Scientia Iranica B (2014) 21(6), 1981–1992

Sharif University of Technology
Scientia Iranica
Transactions B: Mechanical Engineering
www.scientiairanica.com

Research Note

Application of the ball-spine algorithm to design an axial-flow compressor blade

A. Madadi[a], M.J. Kermani[a,*] and M. Nili-Ahmadabadi[b]

a. *Department of Mechanical Engineering, Amirkabir University of Technology, Tehran, P.O. Box 15875-4413, Iran.*
b. *Department of Mechanical Engineering, Isfahan University of Technology, Isfahan, P.O. Box 84156-83111, Iran.*

Received 22 October 2012; received in revised form 26 December 2013; accepted 26 February 2014

KEYWORDS
Ball-spine algorithm;
Inverse design;
Compressor blade profile;
Target pressure distribution;
Computed pressure distribution.

Abstract. In this paper, a novel inverse design algorithm, called a ball spine algorithm, is developed to design the blade to blade flow passage of axial-flow compressors. The algorithm considers the blade surfaces as a set of virtual balls that move freely along the specified directions, called 'spines'. To start the solution, an initial guess for the blade geometry is required. Then, the blade-to-blade flow field is solved using an in-house inviscid solver. Comparing the Computed Pressure Distribution (CPD) with the Target Pressure Distribution (TPD) over the blade surfaces gives guidelines in differential movements for the balls to obtain a modified geometry. For the modified geometry, new grids are automatically generated by an algebraic-elliptic grid-generator. The sequence is repeated until the target pressure is met. An error parameter, ΔPD, indicating the difference between CPD and TPD along the suction and pressure surfaces, is computed, while the blade geometry evolves toward the target geometry. For the initial guess, ΔPD is obtained as 13700 Pa, while, after 100 generations, ΔPD is reduced to 222 Pa.

1. Introduction

The design of hardware involving fluid flow or heat transfer, such as intakes, manifolds, compressors and turbine blades, etc., is defined as the shape determination of solid elements, so that the flow or heat transfer rate is optimized in some senses. Often, both Computational Fluid Dynamics (CFD) and design algorithms are involved in determining an optimal shape. The computational costs in design techniques are usually a challenge, so, an appropriate algorithm for rapid shape determination is always of interest to designers in the field.

In general, the design problems are categorized in two groups; optimization and inverse design. In optimization problems, an objective function, which could be a compound of various targets, is defined, and, using optimization algorithms, the optimum values for the design parameters are determined. Li et al. [1] developed a blading design optimization system using an aeromechanical approach and harmonic perturbation method. They implemented this method on a NASA 67 rotor and improved the efficiency of the rotor by 0.4% considering stress limitations. Verstraete et al. [2] presented a multidisciplinary optimization and applied it to the design of a small radial compressor impeller. In this method, a genetic algorithm and artificial neural network have been used to find geometry with maximum efficiency, regarding a maximum stress limitation in the blades.

The other type of design problem is Surface Shape Design (SSD). Surface Shape Design (SSD) in fluid flow

*. *Corresponding author. Tel.: +98 21 64543421; Fax: +98 21 66419736*
E-mail addresses: Ali.madadi@gmail.com (A. Madadi); mkermani@aut.ac.ir (M.J. Kermani); m.nili@cc.iut.ac.ir (M. Nili-Ahmadabadi)

problems usually involves finding a shape associated with a prescribed distribution of surface pressure or velocity. It should be noted that the solution of an SSD problem is not generally an optimum solution in a mathematical sense. It just means that the solution satisfies a Target Pressure Distribution which resembles a nearly optimum performance [3].

There are basically two different algorithms for solving SSD problems: decoupled (iterative) and coupled (direct or non-iterative) techniques. In the coupled solution approach, an alternative formulation of the problem is used in which the surface coordinates appear (explicitly or implicitly) as dependent variables. In other words, the coupled methods tend to find the unknown part of the boundary values and the flow field unknowns simultaneously in a (theoretically) single-pass or one-shot approach [4]. The governing equations of coupled methods are more complicated than the well-known fluid dynamics equations; hence, these methods are limited to simple flow regimes. In addition, conventional flow field solvers could not be used.

The iterative shape design approach relies on repeated shape modifications, such that each iteration consists of a flow field solution followed by a geometry generation scheme. In other words, a series of sequential problems are solved in which the surface shape is modified (evolved) in each geometry generation so that the desired TPD is eventually achieved [5]. In iterative methods, the governing equations are similar to flow field equations and conventional solvers could be used as a black-box. Hence, iterative methods are applicable for complicated flow regimes.

Iterative methods, such as optimization techniques, have been, by far, the most widely used in solving practical SSD problems. The traditional iterative methods used for SSD problems are often based on trial and error or optimization-search algorithms. The trial and error search process is very time-consuming and computationally expensive, and, hence, needs designer experience to reduce computation costs. Optimization methods [6,7] are commonly used to automate the new generation of geometry in each iteration cycle. In such methods, an objective function (e.g., the difference between the computed surface pressure and the target surface pressure, ΔPD [8]) is minimized, subject to flow constraints which have to be satisfied. Although the iterative methods are general and powerful, they are often computationally costly and mathematically complex [9]. These methods can utilize the method of flow field analysis as a black-box.

Other iterative methods presented so far use physical instead of mathematical algorithms to automate the geometry modification in each iteration cycle. The physical-based methods are easier and quicker than the mathematical (or optimization based) iterative methods. One of these physical algorithms is governed by a transpiration model in which one can assume that the wall is porous. Hence, the mass can be fictitiously injected through the wall in such a way that the new wall satisfies the no flow through the wall boundary condition [10]. This method is aimed to remove nonzero normal velocity on the boundary. A geometry update is determined by applying either the transpiration model based on mass flux conservation [11-15] or the streamline model based on alignments [16]. An alternative algorithm is based on the residual-correction approach. In this method, the key problem is to relate the computed difference between actual pressure distribution on the current estimate of the geometry and the TPD (the residual) to the required changes in the geometry. In this method, the art in the development of a residual-correction method is to find an optimum state between the computational effort (for determining the required geometry correction) and the number of iterations needed to obtain a converged solution. This geometry correction may be estimated by means of a simple correction role, making use of relations between geometry changes and pressure differences known from linearized flow theory. The residual-correction decoupled solution methods try to utilize the analysis methods as a black-box.

Barger and Brooks [17] presented a streamline curvature method in which they considered the possibility of relating a local change in surface curvature to a change in local velocity. Since then, a large number of methods have been developed following that concept [18-26].

Nili et al. presented an iterative inverse design method for internal flows called the Flexible String Algorithm (FSA). They considered the duct wall as a flexible string, frequently deformed under the difference between TPD and CPD, ΔPD = TPD$-$CPD. They developed this method for non-viscous compressible [27,28] and viscous incompressible internal flow regimes [29].

