Proceedings of ASME Turbo Expo 2011
GT2011
June 6-10, 2011, Vancouver, British Columbia, Canada

GT2011-46091

APPLICATION OF AN INVERSE DESIGN METHOD TO MEET A TARGET PRESSURE IN AXIAL-FLOW COMPRESSORS

Ali Madadi
Amirkabir University of Technology(Tehran Polytechnic)
ali.madadi@gmail.com

Mohammad Jafar Kermani
Amirkabir University of Technology(Tehran Polytechnic)
mkermani@aut.ac.ir

Mahdi Nili-Ahmadabadi
Isfahan University of Technology
m.nili@cc.iut.ac.ir

ABSTRACT

Numerous methods have been developed to design axial-flow compressor blades. These methods are generally categorized into inverse or direct approaches. In the inverse design methods, a distribution of an aerodynamic parameter such as pressure or velocity on the blade surfaces is given, and the target blade geometry that can provide the corresponding distribution is to be determined.

In the present work, a novel inverse design algorithm called Ball Spine Algorithm (BSA) is developed to design an axial-flow compressor on the blade to blade surface. In the BSA, the blade surfaces are considered as a set of virtual balls that move freely along the specified directions, called 'spines'. At first, initial blade geometry is guessed and the blade-to-blade flow field is analyzed by an in-house inviscid flow solver based on the Roe scheme. Comparing the computed pressure distribution (CPD) on the blade surfaces with the target pressure distribution (TPD), gives a guideline in a differential movement for the balls to obtain a modified geometry. For the flow field analysis on the modified geometry, new grids are generated by a combined algebraic-elliptic code. The sequence is repeated until the target pressure is reached. For validation, the approach is applied on an arbitrary blade profile.

1 INTRODUCTION

The design of hardware involving fluid flow or heat transfer such as intakes, manifolds, duct reducers, compressor and turbine blades, etc., is defined as the shape determination of the solid elements so that the flow or heat transfer rate is optimal in some sense. Often, both Computational Fluid Dynamics (CFD) and design algorithms are involved in solving an optimal shape design problem. The limitations and computational cost of the design techniques are challenging problems for present time computational technology.

One of the optimal shape design methods is the Surface Shape Design (SSD). Surface Shape Design (SSD) in fluid flow problems usually involves finding a shape associated with a prescribed distribution of surface pressure or velocity. It should be noted that the solution of a 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 [1].

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 and the flow field unknowns simultaneously in a (theoretically) single-pass or one-shot approach [1]. The governing equations of coupled methods are more complicated than well-known fluid dynamics equations; hence these methods are limited to simple flow regimes. In addition, the conventional flow solvers could not be used.

The iterative shape design approach relies on repeated shape modifications such that each iteration consists of flow solution followed by a geometry updating scheme. In other words, a series of sequential problems are solved in which the surface shape is altered between iterations so that the desired TPD is finally achieved [1]. In the iterative methods, the governing equations are similar to the flow field equations and the conventional solver could be used as a black box. Hence the iterative methods are applicable for complicated flow regimes.

Iterative methods, such as optimization techniques, have been by far the most widely used to solve practical SSD problems. The traditional iterative methods used for SSD problems are often based on trial and error or optimization algorithms. The trial and error process is very time-consuming and computationally expensive and hence needs designer experience to reach minimum costs. Optimization methods

[2],[3] are commonly used to automate the geometry modification in each iteration cycle. In such methods, an objective function (e.g., the difference between a computed surface pressure and the target surface pressure [4]) is minimized, subjected to the flow constraints which have to be satisfied. Although the iterative methods are general and powerful, they are often computationally costly and mathematically complex. These methods can utilize the analysis methods for the flow field solution as a black-box.

Other iterative methods presented so far use the physical algorithms instead of the mathematical algorithms to automate the geometry modification in each iteration cycle. The physical 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 and hence the mass can be fictitiously injected through the wall in such a way that the new wall satisfies the slip boundary condition. Aiming removal of nonzero normal velocity on the boundary, a geometry update determined by applying either the transpiration model based on mass flux conservation [5]-[8] or the streamline model based on alignment with the streamlines [10], must be adopted.

An alternative algorithm is based on the residual-correction approach. In this method, the key problem is to relate the calculated differences between the actual pressure distribution on the current estimate of the geometry and the TPD (the residual) to required changes in the geometry. Obviously, the art in developing 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 rule, making use of relations between geometry changes and pressure differences known from linearized flow theory [1].

