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

GT2011-46130

MULTI-ROW INVERSE METHOD BASED ON THE ADJOINT OPTIMIZATION

Jinguang Yang
School of Power and Energy, Northwestern Polytechnical University
Xi'an 710072, Shaanxi, China

Xiuquan Huang
School of Power and Energy, Northwestern Polytechnical University
Xi'an 710072, Shaanxi, China

Hu Wu
School of Power and Energy, Northwestern Polytechnical University
Xi'an 710072, Shaanxi, China

ABSTRACT

The conventional inverse method possesses several disadvantages: experience-dependence, inconvenience for multi-row application and lots of human intervention. This hinders it from becoming a routine design tool. In the present paper, those mentioned shortcomings are conquered by combining the inverse method with a multi-row adjoint method, which could give an automatic and optimized design in a multi-row enviroment. The principles of the inverse method and adjoint method are first introduced. Then a derivation for the adjoint equation based on the Euler inverse method is conducted, and the corresponding adjoint boundary conditions are deeply discussed. After that, the developed inverse method is validated by recovering a turbine stator from a different initial shape. And the validity of the proposed adjoint based inverse method are illustrated by the redesign of a 1-1/2 turbine stage. Finally a comparison is made between the inverse method, the adjoint method based inverse method and the adjoint method based direct method.

NOMENCLATURE

Symbols

f	camber line wrap angle
m	mass flow rate
n	normal
p	static pressure
r	radius
s	area; entropy
t	tangential thickness
u	axial velocity
v	circumferential velocity
v_g	grid velocity
w	radial velocity
A	Jacobian matrix (of axial flux)
B	Jacobian matrix of circumferential flux
C	Jacobian matrix of radial flux
F	axial flux
G	circumferential flux
I	objective function
J	line vector defined by Eq. (25)
M	integrated function of objective function
H	radial flux
M	transform matrix
R	flow equation residual
S	source term of flow equation
U	conservative flow vector
V	volume
$\mathbf{W}$	relative velocity vector
W	primitive flow vector
X	line vector defined by Eq. (41)
α	design variable(s)
φ	adjoint variable corresponding camber line generation equation
λ	adjoint variable
ρ	density
π	total pressure ratio
θ	circumferential coordinate
Δ	difference between blade upper and lower surfaces
Λ	adjoint vector
Θ	camber line generation equation residual

Subscripts

0	baseline value
1-5	vector index
n	normal
bl	blade average value
b	blade
r	derivative in r direction
x	derivative in x direction

Superscripts

+	blade upper surface
-	blade lower surface
-1	inverse matrix

1. INTRODUCTION

Modern aerodynamic design of turbomachinery relies heavily on the use of Computational Fluid Dynamics (CFD), which makes the designer understand the flow physics better than ever before (Denton [1]). With the help of CFD, lots of 3D effects, such as blade sweep, lean and twist, and end wall profiling, are introduced during the design process confidently. However, the main role played by the current CFD is analysis and check. Its direct use in the design mode is rare, and three dimensional designs by CFD don't prevail yet. Researchers have always been endeavoring to develop fully 3D design methods. One of them is the inverse method.

The basic principle of an inverse method lies in that, specifying some aerodynamic parameters initially, a 3D blade shape could be obtained by solving the corresponding governing equations with proper boundary conditions; during the iterations, the blade shape is modified accordingly and the converged blade shape will ensure the flow tangentially aligned with blade surfaces while satisfying the pre-specified aerodynamic parameters. The inverse method enables designers to directly control the blade aerodynamic performance, so that they could use their fluid dynamics insights to improve the designs. Originated from the incompressible flow, the state-of-art inverse method has reached a fairly high level. Tiow and Zangeneh [2] developed an inverse method considering viscous effects by viscous body force, and used the developed methods for the inverse design of NASA rotor 67. Van Rooij and Dang et al. [3] utilized the inverse method to investigate the stage-matching effect. Roidl and Ghaly [4] devised a new inverse method, based on the blade surface virtual movement and time-accurate simulation, which demonstrated to be robust and reliable.

Although the inverse method is starting to be applied in multi-row environment, it is far from becoming a routine tool, because some obstacles still exist. These bottle-neck problems include: a) the inverse method is experience-dependent. How to specify the target aerodynamic parameter distribution is a tough task, and requires the designer to possess lots of experience, both in aerodynamics and in use of the inverse method. b) even if the inverse method is applicable to multi-row environment with the help of mixing plane treatment, it is hard to consider the blade-row matching in advance. Usually, a low order model, such as through-flow model, has to be used to prepare the input data. Here the blade-row matching specifically means a good match of the specified design aerodynamic parameters radial distributions between the adjacent blade rows. c) the initially specified target distribution may not be the optimized one, so several trial-and-error tests are needed in order to get a satisfactory design. This means that manual intervention is necessary in the design process.

The above listed problems of the current inverse method inhibit its wide application. One way to solve this is to combine the inverse method with an optimization technique. This thought is not new. Tiow, Yiu and Zangeneh [5] integrated the simulated annealing method with the inverse method for 2D cascade design; Bonaiuti and Zangeneh [6] reported their combination of the inverse method, response surface technique and multi-objective evolutionary algorithm to realize a multi-row, multi-objective, multi-point optimization design.

The optimization method used here is the adjoint method. Adjoint method originates from control theory, and its wide acceptance in CFD field is largely due to the pioneering work of Jameson [7]. It has the merit that the flow sensitivity is independent of the number of design variables, so it is very suited for large scale optimization. The adjoint method recently had received lots of attention in turbomachinery field. Yang, Wu and Liu [8] presented an optimization design of 2D cascades based on the adjoint method. Wu, Yang and Liu [9] extended its application to three dimensional single row blade. Wang et al [10] further enlarged its application to multi-row environment by adding an adjoint mixing plane treatment to the adjoint solver. A redesign of a 7-row compressor with a total of 1023 design variables resulted in a 2.47% increase in isentropic efficiency, which demonstrated the power of the adjoint optimization. Also a two-point optimization is adopted in the same paper, resulting in improved off-design performance.

