Gradient Span Analysis Method: Application to the Multipoint Aerodynamic Shape Optimization of a Turbine Cascade

Hadrien Montanelli, Marc Montagnac, François Gallard

Gradient Span Analysis Method: Application to the Multipoint Aerodynamic Shape Optimization of a Turbine Cascade

Hadrien Montanelli
NA Group,
Mathematical Institute,
University of Oxford,
Oxford OX26HD, UK
e-mail: montanelli@maths.ox.ac.uk

Marc Montagnac[1]
Research Engineer
Computational Fluid Dynamics Group,
Cerfacs,
42, Avenue G. Coriolis,
Toulouse 31057, CEDEX 1, France
e-mail: marc.montagnac@cerfacs.fr

François Gallard
Research Engineer
IRT Saint Exupéry,
Toulouse 31432, France
e-mail: francois.gallard@irt-saintexupery.com

This paper presents the application of the gradient span analysis (GSA) method to the multipoint optimization of the two-dimensional LS89 turbine distributor. The cost function (total pressure loss) and the constraint (mass flow rate) are computed from the resolution of the Reynolds-averaged Navier–Stokes equations. The penalty method is used to replace the constrained optimization problem with an unconstrained problem. The optimization process is steered by a gradient-based quasi-Newton algorithm. The gradient of the cost function with respect to design variables is obtained with the discrete adjoint method, which ensures an efficient computation time independent of the number of design variables. The GSA method gives a minimal set of operating conditions to insert into the weighted sum model to solve the multipoint optimization problem. The weights associated to these conditions are computed with the utopia point method. The single-point optimization at the nominal condition and the multipoint optimization over a wide range of conditions of the LS89 blade are compared. The comparison shows the strong advantages of the multipoint optimization with the GSA method and utopia-point weighting over the traditional single-point optimization. [DOI: 10.1115/1.4030016]

1 Introduction

Numerical shape optimization in aerodynamics is a design process of geometric shapes following a given criteria of aerodynamic performance, with both geometric and aerodynamic constraints. This technique was intensively studied for a long time in the scientific community. Among many other optimization methods, gradient-based local algorithms are of particular interest if associated with a method that cheaply computes the gradient of the cost function with respect to design variables. The adjoint method [1,2] is one of them, the time cost to compute the gradient being independent of the number of design variables. A gradient-based local algorithm together with the adjoint method can therefore handle a large number of design variables, needed to generate sophisticated industrial configurations. This global framework of shape optimization with adjoint method is widely used in turbomachinery [3–8], ranging from two-dimensional single-point cascade optimization to three-dimensional multipoint multistage optimization, and from steady to unsteady flows.

A turbomachine operates over a continuous range of aerodynamic conditions. Its performance may seriously deteriorate at off-design conditions, if it was designed at a specific operating condition, or even accounting for many operating conditions [9]. The weighted sum model is used to obtain a design optimized not only for one but also for several conditions. Hence, the key point is the selection of the appropriate conditions, and the choice of the associated weights [10].

This paper investigates the shape optimization of the well-known two-dimensional LS89 distributor turbine blade [11], with a local gradient-based algorithm, a cost function given by the weighted sum model, and gradients computed with the discrete adjoint method. The newly introduced GSA method [12], already validated on external aerodynamic optimization cases [13,14], is used to select both the values and the number of operating conditions to use for the scalar cost function. The weights are given by the utopia point method [15].

The paper is organized as follows. The test case is presented in Sec. 2, while Sec. 3 details the design objectives. Numerical methods, and in particular the GSA method, are introduced in Sec. 4. Section 5 is dedicated to numerical results.

2 LS89 Test Case

The test case is the LS89 blade of a highly loaded transonic turbine distributor. This blade was designed by the Von Karman Institute in the early 1990s, and a lot of experimental studies were carried out for very different physical conditions [11]. Figure 1 shows the multiblock structured grid of 140,330 nodes used to perform the numerical experiments. The mesh is split into 35 blocks for parallel computing on 16 cores. The geometrical parameters of the LS89 blade are given by the chord $c = 67.647\,\text{mm}$, the pitch $g/c = 0.850$, the leading edge radius $r_{\text{LE}}/c = 0.061$, and the trailing edge radius $r_{\text{TE}}/c = 0.0105$. The leading edge is at $(x, z) = (0, 0)$. The inlet plane $\partial\Omega_1$ is located at $x = -46.0\,\text{mm}$, and the outlet plane $\partial\Omega_2$ at $x = 100.0\,\text{mm}$.

A turbine nozzle guide vane converts pressure energy into kinetic energy, and guides the flow to the turbine rotor. The flow acceleration is generated by the static pressure ratio $\Pi = P_{s_2}/P_{s_1} < 1, P_{s_1}$ and P_{s_2} being the static pressures on the inlet and outlet boundaries respectively. The static pressure ratio Π characterizes the air flow. The condition MUR235 [11] was chosen as the nominal condition, which corresponds to $\Pi_{\text{nom}} = 0.583$. On the inlet boundary, the total temperature is set to 413.3 K, the total pressure to 182,704.1 Pa, and the axial flow angle to 0.0 deg. The blade surface is supposed to be an adiabatic

[1]Corresponding author.

Contributed by the Heat Transfer Division of ASME for publication in the Journal of Turbomachinery. Manuscript received October 1, 2014; final manuscript received March 3, 2015; published online March 24, 2015. Assoc. Editor: Graham Pullan.