The residual-correction decoupled solution methods try to utilize the analysis methods as a black-box. Barger et al., [11] 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 [12]-[21].

Nili et al. presented an iterative inverse design method for internal flows called Flexible String Algorithm (FSA). They considered the duct wall as a flexible string frequently deformed under the difference between TPD and CPD. They developed this method for non-viscous compressible [22][23] and viscous incompressible internal flow regime [24].

Recently, Nili et al., have presented a novel inverse design method called Ball-Spine Algorithm (BSA). They developed this method for quasi-3D design of meridional plane of centrifugal compressor [25].

In this research, the BSA is used for the 2-D design of axial compressor blading. For the flow field solution, a recently developed in-house code is used.

2 NOMENCLATURE

F	Force imposed on balls
y	y position of balls (m), y coordinate
A	Element area
a	Acceleration of balls
m	Balls mass (kg)
n	Normal direction
t	Tangential direction
f	Filtration coefficient
C	Geometry correction coefficient
P	Static pressure (Pa)
ΔP	Difference between target and computed pressures
Δs	Displacement of each ball
Δt	Time step(s)
TPD	Target Pressure Distribution
CPD	Computed Pressure Distribution
$I_{\max}$	Maximum number of grids in horizontal direction

Subscripts

rel	Relative to the leading edge
target	Target conditions
new	New conditions
old	Old conditions
comp.	Computed conditions
LE	Leading edge

3 FUNDAMENTALS OF THE METHOD

In the present work, the wall is considered as a set of virtual balls, freely moving along the specified direction, shown in Figure 1. Passing fluid flow through the flexible duct causes a pressure distribution to be applied to the wall from the outer side. If a target pressure distribution is applied to the inner side of each duct wall (Figure 2), it is logical that the flexible wall deforms to reach a shape satisfying the target pressure distribution on the wall. In other words, the force due to the difference between the target and current pressure distribution 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. If each virtual ball moves in the same force direction, the adjacent balls may collide together or move away from each other. This can disturb the wall modification procedure. To avoid this problem, each ball should freely move in a specified direction called a spine. In Figure 1, the spines are the normal line connecting the balls with the same x position on two walls. In other words, the horizontal length of duct remains unchanged during the shape modification procedure. In duct inverse design problems, it is essential that a characteristic length be fixed. The direction of the spines depends on what characteristic length should be fixed. Therefore, for different ducts, the spines are differently defined. Another constraint for wall modification is that one point of each wall should be fixed. Typically, the start point of each wall is fixed so that the duct inlet area remains fixed too.

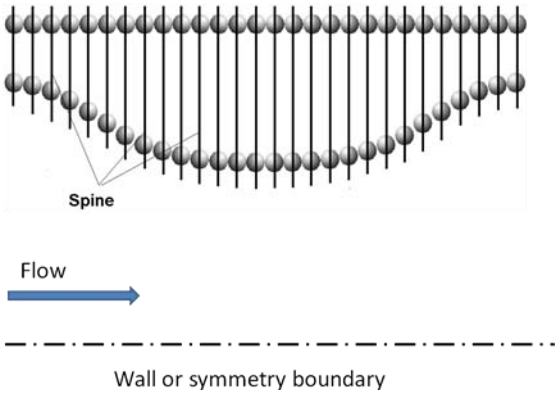


Figure 1. Simulation of a 2-D duct with balls and spines

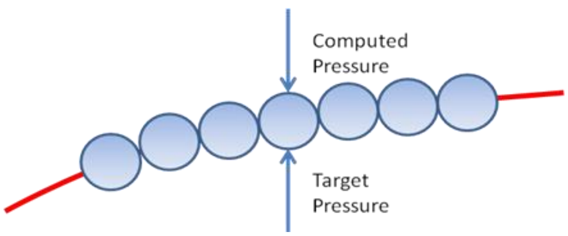


Figure 2. Applying the target and computed pressures on a sample ball

4 MATHEMATICAL APPROACH

4.1 Governing Equations

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

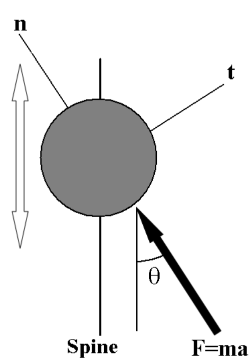


Figure 3. Free body diagram of a ball.