The advantage of the adjoint method makes it a promising candidate for the inverse optimization. Actually, some attempts of combining the inverse method and the adjoint optimization together have been made by researchers. Iollo, Ferlauto and Zanetti [13] successfully integrated the adjoint method with an inverse method based on the meridional plane flow equation, which generated optimized design results for a single rotor and a counter-rotating compressor stage.

In present paper, first an inverse method based on the Euler equation is developed and verified. Then on the basis of the inverse formulation, the corresponding adjoint equation is derived and solved numerically. Finally the developed system is used to optimize a 1-1/2 turbine stage and numerical results demonstrate the ability of the method.

2. METHODS

2.1 Inverse method description

The inverse method adopted here is the same as the one proposed by Tiow et al. [2] and Dang et al.[14], which is based on the steady Reynolds Averaged Navier-Stokes (RANS) equation in the cylindrical coordinate system

$$\frac{\partial(F-F_v)}{\partial x}+\frac{\partial(G-v_gU-G_v)}{r\partial\theta}+\frac{\partial r(H-H_v)}{r\partial r}=S \qquad (1)$$

where U are conservative variables, u, v and w correspond to axial, circumferential and radial velocities respectively, F, G and H are convective fluxes in axial, circumferential and radial direction respectively, F_v, G_v and H_v correspond to viscous flux in each direction, S is the source term considering the centrifugal force in radial momentum equation, and v_g is the circumferential grid moving velocity.

The flow governing equations are spatially discretized by a cell-centered finite volume method framework with inviscid fluxes calculated by the central difference scheme coupled with a blended second- and fourth-order numerical dissipation (Jameson [15]). Time integration is achieved by using the four-stage Runge–Kutta method. The multigrid, local time-stepping techniques and implicit residual smoothing are employed to speed up the convergence of the solution process. Viscous effects are modeled by Baldwin-Lomax turbulence model and Denton's wall laws is used in the approximation of walls. In the direct mode of the flow solver, a slip boundary condition is applied to the end walls and the blade surfaces. At inlet, total pressure and temperature, together with flow angles in circumferential and radial directions, are specified. Static pressure is fixed at hub of the outlet, and the radial pressure distribution is determined by the simple radial equilibrium equation. For multi-row calculation, the mixing plane is used to transfer information between upstream and downstream blade rows.

The inverse mode of the flow solver differs from the direct mode mainly in the imposement of the wall boundary conditions on blade surfaces. In the direct mode, the boundary conditions on the flow surfaces explicitly enforce the flow tangency condition (or no flux condition). However in the inverse method, the imposed blade-surface boundary conditions play two roles. One is to satisfy the given aerodynamic parameters (say pressure loading in the present case), and the other is, combined with the camber line generation process, to ensure the flow tangency condition. Following Tiow et al. [2], the pressure loading is satisfied by letting

$$p^{\pm} = p_{bl} \pm \frac{1}{2}\Delta p \tag{2}$$

where the "+" and "-" represent parameters on the upper and lower blade surface, and the subscript "bl" denotes the blade averaged value, i.e.

$$p_{bl} = \frac{1}{2}\left(p^{+} + p^{-}\right) \tag{3}$$

and the Δ value is defined as

$$\Delta p = p^{+} - p^{-} \tag{4}$$

which is the blade pressure loading.

The flow tangency condition is ensured by

$$v^{\pm} = v_{bl} \pm \frac{1}{2}\Delta v \tag{5}$$

where Δv must be obtained through the flow tangency condition on the upper and lower blade surfaces, which is written as

$$\mathbf{W}^{\pm} \cdot \nabla \alpha^{\pm} = 0 \tag{6}$$

where $\alpha^{\pm} = \theta - \left(f \pm t/2\right)$ define the blade surfaces, and f is the camber line wrap angle, t is the tangential thickness. Their definition is shown in Fig. 1.

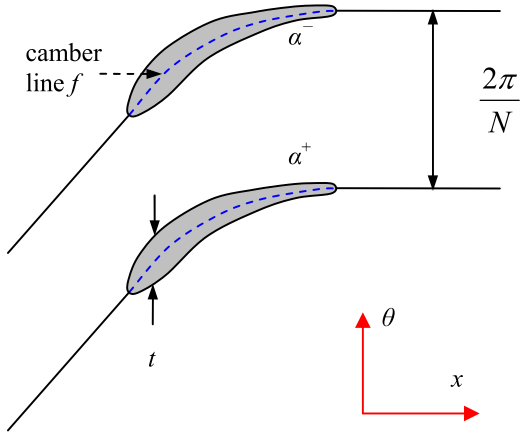

Fig.1 Camber line definition for inverse method

Expanding Eq. (6) and subtracting one from the other yields

$$\Delta v = \left(ru\frac{\partial \alpha}{\partial x} + rw\frac{\partial \alpha}{\partial r}\right)^{+} - \left(ru\frac{\partial \alpha}{\partial x} + rw\frac{\partial \alpha}{\partial r}\right)^{-} \tag{7}$$

The above inverse boundary condition will allow flux on blade surfaces during the iteration process, so it was termed as "transpiration boundary condition". Although the flow may not align with the blade surfaces in the time-marching process, when converged, the flow tangential condition will be ensured, and there will be no flux on the blade surfaces any more.

Adding the expanded expressions of Eq. (6), one has

$$u_{bl}\frac{\partial f}{\partial x} + w_{bl}\frac{\partial f}{\partial r} = \left(\frac{v_{bl}}{r} - \omega\right) - \frac{1}{4}\left(\Delta u\frac{\partial t}{\partial x} + \Delta w\frac{\partial t}{\partial r}\right) \tag{8}$$

which is used for the camber line generation.

The flow chart of the inverse method is illustrated in Fig. 2. The major steps are as follows:

(1) Input the specified pressure loading chordwise and spanwise, together with the stacking grid line j_{stk} and corresponding wrap angle f_{stk}. The inverse method design is then started from an initial guess of the blade shape.

(2) The discretised steady flow equation is marched for one step with the above derived inverse boundary condition on blade surfaces, while the boundary conditions at inlet, outlet and end wall are the same as those in the direct mode.