Recently, Nili et al. presented a novel inverse design method called the Ball-Spine algorithm (BS algorithm). They developed this method for quasi-3D design of the meridional plane of centrifugal compressors [30].

In this paper, the BS algorithm is implemented for the design of a Double Circular Arc (DCA) profile. To do so, corrected geometry for both suction and pressure surfaces have been consecutively generated until the target pressure distributions on both surfaces are achieved. For the flow field solution, a recently developed in-house code, based on the Roe scheme, is used. After each geometry evolution step, the domain grids have been regenerated using an in-house algebraic-elliptic grid generation code. As an initial guess for the geometry, a straight duct (flat plate

airfoil) has been used, which gives the error parameter, $\Delta PD = 13700$ Pa, where ΔPD is the average absolute value of the difference between computed and target pressure distributions, i.e. CPD and TPD, along the blade surfaces. After 100 geometry corrections (evolvement) ΔPD is reduced to 222 Pa. To ensure the uniqueness of the solution, the procedure is repeated using a different initial guess for the geometry, and it is observed that the solution has converged to the same target geometry. To present the effectiveness of the design algorithm, two design problems have also been performed. First, a pressure distribution with a higher loading has been considered as the target pressure distribution, and the corresponding airfoil-shape has been determined. Then, a target pressure distribution with a lower adverse pressure gradient on the suction surface has been used, and the corresponding geometry which satisfies the target pressure has been obtained. The design method has also been applied to a transonic test case and the method is shown to be reliable. As a sensitivity analysis, the effect of the geometry correction coefficient, C, on the stability and convergence rate of the method has also been studied and an optimum value for this coefficient is obtained.

2. The Ball-Spine (BS) algorithm

In the present work, the ball-spine algorithm is applied to the design of a 2D airfoil. At first, the basis of the ball-spine algorithm is detailed. To do so, a two dimensional internal flow through a duct is considered as an example. Here, a target pressure distribution (or profile) along the duct is specified, and the corresponding geometry is determined through the BS algorithm. To start the solution, an initial guess for the duct cross sectional profile is required to be made.

Here, the duct walls are considered as a set of virtual balls (see Figure 1) that are only allowed to freely move along specified directions called spines. For the guessed duct profile (or the duct geometry that does not yet correspond to the target pressure), flow motion through the duct applies a pressure to the wall of the duct, here called computed pressure. For a fluid tube (or say a control volume around the fluid), this pressure is applied at the outer side of the wall, as shown in Figure 2. This pressure, in general, is different from the required target pressure. Hence, the flexible wall (i.e. the balls) is assumed to deform in such a way as to reach a shape satisfying the target pressure. That is, the force due to the difference between the target and computed pressure at each point on the wall is applied to each virtual ball and causes them to move. As the target shape is obtained, this pressure difference vanishes and the balls will stop moving.

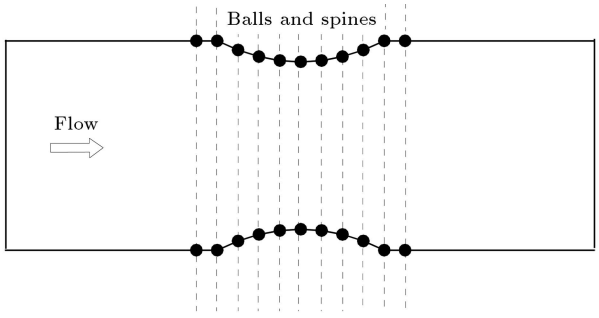

Figure 1. Simulation of a 2-D duct with balls and spine directions.

To derive the kinematic relations of a flexible wall, a uniform mass distribution along the wall is assumed. The free body diagram of the ball is shown in Figure 3.

The net force applied to the ball along the spine direction is computed as:

$$F_{\mathrm{Sp}} = \Delta P.A.\cos\theta, \tag{1}$$

where the subscript Sp indicates the spine direction and:

$$\Delta P = P_{\mathrm{Target}} - P_{\mathrm{Comp.}}, \tag{2}$$

and A is the projected area of each element and θ is the angle between the force vector and spine direction. If a ball moves along the spine direction through a specified time step (Δt), the corresponding displacement is

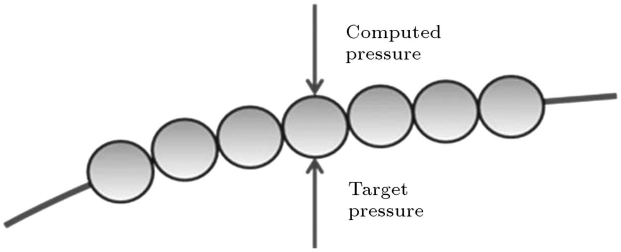

Figure 2. Applying the target and computed pressures on a sample ball.

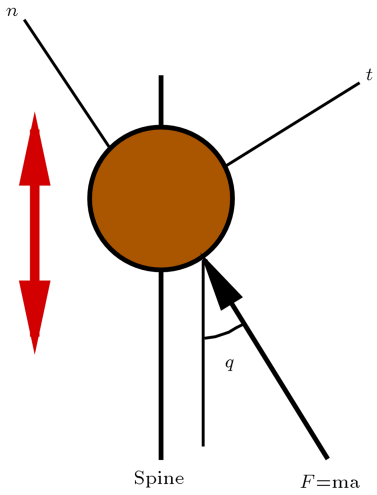

Figure 3. Free body diagram of a sample ball on the duct wall.

computed from the following dynamic relations:

$$\Delta s = \frac{1}{2}a(\Delta t)^2, \qquad a = \frac{F_{\mathrm{Sp}}}{m}, \tag{3}$$

where m and Δs are the mass and displacement of the virtual ball, respectively. Substituting Eqs. (1) and (2) into Eq. (3) yields:

$$\Delta s = \frac{1}{2}\frac{A}{m}(\Delta t)^2 \left(P_{\mathrm{Target}} - P_{\mathrm{Comp.}}\right)\cos\theta, \tag{4}$$

or:

$$\Delta s = C.\Delta P.\cos\theta, \quad \Delta P = \left(P_{\mathrm{Target}} - P_{\mathrm{Comp.}}\right), \tag{5}$$

where $C = A.(\Delta t)^2/2m$ is a constant called the "Geometry Correction Coefficient", with a typical value within the range of 0.0001-0.0005 ($\mathrm{m}^2.\mathrm{s}^2/\mathrm{kg}$). Considering a large value for parameter C causes the balls displacements to increase and the convergence rate to be improved. On the other hand, if parameter C exceeds a limit, the solution becomes unstable. Although a small value of C causes the design procedure to be stable, the convergence rate decreases. Hence, an optimum value for the geometry correction coefficient, which depends on pressure gradients, flow regimes, geometry etc., is usually determined by a trial and error process.