The net force applied on each ball in the spine direction is computed as:

$$F = \Delta P.A.\cos\theta \qquad (١)$$

Where,

$$\Delta P = P_{target} - P_{comp.} \qquad (٢)$$

and A is the area of each element.

If in specified time step (Δt), the ball can move on the spine, the corresponding displacement is computed from the following dynamic relations:

$$a = \frac{F}{m} \quad , \quad \Delta s = \frac{1}{2} a (\Delta t)^2 \qquad (٣)$$

Where, m and Δs are the mass and displacement of the ball, respectively. Substituting Eqns. (١) and (٢) into Eqn. (٣) yields,

$$\Delta s = \frac{1}{2}\frac{A}{m}(\Delta t)^2 \left(P_{target} - P_{comp.}\right)\cos\theta \qquad (٤)$$

Or

$$\begin{aligned} &\Delta s = C.\Delta P.\cos\theta \\ &\Delta P = \left(P_{target} - P_{comp.}\right) \\ &C = \frac{1}{2}\frac{A}{m}(\Delta t)^2 \quad \left[\frac{m^2 s^2}{kg}\right] \end{aligned} \qquad (٥)$$

As seen in Eqn. (٥), the coefficient C is composed of the element area, ball mass, and time step. If a large value is selected for C, the displacements will increase and the convergence rate will be improved. On the other hand, if the parameter C exceeds from a limit, the solution is unstable. Although a small value of C causes the design procedure to be stable, the convergence rate decreases.

The new position of each ball is calculated as follows:

$$y_{new} = y_{old} + \Delta s \qquad (٦)$$

$$x_{new} = x_{old} \qquad (٧)$$

Having updated the wall geometry, the new grids are generated for the internal domain and the flow field over the new domain is solved to compute the wall pressure distribution. The difference between computed and target pressure distributions causes the next shape modification. The procedure is repeated until the pressure distributions are matched. The convergence criteria are defined as

$$\sum \left| P_{target} - P_{computed} \right| \le \varepsilon_P \qquad (٨)$$

Or,

$$\sum \left| y_{new} - y_{old} \right| \le \varepsilon_y \qquad (٩)$$

The design algorithm is shown in Figure 4.

Because the wall is considered as a set of separated balls, during the design process, the wall curvature may be discontinuous in adjacent nodes (balls). An example of such discontinuity is shown in Figure 5. To smooth the wall curvature, a filtration method is applied on the wall y components after each geometry correction step. The filtration method is formulated as follows:

$$y(i,j) = \frac{y(i-1,j) + f.y(i,j) + y(i+1,j)}{f+2} \qquad (١٠)$$

Here, f is the filtration coefficient. Large values for f correspond to low filtration and small values for f results in major filtration. In the present work f is set to 4, i.e.

$$y(i,j)=\frac{y(i-1,j)+4y(i,j)+y(i+1,j)}{6} \qquad (۱۱)$$

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.