(3) A new camber line is obtained by solving the camber line generation equation with the velocities on the blade surfaces. Then the grid is regenerated by an algebraic method. The geometric metrics, such as area and volume, are aslo updated.

(4) Check the residual of the flow field and the camber line wrap angle to determine whether the inverse method is converged. If so, the computation results is post-processed; otherwise go to step (2) for the next design circle.

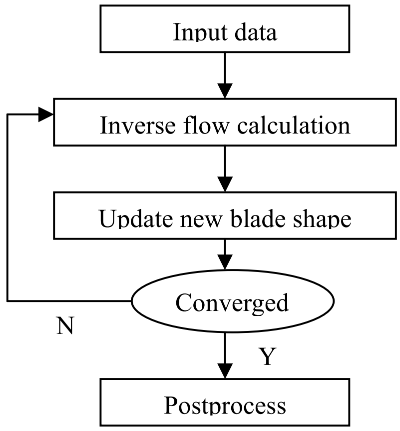

Fig.2 Flow chart of pure inverse method

2.2 Adjoint principle

Before deriving the adjoint equation for the inverse method based on the Euler equation, a concise illustration of adjoint principle is present here.

In general, the objective function I is a function of flow variable U and the design variable α, i.e.

$$I=I(U, \alpha) \tag{9}$$

and the flow governing equation relates to U and α by

$$R=R(U, \alpha) \tag{10}$$

A change of a design variable δα will yield changes in flow variables δU, the flow equation residual δR and the objective function δI. So the gradient of I to α could be written as

$$\frac{dI}{d\alpha}=\frac{\partial I}{\partial \alpha}+\frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha} \tag{11}$$

the term $\partial U/\partial \alpha$ is often named as flow sensitivity, whose computation is time-consuming, involving solving the flow governing equation lots of times when using the general finite difference method. The main idea of adjoint optimization is treating the flow governing equation as an additional constraint, and introducing a Lagrange multiplier, termed adjoint variable, to construct an augmented objective, then the problem become unconstrained. By selecting proper adjoint variables, the dependence of the objective function to the flow sensitivity will be eliminated, so the optimization process will be much more efficient. Linearizing the flow equation with the design variable will give

$$\frac{dR}{d\alpha}=\frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha}+\frac{\partial R}{\partial \alpha}=0 \tag{12}$$

The above equation, multiplied with adjoint variable λ^T, and subtracted from the objective function gradient, becomes

$$\frac{dI}{d\alpha}=\frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha}+\frac{\partial I}{\partial \alpha}-\lambda^T\left(\frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha}+\frac{\partial R}{\partial \alpha}\right) \tag{13}$$

collecting the like terms for $\partial U/\partial \alpha$ will get

$$\frac{dI}{d\alpha}=\left(\frac{\partial I}{\partial U}-\lambda^T\frac{\partial R}{\partial U}\right)\frac{\partial U}{\partial \alpha}+\frac{\partial I}{\partial \alpha}-\lambda^T\frac{\partial R}{\partial \alpha} \tag{14}$$

if we choose the adjoint variables by letting

$$\frac{\partial I}{\partial U}-\lambda^T\frac{\partial R}{\partial U}=0 \tag{15}$$

then the first term on the RHS of Eq. (14) vanishes, and the gradient of objective function will be independent of flow sensitivity. The final reduced gradient of objective function is

$$\frac{dI}{d\alpha}=\frac{\partial I}{\partial \alpha}-\lambda^T\frac{\partial R}{\partial \alpha} \tag{16}$$

which could be calculated without too much labor.

2.3 Adjoint equation for inverse method

Before derivation of the adjoint equation for the inverse method, it is helpful to clarify the relationship between the variables. In practical optimization, the objective function takes on a boundary integral expression, i.e.

$$I=\int_{\partial D} M ds \tag{17}$$

where ∂D are the boundaries of the integrated domain of the flow governing equation, the integrated function M is a function of flow variables M=M(U). In present application, because of using the inverse method, the design variable is chosen to be the pressure loading distribution factor α(x,r). The flow governing equation is expressed as R=R(U, α)=0, where R is a function of the flow variable U and the pressure loading. Also the camber line generation equation could be given as $\Theta=\Theta(U^+,U^-,f_x,f_r)=0$. At the same time, U=U(α), f=f(α).

The derivation process here basically follows Wang and He[10] and Giles and Pierce[19]. Although RANS is used, the derivation is based on the Euler equation, since the principal aim here is to obtain adjoint boundary conditions for inverse methods, and this could be done readily based on the Euler equation. The adjoint counterparts of the viscous terms based on RANS was derived by Wang and He [10].

First introducing the augmented objective function

$$\begin{aligned} I(U,f,\alpha)&=\int_{\partial D} Mds-\int_D \Lambda^T RdV \\ &-\int_{D_{b1}}\phi_1\Theta_1 ds-\int_{D_{b2}}\phi_2\Theta_2 ds \end{aligned} \tag{18}$$

where $\Lambda=(\lambda_1,\lambda_2,\lambda_3,\lambda_4,\lambda_5)^T$, ϕ_1 and ϕ_2 all are Lagrange multipliers, also termed adjoint variables here. The integration domain definition is shown in Fig. 3 and Fig. 4 for the flow equation and camber generation equation respectively. It should be noted that the integration domain for the camber line generation equation is defined on meridional plane, while the integration domain for the flow equation is defined in three dimensional spaces.