Based on the value of for each ball (or better to say, for each computational node along the wall), the new geometry is determined. Then, grids for the revised geometry are generated, and the flow field is solved using a flow solver. The procedure is continued until the computed pressure matches the target pressure. The flowchart of the design algorithm is shown in Figure 4. Because, in the flow field solver, back pressure is imposed at the outlet (which is fixed), and the first point on the walls must be fixed, the target and computed surface pressures are gauged relative to the leading edge pressure, i.e:

$$P_{\mathrm{rel}} = P - P_{\mathrm{LE}}. \tag{6}$$

For the geometry correction, the relative pressure, P_{rel}, is used in Eq. (5).

It should be noted that for supersonic flows, the change of pressure with change of area differs from the subsonic flow regime. Hence, Eq. (5) is rewritten as:

$$\begin{cases} \Delta s = C.\Delta P.\cos\theta & \text{if Mach} < 1 \\ \Delta s = -C.\Delta P.\cos\theta & \text{if Mach} > 1 \end{cases} \tag{7}$$

2.1. The modified wall geometry

Here, the spine directions are selected along the vertical lines, i.e. with the same x-coordinates as illustrated earlier in Figure 1. In other words, the x-coordinates of the balls on the top and bottom walls of Figure 1 remain the same through the shape modification procedure. The modified wall geometry, i.e. the new position of each ball, is calculated as follows:

$$x_{\mathrm{new}} = x_{\mathrm{old}}, \tag{8}$$

and:

$$y_{\mathrm{new}} = y_{\mathrm{old}} + \Delta s. \tag{9}$$

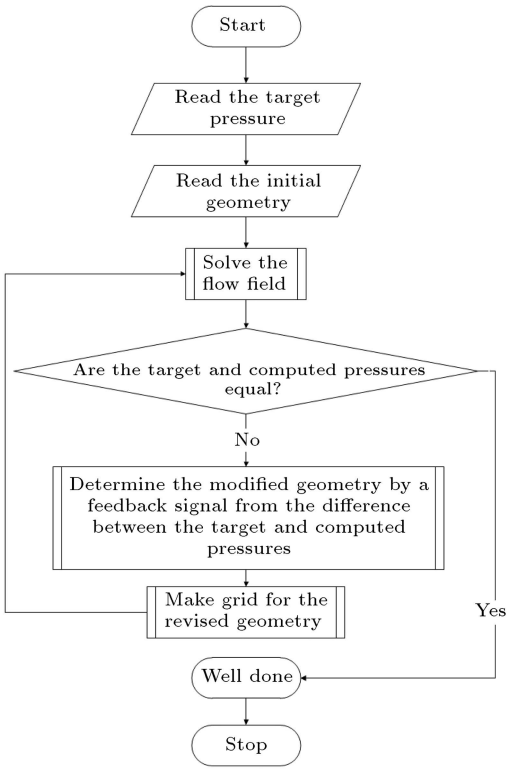

Figure 4. The design flowchart.

2.2. The wall smoothness

In this process, a wall is considered as a set of separated balls. During the design/modification procedure, the curvature of this wall may result in discontinuous shapes in adjacent nodes (balls), as shown in Figure 5. To smooth out the curvature of the wall, a filtration scheme is applied. This process is performed after each geometry correction step. The filtration scheme is formulated as follows:

$$y(i,j) = \frac{y(i-1,j) + f.y(i,j) + y(i+1,j)}{f+2}. \tag{10}$$

Here, f is the filtration coefficient. Large values for f correspond to a low degree of filtration and small values for f result in a greater degree of filtration. In the present work, f is set to 4, i.e:

$$y(i,j) = \frac{y(i-1,j) + 4y(i,j) + y(i+1,j)}{6}. \tag{11}$$

In Figure 5, the filtered geometry is plotted using a dashed line. A higher order of filtration decreases the convergence rate, but improves the stability of the design method.

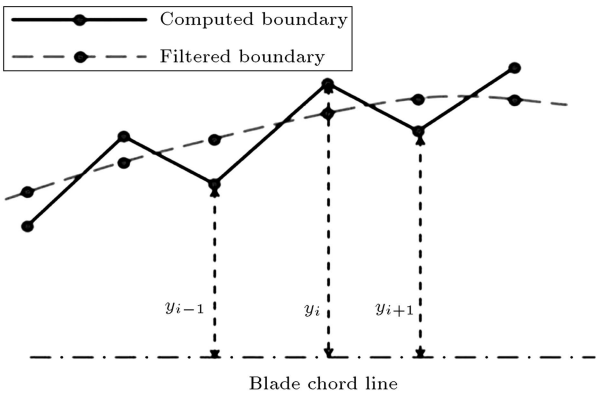

Figure 5. Wall boundary before and after filtration.

2.3. The compressor blades design procedure

In the design procedure of compressor blades, a target pressure distribution is usually provided for each airfoil surface, i.e. for the suction and pressure sides. To enforce the target pressure distributions along the suction and pressure surfaces, geometry correction is done by applying pressure distributions alternatively to the suction and pressure surfaces, in such a way that only one boundary is corrected at each step and the other is kept fixed. Finally, when both pressure distributions match, the geometry is determined.

2.4. Inviscid/viscous design procedures

The ball-spine algorithm can incorporate either an in-house code or a commercial flow solver for the iterative procedure of shape determination. As the flow solver is like a black-box to the BS algorithm, either inviscid or viscous flow solvers can be adopted.

The prescribed BS algorithm uses pressure distribution for geometry correction, by which, the separating regions within the flow fields, usually appearing in regions of large adverse pressure gradients, are avoided. Hence, the trends of pressure distribution determination for inviscid and viscous flows over the surfaces are similar. As a result, first, a designer can use an inviscid flow solver to obtain a primary solution quickly. Then, using a viscous flow solver, the accuracy of the solution can be either checked or improved. If the designed geometry, obtained by the inviscid solver, provides a mild enough adverse pressure gradient, then, the occurrence of separating regions by the viscous flow solver can be checked.

3. Flow field solution

The governing equations for two-dimensional, compressible, unsteady, inviscid flow in the conservative form are given as follows:

$$\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial x}+\frac{\partial G}{\partial y}=0, \tag{12}$$

where:

$$Q=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho e_t\end{bmatrix},\quad E=\begin{bmatrix}\rho u\\ \rho u^2+P\\ \rho uv\\ \rho uh_t\end{bmatrix},\quad G=\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2+P\\ \rho vh_t\end{bmatrix}. \tag{13}$$

