Turbomachinery design by a swarm-based optimization method coupled with a CFD solver

Turbomachinery design by a swarm-based optimization method coupled with a CFD solver

Enrico Ampellio[1c], Francesco Bertini[2b], Andrea Ferrero[1*], Francesco Larocca[1a] and Luca Vassio[3c]

[1]Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, Italy
[2]GE Avio S.r.l., Rivalta di Torino, Italy
[3]Department of Electronics and Telecommunications, Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, Italy

(Received keep as blank , Revised keep as blank , Accepted keep as blank)

Abstract. Multi-Disciplinary Optimization (MDO) is widely used to suitably handle the advanced design in several engineering applications. Such applications are commonly simulation-based, in order to properly capture the physics of the phenomena under study. This framework demands fast optimization algorithms as well as trustworthy numerical analyses, and a synergic integration between the two is required to obtain an efficient design process. In order to meet these needs, an adaptive Computational Fluid Dynamics (CFD) solver and a fast optimization algorithm have been developed and combined by the authors. The CFD solver is based on a high-order discontinuous Galerkin discretization while the optimization algorithm is a high-performance version of the Artificial Bee Colony method. In this work, they are used to address a typical aero-mechanical problem encountered in turbomachinery design. Interesting achievements in the considered test case are illustrated, highlighting the potential applicability of the proposed approach to other engineering problems.

Keywords: MDO, swarm intelligence, discontinuous Galerkin, turbomachinery, CFD

1. Introduction

In advanced engineering, computational tools assume an essential role when dealing with complex design problems (Rao 2009). This field requires efficient improvement strategies, extensive automation of the design process and includes appropriate investigations of the involved physical phenomena. In general, a compromise between accuracy and time/cost is mandatory for the whole design system, especially for the computational simulations adopted to describe the physics. In this regard, a proper tuning and validation of numerical models on reference problems must be accomplished. Furthermore, the adopted optimization strategy should efficiently satisfy the pursued goals while respecting all the constraints. In conclusion, the optimization algorithm and the physical description should be combined by an automatic process which leads to an efficient design platform.

*Corresponding author, PhD, E-mail: andrea_ferrero@polito.it
[a] Professor
[b] Ph.D.
[c] Ph.D. Student

In this framework, several modern applications in engineering require detailed, multipurpose and fast optimizations. The Multi-Disciplinary Optimization (MDO) approach is regularly adopted to manage such problems (Vanderplaats 2007), returning a set of non-dominated candidate solutions, i.e., the Pareto Front. In the present work, the MDO methodology is used to optimize a turbomachinery test case, the well-known T106c low pressure turbine cascade. Its geometry and experimental results are openly available (Michalek *et al.* 2010; Hillewaert *et al*. 2013). A representative simulation-based MDO problem is then set up, estimating both performance and mechanical properties. The objective is to minimize the aero losses while respecting some structural constraints.

In order to address such a problem, a fully automatic MDO platform is developed. It integrates the optimization algorithm directly with the Computational Fluid Dynamics (CFD) simulations, without using surrogates. Although surrogates are exploited in a large class of applications (Jones 2001; Koziel and Leifsson 2013; McCullagh and Nelder 1989) their training phase is a delicate matter (Forrester *et al*. 2008) and hence could not be not cost-effective in terms of Function Evaluations (FEs) when the target functions have high dimensionality and are very complex, noisy and multimodal. Therefore, this work exploits an efficient direct search technique explicitly designed to limit the total number of FEs.

There are several architectures and algorithms suitable for MDO, also when each function evaluation is costly: gradient-based methods, trust region approaches, nature-based algorithms, adjoint techniques and many others (Martins and Lambe 2013; Vestraete and Periaux 2012; Koziel and Yang 2011; Iollo *et al*. 2001). Among them, the meta-heuristic plays a predominant role when dealing with many variables and noisy target functions, whose properties are not known a priori (Talbi 2009). Besides the well-known evolution-based strategies, there are many recent bio-inspired and efficient methods (Yang 2010), such as Particle Swarm Optimization (PSO) and other swarm-based techniques (Bonabeau *et al.* 1999; Kennedy 2010). One of the most promising is the Artificial Bee Colony (ABC) algorithm (Karaboga 2007). Starting from it, the authors developed a high-performance single-objective strategy called AsBeC, which also offers the possibility to provide a partial Pareto set for MDO problems by using the weighted sum approach. The AsBeC algorithm was already successfully employed for real-world turbomachinery multi-objective optimizations (Bertini et al. 2013), giving advantages in comparison to other forefront methods.

The flow field around the T106c blade is studied by means of an adaptive discontinuous Galerkin CFD method, developed by the authors for solving the compressible Reynolds Averaged Navier-Stokes (RANS) equations in two-dimensional domains. This solver can use high-order discretizations in both space and time and it can manage both structured and unstructured meshes and different physical models (Ferrero and Larocca 2013; Ferrero and Larocca 2015).

Details about the adopted CFD solver and the AsBeC algorithm are provided in Section 2 and 3 respectively, while Section 4 covers the engineering test case with the optimization settings and results.

2. Simulation environment

2.1 Discontinuous Galerkin CFD

The blade performance is evaluated by numerically solving the equations of fluid dynamics through a time marching approach. Navier-Stokes equations or Reynolds-Averaged Navier-Stokes equations are integrated on general shaped two-dimensional domains. As far as the RANS

approach is concerned, the low Reynolds k-omega model proposed by Wilcox (2006) is used. This model can give good results for turbomachinery applications in which separation phenomena have to be taken in account (Pacciani *et al*. 2011). The model is implemented in a non-standard manner following the approach of Bassi *et al*. (2005). The physical description is completed by the use of a transition model based on the laminar kinetic energy concept, following the work of Pacciani *et al*. (2011).

The numerical technique is based on a discontinuous Galerkin spatial discretization. The most attractive feature of this choice is the possibility to implement robust high order schemes also in the presence of complex geometries with unstructured meshes. Furthermore, the local nature of the discontinuous Galerkin reconstruction makes it possible to deal with very stretched and distorted elements, such those which are generally used for high Reynolds number flows.

Briefly, the solution inside an element is represented by means of a high order reconstruction projected on a hierarchical modal basis. In this way, several degrees of freedom are used inside each element. In this work a basis obtained from a tensor product of Legendre polynomials is chosen. The basis functions are orthonormalized with the modified Gram-Schmidt procedure, following the approach of Bassi *et al*.(2012). This last aspect is particularly important for preserving the accuracy order when distorted elements are employed (Bassi *et al*. 2012). Curvilinear elements are used at solid walls in order to match the accuracy of the geometrical representation with the accuracy of the solution discretization. In particular, Serendipity mappings are implemented for quadrilateral elements up to the fourth order and mappings up to the third order are used for triangular elements (Onate 2009). The convective fluxes are evaluated according to Osher and Solomon (1982) and Pandolfi (1984).

Diffusive fluxes are computed by a recovery-based scheme (Ferrero *et al*. 2015). For what concerns time integration, both explicit and implicit schemes have been implemented. Steady state simulations, like those considered in this work, are accelerated by means of a fully implicit backward Euler scheme. In this case, the Jacobian of the system is evaluated analytically. The resulting linear system is solved at each time step through the restarted Generalized Minimal Residual algorithm (Saad 2003) with an Incomplete Lower Upper preconditioner.

The time step in the performed simulations is chosen according to the ramping strategy described by Colombo (2011). An additional control is introduced in the present implementation, in order to switch from implicit to explicit (first order) time integration when the time step size drops below a certain threshold. This feature is particularly important during the initial transient in which the cost of the implicit integration would not be compensated by a large time step. The simulations were stopped when the L2 norm of the residuals drops down a fixed tolerance (10e-5).

2.2 Grid generator

The simulation environment is completed by the use of the GNU GPL mesh generation software Gmsh (Geuzaine and Remacle 2009). The meshes employed in this work are composed by a clustered O-type structured grid near the blade, surrounded by an unstructured grid (Fig.1). All the elements are quadrilateral. The distance between the inlet and the leading edge is equal to 0.7 chords, while the outlet boundary is positioned one chord downstream the trailing edge.