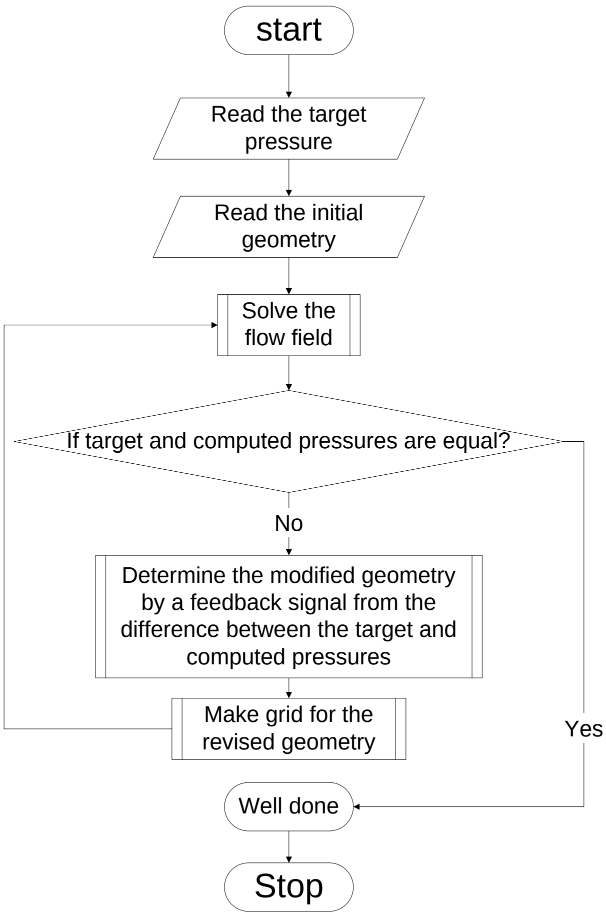


Figure 4. The design flowchart

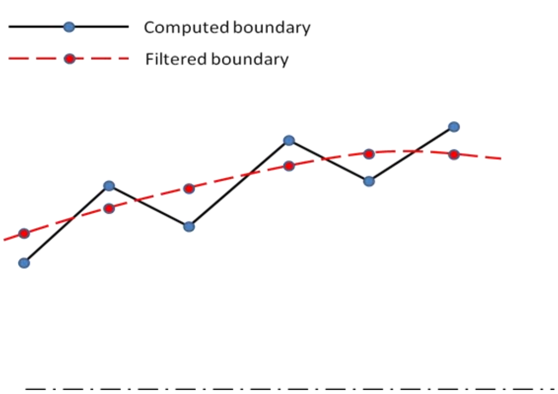


Figure 5. Displacement filtration

4.2 Grid Generation

A combined algebraic-elliptic algorithm is used for grid generation[26]. To impose the grid orthogonality on the blade surface and clustering near the wall, the corresponding control functions are considered in an elliptic algorithm. An example of a generated grid for a blade cascade is shown in Figure 6.

4.3 Flow Field Solution

To solve the inviscid flow field, a recently developed in-house code based on the flux difference splitting (FDS) scheme of Roe [27] is used. The governing equations are discretized in the computational domain using formulations presented by Kermani[28]. 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 [29]. To validate the solver, the numerical results are compared with the experimental data of Emery et al. for a 2-dimensional NACA65-410 cascade[31]. In Figure 7, the pressure coefficient on the blade surfaces is plotted for a cascade with solidity of 1.0, stagger angle of 22.5^0, and 7.5^0 angle of attack.

4.4 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 on the blade suction and pressure surfaces.

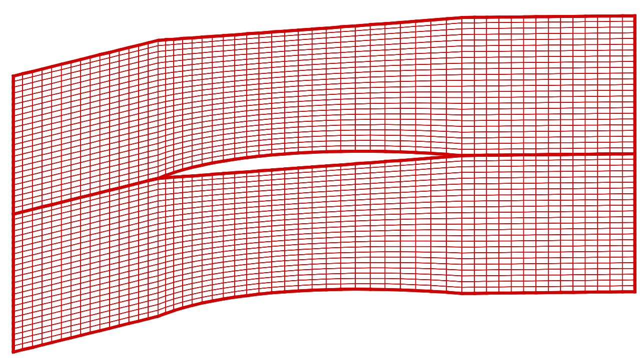


Figure 6. A sample grid generated using developed code

4.5 BSA Design Procedure

For compressor blade design, a target pressure distribution is given for each airfoil surface – suction side and pressure side. To satisfy both distributions, the geometry correction is done by applying the pressure distributions alternatively on the suction and pressure surface in such a way that only one boundary is corrected at each step and the other one is fixed. Finally, when both pressure distributions match, the geometry is fixed.

Because, in the flow field solver, the 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., $P_{rel} = P - P_{LE}$

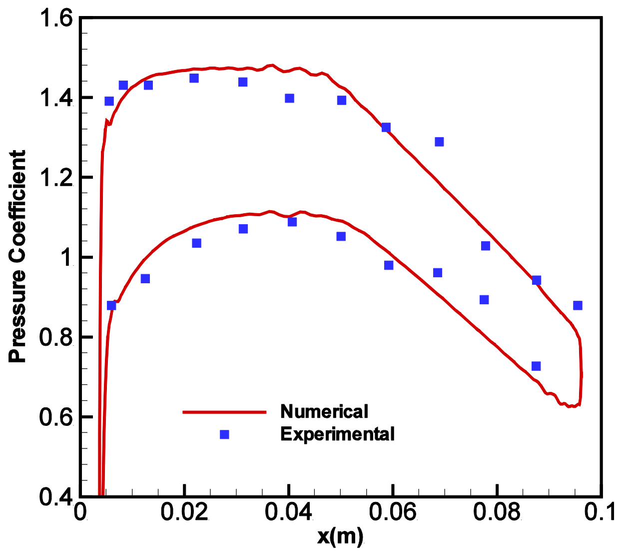


Figure 7. Comparison between numerical and experimental results for 2-D NACA65-410 cascade.

