerodynamic coefficients, only the CAD parameters of each sample are needed for predicting the optimal shape, which is expressed in terms of the CAD parameters. Then, a CFD analysis is performed to verify the results of the new geometry and check if indeed it has a more aerodynamic shape than the existing ones.

The regression toolkit in connection with the evolutionary algorithm (PyGAD) found a new optimal which has lower drag coefficient than the minimum value documented in the closedAGS database (C_d=0.1594). The new shape has physical parameter space:

$z_{physical}$ = [1.0, 1.5, -0.2, -1.0, -1.0, -0.5, -1.0, -1.5, 1.6, -1.0, -1.0, 50.0, 1.0]

and the drag coefficient was predicted to be C_d =0.1436.

A CFD simulation on the optimal configuration identified by the evolutionary algorithm coupled with the predictor is performed, scoring a C_d value of 0.164. Final improvement obtained with respect to the baseline (C_d =0.233) is still very significant, anyway the discrepancy between the prediction and the confirmation is not negligible in this case.

4.2. Reference workflow

Results obtained by the “full-CFD” workflow and “mixed-CFD-RSM” workflow on DrivAer 2D problem are reported in the following paragraphs.

4.2.1. Full CFD-based approach

Overview of the workflow is presented in Figure 8, while the convergence history of the optimizer is shown in Figure 9.