Since the governing equations are discretized by a high-order representation, it is fundamental to employ a sufficiently high-order description for the wall boundaries. In particular, curvilinear elements with cubic edges are used. The high-aspect ratio elements inside the boundary layer region can become significantly distorted when the wall edge is curved. In order to prevent the jacobian of the mapping to become negative, the high-order optimization tool available in Gmsh is

used. In this way, the final mesh contains curvilinear edges not only at wall (Fig.2a) but also inside the first layers near it (Fig. 2b).

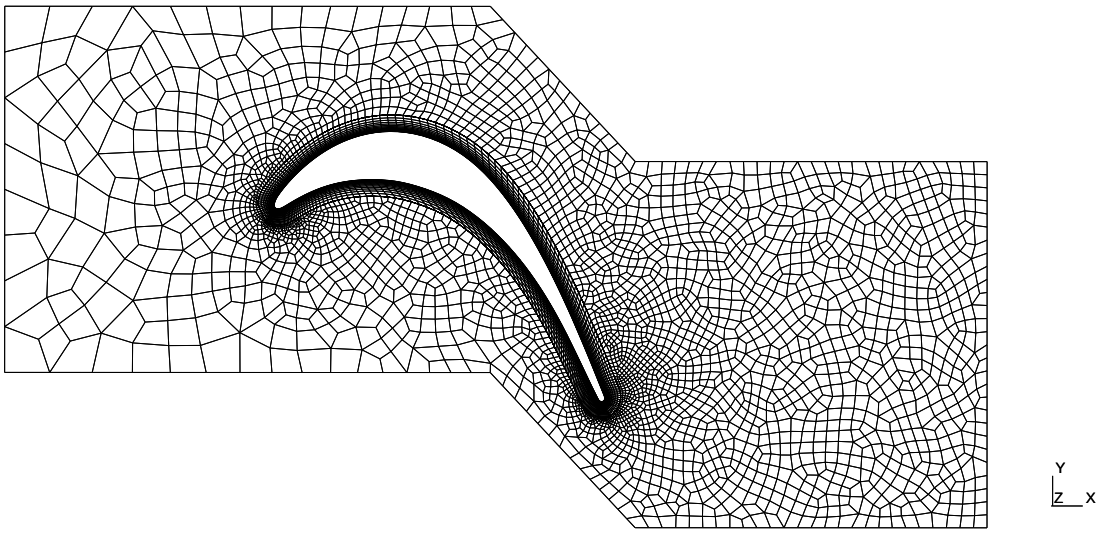
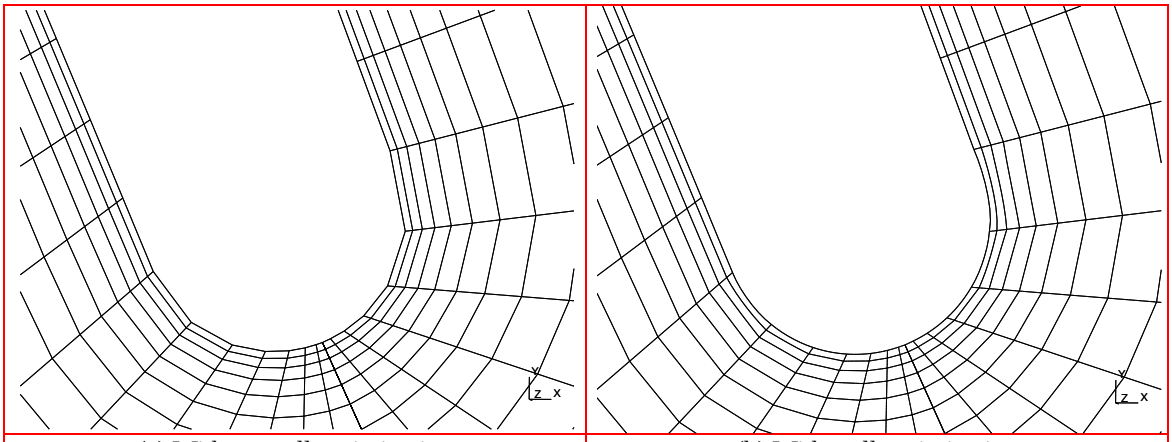


Fig. 1 Example of mesh



(a) Without wall optimization (b) With wall optimization

Fig. 2 Curvilinear mesh generation close to the wall (Detail of the trailing edge)

The hierarchical nature of the chosen basis was exploited to develop a p-adaptive algorithm. This is a useful feature in the framework of automatic optimization problems; indeed, the algorithm refines the solution without the need to change the grid. In particular, an entropy-based sensor is used to detect the boundary layer and the wake and to increase the reconstruction order in the involved elements. In this way it is possible to use a relatively coarse and uniform mesh in the wake region. The order of each element can range from two to four according to the sensor response (see Fig. 7).

3. Optimization environment

3.1 The AsBeC algorithm

The applicative field of engineering optimization is usually characterized by simulation-based problems (Martins *et al.* 2005; Larocca 2008) in which computational analyses are heavily time consuming. Some of the most widespread techniques in the single-objective optimization context lay in the class of meta-heuristic (Glover and Kochenberger 2003). Among them, one of the newest and most promising is the nature-inspired Artificial Bee Colony algorithm (ABC) (Karaboga 2007), which combines the principles of Particle Swarm Optimization theory (Kennedy and Eberhart 1995) and Differential Evolution (Price *et al*. 2005). If compared with other competitive methods, ABC demonstrates high quality, robustness and flexibility for a great variety of optimization problems (Karaboga and Akay 2009). Besides, a lot of technical dissertations, test case applications and improvement works have been done in the recent years (Bolaji *et al*. 2013).

Karaboga firstly developed the Artificial Bee Colony (ABC) algorithm in 2005. The algorithm reproduces the behaviour of a honeybee colony searching for the best nectar source into a target area. Some bees, i.e., the employees, are each assigned to a food source and search the space near it. Then they come back to the hive and communicate the position of the best found food sources to other bees, i.e., the onlookers, that help the employees in the most promising regions. Nectar sources that reveal themselves non-productive are abandoned in place of eventual new fruitful positions, which are investigated by a bee moving in the whole target area, i.e., the scout. In the optimization context, nectar sources represent the input configurations and their nectar amount is the objective value to optimize; non-productive sources are those not improving for some time.

The ABC algorithm is recognized to be simple to implement, easy to be effectively parallelized and hybridized, driven by few control parameters, flexible and robust over a wide range of problems, according to many authors (e.g., Karaboga *et al*. 2014). On the other hand, its local search and refinement skills are less efficient with respect to the global search attitude. Moreover, it does not exploit the history of the best points found.

1: Produce random food sources in the search area
2: Assigm each food source to a different employee
3: **Repeat**
4: **Employee Group phase:** produce new configurations near their food sources updating them whenever any improvement is found
5: Assign each onlooker to one of the food sources, depending on their quality
6: **Onlooker Group phase:** produce new configurations near their food sources updating them whenever any improvement is found
7: **If** a food source is not improving for some time **then** replace it with a new random selected configuration *(scout*)
8: **Until** requirements are met

Fig. 3 ABC/AsBeC pseudocode

Since the original version of the Artificial Bee Colony many researches on the topic were developed. Despite the great amount of available literature on ABC variants and modifications (Bolaji *et al*. 2013), to the best of our knowledge no paper underlines a performance gain even with very few function evaluations. This aspect is of major interest in a CFD-based context like the one herein presented, in which only some hundreds of FEs are available to limit the machine times.

A new algorithm named Artificial super-Bee enhanced Colony (AsBeC) is then proposed to deal with costly optimization. The AsBeC is a modification of the original ABC aimed at improving its speed and solution accuracy for problems where function evaluations have to be limited below 10^3. To accomplish these tasks, enhancements of the basic structure and hybridizations with local interpolation are used. These techniques speed up the convergence of the best solutions in their neighbourhood without clustering the swarm at the same time. As a result, the local search skills of the original ABC (exploitation) is improved without worsening its global attitude (exploration), especially during the first search phases. The improving techniques are classified in the two groups outlined below.