Here, Q is the conservative vector, and E and G are the inviscid flux vectors. The equations are written in the generalized form using metrics of transformation, as:

$$\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial x}+\frac{\partial G}{\partial y}=0, \tag{14}$$

where:

$$Q_1=\frac{Q}{J},$$

$$E_1=\frac{1}{J}\left(\xi_x E+\xi_y G\right),$$

$$G_1=\frac{1}{J}\left(\eta_x E+\eta_y G\right). \tag{15}$$

Eq. (3) is discretized as follows:

$$\frac{\partial Q_1}{\partial t}+\frac{E_{1E}-E_{1W}}{\Delta\xi}+\frac{G_{1N}-G_{1S}}{\Delta\eta}=0, \tag{16}$$

where E_{1E} is the inviscid numerical flux computed in generalized coordinates at the east cell face, E. E_{1W}, G_{1N} and G_{1S} are the flux vectors at west, north and south faces, respectively. To solve the inviscid flow field, a recently developed in-house code, based on the Flux Difference Splitting (FDS) scheme of Roe [31], is used. The governing equations are discretized using formulations presented by Kermani [32]. The inviscid numerical flux, E_{1E}, is written in generalized coordinates, according to Ref. [32], as:

$$E_{1E}=\frac{1}{2}\left(E_{1L}+E_{1R}\right)-\frac{1}{2}\sum_{m=1}^{4}|\lambda_m|\,\delta W_m T_m, \tag{17}$$

where E_{1L} and E_{1R} are the inner and outer value of E_1 at face E, λ is the eigenvalue of the Jacobian matrix determined under Roe averaged conditions, δW is the wave amplitude, and T is the eigenvector corresponding to the eigenvalue, λ. For a complete description of these parameters, the reader is referred to [31,32].

The Roe scheme gives non-physical expansion shocks in the regions where the eigenvalues of the Jacobian matrix vanish, e.g. the sonic regions and stagnation points. To avoid expansion shocks in the regions where the eigenvalues vanish, an entropy correction formula from Kermani and Plett is used here [33]. To validate the solver, the numerical results are compared with the experimental data of Emery et al. for a 2-dimensional NACA65-410 cascade [34]. In Figure 6, the pressure coefficient on the blade surfaces is plotted and compared with experimental data for a cascade with a solidity of 1.0, stagger angle of 22.5°, and 7.5° angle of attack.

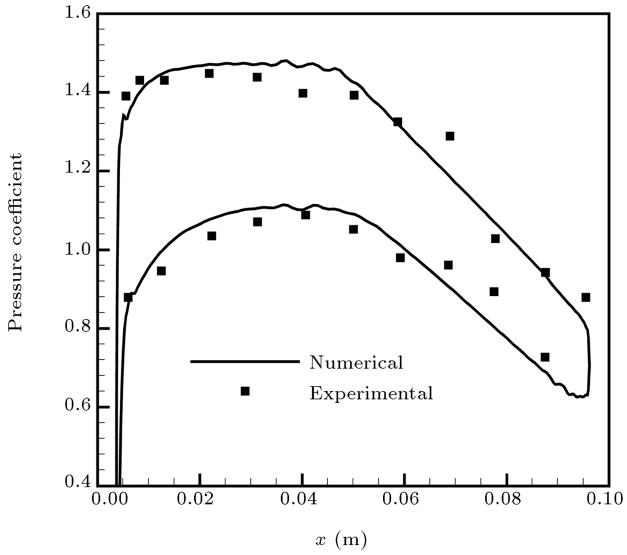

Figure 6. Comparison between numerical results and experimental data for 2-D NACA65-410 cascade.

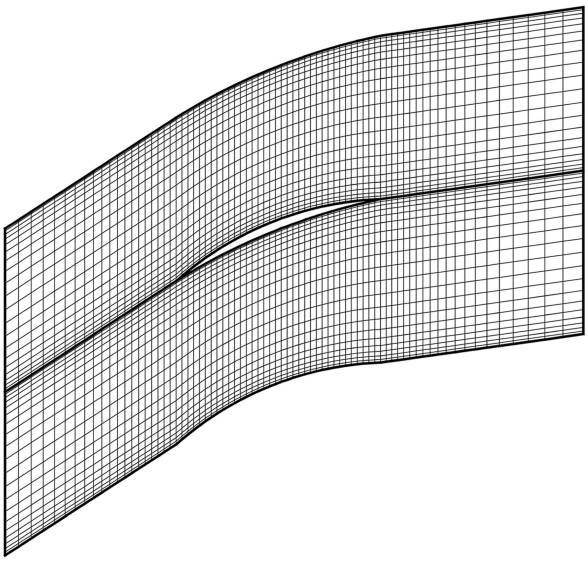

Figure 7. A sample of grid generated by an in-house grid-generator code.

3.1. Grid generation

A combined algebraic-elliptic algorithm is used for grid generation [35]. To impose grid orthogonality on the blade surfaces and clustering near the wall, the corresponding control functions are considered in the elliptic algorithm. An example of a generated grid for a blade cascade is shown in Figure 7.

3.2. Boundary conditions

For compressible flow in a compressor blade cascade, the pressure inlet and pressure outlet boundary conditions are applied at inlet and outlet boundaries, respectively. Two periodic boundaries are considered before and after the blade. The slip boundary conditions are applied to the blade suction and pressure surfaces.

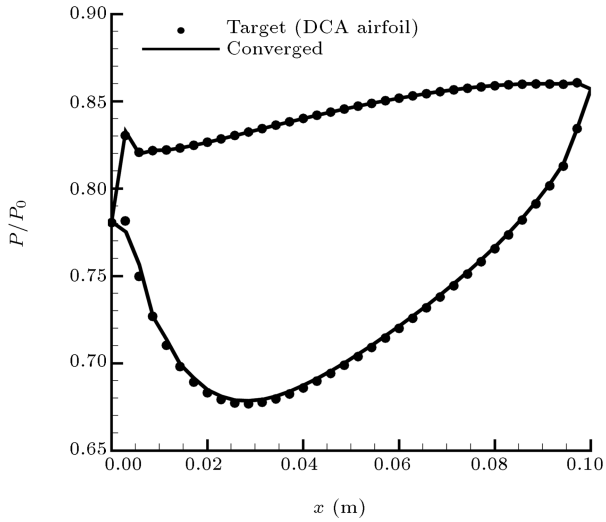

Figure 8. Pressure distributions for the target and the converged geometries for the Double Circular Arc (DCA) blade.

