ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

THE RBF4AERO BENCHMARK TECHNOLOGY PLATFORM

Massimo Bernaschi[1], Alessandro Sabellico[1], Giorgio Urso[1], Emiliano Costa[2], Stefano Porziani[2], Fabrizio Lagasco[2], Corrado Groth[3], Ubaldo Cella[3], Marco Evangelos Biancolini[3], Dimitrios H. Kapsoulis[4], Varvara G. Asouti[4], Kyriakos C. Giannakoglou[4]

[1]National Research Council of Italy - Institute of Applied Computing
via dei Taurini, 19, 00185, Rome
{m.bernaschi, a.sabellico, g.urso}@iac.cnr.it

[2] D'Appolonia S.p.A.
Viale C. Pavese, 305, 00144, Rome
{emiliano.costa, stefano.porziani, fabrizio.lagasco}@dappolonia.it

[3] University of Rome Tor Vergata
Dept. of Enterprise Engineering
Via Politecnico 1, 00133, Rome
corrado.groth@uniroma2.it, ubaldo.cella@designmethods.aero, biancolini@ing.uniroma2.it

[4] National Technical University of Athens, School of Mech. Eng.
Parallel CFD & Optimization Unit
Athens, Greece
jim.kapsoulis@gmail.com, vasouti@mail.ntua.gr, kgianna@central.ntua.gr

Keywords: Optimization, Evolutionary Algorithms, Adjoint Coupling, Mesh Morphing, Radial Basis Functions, Benchmark Technology, Multiphisics

Abstract. *This paper presents the RBF4AERO benchmark technology platform, developed in the framework of the EU-funded RBF4AERO project. The platform enables the so-called Benchmark Management System (BMS) used for benchmark submission and results reporting. The BMS is deployed using three modules, namely the Graphical User Interface (GUI), the Workflow Manager (WM) and the Benchmarking Database System (BDS) which cooperate during the whole optimization benchmark life-cycle. The GUI is the only component which interacts with the end-user. It enables the optimization benchmark submission, along with the progress, results and computational platform resources monitoring. The configuration of the Optimization (OT) and the Morpher Tool (MT) is a pre-requisite for the optimization benchmark submission. In an optimization scenario the WM, which is practically the controller of the system, queries the OT in order to get a table of samples and gives back the results of the simulator (for instance a CFD tool). The evaluated individuals serve as training patterns of a Response Surface Model (RSM) which is, then, used for an Evolutionary Algorithms based optimization. The resulting 'optimal' solution(s) are delivered back to the WM for re-evaluation on the CFD tool. For each evaluation on the CFD tool, when a new geometrical shape is required, the computational grid is morphed using the MT based on radial basis functions.*

1 Introduction

The final goal of the RBF4AERO Project `http://www.rbf4aero.eu` is the development of a Benchmark Technology, namely a dedicated numerical platform and strategy capable to allow aeronautical design engineers to build up the novel optimization environment by using their own numerical models and computing platforms, and achieve the results of multi-objective and multi-disciplinary optimization studies in shorter time with respect to current practices. Besides, the RBF4AERO numerical platform enables to solve other relevant aircraft design studies such as fluid-structure interaction (FSI) and icing growth, and proposes a challenging CFD optimization technique that foresees the adjoint-morphing coupling. The basic idea of the optimization strategy is to make the CFD model parametric through an innovative shape optimization environment based on a high performance meshless morphing technique. The technique is founded on Radial Basis Functions (RBF) approach which offers a number of distinct advantages over the more traditional optimization approaches with no need to face with typical limiting trade-off constraints of speed, accuracy and extent, here the speed is related to the time required by the overall optimization process, the accuracy is achieved using large CFD meshes, and the extent is related to the number of different configurations fully calculated during the optimization process (see Figure 1).