- **Hybridizations: super-bee concept.** These modifications alter the original pseudo-random movement of the bees, accelerating the optimization process and its accuracy through simple local interpolation hybrids.
    a. Whenever one bee does not improve its nectar source with the original pseudo-random mutation, then it tries the opposite movement. This simple linear local estimator is inspired to the concept of opposition based learning (Tizhoosh 2005);
    b. Each bee can estimates the local and directional curvature of the objective function, acting as a second order optimization method with partial Hessian computation. The multi-dimensional parabola passing through three previous positions of the bee (starting food source point, first random movement and opposite position) is calculated and its minimum is evaluated. This local parabolic estimator follows the basic principles of convex optimization (Boyd 2004);
    c. The data knowledge about the history of the best solutions evolving in time is used to make a prediction. A bee is then guided towards the foreseen next best search direction, computed through a weighted average of the last directions of improvement.
- **Enhancements.** These techniques do not alter the architecture of the original ABC, but they make it work differently in order to boost the exploitation from the early optimization phases.
    a. Each group of bees have more time to evolve their nectar sources, having also more chances to use its super-bee skills;
    b. The exploitation of the best food sources is strongly privileged while the worst ones are always penalized. This is particularly beneficial with small colonies;
    c. The scout is relocated in a dynamic range that depends on the current position of the food sources, allowing also exiting the starting boundaries and following the colony movements.

A short pseudo-code of the ABC/AsBeC is outlined in Fig. 3. In the presented test case, the AsBeC algorithm will include the baseline configuration as starting point into the random generated initial food sources, in order to fasten the optimization process.

3.2 Partial pareto

The AsBeC algorithm was originally designed to deal with Single-Objective problems (SO), nevertheless it can be successfully used to extract a partial Pareto Front, which is an approximation of the global one, when solving Multi-Objective (MO) problems. First of all, the weighted sum approach is adopted to transform the MO problem into a SO one (Deb 2001). Then, since the

algorithm stores all the configurations tested and their results, the information about the contributions to the weighted sum is kept and evolved during the entire optimization process. Therefore, an estimated Pareto front is built a posteriori by identifying the non-dominated points from the collective hive memory. This partial Pareto is more reliable and close to convergence in the regions presenting higher weighted sum.

Demonstrative tests on multi-objective analytical functions show that the AsBeC can extract a consistent partial Pareto. In particular, the authors followed the same approach of Deb *et al.* (2002), by using benchmark problems widely adopted in literature to test Multi Objectives Genetic Algorithms (MOGAs). All problems in this testbed have two objective functions to be minimized, they are unconstrained and their dimensionality, *n*, varies from 1 to 30. Six heterogeneous problems over the nine described by Deb at al. (2002), namely SCH, FON, KUR, ZDT3, ZDT4 and ZDT6, have been selected for the purpose of validating the AsBeC. The total number of FEs is chosen to be 25000, as in many other papers on the topic (e.g Toffolo and Benini 2003). The weighted sum approach has been applied to the two functions f_1 and f_2 using unitary weights and the corresponding results are presented in Fig. 4. Each optimization process is repeated 20 times, due to the random nature of the algorithm, then averaged results are shown.

It is evident that the AsBeC partial Pareto is able to approximately describe the Pareto-optimal Front in the zones presenting minimal weighted sum, which are reported in the Table 1 below.

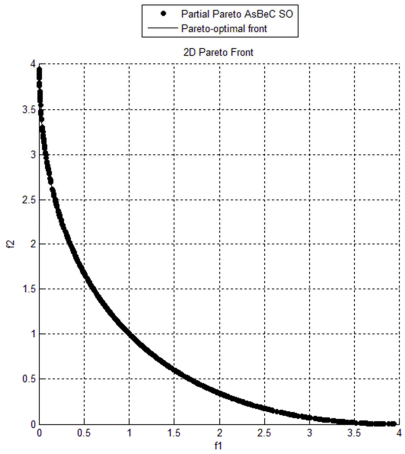
Table 1 Minimal weighted sum regions for the adopted benchmark in the two objectives Pareto space

SCH	FON	KUR	ZDT3	ZDT4	ZDT6
[1.0, 1.0]	[0.002, 0.978] [0.978, 0.002]	[-14.522, -11.583]	[0.850, -0.773]	[0.250, 0.500]	[1.0, 0.0] [0.3884, 0.8492]

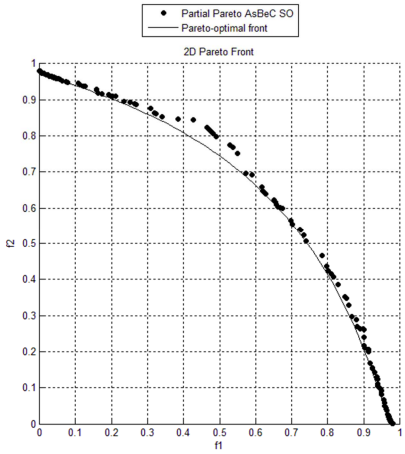
The resulting AsBeC performance seems valuable in comparison with well-known genetic and evolutionary multi objective algorithms, like NSGA, NSGA-II, SPEA, PAES and GeDEA (Deb *et al*. 2002; Toffolo and Benini 2003).

Another important test is the evaluation of the Pareto convergence as function of FEs. This helps to understand how many function evaluations are needed to approach the optimal Pareto, at least with few points. The convergence check for the AsBeC is performed in a 5D search space (same dimensionality of Section 4) by using two of the selected six test functions, FON and ZDT3, which present different types of Pareto Front: one is non-convex and continuous while the other is convex but disconnected. Partial Pareto results are reported in Fig. 5a-c for FON Fig. 5d-f for ZDT3, testing 5000, 600 and 250 FEs. In both cases it is clear that Pareto convergence quickly degrades reducing function evaluations. Nevertheless, even 250 FEs seems to be sufficient to obtain at least some configurations close to the analytical Pareto.

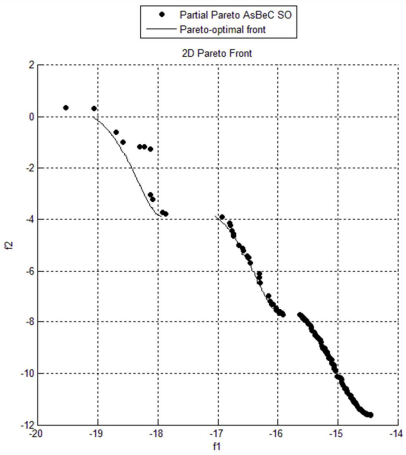
The results of this paragraph support the use of the AsBeC partial Pareto to interpret the optimization results in Section 4.