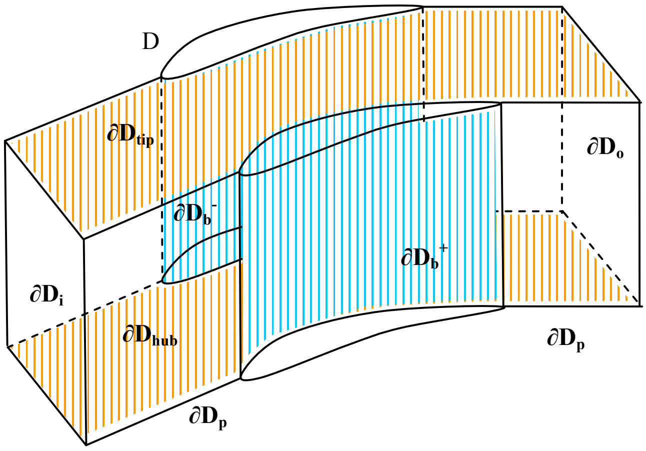

Fig.3 Integration domain for the flow equation

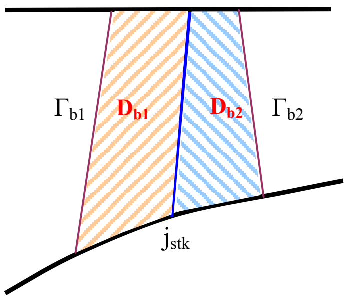

Fig.4 Integration domain for the camber line generation equation

For simplicity, the last two terms in Eq. (18) are combined into one term $-\int_{D_b} \phi \Theta ds$ in the derivation, but it should be always noted that the above equation is integrated in two different domains. Otherwise there will be problems for the adjoint boundary condition imposement.

Computing the gradient of augment objective function to α, one has

$$\frac{dI}{d\alpha} = \int_{\partial D} \frac{\partial M}{\partial U} \tilde{U} ds - \int_D \Lambda^T \frac{dR}{d\alpha} dV$$
$$- \int_{D_b} \phi \left(\frac{\partial \Theta}{\partial U^+} \tilde{U}^+ + \frac{\partial \Theta}{\partial U^-} \tilde{U}^- + \frac{\partial \Theta}{\partial f_x} \tilde{f}_x + \frac{\partial \Theta}{\partial f_r} \tilde{f}_r \right) ds \tag{19}$$

where $\tilde{U} = \frac{\partial U}{\partial \alpha}$ is the flow sensitivity, and $\tilde{f}_x = \frac{\partial f_x}{\partial \alpha}$, $\tilde{f}_r = \frac{\partial f_r}{\partial \alpha}$.

The linearized flow equation could be formulated as

$$\frac{dR}{d\alpha} = \frac{\partial \tilde{F}}{\partial x} + \frac{\partial \left(\tilde{G} - v_g \tilde{U} \right)}{r \partial \theta} + \frac{\partial r \tilde{H}}{r \partial r} - P\tilde{U} - g = 0 \tag{20}$$

where variables with tilde represent perturbed values of corresponding flow variables, and the term "g" can be considered to be the geometric source term, corresponding to $\partial R / \partial \alpha$ in Eq. (12). Categorizing the above equation into two parts, the flow sensitivity part

$$\frac{\partial R}{\partial U} \tilde{U} = \frac{\partial \tilde{F}}{\partial x} + \frac{\partial \left(\tilde{G} - v_g \tilde{U} \right)}{r \partial \theta} + \frac{\partial r \tilde{H}}{r \partial r} - P\tilde{U} \tag{21}$$

and the geometric source term –g, then substituting the above equation into Eq. (19), we will obtain

$$\frac{dI}{d\alpha} = \int_D \Lambda^T g dV + \int_{\partial D} \frac{\partial M}{\partial U} \tilde{U} ds$$
$$- \int_D \Lambda^T \left[\frac{\partial \tilde{F}}{\partial x} + \frac{\partial \left(\tilde{G} - v_g \tilde{U} \right)}{r \partial \theta} + \frac{\partial r \tilde{H}}{r \partial r} - P\tilde{U} \right] dV \tag{22}$$
$$- \int_{D_b} \phi \left(\frac{\partial \Theta}{\partial U^+} \tilde{U}^+ + \frac{\partial \Theta}{\partial U^-} \tilde{U}^- + \frac{\partial \Theta}{\partial f_x} \tilde{f}_x + \frac{\partial \Theta}{\partial f_r} \tilde{f}_r \right) ds$$

The first term on the RHS of Eq. (22) is the final gradient, and the other terms on the RHS constitute the adjoint equation and its boundary condition. Some terms in the third term on the RHS could be further manipulated by integration by parts:

$$- \int_D \Lambda^T \left[\frac{\partial \tilde{F}}{\partial x} + \frac{\partial \left(\tilde{G} - v_g \tilde{U} \right)}{r \partial \theta} + \frac{\partial r \tilde{H}}{r \partial r} \right] dV$$
$$= - \int_{\partial D} \Lambda^T \left[A n_x + \left(B - v_g I \right) n_\theta + C n_r \right] \tilde{U} ds \tag{23}$$
$$+ \int_D \left[\frac{\partial \Lambda^T}{\partial x} A + \frac{\partial \Lambda^T}{r \partial \theta} \left(B - v_g I \right) + \frac{\partial \Lambda^T}{\partial r} C \right] \tilde{U} dV$$

For the camber line generation equation,

$$\frac{\partial \Theta}{\partial U} \tilde{U} = \frac{\partial \Theta}{\partial W} \frac{\partial W}{\partial U} \tilde{U} = \frac{\partial \Theta}{\partial W} M^{-1} \tilde{U} \tag{24}$$

where W is the primitive variable vector $W = (\rho, u, v, w, p)^T$, and $M = \frac{\partial U}{\partial W}$, $M^{-1} = \frac{\partial W}{\partial U}$. According to the camber line generation equation, we have

$$\frac{\partial \Theta}{\partial W^\pm} = \left(0, \frac{1}{2} f_x \pm \frac{1}{4} \Delta u t_x, -\frac{1}{2r}, \frac{1}{2} f_r \pm \frac{1}{4} t_r, 0 \right) = J^\pm \tag{25}$$

then

$$- \int_{D_b} \phi \frac{\partial \Theta}{\partial U^\pm} \tilde{U}^\pm ds = - \int_{D_b} \phi \left(J M^{-1} \tilde{U} \right)^\pm ds \tag{26}$$

because the meridional surface corresponds the circumferential projection of the blade surfaces, so

$$- \int_{D_b} \phi \frac{\partial \Theta}{\partial U^\pm} \tilde{U}^\pm ds = - \int_{\partial D_b^\pm} \phi \left(J n_\theta M^{-1} \tilde{U} \right)^\pm ds \tag{27}$$