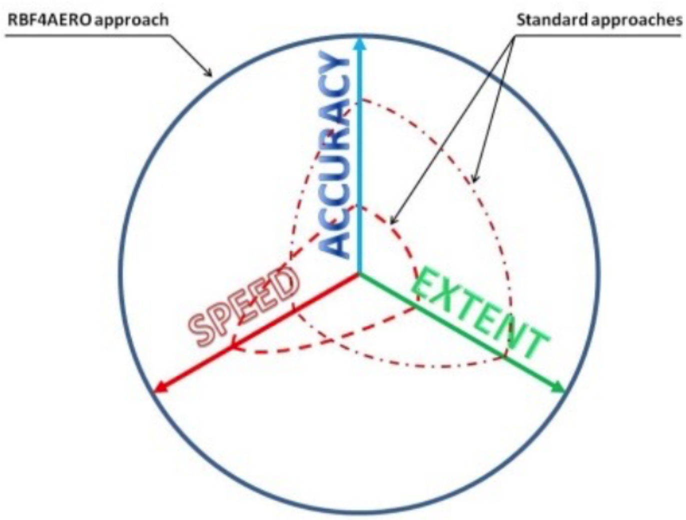

Figure 1: Trade-off between accuracy, time of computing, and number of analysed variants using traditional and the RBF4AERO proposed approach.

2 Description of Work

The scientific and technological enhancements offered by the RBF4AERO techniques require a convincing verification process before entering the industrial practice. For this reason, the Project work plan develops a robust industrial based process divided into three principal tasks:

1. Benchmark Technology Infrastructure Development
   The novel methodological procedure for the computational-driven optimisation proposed in the Project implies the development of an appropriate infrastructure to set-up the op-

timisation environment and enable the simulation of test cases of industrial relevance. A major role in achieving this goal will be played by parallel processing;

2. Benchmark Technology Verification
   Preliminary verification of results accuracy will be achieved on published state-of-the-art reference applications or available industrial cases;

3. Benchmark Technology Numerical Testing
   The optimisation procedure on demanding industrial applications will be carried out and numerically validated by the Project End Users.

Critical analysis of the numerical predictions of morphed configuration with respect to the baseline is supported and complemented by experimental outputs provided within the project. Finally, the effectiveness as well as the efficiency of the overall RBF4AERO optimization procedure is demonstrated.

3 Benchmark Technology Infrastructure capabilites

The architecture of the RBF4AERO numerical platform is thought as a control and communication manager framework where different modules are seamlessly integrated according to the available hardware and software. Looking at the platform as a whole, this integrated system is required to accomplish three principal functions: optimization, mesh morphing and simulation. The first two functions are carried out by software internal to the Benchmark Technology, whereas the latter one by external numerical solvers. An outline of the Benchmark Technology optimization process is depicted in Figure 2.

In specific, the optimization techniques implemented in the RBF4AERO platform are:

1. optimization algorithms based on Evolutionary Algorithms (EAs);

2. optimization based on a gradient algorithm and the continuous adjoint solver;

3. optimization based on Adjoint Self Sculpting.

The EA-based optimization is assisted by metamodels trained on data collected during the Design-of-Experiment (DoE) phase and supported by the Response Surface Models (RSM) that reduces the number of evaluation tool calls [1]. In the case the CFD solver is OpenFOAM and the objective function is the drag, lift or the pressure loss, the user can also exploit the capabilities of the continuous adjoint solver and of the innovative adjoint-morphing coupling. In particular, two algorithms foreseeing the coupling between the adjoint solver and the MT, called gradient-based [2] and Adjoint Self Sculpting algorithms, can be used to perform shape optimization. Moreover the Adjoint Preview feature, in the case multiple shape variations are available, can be adopted to identify the most influent ones. Relating to FSI two methods, respectively referred to as mode-superposition and two-way, are available [3]. According to the mode-superposition method, the CFD model is made flexible imposing structural displacements through mesh morphing by the superposition of its natural modes provided either by an analytic method or by means of a FEM model. The two-way technique foresees instead the coupled use of CFD and structural FEM solvers and the exchange of data between them. Using both approaches the MT allows to apply the deformation to selected wet surfaces only by constraining the other rigid bodies. With regard to icing, two methodological approaches have been designed [4]. According to the first one, termed frozen or constrained, icing simulations are carried out by