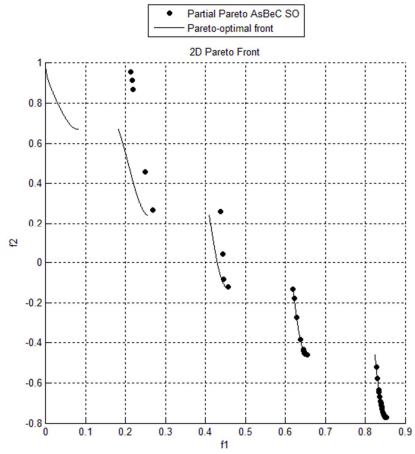
(a) SCH, *n*=1



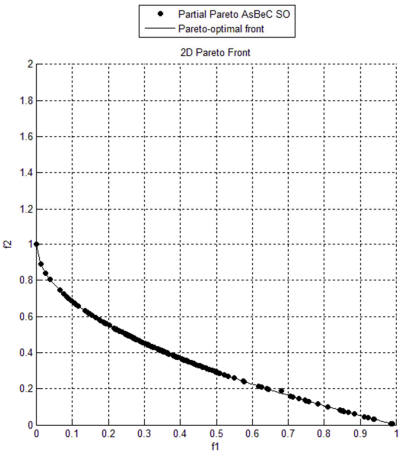
(b) FON, *n*=3



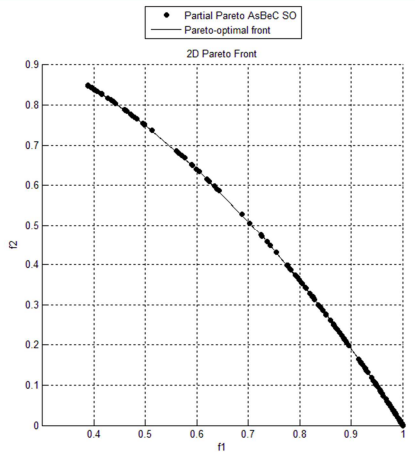
(c) KUR, *n*=3



(d) ZDT3, *n*=30

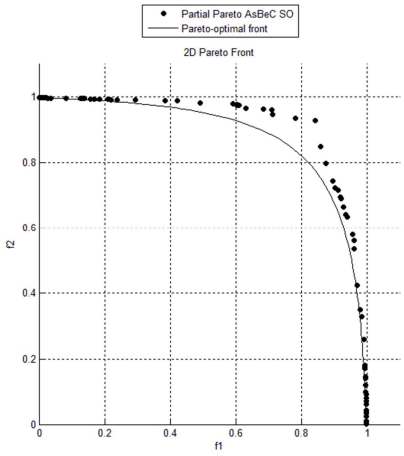


(e) ZDT4, *n*=10

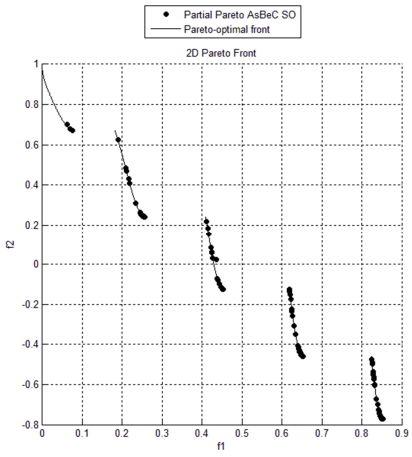


(f) ZDT6, *n*=10

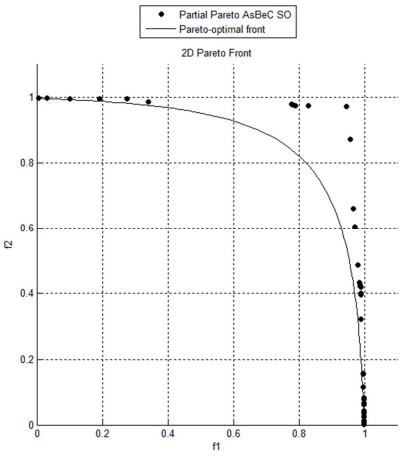
Fig. 4 AsBeC performace on the analytical benchmark, 25000 FEs



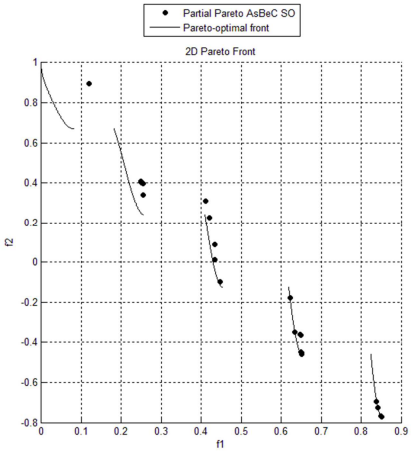
(a) FON, *n*=5, 5000 FEs



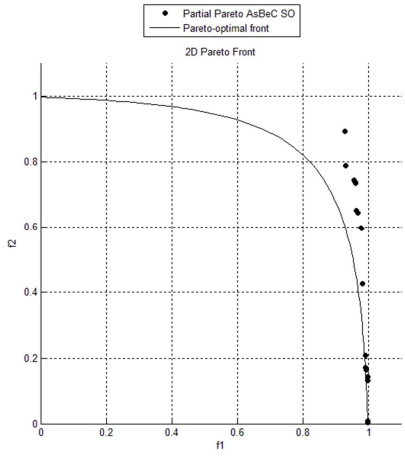
(d) ZDT3, *n*=5, 5000 FEs



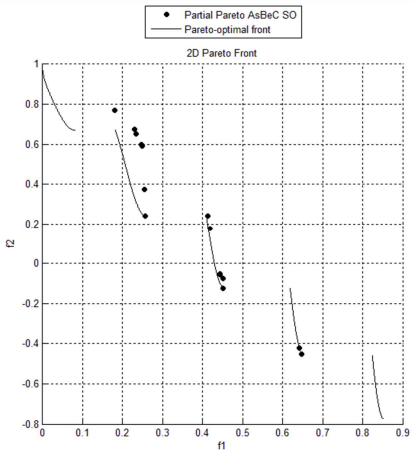
(b) FON, *n*=5, 600 FEs



(e) ZDT3, *n*=5, 600 FEs



(c) FON, *n*=5, 250 FEs



(f) ZDT3, *n*=5, 250 FEs

Fig. 5 AsBeC convergence check as function of FEs by using FON and ZDT3

4. Engineering application

This core section is devoted to the engineering application, from pre-processing details to results presentation.

Airfoil design is a complex topic which still raises a lot of interest in the engineering community. Coupling the airfoil shape optimization with high-fidelity aero analyses such as detailed CFD is an established practice, especially for turbomachinery (e.g. Koiro 1999). In particular, the multi-disciplinary aero-mechanical optimization of turbine blades is standard problem addressed in industrial contexts (Schabowski *et al*. 2010; Bertini *et al*. 2013).

In this work, a renowned turbine test case, i.e. the cascade blade T106c, is chosen as the baseline to exemplify a real-world oriented MDO, based on contemporary numerical tools. For that cascade, experimental data on the 2D blade-to-blade plane for several working conditions are available from the von Karman Institute for Fluid Dynamics (Michalek *et al*. 2010; Hillewaert *et al*. 2013). These data are used in Paragraph 4.3 for validating the simulation environment. In certain conditions this blade is characterized by an evident recirculation bubble and a strong diffusion on the suction side, thus offering large margins of improvement. In particular, the MDO process in this Section considers the inlet swirl angle set to 40 deg while the design value for the T106c is 32.7 deg (see Table 3). The optimization process will then drive the airfoil shape towards new profiles suitable for this severe off-design.

The Paragraph 4.1 briefly describes the parametric system adopted to capture the blade shape; Paragraph 4.2 illustrates the validation of the applied tools, such as the grid convergence and independence, the estimation of the CFD errors and the parameterization efficacy. The optimization platform is described in Paragraph 4.3 while Paragraph 4.4 illustrates the problem definition and settings. Finally, results are reported in Paragraph 4.5.

4.1 Airfoil parametrization

Airfoils have to be finely characterized in order to properly interpret the physical phenomena investigated by computational fluid dynamics. Indeed, the grid generation is one of the most delicate aspects when numerical simulations are involved. The mesh generation requires a large number of points on the profile: they have to be enough to capture all the geometrical peculiarities and closely spaced near to the areas with the larger curvature.

A parameterization system for the airfoil shape is then introduced. This approach enables to draw the profile through its coordinates by means of few curves, whose definition is obtained by defining some fundamental geometrical parameters. The best parameterization is the one with the lower dimensionality among those capable to properly define any geometric detail. In this way the airfoil management would be as simple as possible without losing accuracy. A good parametric system involves only a small number of key parameters, even if a little approximation in the geometry could be present and accepted. Consequently, the adopted parameterization must be a balanced compromise between complexity and quality.

A suitable reference parameterization for turbine airfoils could be the one suggested by Anders and Haarmeyer (2010). It has been adopted In the present work with few simplifications and modifications. The most straightforward idea for a 2D section in Cartesian coordinates would be to implement circumferences for the leading edge and the trailing edge connecting them through polynomial lines, a third degree curve for pressure side and a fourth degree one for suction side. Seven reference points univocally define these four drawing laws: three for the leading edge, three for the trailing edge and one on the throat point. Four reference tangents, which define the wedge

angles for upper and lower sides near leading and trailing edges are then used as conditions to mathematically solve the system. The airfoil curves calculated in this way are continuous and differentiable and they have no discontinuities up to first order, hence they are in $\mathcal{C}^1$ class.

Extra degrees of freedom are added by capturing the same polynomial curves using rational Bezier Splines (BS) of the same order. Then the weights of the BS control points can later introduce further flexibility in defining the shape of the airfoil.

The independent parameters needed to draw a parametric airfoil in the system above described are seventeen (Table 2), assuming that the first and last weights of each BS have to be equal to 1.

Table 2 The seventeen parameters needed to univocally describe a turbine airfoil

Δx, translation of the leading edge along turbine x axis	C_{ax}, axial chord	2nd pressure side BS weight
Δy translation of the leading edge along turbine y axis	C_t, tangential chord	3rd pressure side BS weight
x_{th}, x coordinate of throat point	β_{in}, inlet blade angle	2nd suction side BS weight
y_{th}, y coordinate of throat point	R_{le}, leading edge radius	3rd suction side BS weight
	ω_{in}, inlet wedge angle	4th suction side BS weight
	β_{out}, outlet blade angle	
	R_{te}, trailing edge radius	
	ω_{out}, outlet wedge angle	

This definition is in line with other typical choices for turbomachinery (Wilson 1991; Anders and Haarmeyer 2010).