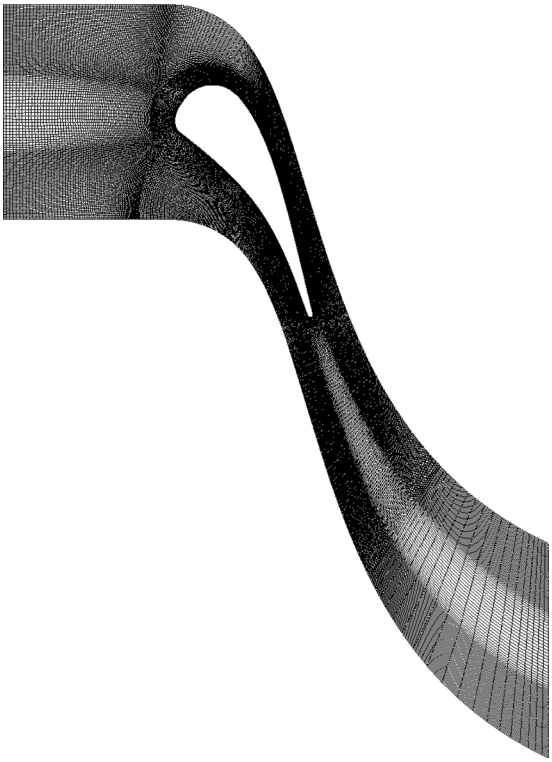

Fig. 1 Multiblock structured grid of the LS89 test case

smooth wall. The lower and upper sides of the mesh are periodic boundary conditions. The Reynolds number Re_1 is 264,170, and the Mach number M_1 is 0.15. This nominal condition corresponds to an isentropic Mach number $M_{2,\mathrm{is}} = 0.927$, and generates a transonic flow. A shock wave is located on the upper surface of the blade, while a large turbulent wake escapes from the trailing edge.

3 Design Objectives

This section describes the design objectives. First, the single-point optimization problem is introduced, and different quantities are defined for a specific operating condition. Then, follows the multipoint optimization problem.

3.1 Design Objective of the Single-Point Optimization Problem. The goal is to minimize the total pressure loss $\mathrm{tpl}(\alpha)$ defined in the below equation at the nominal condition Π_{nom} by modifying the design variables α

$$\mathrm{tpl}(\alpha) = 1 - \frac{P_{t_2}(\alpha)}{P_{t_1}} \tag{1}$$

where $P_{t_2}(\alpha)$ is the averaged total pressure given by

$$P_{t_2}(\alpha) = \frac{1}{|\partial\Omega_2|} \iint_{X\in\partial\Omega_2} P_t(X,\alpha)dS \tag{2}$$

A constraint on the outlet mass flow rate $Q(\alpha)$ defined in Eq. (3) is added to avoid a potential entropy decrease due to a mass flow rate reduction

$$Q(\alpha) = \iint_{X\in\partial\Omega_2} \rho(X,\alpha)U(X,\alpha)\cdot n_2(X)dS \tag{3}$$

where ρ is the density, U is the velocity vector, and n_2 is the normal vector to the surface $\partial\Omega_2$.

The constrained optimization problem is handled with the penalty method. Then, the weighted cost function $J_s(\alpha)$ of the single-point optimization problem is given by

$$J_s(\alpha) = \frac{\mathrm{tpl}(\alpha)}{\mathrm{tpl}(\alpha_0)} + \sigma\left(\frac{Q(\alpha)}{Q(\alpha_0)} - 1\right)^2 \tag{4}$$

in which $\mathrm{tpl}(\alpha_0)$ and $Q(\alpha_0)$ represent the total pressure loss and the outlet mass flow rate of the initial blade, respectively, and σ is the penalty coefficient. The goal of the optimization is to reduce the shock intensity and the wake width in order to minimize the total pressure loss.

3.2 Design Objective of the Multipoint Optimization Problem. As explained in Sec. 1, the aerodynamic design of a turbine cascade is intrinsically multipoint. The goal is to minimize the total pressure loss not only at the nominal condition Π_{nom} but also on a continuous range of operating conditions around Π_{nom}. In practice, this range is defined by m conditions $\{\Pi_k\}_{k=1}^m$. This kind of optimization is classically tackled with the weighted sum model in the literature [6,16–19]. The overall function to minimize is given by

$$J_m(\alpha) = \sum_{k=1}^{m} \omega_k J_s(\alpha,\Pi_k) \tag{5}$$

that is

$$J_m(\alpha) = \sum_{k=1}^{m} \omega_k \left[\frac{\mathrm{tpl}(\alpha,\Pi_k)}{\mathrm{tpl}(\alpha_0,\Pi_k)} + \sigma\left(\frac{Q(\alpha,\Pi_k)}{Q(\alpha_0,\Pi_k)} - 1\right)^2\right] \tag{6}$$

where the ω_k are some weighting coefficients. The two key points of this multipoint approach are the sampling of appropriate operating conditions (number and location) and the choice of the associated weights [10]. We shall say a few words about the way we choose the weights in the next paragraph. The operating conditions are selected with the GSA method, see Sec. 4.4.

The method chosen in this paper to compute the weights is based on the utopia point method. The aim is to force the algorithm to target the Pareto front. If one of the functions, say $J_s(\alpha,\Pi_1)$, could be reduced significantly, then the algorithm would decrease this function, and eventually increase the others, because the descent direction taken to reduce $J_s(\alpha,\Pi_1)$ would be the steepest one. Hence, the idea is to perform a single-point optimization at each of the m conditions, and then choose weights $\omega_k = |1/\Delta\mathrm{tpl}_k|$, $\Delta\mathrm{tpl}_k$ being the variation of total pressure loss at each condition Π_k. In other words, $\Delta\mathrm{tpl}_k$ is considered as the maximum gain that can be obtained at condition Π_k. This method to compute the weights is expensive, and ideally we would like to be able to estimate the potential gains $\Delta\mathrm{tpl}_k$ a priori, as done in wing optimization [13] for drag gains.

4 Numerical Methods

This section presents the numerical methods. The optimizations rely on a full adjoint process.

4.1 Parametrization. A parametric and differentiated computer-aided design (CAD) engine is used to build the parametric model of the surface $S(\alpha)$ of the blade. This in-house code is based on the nonuniform rational basis splines, and the parameters are meaningful to designers (e.g., thickness, curvature, leading edge radius). We shall give more details in Sec. 5.1. Besides, it provides smooth surfaces, and enables to create localized deformations. An analytical reverse mode computes functional gradient $dS/d\alpha$.

4.2 Grid Update. A grid deformation approach is used to update the grid when the CAD model is updated by new design parameters. Subtraction of the initial surface to the updated surface provides a surface deformation field that has to be propagated into the volume grid. The method used relies on an algebraic integral formulation [20]. It is analytically differentiated to provide a reverse mode for the computation of the surface grid sensitivity, from the adjoint volume mesh sensitivity.