4. Results and discussion

The BS algorithm is applied to a Double Circular Arc (DCA) blade to assess the applicability of this method. In this computation, the back pressure ratio is set to $P_{\text{Back}}/P_0 = 0.843$. This guarantees a subsonic flow over the suction and pressure sides of the DCA blade. The blade cascade is shown in Figure 7. To validate the method, the flow field around the DCA airfoil is solved, and pressure distribution on the suction and pressure surfaces of the airfoil is obtained. These pressure distributions are considered as target pressure distributions, as the first assessment of the BS-algorithm. Starting from a flat plate airfoil (straight line) as an initial guess for the geometry, the geometry modification (also called the design process) is continued until the target pressure distribution is reached. It is noted in this first study that the airfoil geometry corresponding to the prescribed loading is the DCA blade. As shown in Figure 8, target and final (converged) pressure distributions present very good agreement. The initial, target and converged geometries are shown in Figure 9. Also shown in this figure are magnifications of viewpoints near the leading and trailing edges for better comparison. To assess the uniqueness of the method, the procedure is repeated using a different initial guess for geometry. That is an airfoil of zero thickness, as shown in Figure 10, which is used as an initial guess for the geometry. This figure shows that for the given target pressure distribution of Figure 8, the BS design algorithm produces the same DCA airfoil geometry as illustrated in Figure 9. In short, it is said that, for a given target pressure distribution, as shown in Figure 8, the final converged geometry using two different initial guesses for the geometries (i.e. a flat-plate airfoil as shown in Figure 9,

$$\Delta PD = \frac{\sum_{i=i_{LE}}^{i_{TE}} \left|(P_{Target} - P_{Comp.})_i\right|_{Suction\ surface} + \sum_{i=i_{LE}}^{i=i_{TE}} \left|(P_{Target} - P_{Comp.})_i\right|_{Pressure\ surface}}{2 \times (i_{TE} - i_{LE} + 1)}. \quad (18)$$

Box I

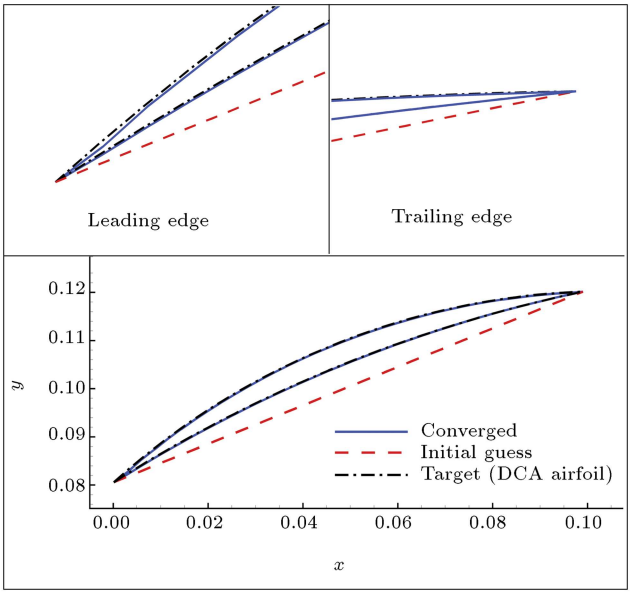

Figure 9. The evolution of the blade geometry for the first assessment of the BS algorithm: Staring from a flat-plate airfoil as the initial guess, it is shown that the final converged geometry matches very well with the DCA airfoil.

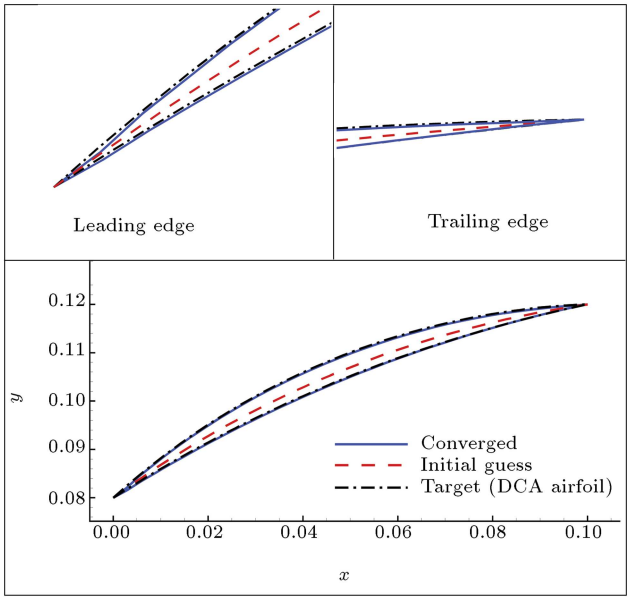

Figure 10. Assessment of the BS-algorithm: Staring from a different initial guess for the DCA airfoil (different from the flat-plate airfoil as noted in Figure 9), it is shown that the BS design algorithm finds the same DCA airfoil geometry.

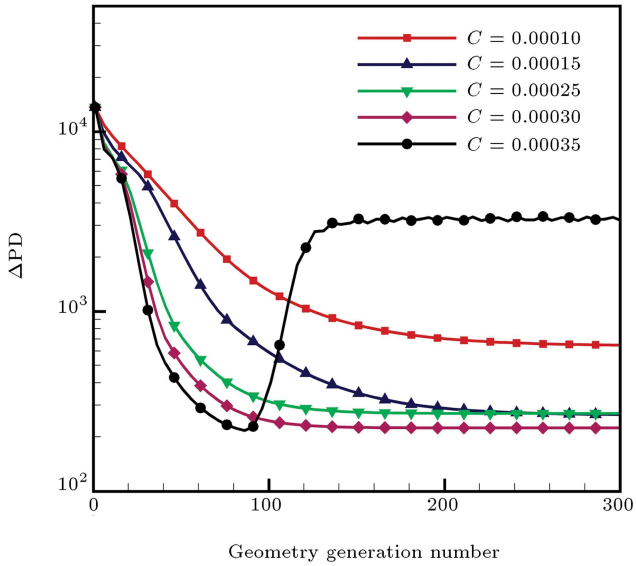

Figure 11. Effect of geometry correction coefficient, C, on convergence rate of the design process; curve of ΔPD with geometry generation number (here ΔPD represents the difference between the computed and target pressure distributions).

and a zero thickness airfoil as shown in Figure 10) produce the same results.

To assess the convergence degree of the solution, an error parameter, called ΔPD, is also defined in Eq. (18) as shown in Box I.

In Figure 11, the convergence history of the design process is shown for different values of geometry correction coefficient, C. The convergence rate is increased as coefficient C is increased to an optimum value of 0.00030 $m^2.s^2/kg$. More increase in values of C results in divergence of the method. It is concluded that for a design process, an optimum or critical value of geometry correction coefficient, C, should be determined.

To assess the accuracy of the BS algorithm for transonic flow regimes, the back pressure for the DCA blade flow-field shown in Figure 7 is decreased to $P_{Back}/P_0 = 0.759$. With reduced back pressure value, the flow in the passage becomes transonic and Mach number increases in the domain. The method is applied, considering the mentioned pressure distribution, using a flat plate airfoil (straight line) as the initial guessed geometry. Comparison between target and converged pressure distributions is shown in Figure 12. Also Mach number distribution is presented

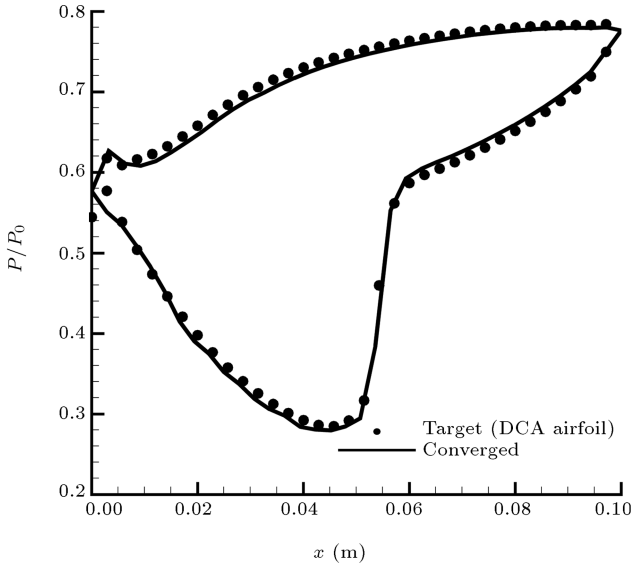

Figure 12. Target and converged pressure distributions for transonic test case.