The drawing laws are used to obtain area and minimum principal inertia moment of the airfoil (Gauss's formulae). Fig. 6 illustrates the T106c baseline airfoil captured in the parametric system (thick curve). BS control points (circular see-through markers), BS polygons (thin lines) and the centroid (black filled marker) are highlighted in the figure.

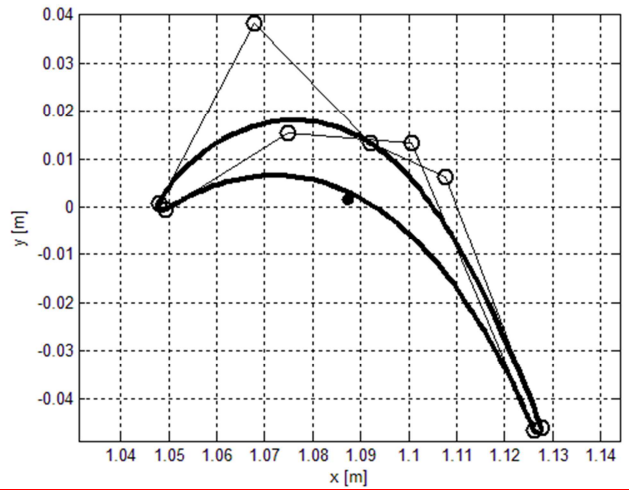


Fig. 6 Baseline airfoil parameterization

4.2 CFD analyses validation

There are several experimental data available in the literature for the T106c blade (Michalek et al. 2010; Hillewaert et al. 2013), however the CFD solver was tested by studying the reference configuration defined in Table 3. The purpose of these preliminary simulations is to estimate the error produced by the CFD solver on the performance evaluation. This is fundamental to identify a tolerance to stop the optimization process and to understand if the improvements showed by the algorithm are meaningful or not.

First, a grid convergence and independence analysis is performed to establish the minimum resolution required to reproduce the performances of the baseline blade with sufficient accuracy. The refinement procedure is stopped by monitoring the convergence of the average exit losses. This study is carry out on the baseline configuration only, then the chosen resolution level is also adopted during the entire optimization process. The final grid contains approximately 7000 quadrilateral elements. If second order (4 degrees of freedom per equations for element) Discontinuous Galerkin elements were used in the entire domain this discretization would be equivalent to a finite volume grid with 28000 cells. Since the p-adaptivity algorithm was employed some elements inside the domain used third order (9 degrees of freedom per equation for each element) and fourth order (16 degrees of freedom per equation for each element) reconstructions, so the total number of degrees of freedom is higher. A typical value obtained in a simulation is approximately equal to 35000 degrees of freedom per equation. Fig. 7 shows an example of the reconstruction order distribution inside the domain.

In Fig. 8, the wall isentropic Mach number distribution calculated by the present discretization is compared with the experimental data. The plot reports the numerical results obtained on both the original geometry and its parametric representation. Notice that the overall behaviour is well described and the main differences introduced by the parametric representation are localized on the leading edge and on the first part of the suction side. This is mostly due to the difficult fitting of the leading edge by a circumference.

In Table 4 the total pressure loss coefficient (Eq. 1) is reported in order to compare experimental results, CFD on the original geometry and CFD on the parametric geometry:

$$\zeta_{P^0} = \frac{P_1^0 - P_2^0}{P_1^0} \tag{1}$$

where P^0 stays for stagnation pressure and subscripts 1 and 2 indicate respectively inlet and outlet stations.

Data in Table 4 refer to a control station positioned 0.465 axial chords downstream the trailing edge, in accord with the experimental results. Loss coefficients are computed through the spatial average of the losses distribution reported by Hillewaert et al. (2013).

This test was used to calibrate the transition model and to estimate the CFD error.

Table 3 Reference condition for CFD validation

Isentropic exit Re number Re_{2s}=1.85e+5	Inlet Turb. Intensity=0.9%
Isentropic exit Mach number M_{2s}=0.65	Inlet angle=32.7 deg
Inlet total temperature T_1^0=298 K	Specific heat ratio γ=1.4
Inlet total pressure P_1^0=7198 Pa	Gas constant R=287.1 J/kgK

Table 4 Comparison on average total pressure loss coefficient

	Loss coefficient ζ_{P^0}
Experiment reported by Hillewart *et al.* (2013)	7.8e-3
CFD original geometry	7.2e-3
CFD parametric geometry	7.0e-3

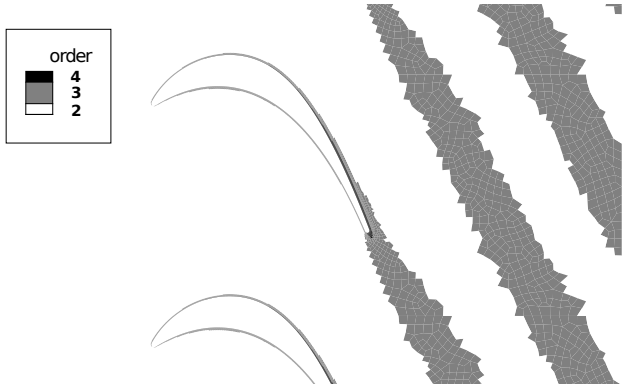


Fig. 7 Reconstruction order distribution

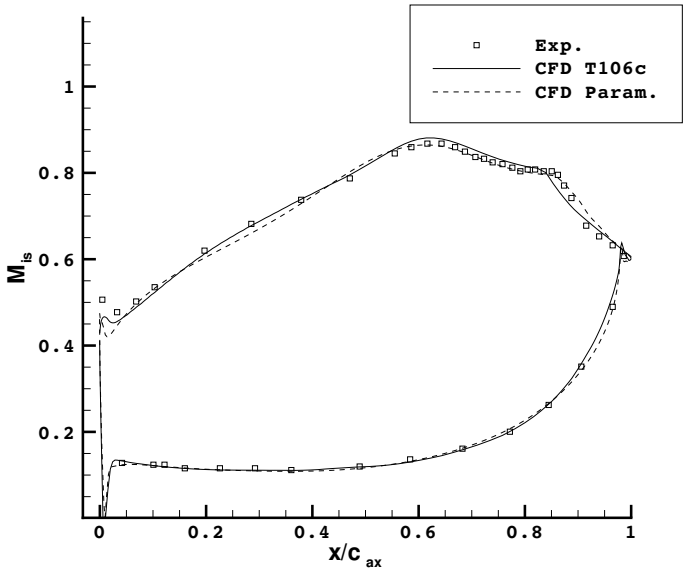


Fig. 8 Comparison on wall isentropic Mach number

4.3 Simulation based optimization platform

An automatic platform is developed for the MDO, in which the optimizer and the CFD solver are integrated. In the specific case, a single airfoil is studied in a 2D fluid environment; however, the same approach can be extended to multi-row 3D simulations or different applications.

Once the original geometry is captured in the parametric system, the optimization can begin. The AsBeC MDO algorithm operates the calls to the airfoil drawing, the grid generation and the CFD solver. These three routines are called through an evaluation function that acts like a "black-box": given a certain set of input variables driven by AsBeC, it returns the output objectives to the main program.

The optimization algorithm settings are: the maximum number of function evaluations, the number of free-to-change variables, the boundaries, the formulation of the targets and the total number of bees. When the process finishes, if the achieved solutions are still not good enough for the designer, the entire procedure can be repeated.

The platform exploits the parallel version of the AsBeC algorithm (see Section 3) hence it is able to evaluate several configuration in parallel. This feature is of great advantage on multi-core machines or clusters, since each CFD simulation can be directly associated to a different core or node. A flow chart of the entire process is reported in Fig. 9.

The AsBeC algorithm, the airfoil drawer/creator, the evaluation function and all the related scripts are written in GNU Octave language. The GNU GPL software Gmsh is used for the grid generation and the CFD code is written in Fortran. The optimization is performed on a 6-cores Intel® i7-3930k machine with GNU/Linux operating system. The authors emphasize that such a platform can work with a personal computer and free software.

