Adjoint Method for Shape Optimization in Real-Gas Flow Applications

M. Pini

Assistant Professor
Propulsion and Power,
Aerospace Engineering Faculty,
Delft University of Technology,
Kluyverweg 1,
Delft 2629 HS, The Netherlands
e-mail: m.pini@tudelft.nl

G. Persico

Assistant Professor
Laboratorio di Fluidodinamica delle Macchine,
Dipartimento di Energia,
Politecnico di Milano,
via Lambruschini, 4,
Milano 20156, Italy
e-mail: giacomo.persico@polimi.it

D. Pasquale

Dipartimento di Ingegneria
Meccanica e Industriale,
Universitá degli studi di Brescia,
via Branze, 38,
Brescia 25123, Italy
e-mail: david.pasquale@ing.unibs.it

S. Rebay

Associate Professor
Dipartimento di Ingegneria
Meccanica e Industriale,
Universitá degli studi di Brescia,
via Branze, 38,
Brescia 25123, Italy
e-mail: stefano.rebay@ing.unibs.it

1 Introduction

Optimization techniques play a key role in the present-day design process of turbomachinery; thanks to the progressive increase of computational capability, high-fidelity solution methods based on computational fluid dynamics (CFD) are now routinely applied within these optimization algorithms, resulting in automatic design tools.

The turbomachinery design process offers optimization challenges at many levels, resulting in a step-by-step procedure of increasing fidelity level. The final turbomachinery layout is then usually a complex combination of the outcomes provided by each optimization step, starting from the preliminary choice of the number of stages and of the stage arrangement [1], to the through-flow (or axisymmetric) design [2–4], up to the detailed blade shape definition [5,6].

In the last decades, inverse design methods [7,8] and CFD-based shape optimization procedures were specifically devel-oped for turbomachinery applications. CFD-based shape opti-mization methods are, in many cases, supported by the application of stochastic methods, such as evolutionary algo-rithms [9]. These techniques usually search for the optimal shape by resorting to genetic algorithms (GA) coupled to sur-rogate models to reduce the overall computational cost of the optimization procedure [10,11]. Such class of methods has a series of advantages, well summarized as follows: (i) the

capability of treating nonsmooth and oscillating fitness func-tions, (ii) the ability of exploring a wide range of possible configurations, (iii) the simultaneous identification of a set of acceptable solutions, and (iv) the extension to multi-objective and multipoint optimization problems in a relatively straight-forward manner [12–15].

However, if the shape is controlled by a large number of design variables, the cost of the overall optimization process may become prohibitive and this practically limits the available design space and the number of possible solutions. Further-more, in many cases the performance improvement of realistic turbomachinery cascades is achieved by optimizing existing blade geometries in presence of mechanical and geometrical constraints; as a consequence, the optimal shape cannot be sig-nificantly different from the initial configuration, thus involving minor geometry modifications. In this context, gradient-based design techniques based on adjoint provide relevant improve-ments in terms of computational cost [16]. These methods automatically determine the optimal configuration by a rela-tively low number of steps, outperforming GAs for these types of problem. In addition, the relatively smoothness of the usual aerodynamic objective functions considerably lowers the risk of blow-up of adjoint-based optimization. As a consequence, the adjoint method really represents a viable and effective choice for the optimization of existing turbomachinery cascades.

The benefits of the adjoint are highly attractive for maximizing turbine performance in ORCs, whereby the high computational resources required to perform accurate real-gas CFD simulations make EAs (GAs) even more demanding to run compared to other

turbomachinery applications. ORC technology is now receiving considerable attention due to its flexibility and competitiveness for small-to-medium power range applications, especially in combination to the exploitation of renewable energy sources [17]. The turbine is recognized to be the most critical component of the system, as the overall system performance is highly sensitive to turbine efficiency. Lot of efforts are currently ongoing to improve the fluid-dynamic design of expanders in order to boost ORCs capabilities, thus contributing to a widespread diffusion of new concept systems for stationary, distributed, and on-board applications (e.g., heat recovery from truck engines [18]). In this regard, due to the shortage of experimental data and, consequently, of reliable efficiency prediction methods, design adaptations and fully novel cascade configurations for ORC turbo-expanders can be effectively devised only through proper real-gas CFD-based design methods. However, despite the large interest around ORCs, very few research works are dedicated to the implementation of automatic shape optimization algorithms specific to ORC turbine cascades. Some pioneering works on this topic can be found in Refs. [19] and [20]. All authors undertake the optimization problem using evolutionary algorithms for the redesign of existing transonic/supersonic blades, whereas no research work has been published yet on adjoint methods for real-gas flows.

This work contributes to draw new perspectives in this research area, proposing an adjoint optimization algorithm for real-gas flow applications. The method combines an inviscid flow solver and an inviscid adjoint solver, both embedding a LuT method for a fast and accurate calculation of fluid thermophysical properties. The turbine geometry is shaped by using state-of-the-art NURBS curves and a preconditioned steepest descent technique is selected as optimizer. The potential of the approach is demonstrated in two ways: first, by improving the performance of an existing converging–diverging turbine cascade operating in supersonic conditions; second, by comparing the performance gain achieved by the full real-gas adjoint method against that obtained through adjoint methods based on ideal EoS. Turbulent simulations are finally accomplished at the last stage of the design process to assess the results attained by the inviscid optimization method.

The paper is organized as follows: Sec. 2 discusses the shape optimization methodology developed in this study with particular emphasis on adjoint theory and design chain construction; Sec. 3 briefly describes the numerical tools employed to resolve the fluid-dynamic and the design problem, while Sec. 4 reports the outcomes of the optimization. Fully turbulent simulations of the baseline and optimized configurations are presented in Sec. 5, whereas the conclusive part of the paper is dedicated to studies on the effect of thermodynamic modeling on the optimized blade performance.

2 Background on Adjoint Method for Fluid Mechanics

In this section, the theoretical basis of the adjoint method is briefly recalled, with particular emphasis on the discrete approach. In a nutshell, the discrete adjoint equations are constructed by differentiating the discretized flow equations. A general mathematical formulation of such an approach, including the grid deformation effects, is now extensively presented. The formulation here described enables to compute the sensitivity derivatives of the fitness function, which depend on the flow equations of motion solved on either structured or unstructured meshes. Notice that all the vectors specified in the following are conventionally assumed as column vectors.

Let J be the cost function to be minimized in an optimization problem. In fluid dynamics, J generally depends on a vector of physical and geometrical design variables α_{phy} and α_{geo}, respectively, state variables $\mathbf{u}$, and grid points $\mathbf{X}$. State variables, in turn, depend on both physical and geometrical design variables, while grid variables are only function of the geometrical design variables. The functional J can therefore be written as

$$J = J[\alpha_{\text{phy}}, \mathbf{u}(\alpha_{\text{phy}}, \alpha_{\text{geo}}), \mathbf{X}(\alpha_{\text{geo}})] \qquad (1)$$

The state variables $\mathbf{u}$ are also constrained to satisfy the fluid-dynamic governing equations. For steady-state problems, the governing equations can be symbolically expressed as

$$\mathbf{R}[\alpha_{\text{phy}}, \mathbf{u}(\alpha_{\text{phy}}, \alpha_{\text{geo}}), \mathbf{X}(\alpha_{\text{geo}})] = 0 \qquad (2)$$

and can be regarded as an equality constraint for the minimization problem.

Notice that Eq. (2) holds for an arbitrary choice of the design variables α_{phy} and α_{geo}. This implies that the total derivatives of Eq. (2) with respect to α_{phy} and α_{geo}, reported in Eq. (3), are identically null, as long as $\mathbf{u}$ is the solution of Eq. (2) for any α_{phy} and α_{geo}

$$\frac{\mathrm{d}\mathbf{R}}{\mathrm{d}\alpha_{\text{geo}}} = \frac{\partial \mathbf{R}}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \alpha_{\text{geo}}} + \frac{\partial \mathbf{R}}{\partial \mathbf{X}} \frac{\mathrm{d}\mathbf{X}}{\mathrm{d}\alpha_{\text{geo}}} = 0$$
$$\frac{\mathrm{d}\mathbf{R}}{\mathrm{d}\alpha_{\text{phy}}} = \frac{\partial \mathbf{R}}{\partial \alpha_{\text{phy}}} + \frac{\partial \mathbf{R}}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \alpha_{\text{phy}}} = 0 \qquad (3)$$

The fitness function and the constraint, defined according to the above expressions, are written considering a general problem, in which both the physical quantities and the shape of the geometry of interest can be modified in the procedure. However, in standard shape optimization problems, the physical variables appearing in expressions (1) and (2) are to be considered as fixed known parameters (i.e., are not considered variables in the design process). The gradient of the fitness function has to be computed with respect to the geometrical design variables α_{geo} only and is therefore expressed as

$$\frac{\mathrm{d}J}{\mathrm{d}\alpha_{\text{geo}}} = \frac{\partial J}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \alpha_{\text{geo}}} + \frac{\partial J}{\partial \mathbf{X}} \frac{\mathrm{d}\mathbf{X}}{\mathrm{d}\alpha_{\text{geo}}} \qquad (4)$$

The relation (1) shows that the gradient is a function of the first-order state sensitivity derivatives $\partial \mathbf{u}/\partial \alpha_{\text{geo}}$. The direct calcu-lation of these terms is usually a very demanding operation in terms of computational cost. The simplest but computationally prohibitive method to compute $\partial \mathbf{u}/\partial \alpha_{\text{geo}}$ is the finite difference technique (FDM). As an alternative, $\partial \mathbf{u}/\partial \alpha_{\text{geo}}$ can be obtained by solving the first equation of Eq. (3), rewritten as

$$\frac{\partial \mathbf{u}}{\partial \alpha_{\text{geo}}} = -\left(\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right)^{-1} \left(\frac{\partial \mathbf{R}}{\partial \mathbf{X}} \frac{\mathrm{d}\mathbf{X}}{\mathrm{d}\alpha_{\text{geo}}}\right) \qquad (5)$$