5 RESULTS AND DISCUSSION

The BSA is applied on an arbitrary blade to assess the applicability of this method. The blade cascade is shown in Figure 8. To validate the BSA, the flow field within the cascade is solved using the flow field solver, and the pressure distribution on the blade surfaces is determined. Then, these distributions are considered as the target pressure distribution and the design process starts from an initial geometry consisting of two straight walls making a constant area passage. Applying the design algorithm results in a designed geometry matching the target geometry, and thus the target pressure distributions are satisfied. In Figure 9, the target and the final pressure distributions show good agreement. The initial, target and designed geometries as three cascades are shown in Figure 10.

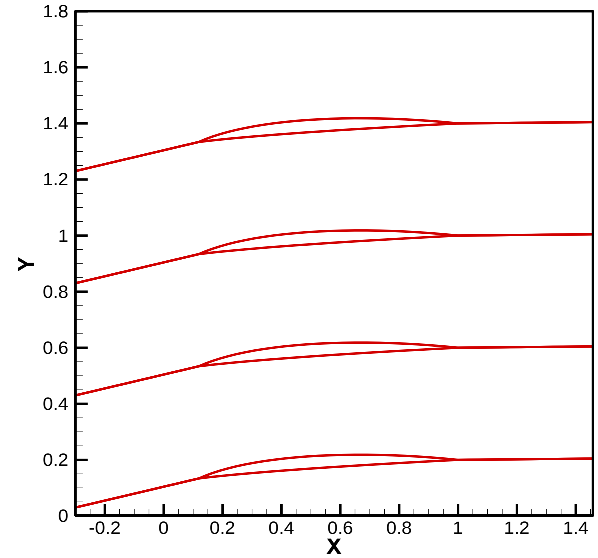
In Figure 11, the history of geometry correction is shown for various intermediate iterations. Similarly, the history of pressure side and suction side pressure distributions are shown in Figure 12 and Figure 13, respectively.

The L1 norm of the design process is calculated as:

$$error = \frac{\sum_{i=1}^{i=I_{\max}} \left|(y_{new} - y_{old})_i\right|_{\substack{\text{Suction}\\ \text{surface}}} + \sum_{i=1}^{i=I_{\max}} \left|(y_{new} - y_{old})_i\right|_{\substack{\text{Pressure}\\ \text{surface}}}}{2 \times I_{\max}} \qquad (١٢)$$

In Figure 14, the convergence history of the design process is shown for different values of coefficient C. The convergence rate is increased as the C is increased to an optimum value of 0.00035. More increase in values of C results in divergence of the method.

In Figure 15 and Figure 16, the Mach number and static pressure contours are shown.