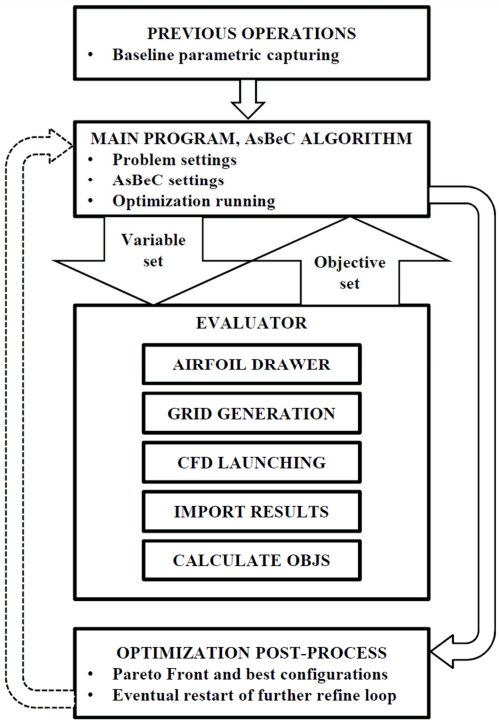


Fig. 9 Flow chart of the parallelized simulation-based optimization platform

4.4 MDO set up

Assuming that the optimization algorithm and the physical simulator have been properly tuned, design parameters and objectives choices are the other key points for the MDO success. The target functions must combine all the multidisciplinary traits needed to be investigated. The authors chose two representative objectives to minimize, associated to the real-world design. They are formulated considering the section under study as a part of the multi-row turbine frame.

Losses

Kinetic energy losses between upstream and downstream averaged quantities are defined as:

$$LOSS = c_1 \cdot \zeta_E[-]; \quad \zeta_E = 1 - \frac{1 - \left(\frac{p_2}{P_2^0}\right)^{\frac{\gamma-1}{\gamma}}}{1 - \left(\frac{p_2}{P_1^0}\right)^{\frac{\gamma-1}{\gamma}}} \tag{2}$$

Where p stays for static pressure, P^0 stays for total pressure, γ is the air specific heat ratio and ζ_E is the kinetic energy loss coefficient calculated at the exit station. The scaling constant c_1 is described later. The exit total pressure value used in Eq (2) is obtained by means of mass flow average.

These losses are directly related to aerodynamic efficiency and then they represent a good performance index.

Dynamic constraints deviations

The 3D shaping of a turbine blade has to be mechanically compliant, so that in each section the area (A) and minimum inertia moment (I_{min}) must satisfy the requirements for blade integrity and safe operability. These two values are set by the structural analyses and are continuously calibrated while aero-mechanical cycles are performed. For example, increasing area and inertia moment improve the mechanical properties of the airfoil, but it could lead to weight or resonance issues. These are related to the 3D blade and are appreciable only by dedicated and comprehensive analyses. The 2D approach if a fast environment for airfoils optimization, but it cannot involve directly such verifications. However, you can assume they have been performed by the 3D blade designer, thus area and moments are already satisfying the margins for structural integrity. Consequently, the optimization should try not to vary A and I_{min} from the values obtained by capturing the baseline airfoil in the parametric system. Then, the second objective is formulated to penalize their deviations.

Another important constraint that should be kept in mind is the mass flow rate elaborated by the blade channel ($\dot{m}$). In order not to alter the working point of the turbine stage to which the blade section belongs, $\dot{m}$ must be kept constant. In fact, the Overall Engine Manufacturer decides the mass flow rate throughout the turbine once the thermodynamic cycle is frozen. The mass flow rate ($\dot{m}_B$) related to the baseline airfoil is then chosen as reference value.

In conclusion, the two previous aspects are combined in the second dimensionless Constrained Objective (CO) that appears as a symmetrical penalization function, including three scaling constants, c_2, c_3 and c_4:

$$CO = c_2 \cdot \left|\frac{A - A_B}{A_B}\right| + c_3 \cdot \left|\frac{I_{min} - I_{min,B}}{I_{min,B}}\right| + c_4 \cdot \left|\frac{\dot{m} - \dot{m}_B}{\dot{m}_B}\right| [-] \tag{3}$$

where $|x|$ operator is the absolute value of x and A_B, $I_{min,B}$ and $\dot{m}_B$ are baseline quantities. Baseline mechanical properties are herein intended as design targets, but the objective in (3) can

be easily reformulated taking care of any specific value for A and I_{min}, for example returned by detailed structural analyses.

The coefficients from c_1 to c_4 are needed in the context of the weighted sum approach. In fact, objectives should be written in a form such that they contribute in equal measure to the global weighted sum, $W_1 \cdot LOSS + W_2 \cdot CO$, when the weights W associated to it are unitary. For this reason, the coefficients c should be tuned in order to adjust LOSS and CO values in such a way that they assume the same order of magnitude during the entire optimization process. It is evident that a sensitivity on each contribute to the two target functions is needed. This sensitivity is guided by the particular optimization problem and by its boundaries, and in the specific case leads to:

$$c_1 = 100;\ c_2 = 10; c_3 = 10; c_4 = 10$$

which makes each target function varies around 1e+0. In this way, the weights W take the role of importance indexes associated to each objective. The weighted sum can put in evidence one or another aspect simply by adjusting these importance indexes.

The variables selection and their boundaries are the last key point to be addressed. The decision to investigate some variables in place of others derives from the common turbine design strategy. Only tools that solve row interactions, like 1D meanline, 2D crosswise or multi-row CFD tools, can change some parameters. These are the absolute positioning of the airfoil, the chord, the throat and the exit blade angle. In addition, the throat regulates the mass flow rate, even if there are blockage effects due to boundary layer thickness to be also considered. The throat point (x_{th},y_{th}) is then maintained constant also to strengthen CO objective. Manufacturing and material limits impose some restrictions, especially for what regards minimum thickness and bending properties near the trailing edge. For this reasons, trailing edge radius and exit wedge angle should be changed cautiously.

As a result, only eight out of seventeen variables are considered for optimization. However, since the significant problems with this airfoil are localized on the suction side, modifying the pressure side can hardly have a positive influence. In order to reduce as much as possible the dimensionality of the problem, the authors have finally selected the following five variables:

- β_{in}, inlet blade angle;
- ω_{in}, inlet wedge angle;
- 2nd suction side BS weight;
- 3rd suction side BS weight;
- 4th suction side BS weight.

During the optimization process these variables are box bounded into an hypercube centred on the baseline values. The upper and lower boundaries of the interval are chosen according to the designer experience and specific problem sensitivity. They must be large enough to accept the changes necessary to achieve the goals, but sufficiently narrow to avoid geometric degeneration. Moreover, it is important to keep in mind that row interactions and 3D phenomena are not resolved at this level. As a consequence, the 2D optimization must fall under the assumption of small perturbations to assure applicability.

For the specific problem boundaries assume the following values:

- $\text{Upper\Lower Bound}_1 = \pm 10$ [deg]
- $\text{Upper\Lower Bound}_2 = \pm 5$ [deg]
- $\text{Upper\Lower Bound}_3 = \pm 0.3$ [-]
- $\text{Upper\Lower Bound}_4 = \pm 1$ [-]
- $\text{Upper\Lower Bound}_5 = \pm 1$ [-]

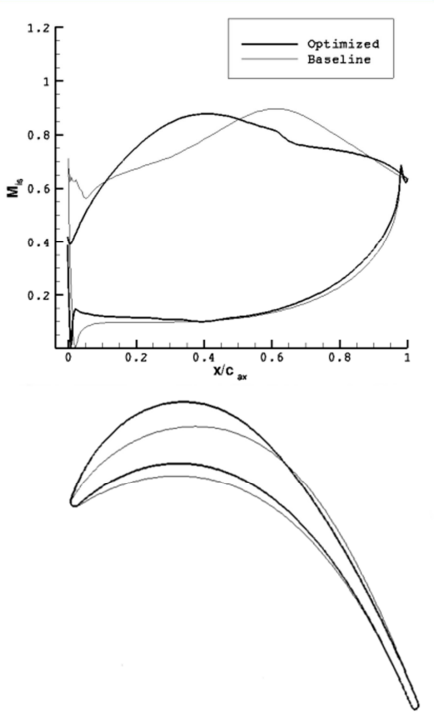
Considering that the CFD simulations are heavily time consuming, the optimization process has to limit the total number of FEs. Hundreds of function evaluations with the AsBeC algorithm are enough to approach the global solution when dealing with problems like the one under exam (see Paragraph 3.2 and Bertini et al. 2013). The optimization is set up by using 12 bees overall, equally divided in 6 employees and 6 onlookers, and 21 cycles for 264 CFD analyses in total. The choice of this number of FEs is supported by the tests described in Section 3.