Now consider the derivatives of Θ with respect to wrap angle,

$$\frac{\partial \Theta}{\partial f_x} = u_{bl} \tag{28a}$$

$$\frac{\partial \Theta}{\partial f_r} = w_{bl} \tag{28b}$$

and

$$\phi \frac{\partial \Theta}{\partial f_x} \frac{\partial \tilde{f}}{\partial x} = \frac{\partial}{\partial x} \left(\phi u_{bl} \tilde{f} \right) - \frac{\partial}{\partial x} \left(\phi u_{bl} \right) \tilde{f} \tag{29}$$

then

$$-\int_{D_b}\phi\frac{\partial\Theta}{\partial f_x}\tilde{f}_x dxdr = -\int_{\Gamma_b}\phi u_{bl}\tilde{f}n_x dl + \int_{D_b}\frac{\partial}{\partial x}(\phi u_{bl})\tilde{f}dxdr \tag{30}$$

in the same way

$$-\int_{D_b}\phi\frac{\partial\Theta}{\partial f_r}\tilde{f}_r dxdr = -\int_{\Gamma_b}\phi w_{bl}\tilde{f}n_r dl + \int_{D_b}\frac{\partial}{\partial r}(\phi w_{bl})\tilde{f}dxdr \tag{31}$$

so

$$-\int_{D_b}\phi\left(\frac{\partial\Theta}{\partial f_x}\tilde{f}_x + \frac{\partial\Theta}{\partial f_r}\tilde{f}_r\right)ds = -\int_{\Gamma_b}\phi\left[u_{bl}n_x + w_{bl}n_r\right]\tilde{f}dl + \int_{D_b}\left[\frac{\partial}{\partial x}(\phi u_{bl}) + \frac{\partial}{\partial r}(\phi w_{bl})\right]\tilde{f}dxdr \tag{32}$$

Γ_b is the boundary of D_b, which means the leading edge, trailing edge, hub and tip region in the bladed area.

Substituting Eq. (23), (27) and (32) into Eq. (22), and eliminating the coefficient of $\tilde{U}$ on domain D, one obtains the adjoint equation for the Euler based inverse method.

$$\frac{\partial\Lambda^T}{\partial x}A + \frac{\partial\Lambda^T}{r\partial\theta}\left(B - v_g I\right) + \frac{\partial\Lambda^T}{\partial r}C + P = 0 \tag{33}$$

after transposition, the expression becomes

$$A^T\frac{\partial\Lambda}{\partial x} + \left(B - v_g I\right)^T\frac{\partial\Lambda}{r\partial\theta} + C^T\frac{\partial\Lambda}{\partial r} + P^T = 0 \tag{34}$$

which is the same as the general Euler based direct adjoint equation.

In order to eliminate the sensitivity $\tilde{f}$,

$$\frac{\partial}{\partial x}(\phi u_{bl}) + \frac{\partial}{\partial r}(\phi w_{bl}) = 0 \tag{35}$$

This equation is named camber line adjoint equation.

2.4 Inverse boundary condition for the adjoint equation

2.4.1 General adjoint boundary conditions

As shown in Fig. 3, the boundary of field integration can be classified into 5 groups, namely the inlet boundary ∂D_i, the outlet boundary ∂D_o, the end walls ∂D_{hub} and ∂D_{tip}, the blade surfaces ∂D_b and the periodic boundary ∂D_p.

For inlet and outlet boundary ∂D_{io}, the adjoint boundary condition is determined by

$$\left\{\frac{\partial M}{\partial U} - \Lambda^T\left[An_x + \left(B - v_g I\right)n_\theta + Cn_r\right]\right\}\tilde{U} = 0 \tag{36}$$

which is also the same as the Euler based direct adjoint equation.

For the end wall surfaces ∂D_{hub} and ∂D_{tip}, considering the normal velocities of the walls are zero, the adjoint boundary conditions are given by (see Wang and He [10] for more detail)

$$\lambda_2 n_x + \lambda_3 rn_\theta + \lambda_4 n_r + \lambda_5 v_g n_\theta = \partial M/\partial p \tag{37}$$

For periodic boundary ∂D_p, the direct copy from the corresponding internal points is used.

The boundary condition on the blade surfaces is discussed in section 2.4.2.

In order to exchange information between different blade rows, Wang and He[10] developed an adjoint mixing plane treatment for multi-row turobmachinery simulation, which is found to be conservative and non-reflecting, so the same technique is used here. The basic process is summarized as follows:

(1) Solve the flux averaged adjoint variables $\hat{\lambda}_i$ (i=1,…,5), i.e. the mixed out variables termed by Wang and He, on both side of the interface;

(2) Compute primitive perturbation of the mixed out variables $\tilde{\lambda}_i = \hat{\lambda}_{i,2} - \hat{\lambda}_{i,1}$, and these perturbations will be used as global perturbations.

(3) Transform the primitive perturbations to the characteristic perturbations using the eigen matrix of the Jacobi matrix.

(4) Correct the perturbation on both sides of the interface according to wave propagation direction. If the wave is incoming, the global characteristic perturbation is used; else if the wave is outgoing, then local characteristic perturbation is used.

(5) Convert the updated characteristic perturbations into primitive perturbations, and add them to the original adjoint variables.

2.4.2 Inverse adjoint boundary conditions on the blade surfaces

In current application, the most important part lies in a compatible boundary condition for the inverse boundary conditions on the blade surfaces. In the inverse flow solver, one boundary condition is specified on each blade surface and the other four boundary conditions are determined by the interior flowfield, so for adjoint boundary conditions, there will be four boundary conditions to be specified, and one determined from internal flowfield.