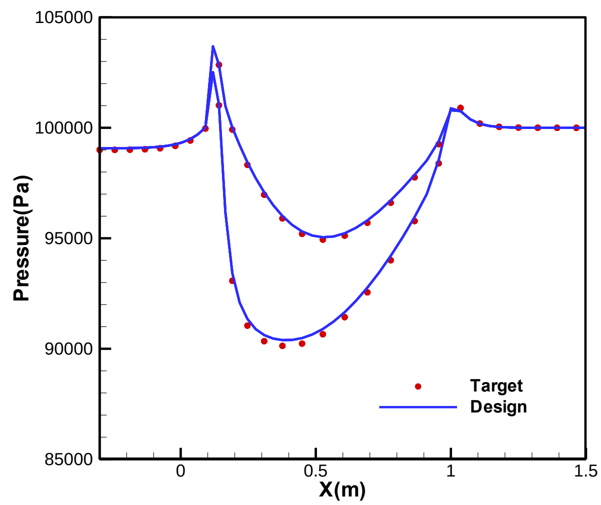


Figure 9. Pressure distributions for the target and the designed geometry

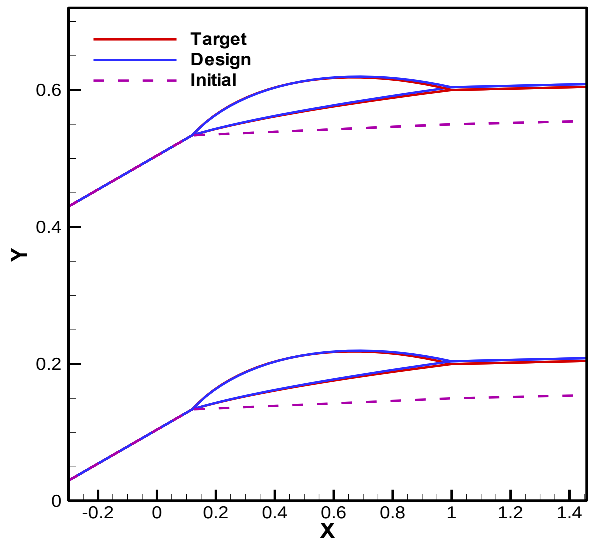


Figure 10. Initial, target and designed geometries

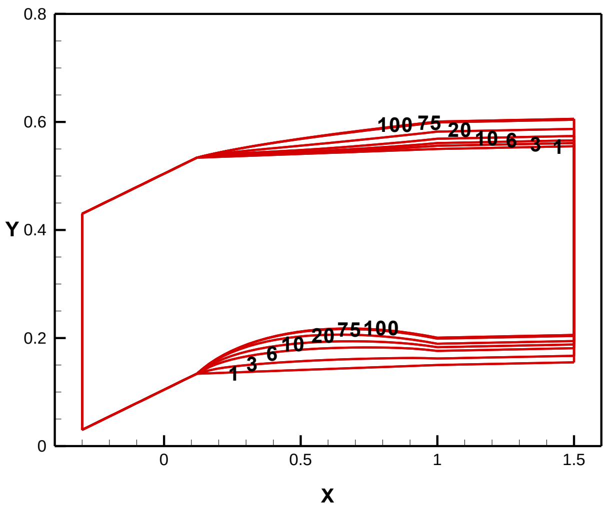


Figure 11. History of geometry correction

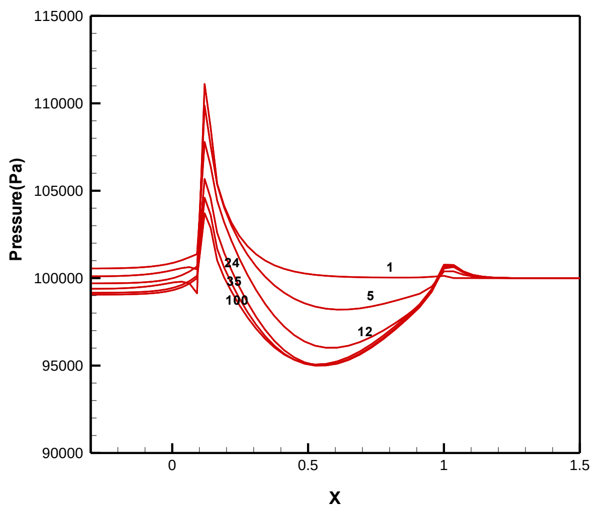


Figure 12. History of pressure side pressure distribution

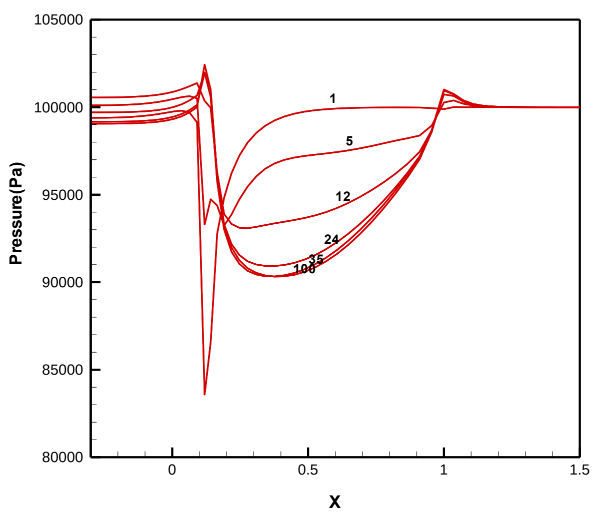


Figure 13 History of suction side pressure distribution

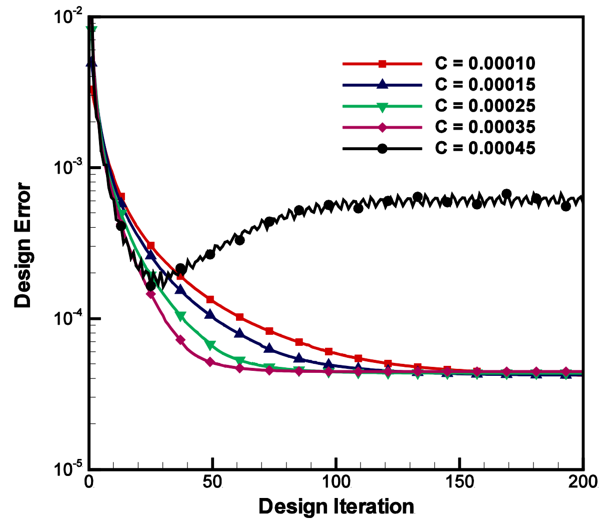


Figure 14. Effect of C coefficient on convergence rate of the design process

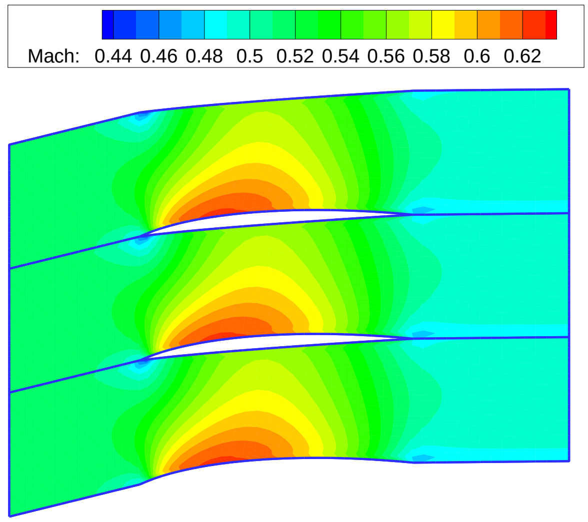


Figure 15 Mach number contours

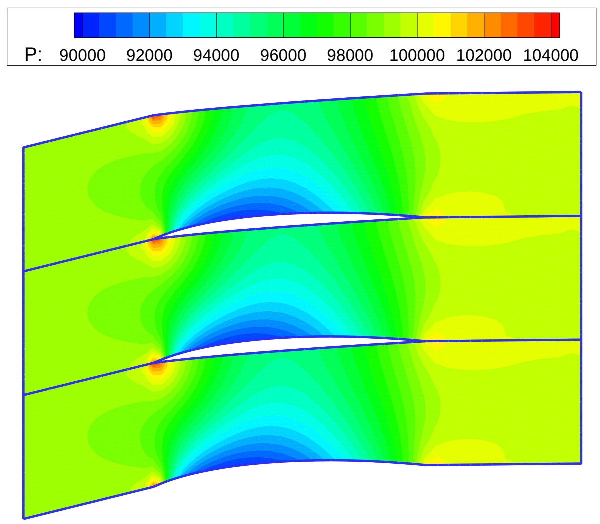


Figure 16. Static pressure contours

6 CONCLUSIONS

The BSA is used to design a compressor blade profile for a target pressure distribution. A recently developed in-house inviscid flow solver is used and for mesh generation a combined algebraic-elliptic algorithm is implemented. Starting from an initial geometry consisting of two straight lines, imposing the target pressure distribution results in a blade geometry which satisfies the target pressure. The BSA is tested on an arbitrary blade profile and the designed geometry shows good agreement with the target geometry.

7 REFERENCES

[1] Ghadak, F., 2005, “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.

[2] Cheng, Chin-Hsiang and Wu, Chun-Yin, 2000, “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), 69-83.

[3] Jameson, A., 1994, “Optimal design via boundary control”, Optimal Design Methods for Aeronautics, AGARD, 3.1 - 3.33.

[4] Kim, J. S., and Park, W. G., 2000, “Optimized Inverse Design Method For Pump Impeller”, Mechanics Research Communications, Vol. 27, No. 4 , pp.465-473.

[5] Dedoussis, V., Chaviaropoulos, P., and Papailiou K. D., 1993, “Rotational Compressible Inverse Design Method for Two-Dimensional, Internal Flow Configurations”, AIAA Journal, Vol. 31, No. 3, pp. 551-558.

[6] Demeulenaere A., and Braembussche, R. van den, 1998, “Three–Dimensional Inverse Method for Turbomachinary Blading Design”, ASME Journal of Turbomachinary, Vol. 120, No. 2, pp. 247–255.