The weights are biased with the aim to drive the optimum search towards the zones with lower losses, assuming:

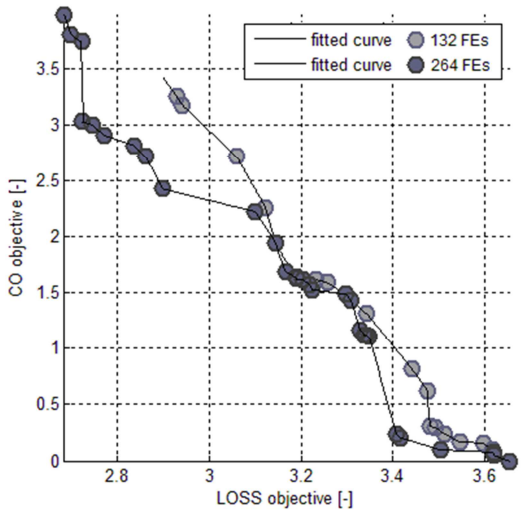
$$W_1 = 5; \quad W_2 = 1$$

4.5 Results

The final optimal solution in terms of weighted sum is reported in Table 5; it is characterized by a relative losses reduction of about 25.5% with respect to the baseline configuration. On the other hand, its area, minimum inertia moment and mass flow differ by 1.56%, 28% and 0.83% respectively from the reference values. Area and mass flow are close to the baseline ones, while I_{min} is much higher. Increasing the minimum inertia moment could improve the mechanical qualities of the airfoil, as it reduces local stresses and enhances blade stiffness. However, only specific analyses as dynamic response, herein not comprised, can point out if the stiffness increase produces side effects.



(a) Isentropic Mach number and geometry for baseline and optimized (best weighted sum) airfoil



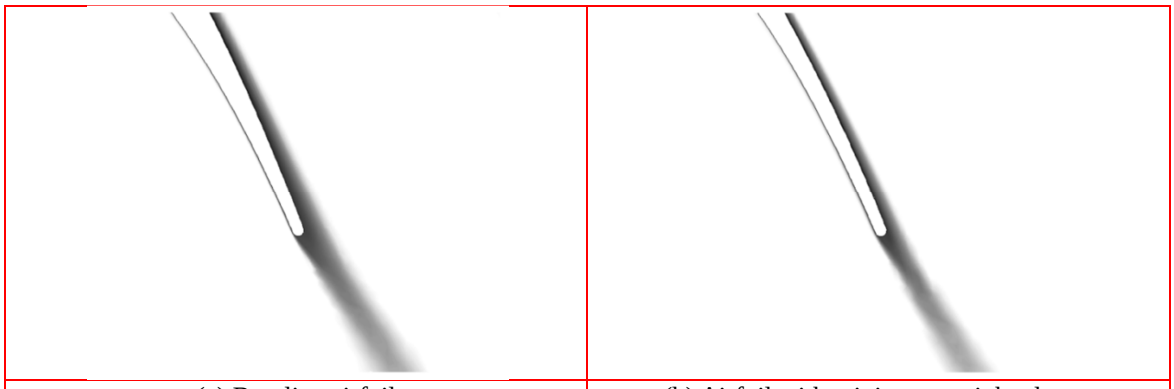
(b) Partial Pareto front

Fig. 10 Results

Fig. 10a shows the comparison in terms of geometries and wall isentropic Mach numbers between the baseline airfoil and the optimized one. Fig. 11a and 11b illustrate the comparison between the entropy fields around the two airfoils: it is clear that the entropy generation is reduced in the second configuration. With the considered augmented incidence, the baseline airfoil is characterized by a large recirculation on the suction side near the leading edge. In contrast, the best weighted sum configuration avoids this separation and regulates the diffusion after the throat station. Fig. 12a and 12b show the streamlines in the leading edge region. As a negative aspect, the best weighted sum airfoil is excessively front loaded with respect to the baseline. To move the configuration towards more after-loaded profiles, interaction phenomena among adjacent turbine rows should be included in the simulation.

Besides the best weighted sum airfoil, it is possible to identify other two solutions which correspond to the best individual goals (Table 5). The one with the best aero performance obtains a 26.5% reduction on losses but that configuration is among the worst ones for CO, while the best for constraints is obviously the baseline.

The obtained partial Pareto Front in Fig. 10b allows identifying all the best compromise configurations. The Pareto points are concentrated near the best weighted sum regions found by the algorithm during the optimization process. The figure also shows the Pareto convergence, extension and population from 132 FEs (light markers) to the final condition of 264 FEs (dark markers). Best weighted sum regions are near (LOSS=2.7, CO=3.0) and (LOSS=3.4, CO=0.2) in the final Pareto, while there was only a best area near (LOSS=3.2, CO=1.6) for the 132 FEs Pareto. This last area and the one around the baseline could be close to convergence, since they are densely populated and do not move forward from 132 to 264 Fes.

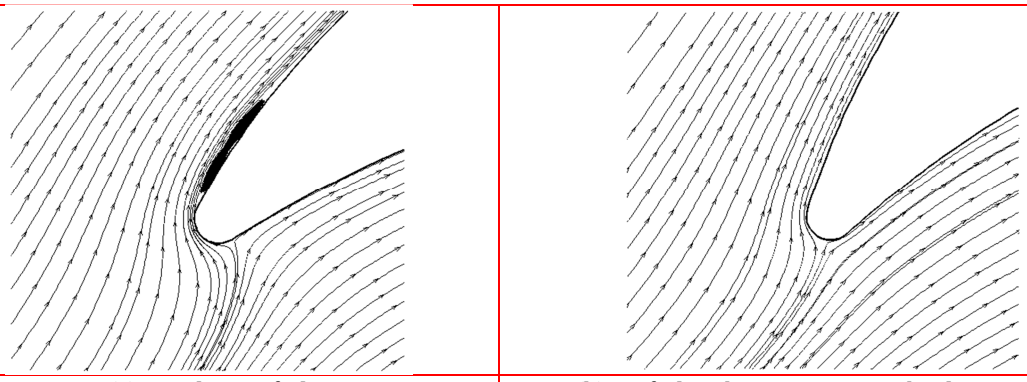


(a) Baseline airfoil | (b) Airfoil with minimum weighted sum

Fig. 11 Entropy field close to the trailing edge

The post processing of the optimization results makes it possible to recognize which are the variables that strongly influence the losses. In particular, the main benefits are related to the increase of the inlet blade angle. In fact, the optimization process changed the inlet blade angle in order to match the inlet flow angle for the chosen working condition. On the other hand, the reduction of the losses leads to an increment of the minimum inertia moment due to the higher flow turning and blade cambering. The only way to contain CO objective divergence is improving efficiency without acting too much on β_{in}, thus regulating BS weights and ω_{in}. However, this leads to minor loss reduction, since β_{in} has a significant impact on aero performance.

Particular care has to be taken in the interpretation of the obtained results. Indeed, both CFD simulations and experimental measurements are affected by uncertainties (see Section 4). For this reason, the main advantage of the optimization is the understanding of the effects and the sensitivity of the different variables on performance. This knowledge makes it possible to establish design rules based on a solid physical background.



(a) Baseline airfoil | (b) Airfoil with minimum weighted sum

Fig. 12 Streamlines around the leading edge

Table 5 Best configurations for individual goals and weighted sum

	LOSS[-]	CO[-]	Weighted Sum[-]
$LOSS_{BEST}$	2.68	3.97	2.90
CO_{BEST}	3.65	0.00	3.04
Weighted Sum_{BEST}	2.72	3.04	2.77

5. Concluding remarks

This paper describes a multidisciplinary approach for turbomachinery design problems. A CFD solver and an optimization platform have been specifically developed for addressing the aeromechanical design of a turbine cascade. In order to assess the reliability of the adopted tools, the meta-heuristic optimization algorithm was tested on analytical functions and the CFD solver was validated on experimental data.