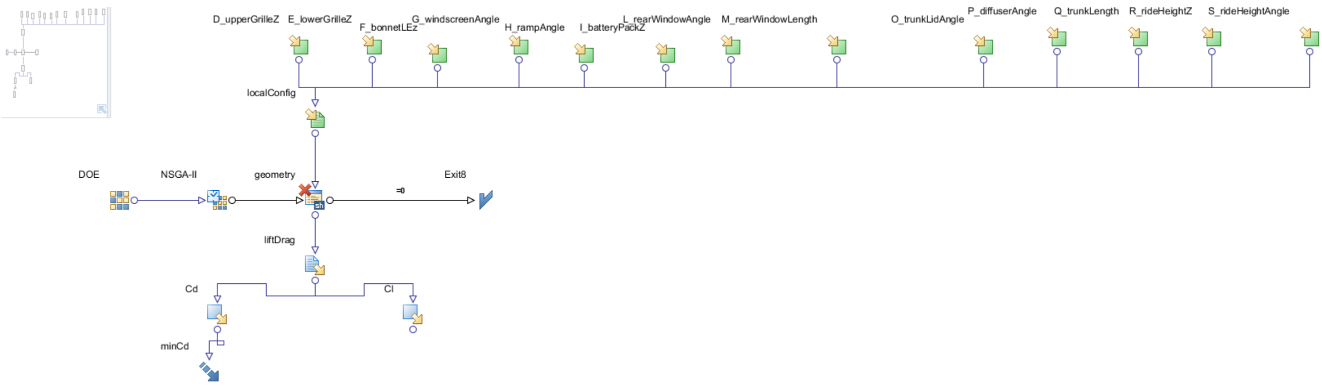

Figure 8: modeFRONTIER workflow for full CFD optimization

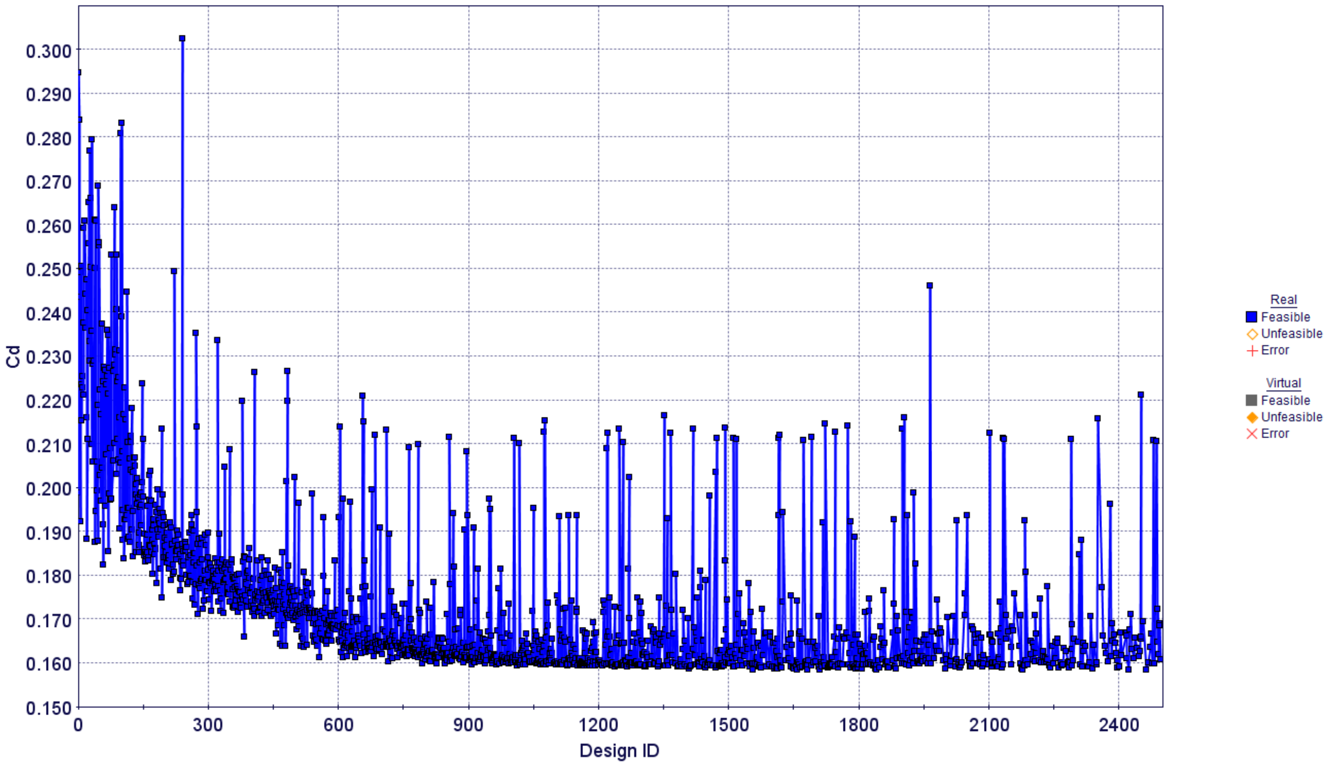

Figure 9: Convergence history of the NSGA-II algorithm

This approach found an optimal configuration (ID 1681) which is characterized by a C_d of 0.159 (evaluated by full CFD simulation). The new shape has physical parameter space:

$z_{physical}$ = [0.6, -0.7, 0.2, -1.0, -0.2, -0.5, 0.6, -1.5, 1.6, -1.0, -1.0, 5.0, 1.0]

A total number of 1733 complete CFD simulations is performed.

4.2.2. Mixed CFD-RSM-based approach

Overview of the workflow is presented in Figure 10, while the convergence history of the optimizer is shown in Figure 11.