[7] De Vito, L., and Braembuussche, R.V.D., 2003, “A Novel Two-Dimentional Viscous Inverse Design Method for Turbomachinery Blading”, Transactions of the ASME, Vol. 125, pp. 310-316.

[8] Leonard, O., and BraemBussche, R., 1997, “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.

[9] Henne, P. A., ١٩٨٠, “An Inverse Transonic Wing Design Method”, AIAA Paper ٨٠-٠٣٣٠.

[10] Volpe, G., 1989, “Inverse design of airfoil contours: Constraints, numerical method applications”, See AGARD, Paper 4.

[11] Barger, R. L., and Brooks, C. W., 1974, “A Streamline Curvature Method for Design of Supercritical and Subcritical Airfoils”, NASA TN D-7770.

[12] Garabedian, P. and McFadden, G., 1982, “Design of supercritical swept wings”, AIAA Journal, Vol. 30, No. 3, pp. 444–446.

[13] Malone, J., Vadyak, J. and Sankar, L.N., 1986, “Inverse aerodynamic design method for aircraft components”, J. Aircraft, Vol. 24, No. 1, pp. 8–9.

[14] Malone, J., Vadyak, J. and Sankar, L.N., 1985, “A technique for the inverse aerodynamic design of nacelles and wing configurations”, AIAA Paper 85–4096.

[15] Campbell, R.L. and Smith, L.A., 1987, “A hybrid algorithm for transonic airfoil and wing design”, AIAA Paper 87–2552.

[16] Bell, R.A. and Cedar, R.D., 1991, “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.

[17] Malone, J.B., Narramore, J.C. and Sankar, L.N., 1989, “An efficient airfoil design method using the Navier–Stokes equations”, AGARD, Paper 5.

[18] Malone, J.B., Narramore, J.C. and Sankar, L.N., 1991, “Airfoil design method using the Navier–Stokes equations”, J. Aircraft, Vol. 28, No. 3, pp. 216–224.

[19] Takanashi, S., 1985, “Iterative three-dimensional transonic wing design using integral equations”, J. Aircraft Vol. 22, pp. 655–660.

[20] Hirose, N., Takanashi, S. and Kawai, N., 1987, “Transonic airfoil design procedure utilizing a Navier–Stokes analysis code”, AIAA J., Vol. 25, No. 3, pp. 353–359.

[21] Dulikravich, G.S. and Baker, D.P., 1999, “Aerodynamic shape inverse design using a Fourier series method”, AIAA Paper 99–0185.

[22] Nili-Ahmadabadi, M., Durali, M., Hajilouy, A. and Ghadak, F., 2009 “Inverse Design of 2D Subsonic Ducts Using Flexible String Algorithm”, Inverse Problems in Science and Engineering, Vol. 17, No. 8, 1037-1057.

[23] Nili-Ahmadabadi, M., Hajilouy, A., Durali, M. and Ghadak, F., 2010, Duct Design in Subsonic and Supersonic Flow Regimes with and without Normal Shock Waves Using Flexible String Algorithm, Scientia Iranica Journal, Vol. 17, No. 3, pp. 179-193.

[24] Nili-Ahmadabadi, M., Hajilouy, A., Ghadak, F. and Durali, M., 2010, “A Novel 2-D Incompressible Viscous Inverse Design Method for Internal Flows Using Flexible String Algorithm”, Journal of Fluids Engineering, ASME, Vol. 132/031401-1-9.

[25] Nili-Ahmadabadi, M., Durali, M., and Hajilouy, A., 2010 “A Novel Quasi-3D Design Method for Centrifugal Compressor Meridional Plane”, Proceedings of ASME Turbo Expo 2010, Glasgow, UK, GT2010-23341.

[26] Hoffmann K.A. and Chiang S.T., Computational Fluid Dynamics

[27] Roe P.L., Approximate Riemann Solvers, Parameter Vectors and Difference Schemes, J. of Computational Physics, Vol. 43, pp. 357-372, 1981.

[28] 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.

[29] Kermani M.J. and Plett E.G., Modified Entropy Correction Formula for the Roe Scheme, AIAA Paper # 2001–0083, 2001.

[30] Chakravarthy S.R. and Szedma K.Y., Euler Solver for Three Dimensional Supersonic Flows with Subsonic Pockets, AIAA Conference, 1987

[31] 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.