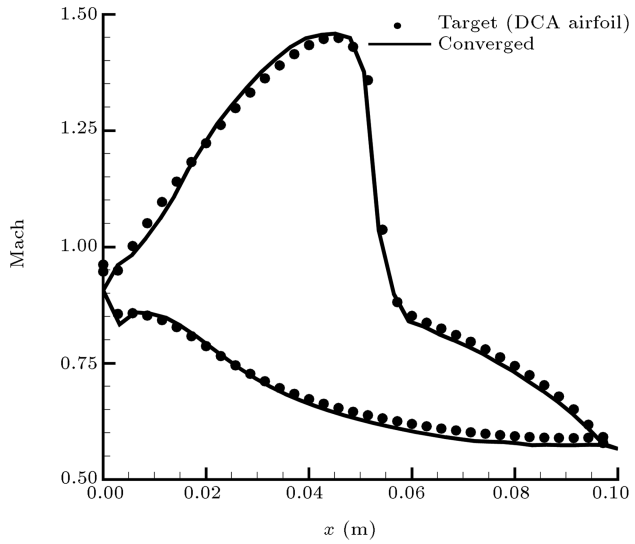

Figure 13. Mach number distribution for the target and the converged geometries for transonic test case.

in Figures 13 and 14. A shock is observed on the suction surface of the blade.

The previous test cases were studied for validation of the ball-spine algorithm. Here, as an applied example, a new target pressure distribution is considered and the accuracy and applicability of the method is assessed. To do this, using the pressure distributions of the DCA airfoil, the target pressure distribution is suggested to have higher loading. In Figure 15, the original, target (with higher loading) and converged pressure distributions are compared. The area between blade surface pressure distributions corresponds to blade loading, which is increased in this test case. In Figure 16, the original and new designed airfoil that has a greater loading is illustrated.

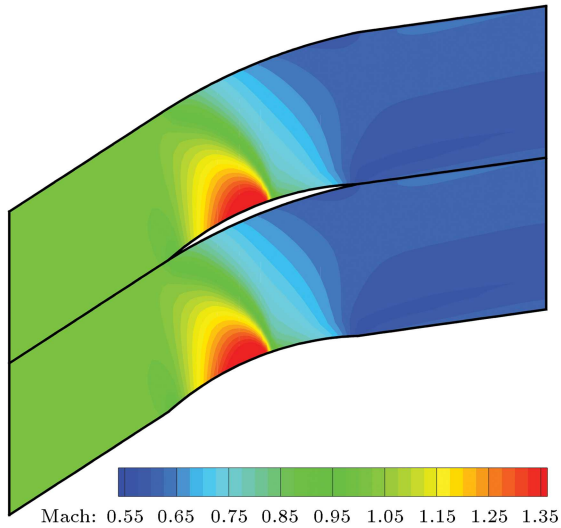

Figure 14. Cascade flow field shown by Mach number contours for the transonic test case.

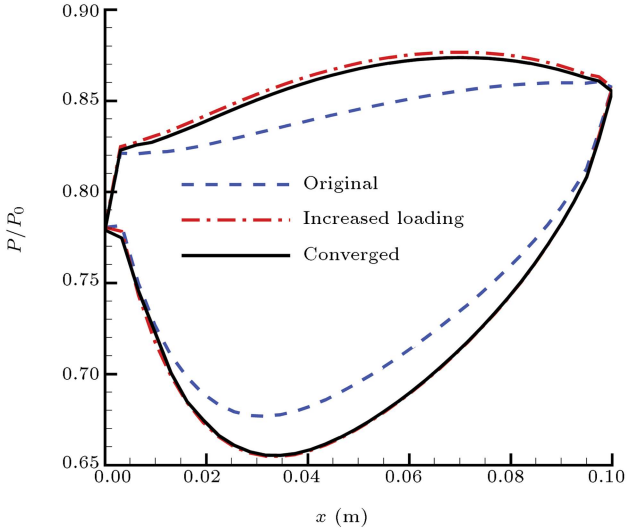

Figure 15. Original, increased loading and converged pressure distributions.

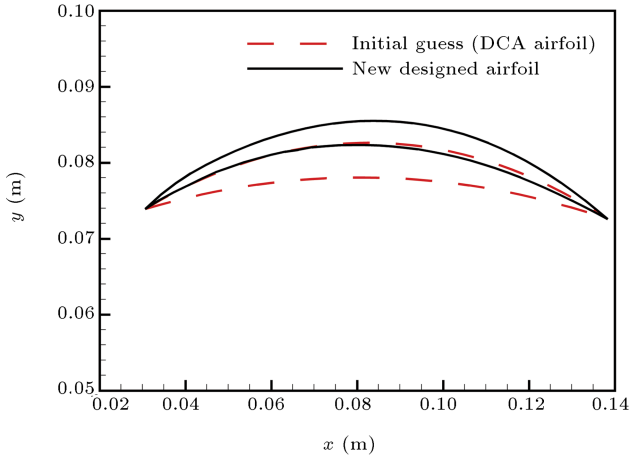

Figure 16. Original and new designed airfoils for increased loading test case.

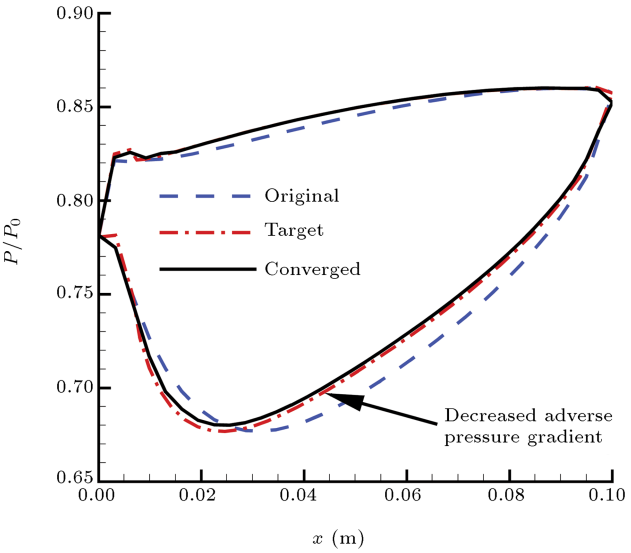

Figure 17. Original, decreased adverse pressure gradient and converged pressure distributions.