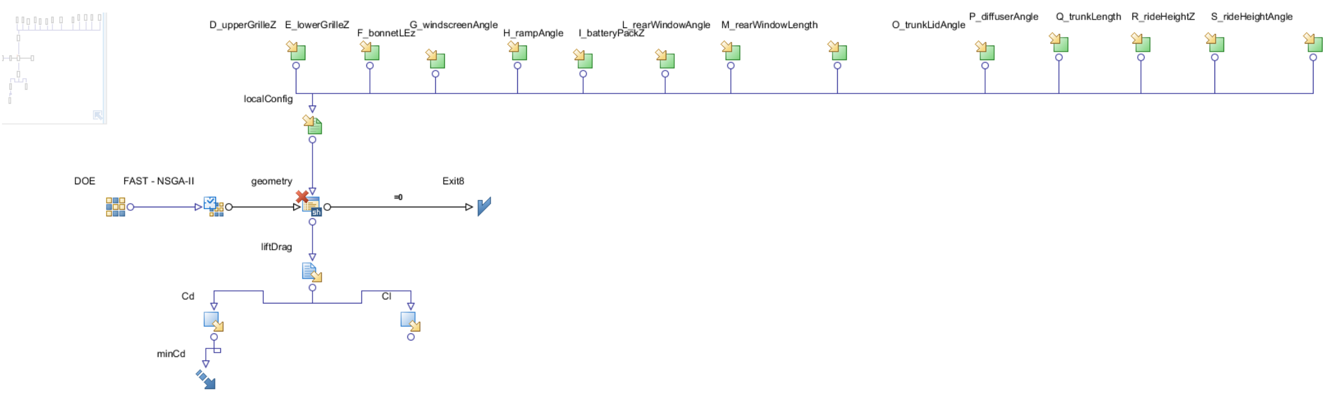

Figure 10: modeFRONTIER workflow for CFD+RSM optimization, using FAST tool

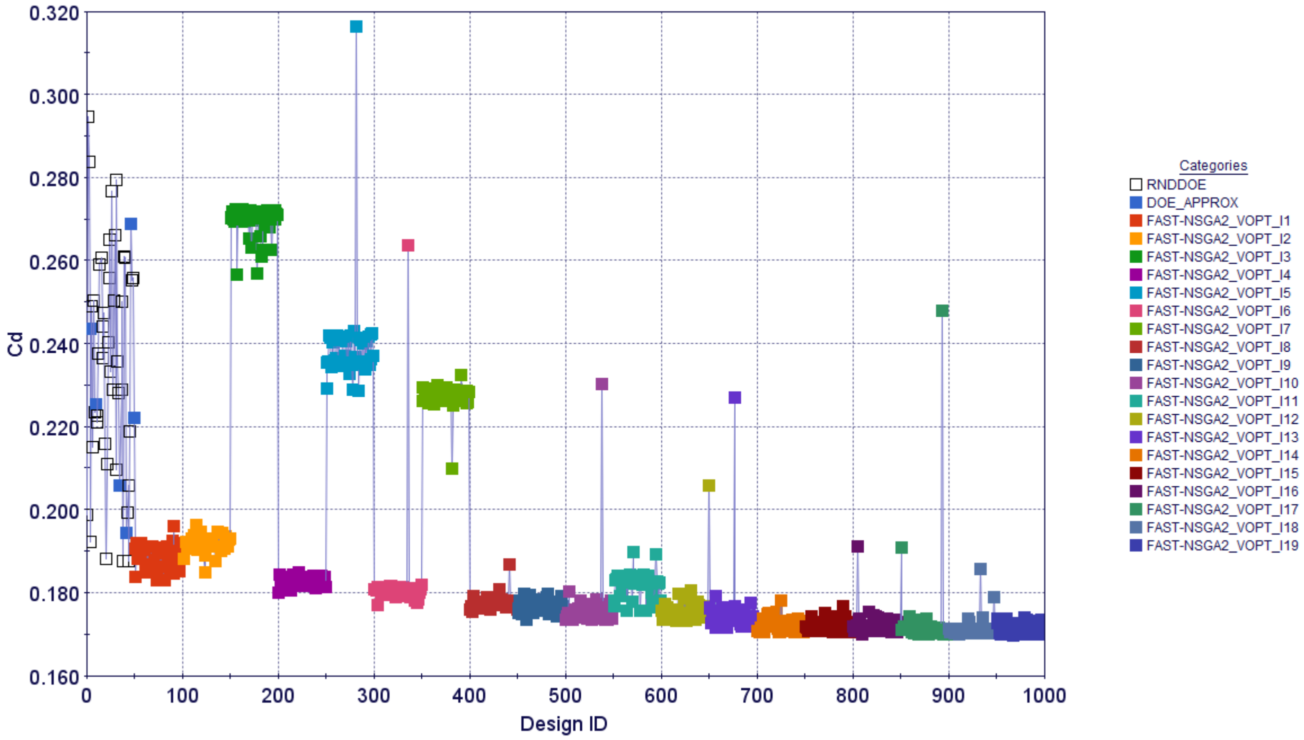

Figure 11: Convergence history of the FAST-NSGA-II algorithm

This approach found an optimal configuration (ID 967) which is characterized by a C_d of 0.170 (evaluated by full CFD simulation). The new shape has physical parameter space:

$z_{physical}$ = [0.0, 0.9, 0.8, -0.2, -0.6, 1.1, 0.6, 1.3, 1.6, -1.0, -1.0, 40.0, 1.0]

A total number of 1000 complete CFD simulations is performed.

5. Methods comparison

A comparison between the different optimization methods is reported in the table below:

Table 2: Comparison of the optimization results

	Optimum C_d (CFD)	Number of high-fidelity evaluations required
UPSCALE workflow	0.164	1000+1
FULL-CFD workflow (ref)	0.159	1733
CFD-RSM workflow (ref.)	0.170	1000
BASELINE	**0.233**	

In terms of optimum identification (lowest C_d configuration) the more conservative approach based on 100% CFD calls confirmed to be the best one, in front of a higher requirement of full CFD evaluations.

The optimum identified with the developed process is instead better with respect to the tested reference mixed CFD+RSM approach, requiring a similar number of tested configurations. It needs to be pointed out that all the CFD evaluations required by the first one are related to the initial dataset generation, with just one additional simulation as final verification. A huge benefit in terms of run-time can be expected, on the other hand, if a parameter free ML environment could be used to exploit already available shapes without the need to create ad-hoc the training data.

An overview of the obtained shapes by the different optimizer is presented in Figure 12, compared with the baseline geometry.

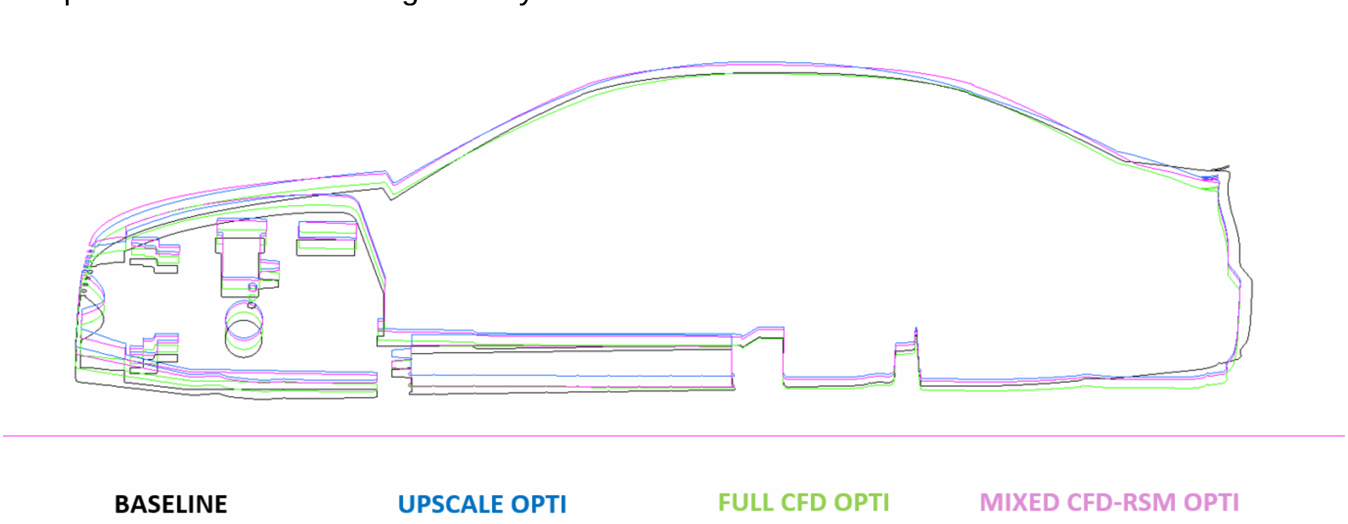

Figure 12: Comparison of the final shapes obtained by different optimization approaches

6. Conclusions and Future Work

An overview of the optimization workflow developed in the UPSCALE project is summarized in this deliverable, combining all the findings and technologies developed in work packages 1 and 2. The tool is delivered to work package 4, in order to verify the capability of the method also on completely industrially-relevant models.