4.3 Flow and Adjoint Solvers. Numerical computations were performed with the ELSA ONERA software [21,22]. This code manages both the flow analysis and the flow sensitivity aspects. It solves the 3D compressible Reynolds-averaged Navier–Stokes equations using a cell-centered finite-volume method on structured grids, and the associated discrete adjoint equations. The turbulence model chosen is the one-equation Spalart–Allmaras [23] model. The spatial convective fluxes of the mean flow are discretized with the upwind Roe scheme [24] and the Harten's entropic correction. A monotone upstream-centered schemes for conservation laws scheme [25] associated with a Van Albada limiter [26] provides second-order accuracy. The spatial convective fluxes of the turbulent flow are discretized with the first-order upwind Roe scheme. Spatial diffusive fluxes are approximated with a second-order central scheme. The turbulent equations are solved separately from the mean flow equations at each time step with the same time-marching method. The backward Euler implicit scheme drives the time integration. The resulting linear systems are solved with the scalar lower-upper symmetric successive over-relaxation (LU-SSOR) method [27]. A standard nonlinear multigrid algorithm [28] combined with local time stepping accelerates the convergence to steady-state solutions.

All of the methods previously described in the flow solver are differentiated by hand. During the differentiation, turbulent eddy viscosity and thermal conductivity are assumed constant [29,30], and a thin shear-layer assumption is made for viscous fluxes. The method to solve the discrete adjoint equation, $A\lambda + b = 0$, is a preconditioned first-degree iterative method, similar to an approximate Newton method [31], given in the following equation:

$$\tilde{A}\left(\lambda^{(l+1)} - \lambda^{(l)}\right) = -A\lambda^{(l)} - b \tag{7}$$

where λ denotes the discrete adjoint vector, and the matrix $\tilde{A}$ is an approximation of the Jacobian matrix A associated with the flow equations, and the vector b. The approximation matrix $\tilde{A}$ comes from the linearization of Roe scheme for the convective flux, and from the linearization of the diffusive flux neglecting the spatial cross derivatives. The resulting linear systems are solved with a few steps of a block LU-SSOR algorithm at each iteration of the iterative method.

4.4 GSA Method. Facing a continuous range of operating conditions, a finite number of them must be selected in order to be able to actually achieve the optimization. These chosen operating conditions must be able to control the performance of the system over the whole range, but must also be limited in number to make the problem computationally tractable.

In this context, what must absolutely be avoided is to find, at the end of the optimization process, an operating condition that has not been selected and at which the performance can be improved without degrading the others. This phenomenon is known as the local maximum behavior [17]. As shown in Ref. [16], it occurs when the gradient of the cost function with respect to the design variables at this operating condition is not linearly dependent on the gradients at the operating conditions initially included in the multipoint optimization problem.

The purpose of the GSA algorithm is to sample the operating condition range—with the smallest possible number of conditions—to set up a multipoint optimization problem so that the local maximum behavior is avoided. It relies on the detection of linear dependencies between the gradients in the range of operating conditions, using a modified Gram–Schmidt algorithm.

From a physical point of view, it is related to the observation that the physics modeled by the equations of fluid dynamics at two different operating conditions can be very similar, or even more, that the sensitivity of the cost function at these two operating conditions can be similar. So, for a given perturbation of the design variables, the cost functions can increase or decrease simultaneously.

In a preprocessing step, a relatively fine sampling of the operating conditions is performed, say with M conditions. The gradients of the objective function with respect to the design variables are computed at these M conditions, with the initial shape. On this initial set, the linear dependencies of the gradients are computed, and a minimal set of $m < M$ linearly independent conditions is selected by Algorithm 1.

Algorithm 1 Gradient Span Analysis Algorithm (GSA)

$\pi_u(v) = \dfrac{\langle v,u\rangle}{\langle u,u\rangle}u$
$indexes \leftarrow \{1,\ldots,M\}$
for $j = 1 \rightarrow M$ **do**
  $c_m \leftarrow 0$
  for $n \in indexes$ **do**
    $q_j \leftarrow \nabla_\alpha J_s(\alpha,\Pi_j)$
    for $i = 1 \rightarrow j-1$ **do** $q_j \leftarrow q_j - \pi_{q_i}(q_j)$ **end for**
    $q_j \leftarrow \dfrac{q_j}{||q_j||}, d \leftarrow 0$
    for $i = 1 \rightarrow M$ **do**
      $v \leftarrow \nabla_\alpha J_s(\alpha,\Pi_i)$
      for $k = 1 \rightarrow j$ **do** $v \leftarrow v - \pi_{q_k}(v)$ **end for**
      if $||v|| < \varepsilon||\nabla_\alpha J_s(\alpha,\Pi_i)||$ **then** $d \leftarrow d+1$ **end if**
    end for
    if $d > c_m$ **then** $q_m \leftarrow q_j, c_m \leftarrow d, n_m \leftarrow n$ **end if**
  end for
  $q_j \leftarrow q_m, indexes \leftarrow indexes\backslash\{n_m\}$
  if $c_m = M$ **then** END **end if**
end for

The approach gives the minimal number of sampling points to control the continuous range of operating conditions [12]. The set of conditions is only valid for the initial shape, and the process should—in theory—be performed for each intermediary shape, or at least for several shapes during the optimization, as in Ref. [17]. In practice, this is only done once, at the setup stage, to preserve the overall computation time. The GSA method requires to compute M flows and gradient solutions, but this number is much smaller than the number of function evaluations needed during the optimization iterations. The GSA algorithm may be performed again at the end of the optimization process to see if the set of conditions is still valid for the final shape. If not, it is easy to restart an optimization process with modified conditions.

From a large number M of operating conditions, the GSA algorithm extracts $m < M$ linearly independent conditions. Note that algorithm 1 depends on a tolerance ϵ. This tolerance has to be carefully chosen, as explained in Ref. [14, Sec. 7.5.2]. If the tolerance ϵ is much higher than the error level observed on the gradient computations, the estimated number m may be much lower than the exact gradient span dimension. Some descent directions may be missed. On the other hand, if the tolerance ϵ is lower than the error level, the number m may be overestimated, and may lead to $m = M$ (very expensive in terms of central processing unit (CPU) cost), i.e., all gradients are seen independent because of the error level. A tolerance $\varepsilon = 0.1$ gives a good compromise in our case.