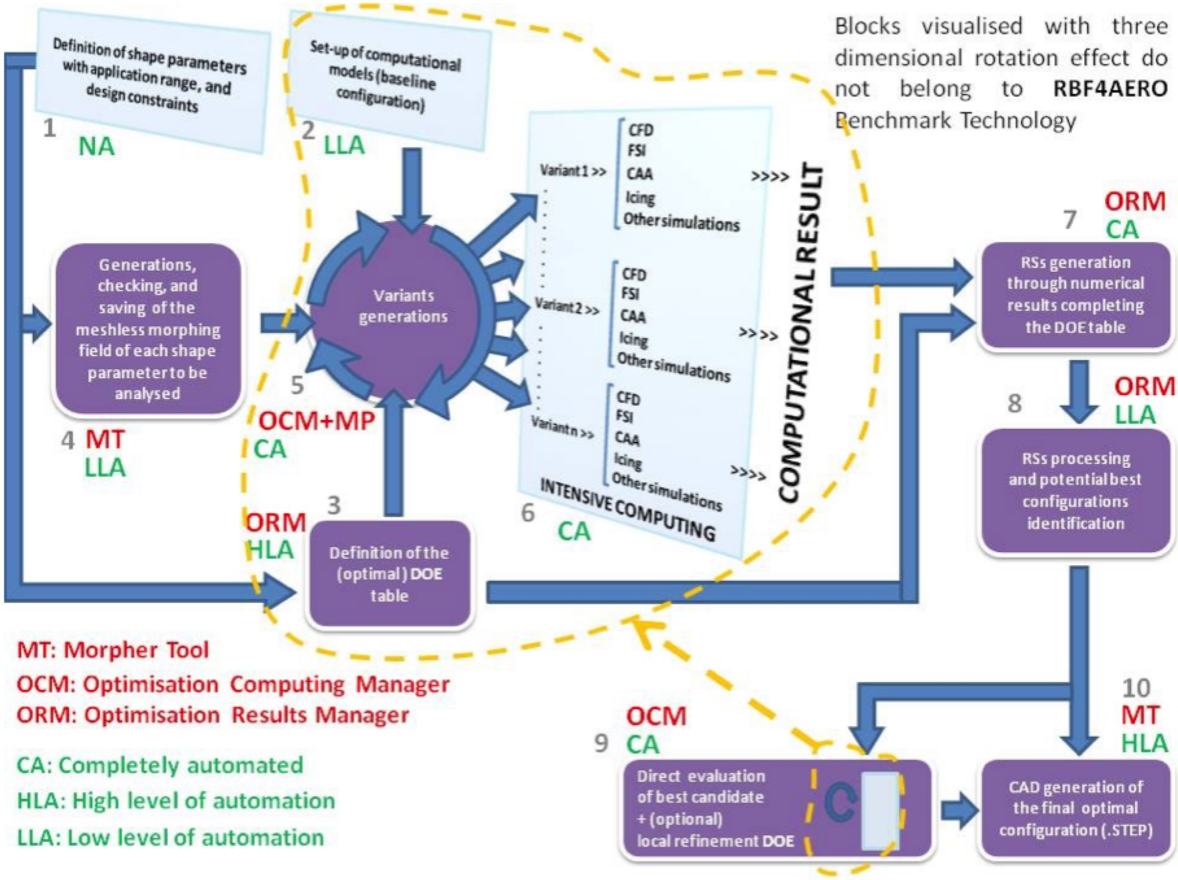

Figure 2: RBF4AERO Benchmark Technology optimization process.

imposing, at specific iterations of the CFD computing, the icing profiles previously calculated by means of an icing accretion tool at predefined instants of time. The second one, referred to as on-the-fly or evolutionary, foresees the use of an accretion code that, in conjunction with a CFD solver, modifies dynamically the numerical grid according to the calculated ice accretion.

4 Benchmark Management System

The **Benchmark Management System (BMS)** is used by the end-user for benchmark editing, benchmark submission, platform monitoring and results reporting. One of the main requirement of the software was to be as modular as possible, in order to be deployed in the largest possible number of system types and be able to manage the largest possible number of solvers. Three software modules have been developed to accomplish all the requirements:

- **Graphical User Interface (GUI)** give the end-user the possibility to interact with the platform by submitting user benchmarks to the BDS;
- **Workflow Manager (WM)** which executes the user submitted benchmarks by evenly distributing workload to platform nodes;
- **Benchmarking Database System (BDS)** which holds all information about user submitted benchmarks; it is the communication channel between GUI and WM.