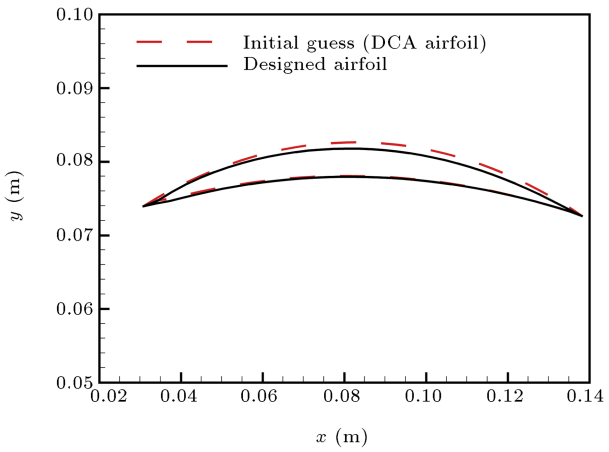

Figure 18. Original and new designed airfoils for decreased adverse pressure gradient test case.

For the second applied test case, the pressure distribution of the DCA airfoil is modified to decrease the adverse pressure gradient on the suction side of the airfoil. If separation occurs in the domain, using a lower adverse pressure gradient decreases the size of the recirculating pocket. In Figure 17, the original, target (with lower adverse pressure gradient) and converged pressure distributions are shown. The converged pressure distribution matches the target pressure distribution. The original and designed airfoils are presented in Figure 18.

5. Conclusions

A Ball-Spine (BS) algorithm has been proposed, in this paper, in order to design a compressor blade profile to match a prescribed target pressure distribution. A recently developed in-house inviscid flow solver has been used for this purpose. In the present study, for mesh generation, a combined algebraic-elliptic algorithm has been developed and used. The proposed BS algorithm has been tested on a DCA (target) blade profile, and the achieved geometry shows a good agreement with the target geometry. Starting from two different initial geometries, the method has been converged to a blade geometry, which satisfies the target pressure. After 100 geometry generation steps, the difference between computed and target pressure distributions are detailed; ΔPD reduces from 13700 Pa to a converged value of 222 Pa. The effect of geometry correction coefficient, C, on the stability and convergence of the design procedure has been investigated, and the optimum correction coefficient has been obtained.

The method has been applied to a transonic test case, and the accuracy of the method has been verified. As an application, the method has been applied using two new pressure distributions. The first distribution has a greater loading and the second has a lower adverse pressure gradient on the suction side. The corresponding airfoils satisfying the target pressure distributions have been designed.

Finally, it should be noted that the present ball-spine approach is not limited to application on axial flow compressors. An identical approach can be used for the design of turbines, centrifugal compressors, nozzles, diffusers and many more. This task can be performed with some changes in boundary conditions, the solver, the grid generator, and the design algorithm steps. It is general enough to be used in both external and internal flow applications.

Nomenclature

A	Element area (m^2)
a	Acceleration of the ball (m/s^2)
C	Geometry correction coefficient (m^2s^2/kg)
CPD	Computed Pressure Distribution (Pa)
E	Inviscid flux vectors in x-direction
E_1	Inviscid flux vectors in ξ-direction
e_t	Total internal energy (J/kg)
F	Force imposed on the ball (N)
f	Filtration coefficient
G	Inviscid flux vectors in y-direction
G_1	Inviscid flux vectors in η-direction
h_t	Total enthalpy (J/kg)
J	Jacobian of transformation
m	Ball mass (kg)
n	Normal direction

P	Static pressure (Pa)
Q	Conservative vector in physical domain
Q_1	Conservative vector in computational domain
T	The eigenvector of Jacobian matrix
t	Time (s), tangential direction
TPD	Target Pressure Distribution (Pa)
u	Velocity component in x-direction (m/s)
v	Velocity component in y-direction (m/s)
x	Position of the ball (m), x coordinate
y	Position of the ball (m), y coordinate
ΔP	Difference between target and computed pressures (Pa)
ΔPD	Difference between target and computed pressure distributions (Pa)
Δs	Displacement of the ball (m)
Δt	Time step (s)
δW	The wave amplitude
η	η-coordinate in computational domain
θ	The angle between force vector and spine direction (rad)
λ	The eigenvalue of Jacobian matrix
ξ	ξ-coordinate in computational domain
ρ	Density (kg/m^3)

Subscripts

Back	At the outlet
Comp.	Computed conditions
E	The East face of the control volume
L	The inner side of the cell face
LE	Leading Edge
N	The north face of the control volume
new	New conditions
old	Old conditions
R	The outer side of the cell face
rel	Relative to the leading edge
S	The South face of the control volume
Sp	Projected on spine direction
Target	Target conditions
TE	Trailing Edge
W	The West face of the control volume
x	x-derivative
y	y-derivative

References

1. Li, H-D., He, L., Li, Y-S. and Wells, R.G. "Blading aerodynamics design optimization with mechanical and aeromechanical constraints", *Proceedings of ASME Turbo Expo 2006*, Barcelona, Spain, GT2006-90503 (2006).
2. Verstraete, T., Alsalihi, Z. and Van den Braembussche, R.A. "Multidisciplinary optimization of a radial turbine for micro gas turbine applications", *Proceedings of ASME Turbo Expo 2007*, Montreal, Canada, GT2007-27484 (2007).
3. Ghadak, F. "A direct design method based on the laplace and euler equations with application to internal subsonic and supersonic flows", Ph.D. Thesis, Sharif University of Technology, Aero Space Department, Iran (2005).
4. Lamm, P.K. "Inverse problems and Ill-posedness", *Inverse Problems in Engineering: Theory and Practice, ASME* (1993).
5. Garabedian, P. and McFadden, G. "Design of supercritical swept wings", *AIAA Journal*, **30**(3), pp. 289-291 (1982).
6. Cheng, C.-H. and Wu, C.-Y. "An approach combining body fitted grid generation and conjugate gradient methods for shape design in heat conduction problems", *Numerical Heat Transfer, Part B*, **37**(1), pp. 69-83 (2000).
7. Jameson, A. "Optimal design via boundary control", *Optimal Design Methods for Aeronautics, AGARD*, pp. 3.1-3.33 (1994).
8. Kim, J.S. and Park, W.G. "Optimized inverse design method for pump impeller", *Mechanics Research Communications*, **27**(4), pp. 465-473 (2000).
9. Frank, P.D. and Shubin, G.R. "A comparison of optimization-based approach for a model computational aerodynamics design problem", *Journal of Computational Physics*, **98**, pp. 74-89 (1992).
10. Dedoussis, V., Chaviaropoulos, P. and Papailiou, K.D. "A fully 3-D inverse method applied to the design of axisymmetric ducts", *Proceeding of TURBO EXPO, 37th International Gas Turbine Aeroengine Congress. Expo.*, Cologne (1992).
11. Dedoussis, V., Chaviaropoulos, P. and Papailiou, K.D. "Rotational compressible inverse design method for two-dimensional, internal flow configurations", *AIAA Journal*, **31**(3), pp. 551-558 (1993).
12. Demeulenaere, A. and Braembussche, R. van den, "Three-dimensional inverse method for turbomachinary blading design", *ASME Journal of Turbomachinary*, **120**(2), pp. 247-255 (1998).
13. De Vito, L. and Braembuussche, R.V.D. "A novel two-dimentional viscous inverse design method for

turbomachinery blading", *Transactions of the ASME*, **125**, pp. 310-316 (2003).