The GSA procedure shows that large dimensional parametric optimization problems can be solved with a crude sampling of the parameter range [13,14], whereas the gradients were supposed

implicitly linearly independent in Ref. [16], leading to much more refined samplings. Besides, authors in Ref. [16] suggest to use a number of operating conditions close to the number of design variables, which is extremely costly for sophisticated shapes.

To summarize, the GSA algorithm gives a minimal set of m well-chosen conditions which are used to set up the cost function of Eq. (6). In this paper, the weights are computed with the utopia point method [15]. If the weights were computed differently, the m conditions selected by GSA would still be able to control the cost function on the whole range of conditions, as shown in Ref. [12]. Moreover, if the cost function were using these m conditions plus some others, it would be fine too. As shown in Ref. [12], the optimality condition on a set of m well-chosen conditions implies the optimality of derived problems with modified weights built with the same former m conditions and any additional one taken from the space of operating conditions.

5 Numerical Results

The computational fluid dynamics code was previously validated in Ref. [32] for the LS89 case.

In the following, the penalty coefficient σ is set to 100. This is in line with what is done by other authors: $\sigma = 50$ in Ref. [4], σ between 50 and 500 in Ref. [6].

5.1 Parametrization. The LS89 blade geometry was parametrized with a CAD model, shown in Fig. 2 on a profile, using 56 parameters. The parametrization controls both lower and upper surfaces. Therefore, it controls the acceleration region, the throat section, and the diffusion section. Six points $\{P_i(X,Z)\}_{i=1}^{6}$ are chosen along the chord, and nine parameters are associated with each of these six points (except the last one, which only has five parameters). Those nine parameters correspond to the position X along the chord, the position Z on the camber line, the thickness T_{cp} of the blade at this point, the tangent angles U_a and L_a, and the tangent modulus U_{lS}, U_{rS}, L_{lS}, and L_{rS}. The last six are the angle of attack, the chord, the pitch, the leading edge radius, and two coordinates dx and dz needed by the CAD model to locate the airfoil. The trailing edge is not included in the CAD model (also frozen in Ref. [4]) because the CAD model comes from wing parametrization, and cannot handle round trailing edges.

The CAD model chosen for the blade optimization retains a subset of 25 design variables, α, taken from the previous model. The other 31 remaining parameters are fixed to the values obtained from the LS89 geometry. The latter include the last six parameters enumerated above, the parameters associated with P_1 and P_2, and all the X-positions and thicknesses. The parameters associated with P_1 and P_2 control the shape of the leading edge, and since these parameters are very sensitive to small variations, they are frozen to avoid geometric discontinuities (note that the leading edge is frozen in Ref. [4] too). The thicknesses are frozen to satisfy possible structural constraints. The design variables of the initial blade are denoted by α_0.

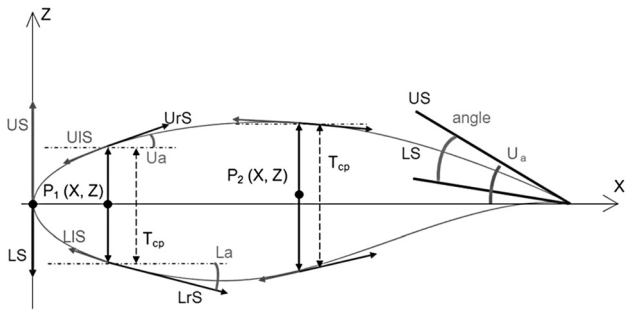

Fig. 2 CAD model parameters

5.2 Gradient Validation With Finite Differences. The gradient of the cost function given by the discrete adjoint method is compared with the one obtained by a second-order finite difference scheme. The number of iterations is set to 2500 for the flow solver, which is enough to decrease the ℓ_2-residual of the fluid density by four orders of magnitude. The number of iterations for the resolution of the adjoint equation (7) is 800. Figure 3 shows the error for every design variable. The design variables #1, #8, and #15 have the highest errors, 14.19%, 9.72%, and 12.41%, respectively. These three variables handle the local camber at different positions, respectively, at around 25%, 50%, and 80% of the chord. The average error is 1.65%, and this is sufficient to show the interest of the GSA method.

The approximations of frozen turbulence and thin shear-layer, made in the differentiation for the discrete adjoint method, are at the origin of these defects. The framework was validated for the 2D Euler equations, for which the difference between adjoint and finite differences vanishes [13]. When the eddy viscosity is fixed in the direct solver, the errors are reduced.

5.3 Selected Conditions. The interval [0.7,1.1] of isentropic Mach numbers $M_{2,is}$ is uniformly sampled with $M = 21$ points. This gives 21 static pressure ratios Π_k in [0.476,0.732], which includes the nominal point $\Pi_{nom} = 0.583$. The algorithm GSA computes the gradient of $J_s(\alpha)$ with respect to α for the 21 conditions Π_k. It outputs $m = 5$ conditions, shown in Table 1, with the corresponding isentropic Mach numbers. The points obtained to sample the interval [0.476,0.732] are very similar to the Chebyshev points of the second kind, used for polynomial interpolation of smooth functions [33]. It reflects the fact that the physics changes continuously with the parameter Π: the total pressure loss and its gradient are smooth functions of Π, and hence a small number of Chebyshev-like points is needed to control their variations. Three points are located in the half of the interval corresponding to the lowest pressure ratio.

The GSA algorithm gives an optimal set of conditions, but it is not unique. Five other conditions could have formed another optimal set, as seen in Sec. 4.4.

5.4 Single-Point Optimization. This section presents the results of the single-point optimization obtained with the L-BFGS-B algorithm [34].