However, the calculation of Eq. (5) requires a considerable CPU memory utilization as a series of N decoupled linear systems (with N the columns of the constant matrix $(\partial \mathbf{R}/\partial \mathbf{X})(\mathrm{d}\mathbf{X}/\mathrm{d}\alpha_{\text{geo}})$) has to be solved. A more efficient strategy, allowing to avoid the computation of Eq. (3), is achieved by plugging the previous relation into Eq. (4). The resulting gradient can be therefore expressed in the form

$$\frac{\mathrm{d}J}{\mathrm{d}\alpha_{\text{geo}}} = -\frac{\partial J}{\partial \mathbf{u}} \left(\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right)^{-1} \left(\frac{\partial \mathbf{R}}{\partial \mathbf{X}} \frac{\mathrm{d}\mathbf{X}}{\mathrm{d}\alpha_{\text{geo}}}\right) + \frac{\partial J}{\partial \mathbf{X}} \frac{\mathrm{d}\mathbf{X}}{\mathrm{d}\alpha_{\text{geo}}} \qquad (6)$$

In the previous relation, the vector $-\partial J/\partial \mathbf{u}(\partial \mathbf{R}/\partial \mathbf{u})^{-1}$ is named adjoint or costate variables vector. The adjoint variables $\mathbf{v}$ are consequently determined through the resolution of the adjoint linear system, rewritten as

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right)^{\mathrm{T}} \mathbf{v} = -\left(\frac{\partial J}{\partial \mathbf{u}}\right)^{\mathrm{T}} \qquad (7)$$

Equation (7) represents the discrete form of the adjoint equation, e.g., Ref. [21]. The gradient vector of the original objective function is finally expressed, through the above relations, as

$$\left(\frac{\mathrm{d}J}{\mathrm{d}\alpha_{\mathrm{geo}}}\right)^{\mathrm{T}} = \left(\frac{\mathrm{d}\mathbf{X}}{\mathrm{d}\alpha_{\mathrm{geo}}}\right)^{\mathrm{T}} \left[\left(\frac{\partial J}{\partial \mathbf{X}}\right)^{\mathrm{T}} + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{X}}\right)^{\mathrm{T}} \mathbf{v}\right] \qquad (8)$$

The term in brackets of Eq. (8) indicates the gradient of the fitness function with respect to the entire set of grid points $\mathbf{X}$, namely when $\mathbf{X} = \alpha_{\mathrm{geo}}$, while being $\left(\mathrm{d}\mathbf{X}/\mathrm{d}\alpha_{\mathrm{geo}}\right)^{\mathrm{T}}$ a purely geometrical sensitivity term representing the rate of grid perturbation once the design parameters are moved from their original position. In fact, during the design process, the set of parameters α_{geo} is iteratively displaced according to a given search direction (i.e., the gradient) and the resulting geometry variation affects in turn the topology of the mesh.

A typical and effective choice to cope with significant geometry alterations is to define it through a set of control points pertaining to some parameterization scheme. In this case, additional information is required to completely establish a relation between the computational grid $\mathbf{X}$ and the parameters α_{geo} of the (parametric) curves. In this work, this further knowledge is specified through the mesh deformation equation. By expressing such equation as $\mathbf{X} = \mathbf{X}(\mathbf{X}_{\mathrm{b_{old}}}, \Delta\mathbf{X}_{\mathrm{b}})$, whereby $\Delta\mathbf{X}_{\mathrm{b}} = \mathbf{X}_{\mathrm{b}} - \mathbf{X}_{\mathrm{b_{old}}}$ indicates the deformation of the shape to be optimized between two consecutive design steps and $\mathbf{X}_{\mathrm{b}}$ the surface grid points, the relation between $\mathbf{X}$ and α_{geo} can be finally written as

$$\mathbf{X} = \mathbf{X}(\mathbf{X}_{\mathrm{b_{old}}}, \Delta\mathbf{X}_{\mathrm{b}}(\alpha_{\mathrm{geo}})) \qquad (9)$$

By deriving Eq. (9) against α_{geo} using the chain rule and plugging the resulting expression in Eq. (8), we conclusively obtain

$$\left(\frac{\mathrm{d}J}{\mathrm{d}\alpha_{\mathrm{geo}}}\right)^{\mathrm{T}} = \left(\frac{\mathrm{d}\mathbf{X}_{\mathrm{b}}}{\mathrm{d}\alpha_{\mathrm{geo}}}\right)^{\mathrm{T}} \left(\frac{\partial \mathbf{X}}{\partial \Delta\mathbf{X}_{\mathrm{b}}}\right)^{\mathrm{T}} \left[\left(\frac{\partial J}{\partial \mathbf{X}}\right)^{\mathrm{T}} + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{X}}\right)^{\mathrm{T}} \mathbf{v}\right] \qquad (10)$$

where $\left(\mathrm{d}\mathbf{X}_{\mathrm{b}}/\mathrm{d}\alpha_{\mathrm{geo}}\right)^{\mathrm{T}}$ is simply equal to $\left(\mathrm{d}\Delta\mathbf{X}_{\mathrm{b}}/\mathrm{d}\alpha_{\mathrm{geo}}\right)^{\mathrm{T}}$.

The gradient components are efficiently evaluated by applying the reverse mode of automatic differentiation at different levels, preliminarily to construct the adjoint system, then to obtain each term of Eq. (10). The methodology used in the present context to compute the first two right-hand side sensitivities is presented in the following. An exhaustive description of the AD principles and some alternative gradient formulations can be found in Refs. [16,22,23].

Surface Representation. In computer-aided design (CAD), the surfaces are generally represented by a series of weighted basis functions. B-splines, NURBS (nonuniform rational B-splines) or Chebyshev Polynomials are examples of parametric curves usually adopted in CFD-based automatic design. NURBS curves are probably the most popular curve and surface in computer graphics and nowadays they are the standard for curve and surface description in CAD software. NURBS curves allow for representing the geometry surface, i.e., the vector $\mathbf{X}_{\mathrm{b}}$, through an algebraic relation synthetically expressed as

$$\mathbf{X}_{\mathrm{b}} = \mathbf{X}_{\mathrm{b}}(\mathbf{u}, \omega, \mathbf{P}) \qquad (11)$$

where $\mathbf{u}$, ω, and $\mathbf{P}$ are, respectively, the knot sequence, the weights, and the control points of parametric curves. A more comprehensive description of NURBS parameterization technique can be found in Refs. [24] and [25]. In the present work, only the control points are considered as design parameters, namely $\alpha_{\mathrm{geo}} = \mathbf{P}$, while assuming the remaining parameters constant throughout the optimization procedure. Equation (11) greatly simplifies and can be then straightforwardly differentiated to obtain the sensitivity $\left(\mathrm{d}\mathbf{X}_{\mathrm{b}}/\mathrm{d}\alpha_{\mathrm{geo}}\right)^{\mathrm{T}}$.

Mesh Deformation. Grid deformation algorithms are still an interesting area of research in the field of computational geometry.

Several methodologies are currently available to accurately deform the mesh for an assigned displacement vector. The most common strategy is based on spring analogy [23], although this method suffers from severe limitations in case of significant grid displacements. More innovative approaches rely on the use of radial basis functions (RBFs) [26] and have been taken as reference in the present work. RBFs have become a well-established technique for interpolating scattered data and they are typically adopted in fluid–structure interaction to transfer information over the discrete fluid–structure interface. The most relevant quality of the RBF-based mesh deformation is the capability of individually moving each mesh node without knowing mesh connectivity informations. This makes the method very suitable for both structured and unstructured grids. RBFs are interpolating functions approximating the displacement of a single mesh point s through a weighted sum of basis functions in the form

$$s(\mathbf{X}) = \sum_{j=1}^{N_{X_{\mathrm{b}}}} w_j \phi\left(\|\mathbf{X} - \mathbf{X}_{\mathrm{b}_j}\|\right) + p(\mathbf{X}_{\mathrm{b}}) \qquad (12)$$

where $\mathbf{X}_{\mathrm{b}_j} = [x_{\mathrm{b}_j}, y_{\mathrm{b}_j}, z_{\mathrm{b}_j}]$ are the surface points where the displacements are known (e.g., blade surface), p an additional linear polynomial which preserves the uniqueness of the solution, $N_{X_{\mathrm{b}}}$ the total number of surface points, and ϕ a given basis function with respect to the Euclidean distance $\|\mathbf{X} - \mathbf{X}_{\mathrm{b}_j}\|$. The choice of the basis function may considerably affect the quality of the deformed mesh, though different classes are suitable for mesh movement problems. In the present work volume splines are used, which equate as $\phi\left(\|\mathbf{X} - \mathbf{X}_{\mathrm{b}_j}\|\right) = \|\mathbf{X} - \mathbf{X}_{\mathrm{b}_j}\|$. The weights $\mathbf{w}$ and the polynomial p coefficients are determined by dictating the interpolating conditions at any point of the surface of interest, yielding

$$s(\mathbf{X}_{\mathrm{b}_i}) = \mathbf{d}_{\mathrm{b}_i} \qquad (13)$$

where $d_{\mathrm{b}_i} = \Delta X_{\mathrm{b}} = X_{\mathrm{b}_i} - X_{\mathrm{b}_{i_{\mathrm{old}}}}$ is the displacement of the i surface point. To close the system, a zero condition on the additional polynomial $p(\mathbf{X}_{\mathrm{b}})$ is usually prescribed, namely

$$\sum_{j=1}^{N_{X_{\mathrm{b}}}} w_j p(\mathbf{X}_{\mathrm{b}_j}) = 0 \qquad (14)$$

Equations (13) and (14) can be assembled in matrix form to quickly find the weights $\mathbf{w}$ and the coefficients β of the polynomial. The resulting linear system is written as

$$\begin{bmatrix} \mathbf{d}_{\mathrm{b}} \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{M}_{\mathrm{b,b}} & \mathbf{P}_{\mathrm{b}} \\ \mathbf{P}_{\mathrm{b}}^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \beta \end{bmatrix} \qquad (15)$$

where $\mathbf{M}_{\mathrm{b,b}}$ indicates a square matrix (with $N_{X_{\mathrm{b}}}$ rows and columns) containing the evaluation of the basis function $\phi_{i,j} = \phi\left(\|\mathbf{X}_{\mathrm{b}_i} - \mathbf{X}_{\mathrm{b}_j}\|\right)$, $\mathbf{P}_{\mathrm{b}}$ a $N_{X_{\mathrm{b}}} \times 4$ matrix with a single row j given by $[1 \quad x_{\mathrm{b}_j} \quad y_{\mathrm{b}_j} \quad z_{\mathrm{b}_j}]$. The spatial displacement of an interior grid node is obtained by evaluating the interpolation function (12) at a generic internal grid point j, whose new position is finally simply given by

$$X_j = X_{j_{\mathrm{old}}} + d_j \qquad (16)$$

Equation (16) can be regarded as an explicit form of equation (9). Notice that all the procedures involved to achieve relation (14) are purely algebraic; therefore, the method is reasonably fast to differentiate for obtaining the sensitivities $\left(\partial \mathbf{X}/\partial \Delta\mathbf{X}_{\mathrm{b}}\right)^{\mathrm{T}}$.