14. Leonard, O. and BraemBussche, R. "A two-dimensional navier stokes inverse solver for compressor and turbine blade design", *Proceeding of the IMECH E part A Journal of Power and Energy*, **211**, pp. 299-307 (1997).

15. Henne, P.A. "An inverse transonic wing design method", *AIAA Paper*, pp. 80-0330 (1980).

16. Volpe, G. "Inverse design of airfoil contours: Constraints, numerical method applications", *AGARD*, Paper 4 (1989).

17. Barger, R.L. and Brooks, C.W. "A streamline curvature method for design of supercritical and subcritical airfoils", NASA TN D-7770 (1974).

18. Malone, J., Vadyak, J. and Sankar, L.N. "Inverse aerodynamic design method for aircraft components", *J. Aircraft*, **24**(1), pp. 8-9 (1986).

19. Malone, J., Vadyak, J. and Sankar, L.N. "A technique for the inverse aerodynamic design of nacelles and wing configurations", *AIAA Paper*, pp. 85-4096 (1985).

20. Campbell, R.L. and Smith, L.A. "A hybrid algorithm for transonic airfoil and wing design", *AIAA Paper*, pp. 87-2552 (1987).

21. Bell, R.A. and Cedar, R.D. "An inverse method for the aerodynamic design of three-dimensional aircraft engine nacelles", in *Proceedings of the Third International Conference on Inverse Design Concepts and Optimization in Engineering Sciences*, ICIDES-III, G.S. Dulikravich, Ed, Washington, D.C., 23-25, October, pp. 405-417 (1991).

22. Malone, J.B., Narramore, J.C. and Sankar, L.N. "An efficient airfoil design method using the Navier-Stokes equations", *AGARD*, Paper 5 (1989).

23. Malone, J.B., Narramore, J.C. and Sankar, L.N. "Airfoil design method using the Navier-Stokes equations", *J. Aircraft*, **28**(3), pp. 216-224 (1991).

24. Takanashi, S. "Iterative three-dimensional transonic wing design using integral equations", *J. Aircraft*, **22**, pp. 655-660 (1985).

25. Hirose, N., Takanashi, S. and Kawai, N. "Transonic airfoil design procedure utilizing a Navier-Stokes analysis code", *AIAA J.*, **25**(3), pp. 353-359 (1987).

26. Dulikravich, G.S. and Baker, D.P. "Aerodynamic shape inverse design using a fourier series method", *AIAA Paper*, pp. 99-0185 (1999).

27. Nili-Ahmadabadi, M., Durali, M., Hajilouy, A. and Ghadak, F. "Inverse design of 2D subsonic ducts using flexible string algorithm", *Inverse Problems in Science and Engineering*, **17**(8), pp. 1037-1057 (2009).

28. Nili-Ahmadabadi, M., Hajilouy, A., Durali, M. and Ghadak, F. "Duct design in subsonic and supersonic flow regimes with and without normal shock waves using flexible string algorithm", *Scientia Iranica Journal*, **17**(3), pp. 179-193 (2010).

29. Nili-Ahmadabadi, M., Hajilouy, A., Ghadak, F. and Durali, M. "A novel 2-D incompressible viscous inverse design method for internal flows using flexible string algorithm", *Journal of Fluids Engineering, ASME*, 132/031401-1-9 (2010).

30. Nili-Ahmadabadi, M., Durali, M. and Hajilouy, A. "A novel quasi-3D design method for centrifugal compressor meridional plane", *Proceedings of ASME Turbo Expo 2010*, Glasgow, UK, GT2010-23341 (2010).

31. Roe, P.L. "Approximate riemann solvers, parameter vectors and difference schemes", *Journal of Computational Physics*, **43**, pp. 357-372 (1981).

32. Kermani, M.J. "Development and assessment of upwind schemes with application to inviscid and viscous flows on structured meshes", Ph.D. Thesis, Department of Mechanical & Aerospace Engineering, Carleton University, Canada (2001).

33. Kermani, M.J. and Plett, E.G. "Modified entropy correction formula for the roe scheme", *AIAA Paper*, pp. 2001-0083 (2001).

34. Emery, J.C., Herrig, L.J., Erwin, J.R. and Felix, A.R. "Systematic two dimensional cascade tests of NACA 65-series compressor blades at low speeds", *NACA Reports*, 1368 (1957).

35. Hoffmann, K.A. and Chiang, S.T., *Computational Fluid Dynamics*, **I**, 4th Ed (2000).

Biographies

Ali Madadi obtained his BS and MS degrees from Sharif University of Technology, Tehran, Iran. His MS thesis concerned the field of axial flow compressor performance prediction. Currently, he is a PhD degree candidate of Mechanical Engineering at Amirkabir University of Technology, Tehran, Iran, working on the inverse design of axial flow compressor blades. During his PhD program, he and his colleagues developed a three dimensional inviscid flow solver based on the Roe scheme for flow field analysis of axial flow compressors. The solver is applicable for both rotors and stators, and he has used the solver in the inverse design method to design axial flow compressor blade profiles. His research interests include CFD, inverse design and compressible flows.

Mohammad Jafar Kermani obtained his BS degree in Mechanical Engineering (Thermal-Fluid) from Shiraz University, Iran, his MS degree in Applied

Mathematics from Manchester University, UK, and his PhD degree from Carleton University, Canada. He also pursued a $2^{1/2}$ year PDF in UNB, Canada, on steam turbines and PEMFCs.

Currently he is Associate Professor in Amirkabir University of Technology, Tehran, Iran, and is also taking part in the Alexander von Humboldt Fellowship for Experienced Researchers in the Center for Solar Energy and Hydrogen Research, in Ulm, Germany.

He has authored 94 papers and supervised 33 graduate students. He is also the recipient of MITACS, first prize, in 2003, in Halifax, Canada.

Mahdi Nili-Ahmadabadi received his MS and PhD degrees from Sharif University of Technology, Tehran, Iran, in 2005 and 2010, respectively. Currently, he is Assistant Professor and faculty member of the Mechanical Engineering Department of the Isfahan University of Technology, in Iran. His major research interests are inverse design, turbomachinery, experimental aerodynamics, and PIV measurement.