The convergence is reached after 20 iterations of the optimization process, and for a total CPU cost of 144,000 s. Figure 4 shows the evolution of the total pressure loss and the normalized mass flow rate $Q(\alpha)/Q(\alpha_0)$. The process could have probably been stopped around 10 cycles since no more decrease is gained afterward. Table 2 synthesizes the results. The total pressure loss is reduced by 0.63%. The decrease in mass flow rate is much smaller

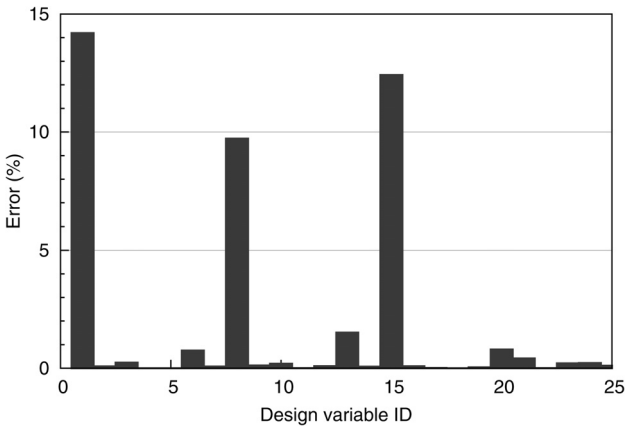

Fig. 3 Error on the gradient computed by the discrete adjoint method

Table 1 Static pressure ratios selected by GSA, and corresponding isentropic Mach numbers

	Π_1	Π_2	Π_3	Π_4	Π_5
Π	0.732	0.680	0.562	0.524	0.476
$M_{2,is}$	0.70	0.78	0.96	1.02	1.10

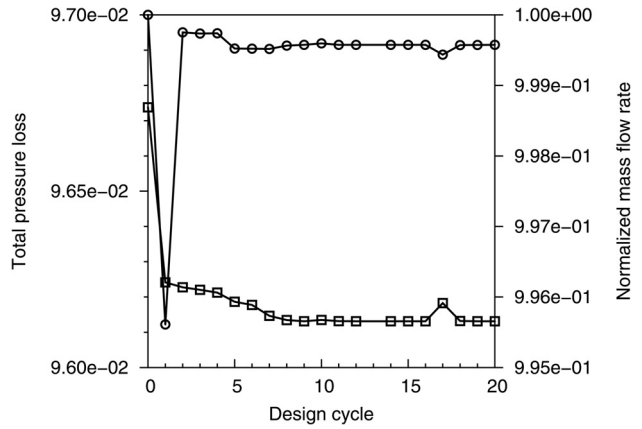

Fig. 4 Evolution of the total pressure loss (square symbol), and correlative evolution of the normalized mass flow rate (circle symbol)

than the maximum acceptable fluctuation ±0.5% defined in Ref. [6]. Figure 5 shows the initial blade geometry and the difference Δz of z-coordinates to subtract to the initial blade geometry to get the final one. Major differences are located in the shock wave region.

The interesting point is to see how the new shape behaves on the range of operating conditions, knowing that it has only been designed taking into account one operating condition. Pressure ratios are sampled uniformly with 21 points in the interval [0.476, 0.732] to draw the polar curve. Figure 6 shows the normalized total pressure loss $\text{tpl}/\text{tpl}(\alpha_0)$, and the normalized mass flow rate $Q/Q(\alpha_0)$, with respect to the static pressure ratio Π, for the initial LS89 shape and the optimized shape. The vertical line locates the nominal condition. Table 3 gives the results obtained at the five conditions that are used for the next multipoint optimization. The total pressure loss decreases for large pressure ratios, but increases for small ones, and the mass flow rate varies a lot. This is what designers call a poor design: the new shape is optimal for the nominal condition, but the performance gets worse at some off-design conditions. A way to obtain better designs is given by the multipoint optimization.

5.5 Multipoint Optimization. This section presents the results of the multipoint optimization.

A single-point optimization at each of the five conditions given by the GSA method in Sec. 5.3 was performed to evaluate the total pressure loss Δtpl_k at each condition. The weights have been set to $\omega_k = |1/\Delta\text{tpl}_k|$ as explained in Sec. 3.2. Table 4 gives the

Table 2 Total pressure loss and normalized mass flow rate for initial and optimal blades, for the single-point optimization at the nominal condition

	Total pressure loss	Normalized mass flow rate
Initial	0.09674	1.0
Optimal	0.09613	0.9996
Variation (%)	−0.6306	−0.0400

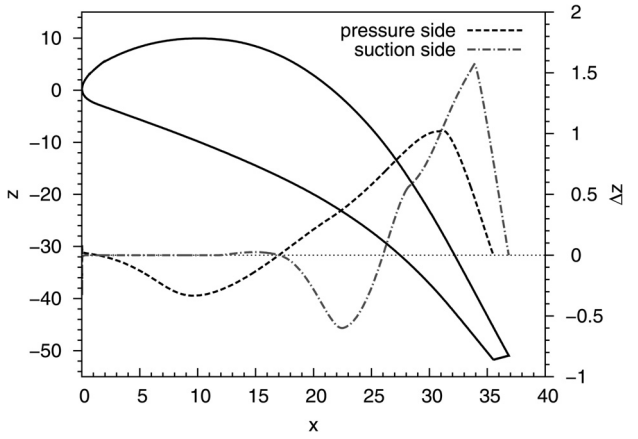

Fig. 5 Initial blade geometry in a (x, z) plane, and difference Δz of z-coordinates between final and initial shapes (frozen trailing edges not shown)

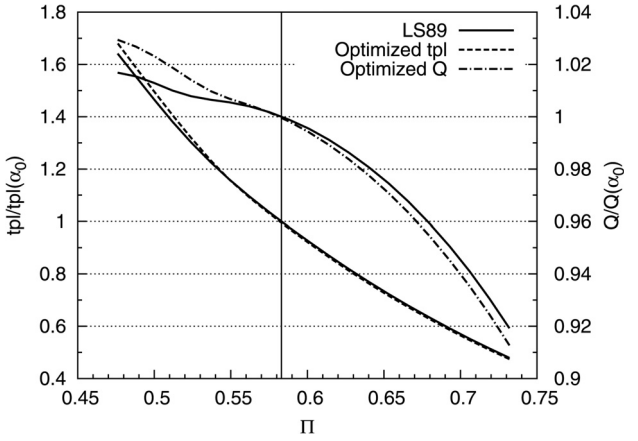

Fig. 6 Normalized mass flow rate and total pressure loss with respect to the pressure ratio Π for initial and optimized shapes

variations of total pressure loss for the five single-point optimizations, and the associated weights.

The cost of one iteration of the multipoint optimization process is much higher than the cost of one iteration of the single-point optimization. One iteration of the multipoint optimization corresponds to five iterations of the single-point optimization process, since the evaluation of the multipoint cost function requires five single-point cost function evaluations. On top of that, the cost of the computation of the weights has to be added.