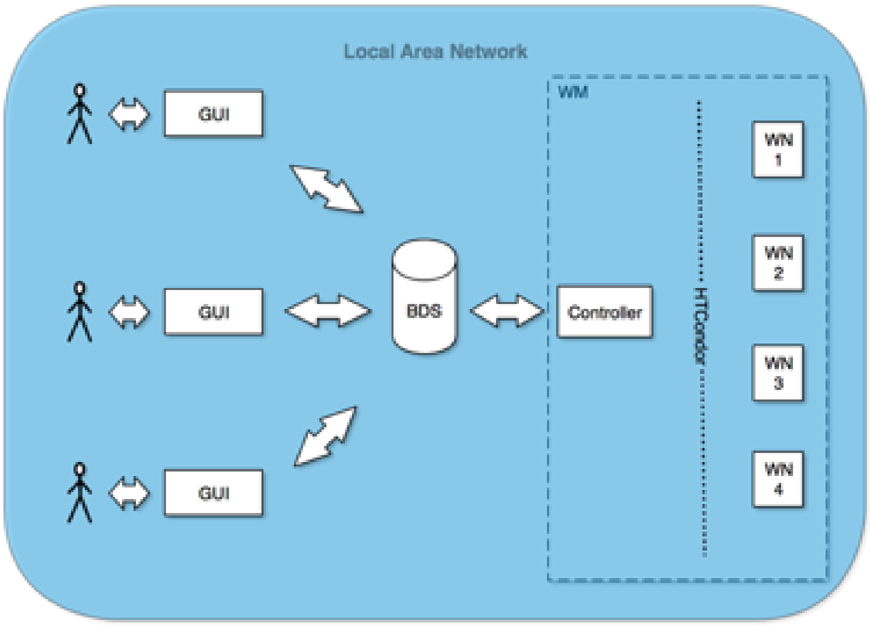

Figure 3: Benchmark Management System simplified scheme.

These three modules cooperate in order to assist the end-user during the whole optimization benchmark life-cycle.

4.1 Graphical User Interface

In order to guarantee compatibility on the majority of operating systems, GUI is based on the multi-platform Qt framework. The Graphical User interface is the only component which interacts with the end-user; it basically gives the possibility to:

- submit a generic benchmark
- show benchmark progresses and results
- monitor the available platform resources load

Benchmark submission is the procedure for the end-user to setup an optimization benchmark. The benchmark can be a shape optimization procedure based on the methods previously described, a multi-physics simulation or both. During this phase the user will input: the computational model prepared for a generic simulator/CFD, the RBF material needed during the morphing phase, all the optimization parameters needed for the specific benchmark type. Once the input is finished, the user can submit the benchmark which is then delivered to the Benchmarking Database System.

Benchmark visualization After the benchmark is submitted, it will be dispatched by the Workflow Manager. The user can follow all the progresses within the GUI, which is automatically updated while the information is inserted in the BDS. Benchmark visualization includes tabular views of data, various post processing graphs drawn using live data, platform logs for all messages and outputs/errors generated during the benchmark execution.

Resources monitoring For each platform computation node, the Workflow Manager is also responsible for updating node load information inside the BDS. This information is then received by the GUI which shows it to the end-user.

4.2 Workflow Manager

The workflow manager is the module which coordinates all the computing activities in the system. It queries the optimizer in order to get a table of candidate solutions, then represents the generated candidate solutions inside the BDS, executes every single candidate solution by scheduling jobs to be ran on the installed working nodes and gives back the results of the various jobs into the optimizer, writing the results also in the BDS as they can be read by the GUI and shown to the end-users. During jobs execution all the log information is stored by the WM on its log files, and all the important log messages produced by the running jobs are also stored in the BDS. When the user requests for these informations, the GUI will query the BDS and show these messages on a dedicated panel.

WM runs as a service on the hosting server and can be stopped and resumed at any moment, as all the benchmark and computed data are stored in the BDS and can be retrieved when the service is restarted. This means a lot of time saved when a running benchmark is stopped by the user or by the hosting machine malfunctioning.