3 Computational Models

Flow Solver. The discrete form of the Euler equations is solved using the in-house code zFlow, which approximates the model

equations by the so-called finite element/finite volume method [27] with implicit time integration. This results in a very efficient solution method, which is a crucial feature when the code is introduced within a design methodology. By virtue of a generalized thermodynamic treatment, zFlow allows for dealing with real-gas flows governed by complex (but very accurate) equations of state and can be applied to steam turbines, refrigerating compressors, and ORCs turbines, see e.g., Refs. [1] and [28]. The thermodynamic capabilities of the solver have been recently extended to cope with LuT approaches. Herein, the thermodynamic properties are computed by resorting to a consistent interpolation method suitable for real-gas flow simulations. A brief description of the present LuT method is given in the following.

LuT Method. LuT approach allows to model the thermophysical behavior of any pure fluids by properly interpolating thermodynamic data stored within the nodes of a computational grid. A fairly simple method has been proposed in Ref. [29], whereby each quantity is obtained by separately interpolating the corresponding nodal values using surrogate functions of increasing accuracy. Despite its simplicity and easy implementation, the method does not satisfy thermodynamic consistency, and this may induce spurious oscillations during the convergence process of the solver. In the present work, a different approach is considered, in which consistency is intrinsically fulfilled by constructing a fundamental relation $e(v, s)$ for each cell of the thermodynamic grid. Bicubic polynomial form is used, whose coefficients are evaluated by resorting to entropy, specific volume, and internal energy of the surrounding 16 nodes.

The whole set of thermodynamic quantities normally involved in CFD calculations, i.e., P, T, h, c, is then provided by conveniently deriving the analytical expression of the fundamental relation. As a result, the thermodynamic consistency of the method is automatically satisfied. The full set of thermodynamic properties as a function of s, v can be inferred as

$$
P = -\left(\frac{\partial e}{\partial v}\right)_s, \quad T = \left(\frac{\partial e}{\partial s}\right)_v
$$
$$
h = e - \left(\frac{\partial e}{\partial v}\right)_s v, \quad c = v\sqrt{\left(\frac{\partial^2 e}{\partial v^2}\right)_s} \tag{17}
$$

The construction of the thermodynamic mesh is based on the discretization of the saturation line according to a given temperature interval. The points can be uniformly or variably spaced through a spline-based reconstruction method, allowing also local refinements close to the critical point. The resulting one-dimensional grid represents the support line on which the LuT is stemmed by proceeding along directions locally orthogonal to the line using $\log(v)$, s as independent variables. An example of tabulated region constructed around an isentropic transformation ideally experienced by an ORC turbine is sketched in the left Fig. 1. The same thermodynamic area can delimit grids of increasing refinement, as shown in the right Fig. 1 for a mesh of 10^4 elements.

For a given input pair of thermodynamic properties (e.g., vs, ve, Ps, PT), the calculation strategy is divided into three main steps: the mesh cell containing the desired point is initially identified through a *kd-tree search* algorithm [30] together with the corresponding set of coefficients; then, the given pair is converted into the explicit state vs by iteratively solving a nonlinear equation (system); finally all thermodynamic properties are calculated by means of the set of equations (17). The accuracy and the computational efficiency of the method have been proved by an extensive numerical campaign. The interested reader is referred to Ref. [31] for full details.

Adjoint Solver. Basically the adjoint solver implements a linearization of the governing equations, thus inheriting the numerical features of the native flow solver. The adjoint problem requires the calculation of a linear algebraic system in the form $\mathbf{Av} = \mathbf{b}$, where $\mathbf{A} = (\partial \mathbf{R}/\partial \mathbf{u})^T$, $\mathbf{b} = -(\partial J/\partial \mathbf{u})^T$. In order to solve the adjoint system, iterative matrix-free techniques are preferred, as they do not require the explicit calculation of the Jacobian matrix $\partial \mathbf{R}/\partial \mathbf{u}$, but they only need to evaluate the product $(\partial \mathbf{R}/\partial \mathbf{u})^T \mathbf{v}$ at each iteration. In particular, in the present research, the iterative flexible generalized minimal residual method (FGMRES), see Ref. [32], is adopted. The adjoint system of equations (7) is then solved by constructing the vectors $(\partial \mathbf{R}/\partial \mathbf{u})^T \mathbf{v}$ and $(\partial J/\partial \mathbf{u})^T$ by exploiting AD in reverse mode, see Ref. [33]. In this way, the high-resolution of the numerical method is intrinsically preserved. The efficiency of the FGMRES is further enhanced by preconditioning the linear system through the ILU0 (no fill-in allowed) or ILUT (fill-in allowed by fixing a threshold value) incomplete factorization techniques applied to the first-order Jacobian matrix.

Extension of Adjoint Method to Arbitrary Equations of State. In close analogy with the zFlow solver, the adjoint counterpart is capable to deal with real-gas effects using the same thermodynamic framework. Nevertheless, its generalization to arbitrary equations of state implies further issues in the construction of the adjoint equations. These are mostly related to the complexity of the thermodynamic relations, which strongly impacts on the formulation of numerical methods for generalized EoS. The latter
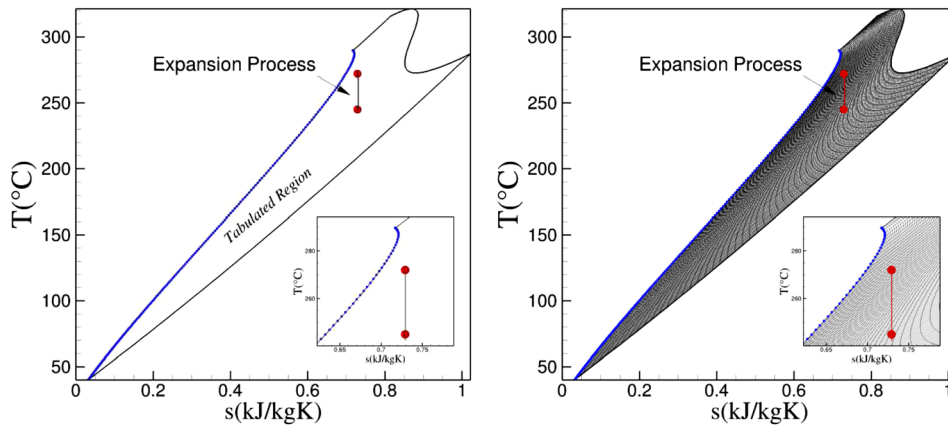

Fig. 1 Left: thermodynamic region covered by a LuT for the siloxane MDM in the $T-s$ diagram. Blue dots located on the vapor saturation line represent the support points of the grid. Right: thermodynamic grid of 10,000 elements built within the tabulated region. Notice the high distortion of the mesh, as it is originally generated using different independent variables.

are in fact a rather involved evolution of schemes originally developed for perfect gases [34].

As an example, most of the flux splitting methods successfully used in real-gas flows need the computation of secondary thermodynamic properties to accurately evaluate the fluxes among adjacent cells. These are usually limited to the derivatives of pressure with respect to density $\nabla \mathbf{p}_\rho$ and internal energy $\nabla \mathbf{p}_e$ in case of inviscid flows. As outlined in Ref. [27], the speed of sound of complex molecule, which is a crucial property also for centered schemes, can be recovered by properly combining pressure derivatives. Furthermore, for implicit time integration, derivatives of pressures are also necessary to obtain Jacobian matrices. In this case, the dependence of the interface speed of sound on the conserved variables $\mathbf{u}$ is often neglected, resulting in a frozen speed of sound approach.

Since exact fully linearized adjoint equations are of concern in the present research, second-order derivatives emerge in the Jacobian matrices. According to the features of the real-gas numerical schemes and by focusing on the dependency of $\mathbf{R}$ from $\mathbf{u}$, the set of discretized governing equations $\mathbf{R}$ becomes a function of the conservative variables through an enlarged state $\mathbf{q}$ and can be concisely written as

$$\mathbf{R} = \mathbf{R}[\mathbf{q}(\mathbf{u})] \tag{18}$$

where $\mathbf{q} = (\mathbf{u}, \mathbf{p}, \nabla \mathbf{p}_e, \nabla \mathbf{p}_\rho)$. As pressure $\mathbf{p}$ and its derivatives $\nabla \mathbf{p}_e, \nabla \mathbf{p}_\rho$ are functions of the conservative state $\mathbf{u}$, the following relation for the Jacobian holds:

$$\left(\frac{d\mathbf{R}}{d\mathbf{u}}\right) = \left(\frac{\partial \mathbf{R}}{\partial \mathbf{q}}\right)\left(\frac{d\mathbf{q}}{d\mathbf{u}}\right) \tag{19}$$

The second matrix of the right-hand side of Eq. (19) involves second-order derivatives of pressure with respect to density and internal energy. By plugging relation (19) in Eq. (7), the final expression of the left-side term of the adjoint system can be rewritten as

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right)^T \mathbf{v} = \left(\frac{d\mathbf{q}}{d\mathbf{u}}\right)^T \left(\frac{\partial \mathbf{R}}{\partial \mathbf{q}}\right)^T \mathbf{v} \tag{20}$$

A first solution to comply with exact Jacobians is to automatically linearize the flux function and the thermodynamic equations in a fully coupled manner. When this is done, a number of shortcomings may arise (e.g., differentiation errors), as AD technique is not still mature for simultaneously processing large parts of CFD solvers. Furthermore, this constrains the adjoint equations to be dependent on a specific thermodynamic model. An alternative strategy is then preferred in this research, consisting in the separate computation of the terms $(d\mathbf{q}/d\mathbf{u})^T$ and $(\partial \mathbf{R}/\partial \mathbf{q})^T$. More precisely, reverse AD is only applied to the numerical fluxes of the flow solver, i.e., to compute the second term $(\partial \mathbf{R}/\partial \mathbf{q})^T \mathbf{v}$, the transposed Jacobian $(d\mathbf{q}/d\mathbf{u})^T$ being analytically achievable once the analytical expressions of the EoS or of the fundamental relation are available. For bicubic relations, as the ones provided by the present LuT method, second-order derivatives are particularly easy to determine and can be added as outputs to the set of thermodynamic properties (17).