Figure 7 shows the evolution of the total pressure loss at the five conditions given by the GSA algorithm, and shows that all curves decrease. Figure 8 shows the evolution of the mass flow rate, and shows that all constraints are reasonably respected. In Figs. 7 and 8, the quantities are normalized by their respective initial values.

Table 3 Variations of total pressure loss and mass flow rate at different conditions

	Π	Δtpl (%)	ΔQ (%)
Π_1	0.732	−1.0126	−0.7076
Π_2	0.680	−0.8990	−0.4934
Π_{nom}	0.583	−0.6306	−0.0426
Π_3	0.562	−0.3543	+0.0516
Π_4	0.524	+1.3258	+0.6168
Π_5	0.476	+2.4136	+1.2382

Table 4 Variations of total pressure loss for the five single-point optimizations, and associated weights for the multipoint optimization

	Π_1	Π_2	$\Pi 3$	Π_4	Π_5
Δtpl (%)	−1.038	−0.977	−0.384	−1.026	−2.744
1/|Δtpl|	96.33	102.3	260.3	97.46	36.44

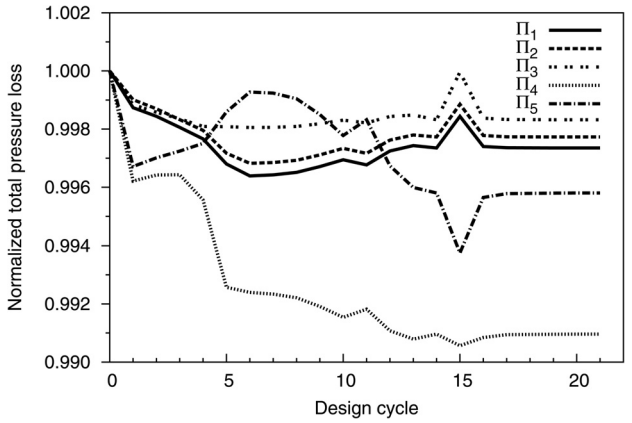

Fig. 7 Evolution of the total pressure loss at the five conditions

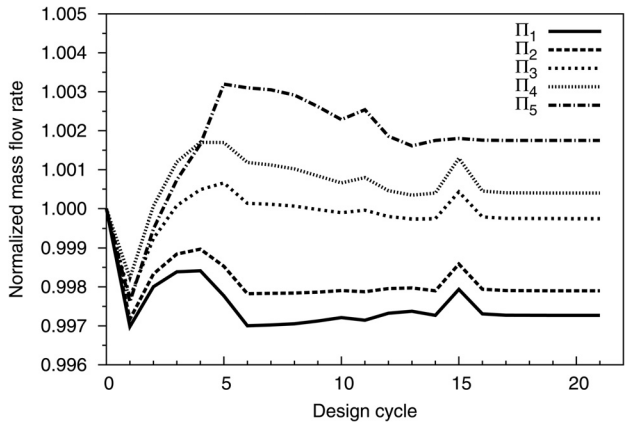

Fig. 8 Evolution of the mass flow rate at the five conditions

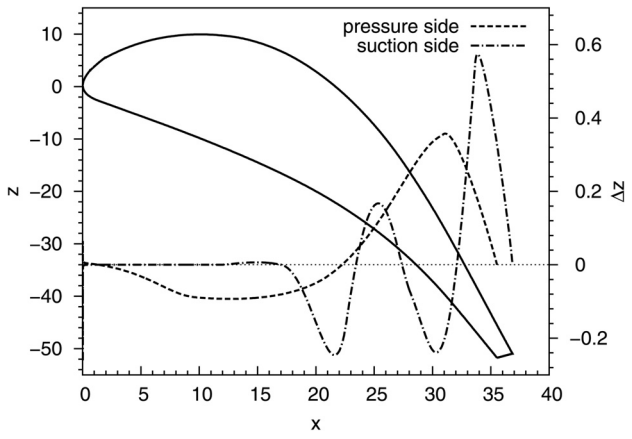

Fig. 9 Initial blade geometry in a (*x*, *z*) plane, and difference Δz of *z*-coordinates between final and initial shapes (frozen trailing edges not shown)

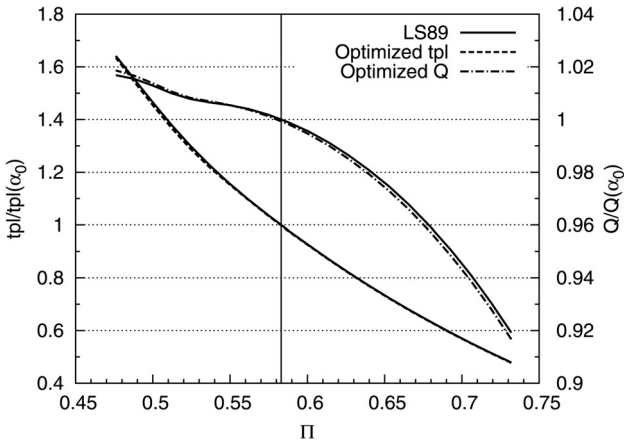

Fig. 10 Normalized mass flow rate and total pressure loss with respect to the pressure ratio Π for initial and optimized shapes

Table 5 Variations of total pressure loss and mass flow rate at different conditions

	Π	Δtpl (%)	ΔQ (%)
Π_1	0.732	−0.2648	−0.2732
Π_2	0.680	−0.2271	−0.2101
Π_{nom}	0.583	−0.2024	−0.0716
Π_3	0.562	−0.1678	−0.0255
Π_4	0.524	−0.9039	+0.0400
Π_5	0.476	−0.4197	+0.1749

Figure 9 compares the initial and the optimized shapes. As in Fig. 5, the first 25% of the suction side remains unchanged due to the parametrization. In our case, both curves in Figs. 5 and 9 have a minimum near $x = 22$, and a maximum near $x = 34$. The main difference for the suction side is located near $x = 29$, where there is an inflection point in Fig. 5 and a completely different pattern in Fig. 9. The curves are quite different for the pressure side, except that the maximum are both located at $x = 31$.