For the blade surfaces ∂D_b in the inverse method, the boundary conditions to be satisfied on the blade surfaces are

$$\int_{\partial D_b^+}\left\{\frac{\partial M}{\partial U} - \Lambda^T\left[An_x + \left(B - v_g I\right)n_\theta + Cn_r\right] - \phi JM^{-1}n_\theta\right\}\tilde{U}ds + \int_{\partial D_b^-}\left\{\frac{\partial M}{\partial U} - \Lambda^T\left[An_x + \left(B - v_g I\right)n_\theta + Cn_r\right] - \phi JM^{-1}n_\theta\right\}\tilde{U}ds = 0 \tag{38}$$

Supposed that the objective function is only applied at inlet and outlet, so $\frac{\partial M}{\partial U}$ is zero on the blade surfaces, then Eq. (38) could be rewritten as

$$\left(\Lambda^T\tilde{F}_n + \phi Jn_\theta\tilde{W}\right)^+ + \left(\Lambda^T\tilde{F}_n + \phi Jn_\theta\tilde{W}\right)^- = 0 \tag{39}$$

expressing F_n by the primitive variables, one obtains

$$\left[\left(\Lambda^T A_n M + \phi Jn_\theta\right)\tilde{W}\right]^+ + \left[\left(\Lambda^T A_n M + \phi Jn_\theta\right)\tilde{W}\right]^- = 0 \tag{40}$$

the expression of F_n and A_n are listed in the appendix. Defining

$$X = (X_1, X_2, X_3, X_4, X_5) = \Lambda A_n M + \phi J n_\theta \quad (41)$$

then the above equation could be re-written as

$$\left(X_1^+, X_2^+, X_3^+, X_4^+, X_5^+\right)\left[(\tilde{\rho}, \tilde{u}, \tilde{v}, \tilde{w}, \tilde{p})^+\right]^T + \left(X_1^-, X_2^-, X_3^-, X_4^-, X_5^-\right)\left[(\tilde{\rho}, \tilde{u}, \tilde{v}, \tilde{w}, \tilde{p})^-\right]^T = 0 \quad (42)$$

Expanding it generates

$$X_1^+\tilde{\rho}^+ + X_2^+\tilde{u}^+ + X_3^+\tilde{v}^+ + X_4^+\tilde{w}^+ + X_5^+\tilde{p}^+ + X_1^-\tilde{\rho}^- + X_2^-\tilde{u}^- + X_3^-\tilde{v}^- + X_4^-\tilde{w}^- + X_5^-\tilde{p}^- = 0 \quad (43)$$

The inverse boundary conditions in the inverse flow solver equate to

$$\tilde{p}^+ - \tilde{p}^- = 0 \quad (44)$$

$$\tilde{v}^+ - \tilde{v}^- = r^+\left(f_x^+\tilde{u}^+ + f_r^+\tilde{w}^+\right) - r^-\left(f_x^-\tilde{u}^- + f_r^-\tilde{w}^-\right) \quad (45)$$

substituting the above two equation into the Eq. (43) gives

$$X_1^+\tilde{\rho}^+ + \left(X_2^+ + r^+ f_x^+ X_3^+\right)\tilde{u}^+ + \left(X_4^+ + r^+ f_r^+ X_3^+\right)\tilde{w}^+ + X_1^-\tilde{\rho}^- + \left(X_2^- - r^- f_x^- X_3^+\right)\tilde{u}^- + \left(X_4^- - r^- f_r^- X_3^+\right)\tilde{w}^- + \left(X_3^- - X_3^+\right)\tilde{v}^- + \left(X_5^- - X_5^+\right)\tilde{p}^- = 0 \quad (46)$$

In order to eliminate the perturbations of p and circumferntial velocity v, it is sufficient and necessary to let

$$X_5^+ + X_5^- = 0 \quad (47)$$

$$X_3^+ + X_3^- = 0 \quad (48)$$

also setting

$$X_1^+ = 0 \quad (49)$$

$$X_2^+ + r^+ f_x^+ X_3^+ = 0 \quad (50)$$

$$X_4^+ + r^+ f_r^+ X_3^+ = 0 \quad (51)$$

to eliminate density, axial and radial velocity perturbations on the upper surface, and setting

$$X_1^- = 0 \quad (52)$$

$$X_2^- - r^- f_x^- X_3^- = 0 \quad (53)$$

$$X_4^- - r^- f_r^- X_3^- = 0 \quad (54)$$

to eliminate density, axial and radial velocity perturbations on the lower surface. The specific numerical implementation of this boundary condition is given in the appendix.

The adjoint equation corresponding to the camber line generation equation is determined by

$$\phi\left[u_{bl} n_x + w_{bl} n_r\right]\tilde{f} = 0 \quad (55)$$

When deriving Eq. (55), the zero normal velocity condition along the hub and tip is used. For a general stacking lime, which is in aligned with j=j_{stk} grid line, $\phi_1 = 0$ should be applied at the leading edge, and $\phi_2 = 0$ at the trailing edge.

2.5 Design optimization

The objective function used in present study is the entropy generation rate,

$$I = \Delta s/\Delta s_0 + \sigma_1\left(m/m_0 - 1\right)^2 + \sigma_2\left(\pi/\pi_0\right)^2 \quad (56)$$

where the subscript "0" denotes the baseline value and σ are weight factors. For the application in present study, both weight factors are set to 100. The same objective function is used by Wang and He [10]. The pressure loading perturbation is parameterized using Hicks-Henne function. After getting gradients by solving the adjoint equations, a deepest decent method is used to give the change of design variables. Actually the gradients calculation and optimization process is termed as adjoint method in present paper.

A typical flowchart of the optimization application is illustrated in the Fig. 5. The problem is set up by supplying the design condition, the tangential thickness and an initial blade shape, also an initial aerodynamic parameter distribution is needed to start the calculation, which could be obtained by former direct calculation. After solving the inverse flow equation and adjoint equation, the gradients and the changes of the design variables are obtained. Then a check is made to see whether the design variables change is within some pre-specified convergence criteria. If satisfied, then the computation will stop. Otherwise, a new blade shape is computed and problem is reset up to start the next round computation.