The opposite strategy has been conversely devised to obtain the left-hand side of adjoint equations, namely $(\partial J/\partial \mathbf{u})^T$. As a matter of fact, typical objective functions J for turbomachinery applications (e.g., efficiency) are frequently obtained by resorting to averaged quantities (e.g., enthalpies) upstream and downstream of the cascade. For real-gas flows, this usually leads to the solution of implicit algebraic nonlinear systems (for instance in mixed-out averaging procedures), thereby using a coupled differentiation notably relieves the effort to achieve $(\partial J/\partial \mathbf{u})^T$. However, such vector is computed once and stored, allowing to accomplish the reverse differentiation process in a fully customized way at a pre-processing stage.

4 Optimization Algorithm

The zFlow code and the adjoint solver have been implemented within a fully automated optimization strategy, whose block diagram is shown in Fig. 2. In order to implement a flexible and efficient optimization process, a complete package of specific numerical tools has been developed. The numerical suite embeds an unstructured grid generator based on an advancing front/Delaunay technique described in Ref. [35], the gradient calculator and a multidimensional simulation initializer. The latter tool is employed to properly initialize both the flow and adjoint calcula-tion by interpolating the flowfields computed at the preceding iter-ation on the new grid, thus accelerating the convergence of both solvers. Since the gradient is generally less smooth than the objec-tive function, it is normally postprocessed to prevent the blow-up of the design procedure. In particular implicit smoothing techni-ques, as in Ref. [36], and projection operations (e.g., along the surface normals) can be applied to the original gradient. In case of NURBS-based optimization, such a regularization can be usually avoided as parametric curves intrinsically preserve the geometry smoothness. However, in this study, the smoothing procedure is applied at (surface) grid points level to regularize the sensitivities before computing the gradient with respect to control points. This results in a low-pass filtering operation, thus further enhancing the stability of the design process.

The optimization is updated through a preconditioned steepest descent algorithm. The spatial displacements of the NURBS
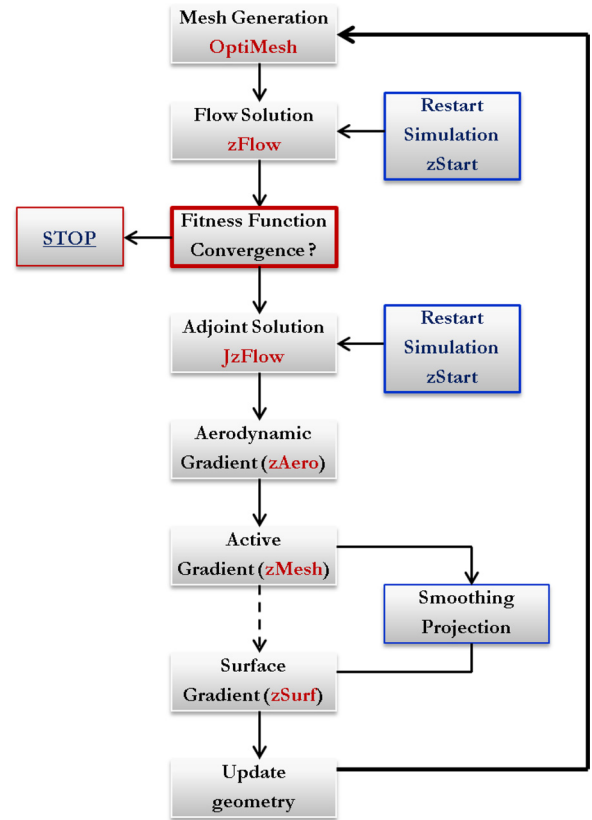

Fig. 2 Fully automated adjoint-based optimization algorithm. The aerodynamic gradient refers to the sensitivities with respect to all grid points, the active gradient is the vector of derivatives against surface grid points, whereas the surface gradient indicates the sensitivities with respect to NURBS control points. Red labels denote the names of the numerical solvers.

control points pertaining to a certain iteration are then computed as

$$\delta \mathbf{P}^{\mathrm{T}} = -\lambda \left(\widehat{\frac{\mathrm{d}J}{\mathrm{d}\mathbf{P}}} \right)^{\mathrm{T}} \qquad (21)$$

where λ is the step-length required to convert the (projected and smoothed) gradients into spatial displacements. Since the use of a line search method to determine λ at each iteration may result computationally demanding, see e.g. Ref. [37], this factor is herein assumed as a constant parameter. For 2D computations, the cost associated to grid generation is a small fraction of the time required by the flow and adjoint solvers to converge. As a result, in this work, the geometry is meshed at each optimization step instead of adapting the old grid to the new geometry configuration by means of the RBF-based deformation tool.

5 Shape Optimization of a Supersonic Turbine Cascade

The design algorithm extensively discussed in the previous section is now applied to the redesign of the profile of a converging–diverging turbine cascade operating under supersonic conditions. The blade geometry, shown in Fig. 3, has been originally designed by means of the method of characteristics (MOC) for the diverging part and constructing a highly smooth leading-edge portion for ensuring acceptable performance in case of relevant variations of the incidence angle.

The optimization process aims at maximizing the performances of the cascade operating under the thermodynamic conditions listed in Table 1.

Geometry Construction. Both pressure and suction side profile of the turbine blade are defined by a set of 11 control points, see Fig. 3. The exact position of such points is iteratively obtained by minimizing the distance between the reference profile and the profile generated by the NURBS curves. The procedure is repeated for both sides and converges in a few steps.

Simulation Setup. In this work inviscid governing equations are considered, the choice being appropriate for supersonic cas-cades in which shock waves represent the major loss mechanism. As the final purpose is the optimization of planar profiles, 2D blade-to-blade flow is considered. Inflow conditions are computed starting from total pressure and temperature, while nonreflecting
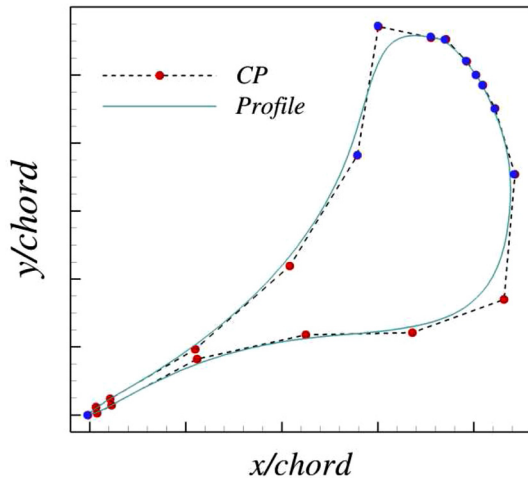
Fig. 3 Profile contour and control points distribution of the converging–diverging blade. Red circles indicate the design variables, while blue ones are kept fixed during optimization.

Table 1 Total upstream conditions, static outlet pressure, and inlet compressibility factor for shape optimization at reference condition

Fluid	P_{T}	T_{T}	p_{s}	z
MDM	8 bar	272 °C	1.0 bar	0.7

boundary conditions are applied at the domain outflow in order to capture the realistic flow pattern. High-resolution TVD spatial discretization scheme is used for both flow and adjoint equations. A preliminary analysis has been conducted to assess the independence of the solution from the grid spacing, leading to the final grid size of about 20,000 triangular elements. The working compound is the siloxane MDM, whose thermophysical properties are computed by using the Span–Wagner model via the LuT method on a thermodynamic grid of 10,000 elements, see Fig. 1.

Results and Discussion. The predicted flow-field of the baseline cascade configuration, shown in the left frame of Fig. 4 in terms of Mach number, highlights a strong shock generated downstream on the blade. Such shock wave is actually due to the coalescence of the compression fan produced on the curved rear suction side of the blade with the fishtail shock pattern generated at trailing edge. This is consistent with the design cascade outlet Mach number, equal to about 1.88.

The resulting effect is a nonuniform flow at the blade outlet section which is highly detrimental for the efficiency of the cascade and of the whole stage. Therefore, main aim of the shape optimization is the achievement of a more uniform flow in terms of pitchwise distribution of discharge Mach number at the outlet bound of the computational domain. Furthermore, achieving a more uniform flow necessarily leads to a weakening of the shock strength, and therefore to a considerable reduction of the total pressure losses. This is then expected as a further outcome of the design process. Moreover shock waves/boundary layer interactions are also expected to be significantly attenuated, reducing the risk of boundary layer separations and hence the wake viscous losses. Eventually, a more uniform flow leaving the stator contributes to improve the efficiency of the subsequent rotor and of the full turbine stage. The cost functional to be minimized is therefore written as

$$J(\mathbf{P}) = \left[\frac{\sum_{i=1}^{n_{\mathrm{ob}}} (M_i - M_{\mathrm{mix}})^2}{n_{\mathrm{ob}}} \right]^{\frac{1}{2}} \qquad (22)$$

where M is the Mach number at each boundary node, n_{ob} the number of outflow boundary nodes, and subscript mix indicates the pitchwise mixed-out averaged value.