The results obtained for the chosen test case suggest that this approach is able to efficiently adapt the airfoil shape according to the working conditions. Above all, some guidelines for multidisciplinary improvements are derived. Indeed, a Pareto Front approximation is provided and analysed, obtaining the population of best candidate solutions.

The achievements presented in this paper pave the way for an effective application of such kind of design platform to other engineering problems, even different from airfoil design. Indeed, the key features constituting the backbone of the described approach might have a good all-purpose applicability. Future works on the topic will cover different kinds of engineering applications in order to assess what are the advantages and limitations when generalizing this methodology.

References

1. Anders, J.M., Haarmeyer, J. (2010). *A parametric blade design system*. VKI lecture series.
2. Bassi, F., Botti, L., Colombo, A., Di Pietro, D.A., Tesini, P. (2012). On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations. *Journal of Computational Physics,* **231**(1), 45-65.
3. Bassi, F., Crivellini, A., Rebay, S., Savini, M. (2005). Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and k-omega turbulence model equations. *Computers and Fluids*, **34**, 507-540.
4. Bertini, F., Dal Mas, L., Vassio, L., Ampellio, E. (2013). *Multidisciplinary optimization for gas turbines design*. AIDAA XXII conference 2013, 9-12 September, Naples.
5. Bonabeau, E., Dorigo, M., Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press US.
6. Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.
7. Binitha, S., Siva Sathya, S. (2012). A Survey of Bio inspired Optimization Algorithms. *International Journal of Soft Computing and Engineering*, **2**(2), 137-151.
8. Bolaji, A., Khader, A., Al-Betar, M., Awadallah, M. (2013). Artificial Bee Colony Algorithm, its variants and applications: a survey. *Journal of Theoretical and Applied Information Technologhilley*, **47**(2), 434-459.
9. Colombo, A. (2011). An agglomeration-based Discontinuous Galerkin method for compressible flows. *PhD Thesis, University of Bergamo, Italy.*
10. Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley, ISBN: 978-0-471-87339-6
11. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on,* **6**(2),182-197
12. Ferrero, A., Larocca, F. (2013). *Test cases C1.1, C1.2 and C1.6*, Second International Workshop on High-Order CFD Methods, Cologne, Germany, May 2013, http://www.dlr.de/as/hiocfd (Accessed June 2015).
13. Ferrero. A., Larocca, F. (2015). *Test cases C1.2 and C1.3*, Third International Workshop on High-Order CFD Methods, Orlando, Florida, USA, January 2015, https://www.grc.nasa.gov/hiocfd/ (Accessed June 2015).
14. Ferrero, A., Larocca, F., Puppo, G. (2015) A robust and adaptive recovery-based discontinuous Galerkin method for the numerical solution of convection-diffusion equations. *Journal for Numerical Methods in Fluids*, **77**(2), 63-91.
15. Forrester, A., Sobester, A., Keane, A. (2008). *Engineering Design via Surrogate Modelling: A Practical Guide*. Springer, ISBN: 978-0-470-06068-1.
16. Geuzaine, C., Remacle, J.F. (2009). Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, **79**(11), 1309-1331.
17. Glover, F., Kochenberger, G.A. (2003). *Handbook of Metaheuristics*. Springer, International Series in Operations Research & Management Science. ISBN 978-1-4020-7263-5.
18. Hillewaert, K., Carton de Wiart, C., Arts, T. (2013). *Test cases C3.7*, Second International Workshop on High-Order CFD Methods, Cologne, May 2013, http://www.dlr.de/as/hiocfd (Accessed June 2015).
19. Iollo, A., Ferlauto, M., Zannetti, L. (2001). An Aerodynamic Optimization Method based on the Inverse Problem Adjoint Equations, *Journal of Computational Physics*, **173**(1), 87-115.
20. Jones, D.R. (2001), A taxonomy of global optimization methods based on response surfaces, *Journal of Global Optimization*, **21**(4)**,**345-383
21. Karaboga, D. (2005). *An Idea Based on Honey Bee Swarm for Numerical Optimization*. Technical Report TR06, Erciyes University, Turkey.
22. Karaboga, D. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, **39**(3), 459-471.

23. Karaboga, D., Akay, B. (2009). A comparative study of Artificial Bee Colony Algorithm. *Applied Mathematics and Computations*, **214**(1),108-132.
24. Karaboga D., Gorkemli, B., Ozturk, C., Karaboga, N. (2014). A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, **42**(1), 21-57.
25. Kennedy, J., Eberhart, R. (1995). Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks IV, 1942-1948.*
26. Kennedy, J. (2010). Particle Swarm Optimization. *Springer, Encyclopedia of Machine Learning*, pp *760-766*
27. Koiro, M.J., Myers, R.A., Delaney, R.A. (1999). *TADS-A CFD-Based Turbomachinery Analysis and Design System With GUI.* NASA technical report.
28. Koziel, S., Yang, X.S. (2011). Computational Optimization, Methods and Algorithms. *Springer.*
29. Koziel, S. and Leifsson, L. (2013). Surrogate-Based Modeling and Optimization, Applications in Engineering. *Springer, New York.*
30. Larocca, F. (2008). Multiple objective optimization and inverse design of axial turbomachinery blades. *Journal of Propulsion and Power*, **24**(5),1093-1099.
31. McCullagh, P., Nelder, J. (1989). Generalized Linear Models, Second Edition. *Chapman and Hall/CRC. ISBN0-412-31760-5*.
32. Martins, J.R.R.A., Alonso, J.J., Reuthes, J.J. (2005*).* A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design. *Optimization and Engineering*, ***6****(1), 33-6.*
33. Martins, J.R.R.A., Lambe, A.B. (2013) Multidisciplinary design optimization: A Survey of architectures. *AIAA Journal*, **51**(9), 2049-2075.
34. Michalek, J., Monaldi, M., Arts, T. (2010). Aerodynamic performance of a very high lift low pressure turbine airfoil (T106C) at low Reynolds and high Mach number with effect of free stream turbulence intensity. *ASME paper GT2010-22884*, Glasgow, UK, 14–18 June 2010.
35. Onate, E. (2009). *Structural Analysis with the Finite Element Method: Linear Statics, Volume 1: Basis and Solids.* Springer, Berlin.
36. Pacciani, R., Marconcini, M., Arnone, A., Bertini, F. (2011). An assessment of the laminar kinetic energy concept for the prediction of high-lift, low-Reynolds number cascade flows, *Proceedings of the Institution of Mechanical Engineers Part A Journal of Power and Energy*, **225**, 995-1003.
37. Pandolfi, M. (1984). A contribution to the numerical prediction of unsteady flows. *AIAA Journal*, **22**(5), 602-610.
38. Panigrahi, B.K., Shi, Y., Lim, M.H. (2011). Handbook of Swarm Intelligence. *Springer*.
39. Price, K., Storn, R., Lampinen, J. (2005). *Differential Evolution - A Practical Approach to Global Optimization*. Springer, Berlin.
40. Rao, S.S. (2009). *Engineering Optimization: Theory and Practice*, John Wiley & Sons.
41. Schabowski, Z., Hodson, H., Giacche, D., Power, B., Stokes, M.R. (2010). Aeromechanical Optimisation of a Winglet-Squealer Tip for an Axial Turbine. ASME Turbo Expo 2010, **7**, 1593-1607
42. Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*, 2nd Edition, SIAM.
43. Talbi, E.G. (2009). *Metaheuristics: from design to implementation*. Wiley. ISBN 0-470-27858-7.
44. Toffolo, A., Benini, E. (2003). Genetic Diversity as an Objective in Multi-Objective Evolutionary Algorithms. *Evolutionary Computation* **11**(2), 151-167.
45. Tizhoosh, H. (2005). Opposition-based learning: A new scheme for machine intelligence. *Proceedings of international Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA.*
46. Vanderplaats, G.N. (2007). Multidiscipline Design Optimization. Vanderplaatz R&D, Inc.
47. Vestraete, T., Periaux, J. (2012). Introduction to optimization and multidisciplinary design in Aeronautics and Turbomachinery. *VKI Lecture series*.
48. Wilcox, D.C. (2006). *Turbulence Modeling for CFD.* 3rd edition, DCW Industries.
49. Wilson, D.G. (1991). *The design of high-efficiency turbomachinery and gas turbines.* MIT press, Cambridge, Massachusetts, 5th printing.
50. Yang, X.S. (2010). *Nature-Inspired Metaheuristic Algorithms – Second Edition.* Luniver press.