Using again 21 points uniformly sampled in the interval [0.476, 0.732] of pressure ratios, Fig. 10 compares the normalized total pressure loss and the normalized mass flow rate between both shapes over the whole range of operating conditions. Table 5 synthesizes the results, while Fig. 11 shows the differences of total pressure losses between the initial shape and the optimized shapes of the single-point and multipoint optimizations.

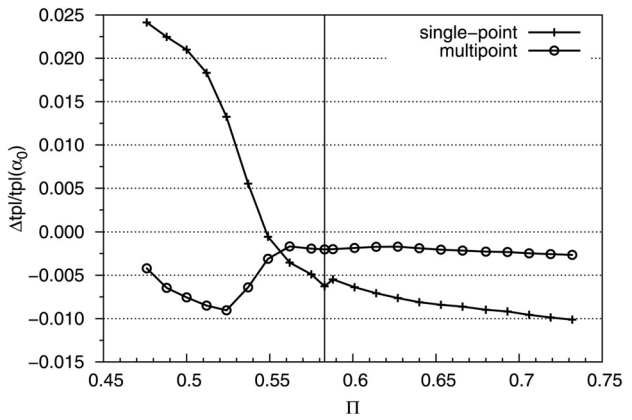

Fig. 11 Differences of total pressure losses between initial and optimized shapes, with respect to the pressure ratio Π

As opposed to Fig. 6, Fig. 10 shows that the mass flow rate fluctuations are kept under 0.3%. Figure 11 shows that the total pressure loss decreases this time at all the operating conditions for the optimized shape obtained by the multipoint optimization, even if the gains are lesser than those of the single-point optimized shape for high pressure ratios. Moreover, note that the single-point curve shows a slight local dip at the nominal condition, which is characteristic of single-point optimizations. It is not observed for the multipoint optimization.

At the initial setup phase, the GSA automatically gave three conditions for low pressure ratios and two for high ones. The GSA applied on the final optimized shape also gives five conditions, which means that the multipoint optimization has not missed any potential performance improvement on the optimal shape.

6 Conclusions

The performance of a system optimized at a nominal condition often significantly deteriorates when the system operates at other operating conditions. Consequently, when the system operates over a range of operating conditions, the designer should control its performance over this range during the design phase, and not a posteriori. The weighted sum method is typically used to perform multi-objective optimizations. Hence, the range of operating conditions has to be discretized, and a preprocessing step needs to select the conditions to include in the weighted sum. The recent GSA method is used to automate the selection of the operating conditions. It guarantees a minimal sample number and therefore a minimal cost, and relies on mathematical theorems. The method also ensures that no performance gain opportunities are missed, i.e., that adding any other operating condition to the weighted sum would not lead to a performance improvement without a degradation at another operating condition.

The methodology is applied to the redesign of the LS89 blade over a wide range of pressure ratios. Our function of interest is the total pressure loss, and the penalty method is used to include the constraint on the mass flow rate into an unconstrained problem formulation. The gradient of the cost function with respect to design variables is computed with the discrete adjoint method, whose computation time does not depend on the number of design variables.

Although the variations of total pressure losses during all the performed optimizations are quite modest, it is clear that the single-point optimization increases the performance at the nominal condition and for higher pressure ratios, but decreases it dramatically at lower pressure ratios. The GSA method only gives five operating conditions to build the multipoint optimization problem. The multipoint optimization performs best at all conditions, and the deformations of the final shape are less marked than those resulting from the single-point optimization. These results shows the interests of the combined GSA and utopia point method applied to turbomachinery design.

Future works are under consideration to improve the present methodology (parametrization, gradient accuracy, and so on). The method is cost-efficient since the adjoint method is used both to setup the optimization problem and to solve it. Therefore, the simple LS89 demonstration opens the perspective of more interesting industrial configurations such as multistage turbomachinery or counter rotative open rotors, with hundreds of design parameters and multiple operating condition ranges.

Acknowledgment

Airbus Group Company and Matthieu Meaux are gratefully acknowledged for making available the optimization chain and the associated tools.

Nomenclature

c_m = dimension of the gradient space computed by GSA
i, j, k, n, d = indices
J_{m} = multipoint cost function
J_{s} = single-point cost function
m = number of vectors in the gradient space given by GSA
M = size of the initial sampling of the condition range
M_1 = Mach number at the inlet plane
$M_{2,\mathrm{is}}$ = isentropic Mach number at the outlet plane
P_{s_i} = static pressure at section i
P_{t_i} = averaged total pressure at section i
Q = mass flow rate
q_j = j^{eme} vector of an orthonormal basis
Re = Reynolds number
tpl = total pressure loss
u, v = gradient vectors
x, y, z = Cartesian coordinates
α = vector of design variables
ε = threshold for GSA
λ = adjoint variables
Π_k = static pressure ratio at condition k
σ = penalty coefficient
ω_k = weights in the multipoint cost function
$\partial\Omega_1$ = inlet plane
$\partial\Omega_2$ = outlet plane