As a preliminary step, the surface gradient of the objective function, computed at the first iteration of the design process, properly smoothed and normally projected, is analyzed. This initial operation, whose result is represented in the right Fig. 5, can be regarded as a very useful strategy to better understand how the design process will proceed and, possibly, to decide which, if any, design variables can be removed from the optimization. As expected, the gradient distribution shows that the value of the objective J is relevantly influenced by the blade region located downstream of the throat, while a minor contribution is due to the upstream part. As a matter of fact, in supersonic cascades the physical features of the shock waves stemming in the turning region are greatly affected by the shape of the semibladed channel. Therefore, in the present study, the control points drawn with blue circles in Fig. 3 are kept fixed during the design process (Fig. 6).
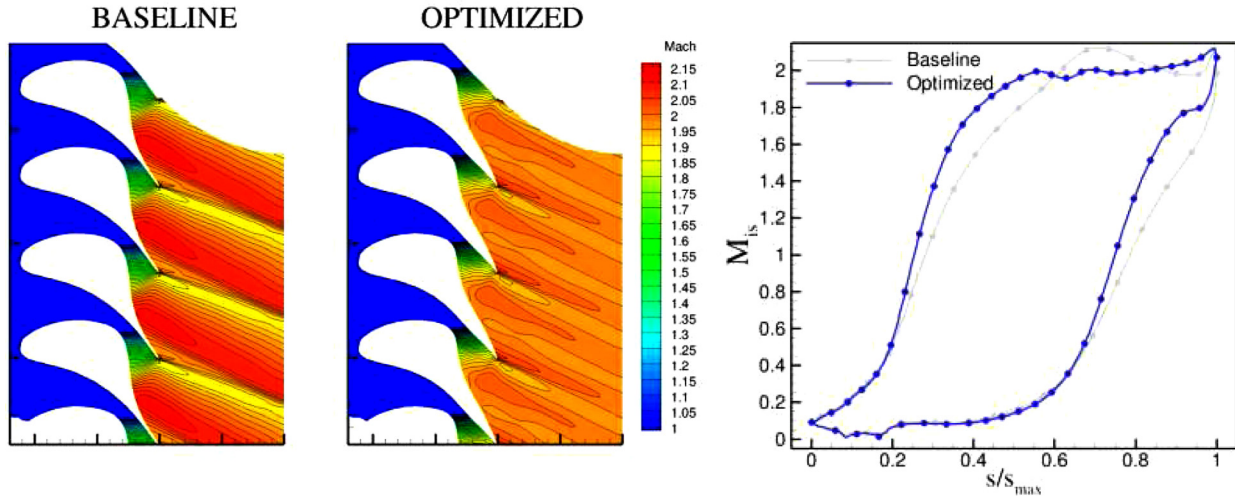
Fig. 4 Left: predicted Mach flow-field using the baseline and optimized cascade. Right: comparison of blade loading between the baseline and optimized cascade.

The convergence of the optimization procedure is achieved in about 15 iterations, see Fig. 7, requiring a total computational cost of about 30 min on a standard PC (Opteron dual-core worksta-tion). After some trial and error, the step-length λ and smoothing parameters ϵ were set to -0.05 and 30, respectively. Less smooth and slightly faster optimizations have been achieved by reducing the ϵ value, although the final solution is found to be very similar in all cases. The final results of the simulation are reported in Table 2.

The optimized blade shape, depicted in the right frame of left Fig. 4, features a more straight rear suction side with respect to the initial configuration, essentially in the semibladed region. The redesign of the rear passage geometry entails significant changes of the flow behavior inside both bladed and turning (postexpan-sion) regions. By looking at the results reported in Table 2, it should be first noted that the unconstrained optimization process involves a rather significant change of the throat size while enlarg-ing the blade outlet section. However, the resulting mass flow rate variation between the two configurations, in the order of 8%, can be promptly adjusted by acting on the blade span, with a minor impact on the aerodynamic losses.

The modifications imposed by the optimization algorithm to the diverging portion of the baseline blade led to an optimal bladed channel characterized by higher passage area ratios. As a consequence, the fluid stream is subject to larger accelerations on both pressure and suction sides of the diverging channel, reaching higher outlet Mach numbers. As a result, the flow is not further accelerated at the exit of the bladed region due to the straight designed rear suction profile. The reduction of the suction side curvature enables to avoid the over-acceleration present in the original flow pattern, which is the cause of the compression fan formation just before the trailing-edge of the baseline cascade. As well visible from the pressure distribution on the suction side of the baseline and optimized blading, reported in the right frame of Fig. 4, the concave rear shape of the baseline suction side induces an isentropic deceleration of the flow, while the conjunction of the fan characteristic waves promotes the onset of a strong shock in the mixing zone. The shock is instead completely removed in the optimized configuration and the efficiency decay remains just related to the fishtail shock detaching from the trailing-edge.

A more uniform flow is therefore achieved in the downstream region, leading to a relevant decrease of the total pressure loss coefficient, which reduces from about 12–3.6% for the optimal configuration.
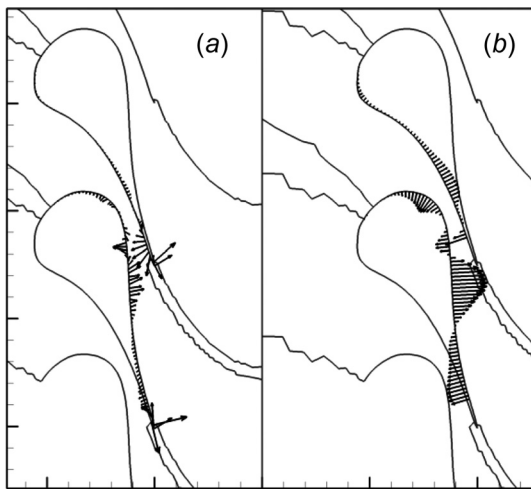
6 Assessment of Cascade Performance Through Turbulent Simulations

The proposed optimization procedure is based on an inviscid flow model. For supersonic turbines, in absence of significant flow
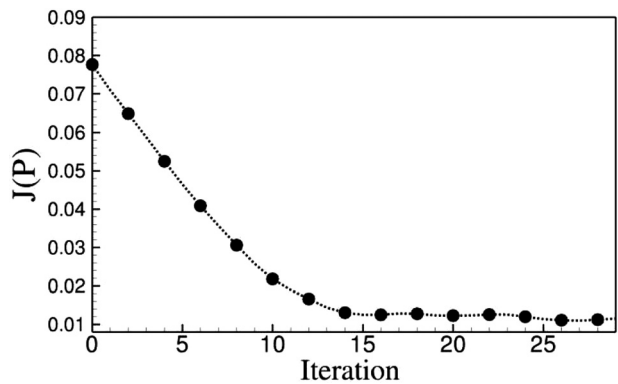

Fig. 5 Sketch of the nonsmoothed (*a*) and smoothed and normally projected (*b*) gradient vectors with respect to blade surface mesh nodes. The gradient components are properly scaled to be better visualized. Notice that flow uniformity, i.e., the cost functional, highly depends on the deflection imparted by the rear suction profile.

Fig. 6 Convergence history of the design process
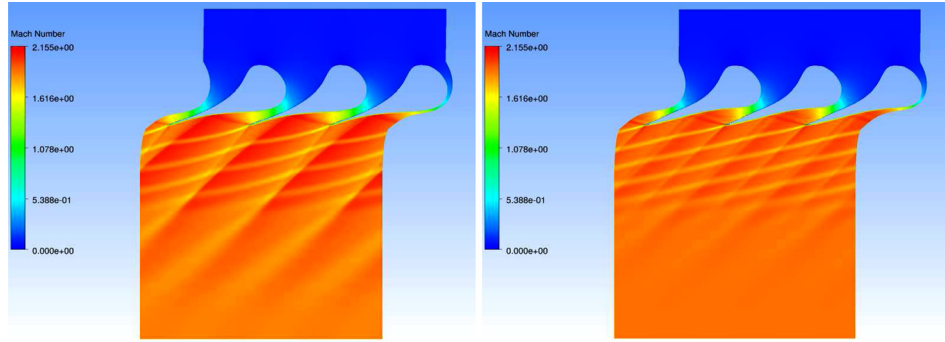
Fig. 7 Left: predicted Mach number distribution of the baseline cascade. Right: predicted Mach number distribution of the optimized cascade.

separations, this assumption leads to acceptable estimates of the cascade performance and significantly reduces the cost of the optimization process. Even if high-fidelity flow predictions require fully turbulent CFD simulations, their direct application in optimization runs for real-gas flows suffers from severe computational demands and then is still far from being considered as a standard approach. In case of real-gas flows, the intrinsic drawbacks of fully inviscid (e.g., limited fidelity) or fully turbulent (e.g., large overall cost) optimizations can be partly compensated using a pseudohybrid strategy, consisting in the exploitation of viscous simulations at the final stage of the design procedure to assess the outcomes of the inviscid optimization process. In this respect, turbulent simulations are generally able to provide very useful insights for further refinements of the optimized geometry.

6.1 Fully Turbulent CFD Model. In the present work, the performances of both baseline and optimized blade profiles are quantitatively assessed by means of fully turbulent CFD simulations performed with the ANSYS-CFX solver. As only blade-to-blade effects are of interest in this study, quasi-3D calculations are carried out considering a straight stream-tube around midspan.

A structured grid composed by 100,000 hexahedral cells (with two cells in spanwise direction) is used throughout the calculations. Turbulence effects are modeled using the k–ω SST model, ensuring wall y^+ well below the unity all along the blade; in full coherence with the inviscid simulations, siloxane MDM is treated as a real gas through a LuT interpolation method suitably defined for the CFX solver. High resolution numerical schemes and implicit time integration are used. Total conditions, flow angles, and turbulence quantities are assigned at the inlet, static pressure is given at the outlet. Spurious pressure wave reflections are avoided by placing the outflow domain at a distance of about three axial chords from the trailing-edge.

6.2 Results and Discussion. The flow fields computed by the turbulent flow model for the baseline and the optimized blade configurations are now discussed in detail. Most of the flow features observed in the inviscid simulations are retained in turbulent calculations, proving the acceptable fidelity of the inviscid model for this supersonic cascade. As visible in Fig. 7, where the Mach number distributions are presented, the strong shock generated downstream of the baseline cascade is highly weakened in the

Table 2 Results of shape optimization. Mass flow rates are reported in nondimensional form.