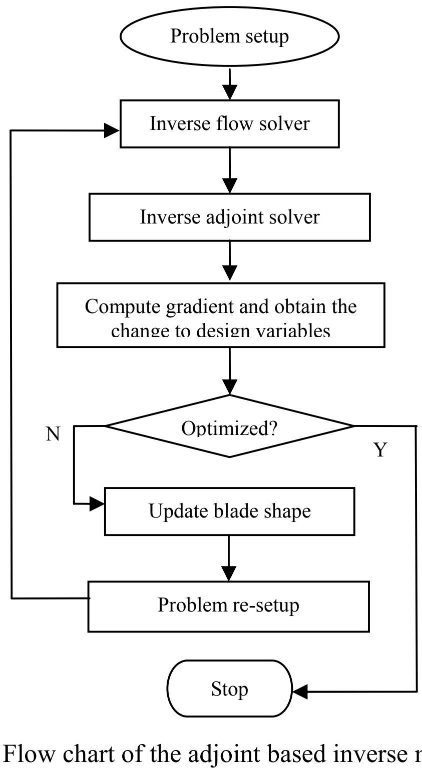

Fig.5 Flow chart of the adjoint based inverse method

3. RESULTS AND DISCUSSIONS

3.1 Aachen 1-1/2 stage turbine

The Aachen turbine, illustrated in Fig. 6, comprises of an IGV, a rotor and a stator, operating in an axial configuration at modest Mach numbers of ~0.5. The low aspect ratio of the blades, and the constant tip and hub end wall contours enhance strong secondary flow phenomena. Both the 36 vanes and 41 blades are cylindrical and untwisted. The rotor is unshrouded with a tip clearance of 0.4mm, but in the simulation, the tip clearance does not be modeled.

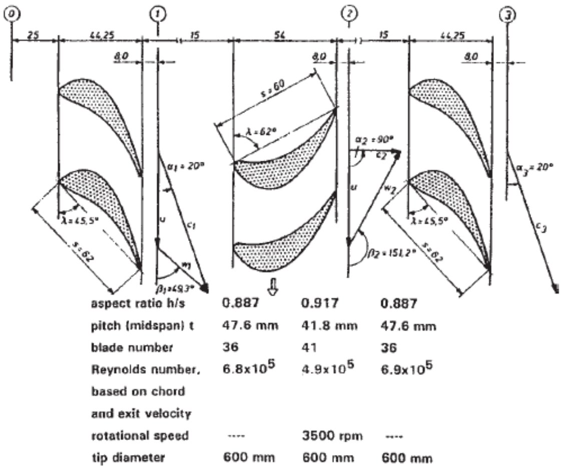

Fig. 6 Schematic of the Aachen turbine stage used in the study

The IGV of the turbine is used to validate of the inverse method, and the rotor and stator of the stage are redesigned using the presnet adjoint method based multi-row inverse method.

3.2 Validation of the inverse method

In order to validate the developed inverse method, the IGV of the turbine stage was recovered starting from a different geometry, with the target pressure loading and tangential thickness specified.

Figure 7 compares the initial blade profiles, the redesigned and the target ones at mid span position. The target and redesigned pressure distributions are shown in Fig.8. The consistence of comparison is very good except some subtle difference at the trailing edges due to the large grid skewness there.

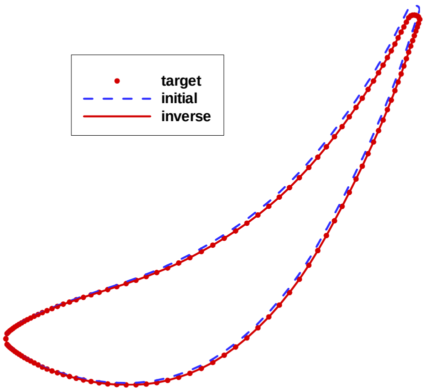

Fig. 7 Initial, inverse and target blade profile comparison

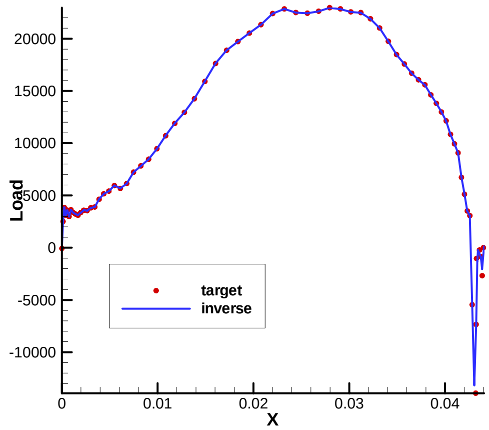

Fig. 8 Target and inversely computed pressure comparison

3.3 Redesign using the developed adjoint method based multi-row inverse method

The rotor and stator of the turbine stage are optimized simultaneously to verify the developed method, and the IGV is kept unchanged. A 105×41×41 grid density for each row is used throughout the calculation. Figure 9 shows the computational grid on meridional plane and quasi-stream surface. 11 control points in chordwise and 11 in radial direction are distributed to parameterize the pressure loading distribution.

After 24 optimization cycles, the objective function decrease about 10%. The objective function evolution history is shown in Fig. 10. The optimized performance is compared with the baseline in Table 1. According to the computation, the efficiency was increased about 1 point, while keeping the mass flow rate and pressure ratio change less than 0.6%.