Job scheduler

The **Job Scheduler** layer separates the WM component in two sub-layers: controller side and worker side. In the former, jobs are defined and delivered to a **Job Sender** that packs and entrusts them to the **Job Scheduler** layer; in the latter, executed on a potentially different system node, all the CFD/solver specific computation is done.

This way the controller node does not have to deal with node selection in job assignment, it has just to assure right dependencies between jobs. Before the assignment of a job to a particular node, the **Job Scheduler** checks if the node has access to all the solvers and resources needed in order to complete the job. Once the node has been selected, resources on that node will be allocated and the job will be prepared to be sent.

Finally, on the worker side, the workflow manager will receive the job from the **Job Scheduler** layer, will unpack the job and dispatch it to the right helper. When the job is completed, **Job Scheduler** will take care of copying the output results (a set of files on disk) to the controller node, which then parses them and updates the information in the **BDS**. Also, on a benchmark type basis, some important folders in the job working directory will be saved and referenced as a *Resource* inside the **BDS**. The end-user can request these *Resources* from a dedicated panel in the **GUI**. In our implementation we used the open-source workload management system **HTCondor**[1], which handles all the job scheduling operations during node selection and job sending.

[1] HTCondor software website: `https://research.cs.wisc.edu/htcondor/index.html`

Worker-side helpers

Each of the platform nodes communicates with the Controller node only through HTCondor communication protocols and can have any number and type of simulators/CFDs installed. The only constraint is that for each simulator, an Helper must be defined and installed. An Helper is a Python script with a predefined command line arguments set, defined in the Workflow Manager specifications. As the Python interpreter and HTCondor software are implemented in various operating systems, an RBF4AERO platform installation can have an heterogeneous set of worker nodes (WN) and can accomodate a huge number of simulators.

4.3 Benchmarking Database System

The Benchmarking Database System (BDS) is a relational database defined in a PostgreSQL DBMS installation. This module is the glue between the GUI and the WM, as they only communicate through queries to the DBMS. We have chosen this particular DBMS because it natively implements some triggering procedures which can notify tables update to both, GUI and Workflow Manager sides. The fact that all the information about benchmark management is stored inside BDS will also allow future exploitation of the platform using a web app. This will enable end-users to access RBF4AERO platform also outside the LAN where the WM and BDS are installed.

5 Conclusions

In this paper, an overview of the RBF4AERO benchmark technology platform and its working logic was presented. The complex modular structure that compose it was designed in order to be deployed in heterogeneous architectures and to be as much customisable as possible. Multi-objectives and multi-physics optimisation studies can be accomplished in a dramatically shorter times with respect to current practices.

6 Acknowledgements

This work was financially supported by the RBF4AERO project funded in part by the EUs 7th Framework Programme (FP7-AAT, 2007 - 2013) under Grant Agreement no. 605396.

REFERENCES

[1] D.H. Kapsoulis, V.G. Asouti, K.C. Giannakoglou, S. Porziani, E. Costa, C. Groth, U. Cella and M.E. Biancolini *Evolutionary Aerodynamic Shape Optimization through the RBF4AERO Platform*. ECCOMAS Congress 2016 VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete Island, Greece, 510 June 2016.

[2] E.M. Papoutsis-Kiachagias, M. Andrejašič, S. Porziani, C. Groth, D. Eržen, M.E. Biancolini, E. Costa, and K.C. Giannakoglou *Combining an RBF-based Morpher with Continuous Adjoint for Low-Speed Aeronautical Optimization Applications*, ECCOMAS Congress 2016 VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete Island, Greece, 510 June 2016.

[3] M. Andrejašič, D. Eržen, E. Costa, S. Porziani, M. E. Biancolini and C. Groth *A mesh morphing based FSI method used in aeronautical optimization applications*. ECCOMAS

Congress 2016 VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete Island, Greece, 510 June 2016.

[4] M.E. Biancolini, E. Costa, C. Groth, G. Travostino and G. D'Agostini *Reliable mesh morphing approach to handle icing simulations in the aviation sector*. Aircraft Engineering and Aerospace Technology (submitted).