Blade	J(P)	Y (%)	m_{flow}	α_{flow} (deg)	$Mach_{mix}$
Baseline	0.078	11.78	1	74.95	1.95
Optimized	0.014	3.6	0.92	76.68	1.98

optimized configuration. This is a direct consequence of the full redesign of the blade rear suction side, which affects the shape of both the diverging channel and the semibladed region. As already observed for the inviscid simulation, the higher passage area ratio of the optimal cascade leads to a larger acceleration in the bladed regions. As a result, and thanks to the almost straight shape of the rear suction side, no significant over-speed is observed in the semibladed region of the optimal blade, as clearly visible in the isentropic Mach number distributions provided in the left frame of Fig. 8.

The Mach number and flow angle pitchwise distributions are therefore more uniform in the optimized configuration, as visible in the right frame of Fig. 8 for the Mach number, thus assessing the successful optimization strategy proposed in this work.

The final results of the simulations are reported in Table 3. Also in the fully turbulent simulations, the higher degree of uniformity provided by the optimized cascade leads to a considerable improvement of the aerodynamic performance. The decrease of total pressure loss coefficient is, indeed, in accordance with the reduction provided by the inviscid simulations. This suggests that the improvement of the turbine performance is mostly due to the weakening of the shocks achieved with the optimized configuration. Notice that turbulent calculations predict an 8% higher total pressure loss coefficient than that estimated from inviscid simulations for both the configurations. Most of these differences origin in the mixing process downstream of the cascade, which are now affected by the simultaneous presence of the wake interacting with the shock waves, thus inducing more dissipative effects.

To better highlight the impact of viscous phenomena on the aerodynamics of the present cascade, and hence to capture significant fluid-dynamic deviations stemming from the inviscid flow model, the isentropic Mach number distributions for inviscid and viscous calculations are compared in Fig. 9, while the detailed pressure fields in the supersonic regions downstream of the throat for the baseline and optimized configurations are provided in Figs. 10 and 11, respectively.

Left and right frames of Fig. 9, referred to the baseline and optimized configurations, respectively, indicate that the viscous and the inviscid profiles match all along the pressure side and for the large part of the suction side, namely up to the point of maximum over-speed. This means that the inviscid flow model is able to accurately capture the most critical design feature of the present cascade.

Downstream of the over-speed point, the inviscid predictions do not capture the sudden recompression predicted by the turbu-lent calculations. This compression wave is, in fact, the impinge-ment on the blade wall of the pressure side leg of the fishtail shock pattern generated at the adjacent blade trailing edge. Figures 10 and 11 clearly indicate that the trailing edge shocks in viscous calculation are by far stronger than those in inviscid ones. This is mainly due to two reasons: first, while a sharp trailing edge has been used for the inviscid calculations, a more realistic
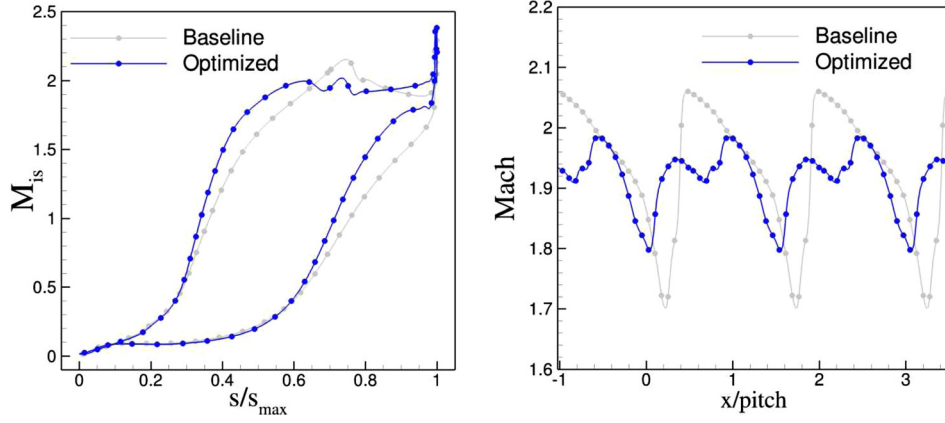
Fig. 8 Left: isentropic Mach number distribution along the blade surface of the baseline and optimized configurations. Right: predicted spanwise Mach number distribution at outflow boundary.

Table 3 Results of turbulent simulations of the baseline and optimized configurations. Mass flow rates are reported in non-dimensional form.

Blade	Y (%)	m_{flow}	α_{flow} (deg)	Mach$_{mix}$
Baseline	20.32	1	76.53	1.886
Optimized	12.24	0.94	77.65	1.911

round trailing edge has been considered for viscous ones; second, the displacement thickness of the blade boundary layers acts so to artificially reduce the cross section available for the fluid in viscous calculations. The combination of these two effects results in a severe flow rotation on both the sides of the trailing edge region, which subsequently leads to the generation of two strong trailing edge shocks.

In the inviscid calculation of the baseline cascade, the pressure side shock has been found to be negligible, and the downstream shock is formed just by the coalescence of isentropic compression waves formed on the rear blade suction side; in the viscous simulation, instead, the generation of the downstream shock is also promoted by the reflection of the pressure side shock on the blade wall. In the optimized configuration, the straight rear shape inhibits the shock generation, and hence the (weak) downstream shock is just given by the shock reflected on the blade wall. In both the cases, the shock reflection causes the local drop in isentropic Mach number observed in viscous simulations, which definitively entails a global reduction of the flow velocity along the rear suction side. This flow physics is fully consistent with the significant increase of total pressure losses observed in viscous simulations, as the shock reflection process causes a significant increase of boundary layer momentum thickness.

Despite the shock reflection not properly captured in the inviscid simulations, the comparison with the viscous model shows that the shock reflection does not alter the pressure trend in the rear part of the blade, which remains decelerating in the baseline cascade and almost uniform in the optimized one. This is the key-reason why the proposed inviscid optimization method results reliable for this configuration; the method is hence expected to be generally valid for other supersonic cascade configurations, pro-vided that flow separations are avoided.

7 Influence of Thermodynamic Model Accuracy on the Optimized Blade Shape

Most of the ORC turbines operate in thermodynamic regions characterized by strong real-gas effects. The importance of accurate fluid thermophysical descriptions in ORC turbomachinery has been underlined in a number of seminal works, see for instance Refs. [27], [38], and [39]. These works demonstrated that neglecting nonideal thermodynamic effects in the simulation process can lead to unsatisfactory predictions of the main flow features, such as Mach numbers and pressure coefficient distributions along the blade. Hence, automated design methods not considering real effects are expected to provide misleading design indications in thermodynamic conditions where strong nonideal effects take
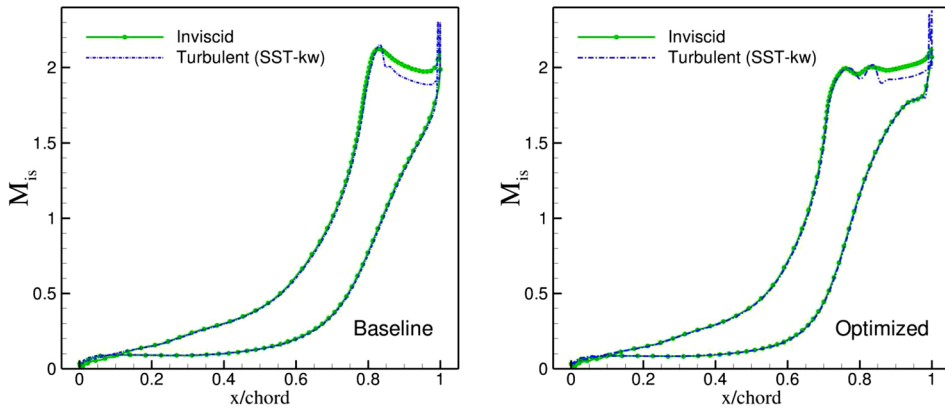

Fig. 9 Left: predicted isentropic Mach number of the baseline blade surface for inviscid and turbulent flows. Right: predicted isentropic Mach number of the optimized blade surface for inviscid and turbulent flows.
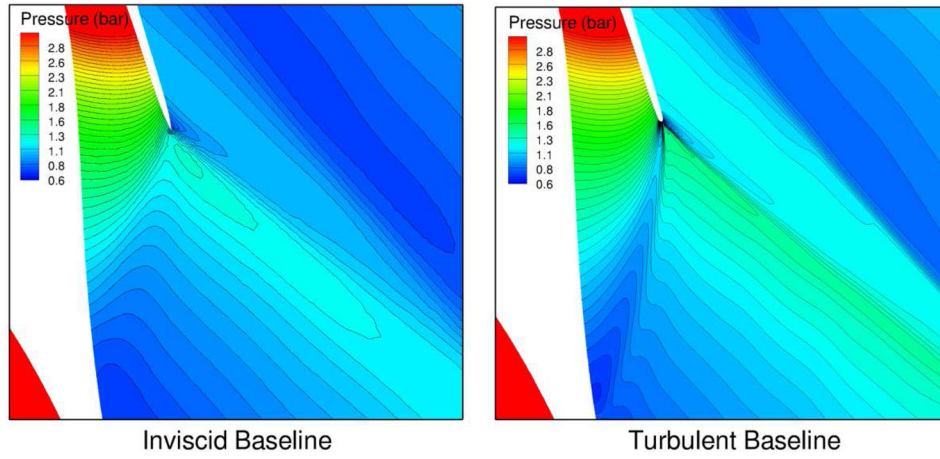
Fig. 10   Left: predicted inviscid fishtail wave pattern of the baseline configuration. Right: predicted turbulent fishtail shock pattern of the baseline configuration.
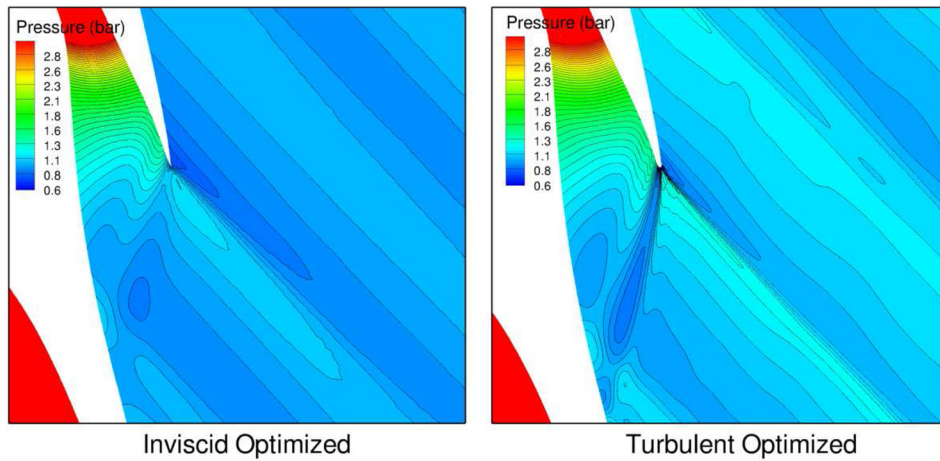