In addition to that, a comparison of the workflow is performed with respect to typical optimization approaches currently in use among OEMs to optimize the aerodynamic performance of the vehicles. The test case is performed on a simplified 2D DrivAer shape, but conclusions can be extended to most complex geometries with respective scaling factors in terms of run-time.

The comparison underlines how the developed method is able to carry out an optimization of the shape of the vehicle, leading to significant improvements in terms of drag coefficient. On the other hand, some limitations are still visible from the process:

- Even if a strong improvement with respect to the baseline is found, the predicted optimum is not completely confirmed by the higher-resolution method, not leading to the global

optimum (to the best of our knowledge) of the problem. The comparison with respect to the FAST algorithm used in this application, on the other hand, is definitely encouraging.
- One of the bottleneck in terms of run-time of the methodology remains the need to create a big initial set of data with high-resolution methods to train the predictor. A huge benefit in terms of run-time can be expected, on the other hand, if a parameter free ML environment could be used to exploit already existing results. This point can be indicated as one of the most interesting area of research for future activities and projects.

ACKNOWLEDGEMENT

The author(s) would like to thank the partners in the project for their input, valuable comments on previous drafts and for performing the review.

Project partners:

PARTICIPANT Nº	PARTICIPANT ORGANISATION NAME	COUNTRY
1 (Coordinator)	IDIADA AUTOMOTIVE TECHNOLOGY SA (IDIADA),	Spain
2	VOLVO PERSONVAGNAR AB (Volvo Cars)	Sweden
3	VOLKSWAGEN AG (VW)	Germany
4	CENTRO RICERCHE FIAT SCPA (CRF)	Italy
5	ESI GROUP (ESI GROUP)	France
6	ENGYS LTD (ENGYS LTD)	United Kingdom
7	Kompetenzzentrum - Das Virtuelle Fahrzeug, Forschungsgesellschaft mbH (VIF)	Austria
8	VRIJE UNIVERSITEIT BRUSSEL (VUB)	Belgium
9	ECOLE NATIONALE SUPERIEURE D'ARTS ET METIERS (ENSAM PARISTECH)	France
10	ALGORITHMICA TECHNOLOGIES GMBH (ALGORITHMICA)	Germany
11	F INICIATIVAS ESPAÑA I MAS D MAS I SLU (FI GROUP)	Spain

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement no. 824306

7. References

1. **HELYX Open-source CFD for Enterprise. [Online] ENGYS, April 2020. https://engys.com/products/helyx.**

2. ***Introduction of a New Realistic Generic Car Model for Aerodynamic Investigations.*** **A. Heft, T. Indinger, N. Adams. Detroit, Michigan, USA : s.n., April 23-26, 2012.**

3. ***PyGAD: An Intuitive Genetic Algorithm Python Library.*** **Gad, A. F. s.l. : arXiv, 2021. 2106.06158.**

4. ***Optimizing the External Shape of Vehicles at the Concept Stage.*** **Ribaldone, Enrico, et al. 2, s.l. : SAE International Journal of Engines, 2011, Vol. 4. 2011-37-0014.**

5. ***Application of Adjoint Methods on Drag Reduction of Current Production Cars.*** **Francesconi , Gianluca, et al. s.l. : SAE, 2018. 2018-37-0016.**

6. ***Application of the Adjoint Method for Vehicle Aerodynamic Optimization.*** **Blacha, Thomas, et al. s.l. : SAE, 2016. 2016-01-1615.**

7. ESTECO. modeFRONTIER User Guide. 2016.