References

[1] Pironneau, O., 1974, "On Optimum Design in Fluid Mechanics," J. Fluid. Mech., **64**(1), pp. 97–110.
[2] Jameson, A., 1988, "Aerodynamic Design Via Control Theory," J. Sci. Comput., **3**(3), pp. 233–260.
[3] Luo, J., Xiong, J., Liu, F., and McBean, I., 2010, "Three-Dimensional Aerodynamic Design Optimization of a Turbine Blade by Using an Adjoint Method," ASME J. Turbomach., **133**(1), p. 011026.
[4] Li, H., Song, L., Li, Y., and Feng, Z., 2010, "2D Viscous Aerodynamic Shape Design Optimization for Turbine Blades Based on Adjoint Method," ASME J. Turbomach., **133**(3), p. 031014.
[5] He, L., and Wang, D. X., 2010, "Concurrent Blade Aerodynamic-Aero-Elastic Design Optimization Using Adjoint Method," ASME J. Turbomach., **133**(1), p. 011021.
[6] Wang, D. X., He, L., Li, Y. S., and Wells, R. G., 2010, "Adjoint Aerodynamic Design Optimization for Blades in Multistage Turbomachines—Part II: Validation and Application," ASME J. Turbomach., **132**(2), p. 021012.
[7] Walther, B., and Nadarajah, S., 2012, "Constrained Adjoint-Based Aerodynamic Shape Optimization of a Single-Stage Transonic Compressor," ASME J. Turbomach., **135**(2), p. 021017.
[8] Luo, J., Zhou, C., and Liu, F., 2013, "Multipoint Design Optimization of a Transonic Compressor Blade by Using an Adjoint Method," ASME J. Turbomach., **136**(5), p. 051005.
[9] Drela, M., 1998, "Pros and Cons of Airfoil Optimization," *Frontiers of Computational Fluid Dynamics*, World Scientific, Singapore, pp. 363–381.
[10] Das, I., and Dennis, J. E., 1997, "A Closer Look at Drawbacks of Minimizing Weighted Sums of Objectives for Pareto Set Generation in Multicriteria Optimization Problems," Struct. Multidiscip. Optim., **14**(1), pp. 63–69.
[11] Arts, T., and Lambert de Rouvroit, M., 1992, "Aero-Thermal Performance of a Two-Dimensional Highly Loaded Transonic Turbine Nozzle Guide Vane: A Test Case for Inviscid and Viscous Flow Computations," ASME J. Turbomach., **114**(1), pp. 147–154.
[12] Gallard, F., Mohammadi, B., Montagnac, M., and Meaux, M., "An Adaptive Multipoint Formulation for Robust Parametric Optimization," J. Optim. Theory Appl., **165**(1) (in press).
[13] Gallard, F., Meaux, M., Montagnac, M., and Mohammadi, B., 2013, "Aerodynamic Aircraft Design for Mission Performance by Multipoint Optimization," AIAA Paper No. 2013-2582.
[14] Gallard, F., 2014, "Aircraft Shape Optimization for Mission Performance," Ph.D., thesis, Université de Toulouse-ISAE, Toulouse, France.
[15] Marler, R. T., and Arora, J. S., 2004, "Survey of Multi-Objective Optimization Methods for Engineering," Struct. Multidiscip. Optim., **26**(6), pp. 369–395.
[16] Li, W., Huyse, L., and Padula, S., 2002, "Robust Airfoil Optimization to Achieve Drag Reduction Over a Range of Mach Numbers," Struct. Multidiscip. Optim., **24**(1), pp. 38–50.
[17] Zingg, D., and Elias, S., 2006, "Aerodynamic Optimization Under a Range of Operating Conditions," AIAA J., **44**(11), pp. 2787–2792.
[18] Reuther, J. J., Jameson, A., Alonso, J. J., Rimlinger, M. J., and Saunders, D., 1999, "Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers, Part 1," J. Aircr., **36**(1), pp. 51–60.
[19] Marler, R. T., and Arora, J. S., 2010, "The Weighted Sum Method for Multi-Objective Optimization: New Insights," Struct. Multidiscip. Optim., **41**(6), pp. 853–862.

[20] Meaux, M., Cormery, M., and Voizard, G., 2004, "Viscous Aerodynamic Shape Optimization Based on the Discrete Adjoint State for 3D Industrial Configurations," 4th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS), Jyväskylä, Finland, July 24–28.
[21] Cambier, L., Gazaix, M., Heib, S., Plot, S., Poinot, M., Veuillot, J.-P., Boussuge, J.-F., and Montagnac, M., 2011, "An Overview of the Multi-Purpose elsA Flow Solver," Aerosp. Lab., **2**, p. AL02-10.
[22] Puigt, G., Gazaix, M., Montagnac, M., Le Pape, M.-C., de la Llave Plata, M., Marmignon, C., Boussuge, J.-F., and Couaillier, V., 2011, "Development of a New Hybrid Compressible Solver Inside the CFD elsA Software," AIAA Paper No. 2011-3379.
[23] Spalart, P. R., and Allmaras, S. R., 1992, "A One-Equation Turbulence Transport Model for Aerodynamic Flows," AIAA Paper No. 92-0439.
[24] Roe, P. L., 1981, "Approximate Riemann Solvers, Parameter Vectors and Difference Schemes," J. Comput. Phys., **43**(2), pp. 357–372.
[25] van Leer, B., 1979, "Towards the Ultimate Conservative Difference Scheme. V. A Second-Order Sequel to Godunov's Method," J. Comput. Phys., **32**(1), pp. 101–136.
[26] van Albada, G. D., van Leer, B., and Roberts, W. W., 1982, "A Comparative Study of Computational Methods in Cosmic Gas Dynamics," Astron. Astrophys., **108**(1), pp. 76–84.
[27] Yoon, S., and Jameson, A., 1988, "Lower-Upper Symmetric-Gauss–Seidel Method for the Euler and Navier–Stokes Equations," AIAA J., **26**(9), pp. 1025–1026.
[28] Jameson, A., 1982, "Steady State Solutions of the Euler Equations for Transonic Flow by a Multigrid Method," *Transonic, Shock, and Multidimensional Flows: Advances in Scientific Computing*, Academic, New York, pp. 37–70.
[29] Jameson, A., Martinelli, L., and Pierce, N. A., 1998, "Optimum Aerodynamic Design Using the Navier–Stokes Equations," Theor. Comput. Fluid Dyn., **10**(1), pp. 213–237.
[30] Nielsen, E. J., and Anderson, W. K., 1999, "Aerodynamic Design Optimization on Unstructured Meshes Using the Navier–Stokes Equations," AIAA J., **37**(11), pp. 1411–1419.
[31] Pinel, X., and Montagnac, M., 2013, "Block Krylov Methods to Solve Adjoint Problems in Aerodynamic Design Optimization," AIAA J., **51**(9), pp. 2183–2191.
[32] Gourdain, N., Gicquel, L. Y. M., and Collado, E., 2012, "Comparison of RANS and LES for Prediction of Wall Heat Transfer in a Highly Loaded Turbine Guide Vane," J. Propul. Power, **28**(2), pp. 423–433.
[33] Trefethen, L. N., 2013, *Approximation Theory and Approximation Practice*, SIAM, Philadelphia.
[34] Byrd, R., Lu, P., Nocedal, J., and Zhu, C., 1995, "A Limited Memory Algorithm for Bound Constrained Optimization," SIAM J. Sci. Comput., **16**(5), pp. 1190–1208.