Fig. 11   Left: predicted inviscid fishtail wave pattern of the optimized configuration. Right: predicted turbulent fishtail shock pattern of the optimized configuration.
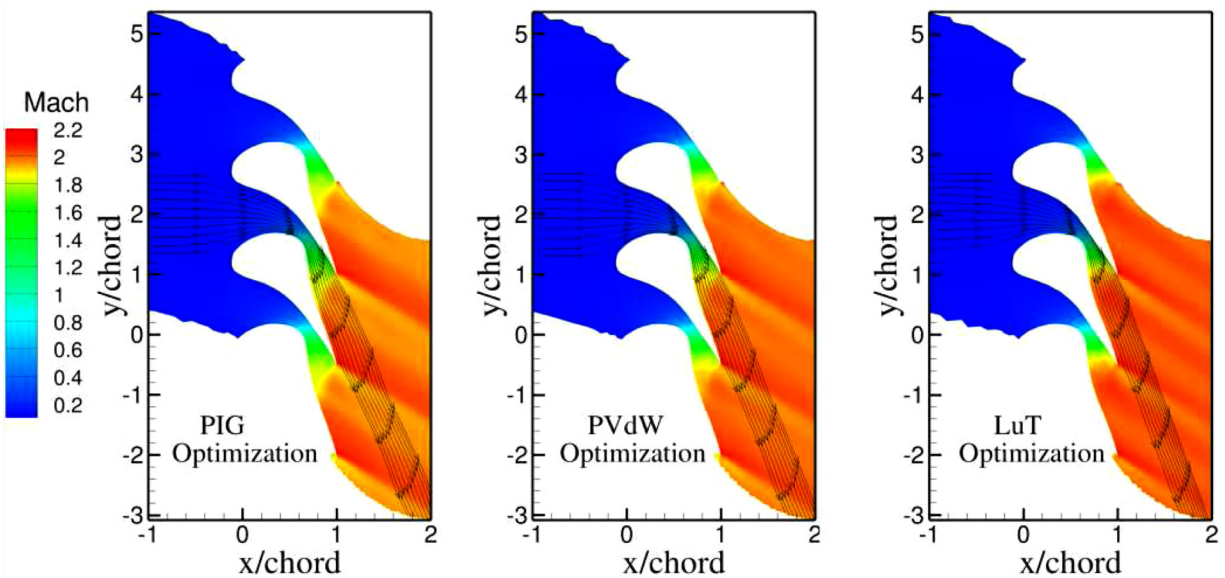

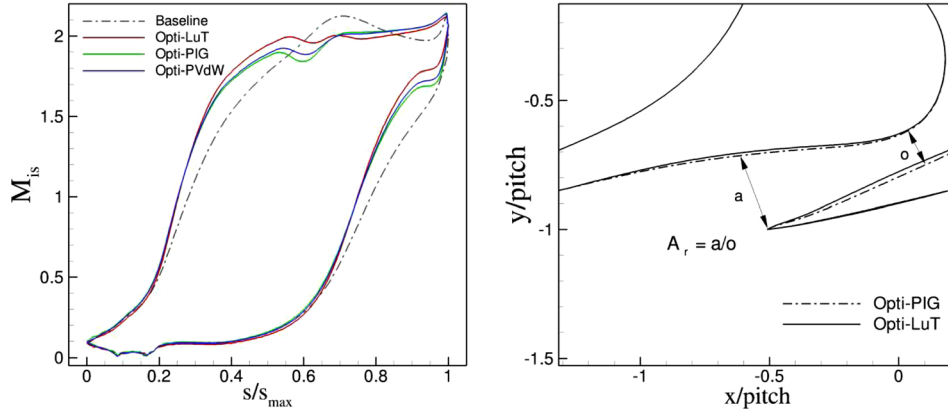**Fig. 12   Mach flowfield obtained through the three optimized blades (PIG, PVdW, and LuT)**

Fig. 13 Left: predicted isentropic Mach number distribution along the blade profile computed through the Span–Wagner EoS for the three optimized blades (PIG, PVdW, and LuT). Notice that curvilinear abscissa s/s_{max} is used to plot the trends in place of the nondimensional streamwise coordinate. Right: area ratio $(A_r = a/o)$ for the optimized PIG and LuT cascades. Notice that $A_{r_{opt}} = A_{r_{LuT}} > A_{r_{PIG}}$ and correspondingly results that $A_{r_{LuT}} > A_{r_{PVdW}}$ leading to the formation of an over-expanded freejet outside the bladed channel.
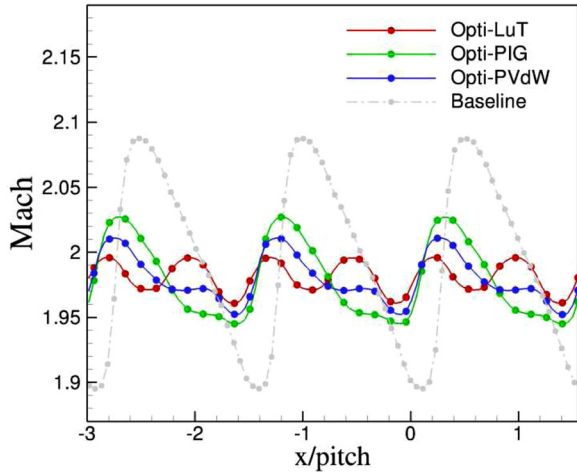

Fig. 14 Outlet Mach number pitchwise distribution obtained with the three optimized blades (PIG, PVdW, and LuT)

place. In the context of the present research, the use of ideal thermodynamic relations in both flow and adjoint solver may lead to gradient values different from those observed using more accurate EoS, and, therefore, to suboptimal geometries. For turbines working close to the critical point, the deviation can be even more amplified.

The nominal operating conditions of the supersonic turbine lead to an expansion process characterized, at least in the initial part, by relevant real-gas effects (inlet compressibility factor close to 0.7). Therefore, the present section investigates the advantages offered by a fully real-gas adjoint-based design methodology compared to approaches based on simplified equations of state.

Beside the LuT-based optimization, two further shape optimizations are carried out based, respectively, on the polytropic ideal gas law (PIG) and the polytropic Van der Waals model (PVdW).

Table 4 Predicted fitness function J values and total pressure loss coefficient $Y = P_{T0} - P_{T1}/P_{T0} - p_{s1}$ for the baseline and optimized cascades using the Span–Wagner model

	Baseline	Optimized (PIG)	Optimized (PVdW)	Optimized (LuT)
J	0.077	0.02	0.018	0.014
Y (%)	11.78	4.8	4.5	3.6

The same geometry parameterization, computational grid, and optimization parameters of the LuT test case are used. Similar convergence rates are achieved, with a 30% computational expense saved with respect to the LuT case by virtue of the lower cost associated to PIG and PVdW computations. The performance of the resulting optimized configurations is finally investigated by means of the multiparameter Span–Wagner model. The resulting Mach number flowfields are depicted in Fig. 12.

As highlighted by the isentropic Mach number distribution shown in the left frame of Fig. 13, the turbine nozzles designed using PIG and PVdW models are characterized by a lower Mach number at the outlet section. This means that the optimal area ratio $A_{r_{opt}}$ calculated through ideal gas laws constrains the real-gas stream to be under-expanded within the turbine blade passage, i.e., $\beta_{r_{id}} > \beta_{r_{re}}$, meaning that a larger outlet section would be necessary to achieve the same expansion ratio, namely $\beta_{r_{id}} = \beta_{r_{re}}$. As a matter of fact, during nozzle expansions, real vapor molecule, whose size depends on the molecular complexity and molar mass (generally higher for organic compounds characterized by several degrees of freedom), tends to highly attract each other (due to intermolecular attractive forces) and thus to reduce their pressure with major difficulty than ideal gases. The real flow leaving an ideal-designed nozzle is therefore over-pressurized, leading to the formation of an over-expanded freejet outside the channel for reaching the given ideal pressure.

A corresponding behavior is observed in the cascades optimized by using PIG/PVdW, characterized by $A_r < A_{r_{opt}}$; when the PIG-designed cascade is operated with the real-gas model, it behaves as in off-design conditions leading to strong postexpansion phenomena in the semibladed region with respect to the prescribed static backpressure. The effect of a greater acceleration on the rear suction profile of the PIG and PVdW optimal configurations also contributes to increase the Mach number just upstream of the trailing-edge, strengthening the intensity of the fishtail shock detaching from it. This in turn strongly affects the flow uni-formity at the outlet domain, shown in Fig. 14, and the resulting performance of the cascades, see Table 4.

The use of a fully real-gas adjoint method for shape optimization problems provides the best adaptation of the designed blade to the cascade pressure ratio, thus avoiding the onset of gas dynamic phenomena that would deteriorate turbine performance, especially for transonic and supersonic configurations.

8 Conclusions

This research work has proposed a novel adjoint-based shape optimization approach for real-gas flow applications. A key

novelty of the present approach is represented by the extension of the adjoint method to flows governed by arbitrarily complex thermodynamic models through the implementation of an efficient LuT methodology. Valuable features of this latter methodology are, among others, the possibility of using EoS of any kind and the capability of analytically providing secondary thermodynamic properties, leading to a great simplification of the differentiation process of the flow solver.

Around the adjoint method, an unconstrained gradient-based optimization has been conceived. NURBS parametric curves have been implemented to handle large geometry deformations while simultaneously preserving surface smoothness. A robust preconditioned steepest descent has been selected as optimizer. The design methodology has been applied to the redesign of an existing supersonic turbine cascade expanding siloxane vapors characterized by high real-gas effects. Considerable improvement of the performance, in terms of flow uniformity at the cascade outlet and, especially, of reduction of total pressure losses, has been achieved with very limited computational costs. These outcomes have been further assessed by means of high-fidelity simulations based on a fully turbulent CFD model.

Finally, the potential of the method has been further demonstrated by comparison, on the same turbine test case, against simplified approaches do not accounting for real-gas effects. Results revealed that fully real-gas adjoint methods are inherently capable to provide significant efficiency gain with respect to ideal methods while maintaining a comparatively similar computational effort. As a consequence, they can be considered the most promising and effective methodologies for shape optimization of ORC turbines.

Coherently with the attained indications, future research will be directed toward improving the current approach by extension to viscous flows and constrained optimization.

Nomenclature

a = nozzle outlet section
A_r = area ratio
M = Mach number
m_{flow} = mass flow rate
n = number of cells
J = fitness function
$\bar{J}$ = fitness function mean value
o = nozzle throat section
p = static pressure
P = control points/pressure
P_r = pressure ratio
R = flow equations
s = curvilinear abscissa
T = total temperature
u = state variables, NURBS knot sequence
v = adjoint variables
X = grid variables
Y = total pressure loss
Y = total pressure loss coefficient
α = independent variables of the optimization problem
α_{flow} = outflow angle
β = coefficients of the linear polynomial of the RBF interpolation, nozzle expansion ratio
λ = step-length
ϵ = smoothing parameter
ω = weights of the parametric curves

Acronyms

EA = evolutionary algorithm
EoS = equation of state
GA = genetic algorithm
ORC = organic rankine cycle
LuT = look-up table
PIG = polytropic ideal gas

PVdW = polytropic Van der Waals
RBF = radial basis function
SW = Span–Wagner

Subscripts

geo = geometrical
id = ideal
is = isentropic
mix = mixed-out averaged quantities
ob = outflow boundary nodes
phy = physical
r = ratio
re = real
s = static
T = total

References

[1] Pini, M., Persico, G., Casati, E., and Dossena, V., 2013, "Preliminary Design of a Centrifugal Turbine for ORC Applications," ASME J. Eng. Gas Turbines Power, **135**(4), p. 042312.
[2] Petrovic, M. V., Dulikravich, G. S., and Martin, T. J., 2001, "Optimization of Multistage Turbines Using a Through-Flow Code," J. Power Energy, **215**(5), pp. 559–569.
[3] Larocca, F., 2008, "Multiple Objective Optimization and Inverse Design of Axial Turbomachinery Blade," J. Propul. Power, **24**(5), pp. 1093–1099.
[4] Pasquale, D., Persico, G., and Rebay, S., 2014, "Optimization of Turbomachinery Flow Surfaces Applying a CFD-Based Troughflow Method," ASME J. Turbomach., **136**(3), p. 031013.
[5] Pierret, S., and Van Den Braembussche, R., 1999, "Turbomachinery Blade Design Using a Navier–Stokes Solver and Artificial Neural Network," ASME J. Turbomach., **121**(2), pp. 326–332.
[6] Rai, M., 2000, "Aerodynamic Design Using Neural Networks," AIAA J., **38**(1), pp. 173–182.
[7] Leonard, O., and Van Den Braembussche, R., 1992, "Design Method for Subsonic and Transonic Cascade With Prescribed Mach Number Distribution," ASME J. Turbomach., **114**(3), pp. 553–560.
[8] Demeulenaere, A., Leonard, O., and Van Den Braembussche, R., 1997, "A Two-Dimensional Navier–Stokes Inverse Solver for Compressor and Turbine Blade Design," J. Power Energy, **211**(4), pp. 299–307.
[9] Coello, C., 2000, "An Updated Survey of GA-Based Multiobjective Optimization Techniques," ACM Comput. Surv., **32**(2), pp. 109–143.
[10] Verstraete, T., Alsalihi, Z., and Van Den Braembussche, R., 2010, "Multidisciplinary Optimization of a Radial Compressor for Microgas Turbine Applications," ASME J. Turbomach., **132**(2), p. 031004.
[11] Pasquale, D., Ghidoni, A., and Rebay, S., 2013, "Shape Optimization of an Organic Rankine Cycle Radial Turbine Nozzle," ASME J. Eng. Gas Turbines Power, **135**(4), p. 042308.
[12] Lee, S. Y., and Kim, K. Y., 2000, "Design Optimization of Axial Flow Compressor Blades With Three-Dimensional Navier–Stokes Solver," KSME Int. J., **14**(9), pp. 1005–1012.
[13] Oyama, A., 2004, "Transonic Axial-Flow Blade Optimization: Evolutionary Algorithms/Three-Dimensional Navier–Stokes Solver," J. Propul. Power, **20**(4), pp. 612–619.
[14] Chen, N., Zhang, H., Huang, W., and Xu, Y., 2005, "Study on Aerodynamic Design Optimization of Turbomachinery Blades," J. Therm. Sci., **14**(4), pp. 298–304.
[15] Pierret, S., Coelho, R., and Kato, H., 2006, "Multidisciplinary and Multiple Operating Points Shape Optimization of Three-Dimensional Compressor Blades," Struct. Multidiscip. Optim., **33**(1), pp. 61–70.
[16] Peter, J., and Dwight, R., 2010, "Numerical Sensitivity Analysis for Aerodynamic Optimization: A Survey of Approaches," Comput. Fluids, **39**(3), pp. 373–391.
[17] Quoilin, S., Broek, M. V. D., Declaye, S., Dewallef, P., and Lemort, V., 2013, "Techno-Economic Survey of Organic Rankine Cycle (ORC) Systems," Renewable Sustainable Energy Rev., **22**(0), pp. 168–186.
[18] Lang, W., Almbauer, R., and Colonna, P., 2013, "Assessment of Waste Heat Recovery for a Heavy-Duty Truck Engine Using an ORC Turbogenerator," ASME J. Eng. Gas Turbines Power, **135**(4), p. 042313.
[19] Pasquale, D., Ghidoni, A., and Rebay, S., 2013, "Shape Optimization of an Organic Rankine Cycle Radial Turbine Nozzle," ASME J. Eng. Gas Turbines Power, **135**(4), p. 042305.
[20] Cinnella, P., and Congedo, P., 2006, "GA-Hardness of Dense-Gas Flow Optimization Problems," Applied Simulation and Modelling, Rhodes, Greece, June 26–28.
[21] Giles, M., and Pierce, N., 2000, "An Introduction to the Adjoint Approach to Design," Oxford Computing Laboratory, Oxford, UK.
[22] Giles, M., Ghate, D., and Duta, M., 2005, "Using Automatic Differentiation for Adjoint CFD Code Development," Oxford Computing Laboratory, Oxford, UK.
[23] Carpentieri, G., Koren, B., and van Tooren, M., 2007, "Adjoint-Based Aerodynamic Shape Optimization on Unstructured Meshes," J. Comput. Phys. **224**(1), pp. 267–287.

[24] Hoschek, J., Lasser, D., and Schumaker, L., 1996, *Fundamentals of Computer Aided Geometric Design*, A.K. Peters, Natick, MA.

[25] Farin, G., 2002, *Curves and Surfaces for CAGD: A Practical Guide*, 5th ed., Academic Press, Waltham, MA.

[26] de Boer, A., van der Schoot, M., and Bijl, H., 2007, "Mesh Deformation Based on Radial Basis Function Interpolation," Comput. Struct., **85**(11), pp. 784–795.

[27] Colonna, P., and Rebay, S., 2004, "Numerical Simulation of Dense Gas Flows on Unstructured Grids With an Implicit High Resolution Upwind Euler Solver," Int. J. Numer. Methods Fluids, **46**(7), pp. 735–765.

[28] Harinck, J., Colonna, P., Guardone, A., and Rebay, S., 2010, "Influence of Thermodynamic Models in 2D Flow Simulations of Turboexpanders," ASME J. Turbomach., **132**(1), p. 011001.

[29] Pecnik, R., Rinaldi, E., and Colonna, P., 2012, "Computational Fluid Dynamics of a Radial Compressor Operating With Supercritical CO_2," ASME J. Eng. Gas Turbines Power, **134**(12), p. 122301.

[30] Boger, A., 2001, "Efficient Method for Calculating Wall Proximity," AIAA J., **39**(12), pp. 2404–2406.

[31] Pini, M., Spinelli, A., Persico, G., and Rebay, S., 2014, "Consistent Look-Up Table Interpolation Method for Real-Gas Flow Simulations," Comput. Fluids (in press).

[32] Saad, Y., and Schultz, M., 1986, "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," SIAM J. Sci. Stat. Comput., **7**(3), pp. 856–869.

[33] Hascoet, L., and Pascual, V., 2004, *Tapenade 2.1 User's Guide*, Unité de recherche INRIA Sophia Antipolis, Cedex, France.

[34] Rinaldi, E., Pecnik, R., and Colonna, P., 2014, "Implicit Schemes for Compressible Flows of Dense Gases," J. Comput. Phys. (in press).

[35] Ghidoni, A., Pelizzari, E., Rebay, S., and Selmin, V., 2006, "3D Anisotropic Unstructured Grid Generation," Int. J. Numer. Methods Fluids, **51**(9–10), pp. 1097–1115.

[36] Jameson, A., and Kim, S., 2003, "Reduction of the Adjoint Gradient Formula in the Continuous Limit," AIAA Paper No. 2003-0040.

[37] Kim, S., Hosseini, K., Leoviriyakit, K., and Jameson, A., 2005, "Enhancement of the Adjoint Design Methods Via Optimization of Adjoint Parameters," AIAA Paper No. 2005-0448.

[38] Colonna, P., Harinck, J., Rebay, S., and Guardone, A., 2008, "Real-Gas Effects in Organic Rankine Cycle Turbine Nozzles," J. Propul. Power, **24**(2), pp. 282–294.

[39] Rebay, S., Colonna, P., Pasquale, D., and Ghidoni, A., 2009, "Simulation of the Turbulent Dense Gas Flow Through The Nozzle of an Organic Rankine Cycle Turbine," 8th European Conference on Turbomachinery, Fluid Dynamics and Thermodynamics, Graz, Austria, Mar. 23–27, pp. 1137–1148.