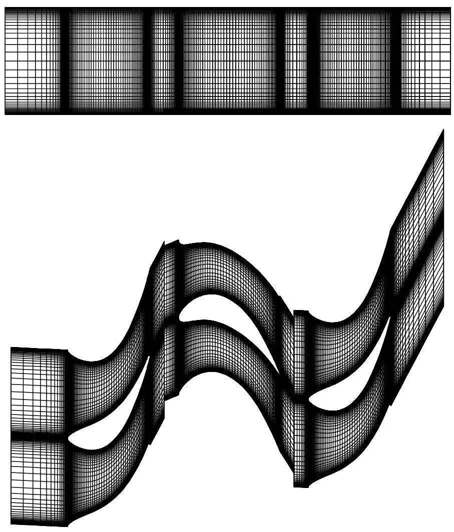
Fig. 9 Meridional and quasi-stream surface computational grid for Aachen turbine stage

Table 1. Original and optimized performance of the Aachen turbine stage

	mass flow rate	total pressure ratio	isentropic entropy
original	8.2214 kg/s	0.8283	83.26%
optimized	8.2059 kg/s	0.8233	84.27%

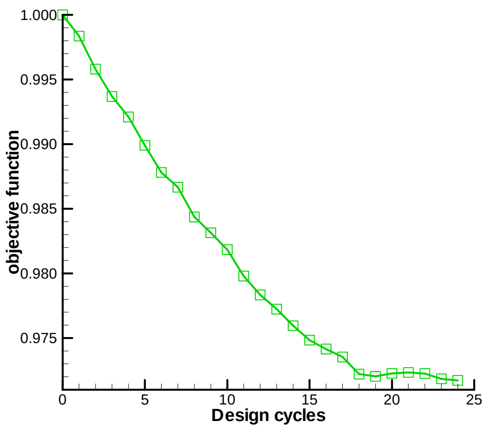

Fig. 10 Objective function evolution history

Figures 11-13 show the geometric changes of the optimized from the original blade profiles at the 10%, 50% and 90% span respectively. All the original blades are ruled blades, which means the blades can be defined by two sections at hub and tip respectively. It is straightforward to see that, after optimization, the geometry change is nonlinear between hub and tip. For both the rotor and the stator, the largest geometry change happens at the mid-span section. The geometric changes of blades make the passage curvature decrease at the rear part.

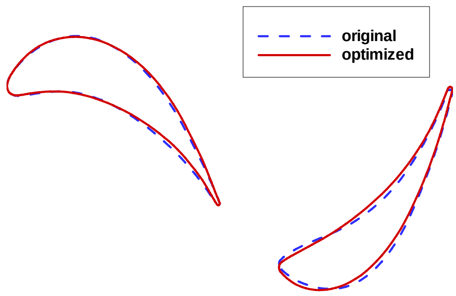

Fig.11 Geometry change of the original blade profile and the optimized one at 10% span position from hub

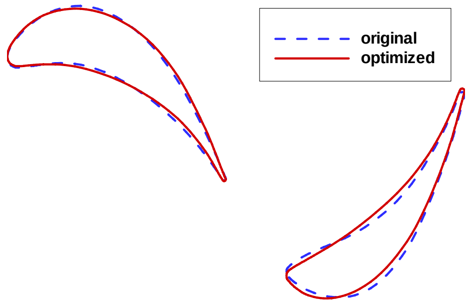

Fig.12 Geometry change of the original blade profile and the optimized one at 50% span position from hub

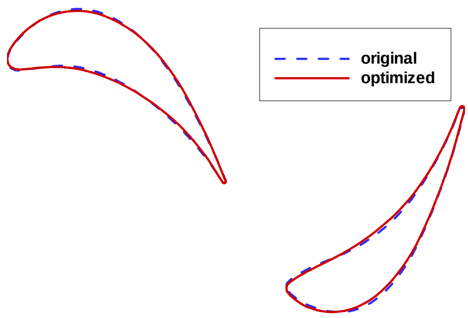

Fig.13 Geometry change of the original blade profile and the optimized one at 90% span position from hub

Figures 14-16 compare the flowfiled on quasi-stream surface. Examining those figures, it it easy to discover that there is elevated Mach number distribution in the optimized blade passage, which is consistent with the geometric change. It is believed that the efficiency increase is due to the higher passage velocity which improves the boundary layer behavior. Also, for the stator, there is a little change of incidence, which makes the rotor and stator matching better than the original.

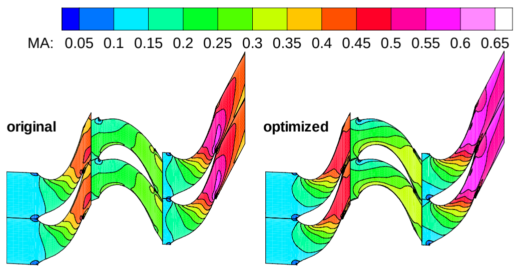

Fig.14 Relative mach contour at 10% span position starting from the hub

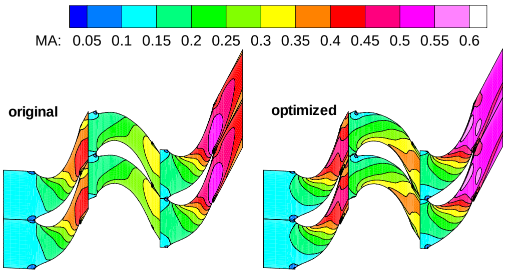

Fig.15 Relative mach contour at 50% span position starting from the hub

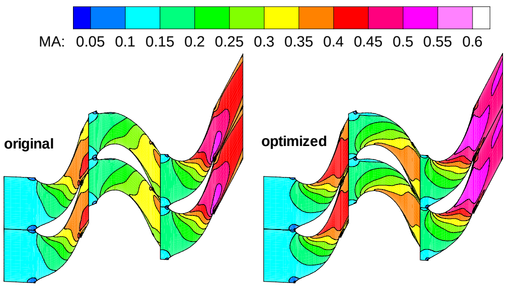

Fig.16 Relative mach contour at 90% span position starting from the hub

3.4 Comparison between the inverse method, adjoint based direct optimization method, and adjoint based inverse method

At present, there are generally three kinds of methods to realize three-dimensional turbomachinery aerodynamic designs. The first is the optimization based on using the CFD in direct mode which is called direct optimization method (DOM) for short. The second is the pure inverse method (PIM) which utilize flow tangency condition and re-adjusting the camber line to satisfy the provided loading. The third is the optimization method based on inverse method, such as one developed in present paper, which is termed Inverse Optimization Method (IOM) for short. The following table compares the advantages and disadvantages among those methods. It is easy to conclude that, the DOM and IOM are counterparts, which basically share the same properties. The exception is that DOM deals with geometric design variables and it is easy to apply geometric constraints, while IOM takes aerodynamic parameters as design variables and can impose aerodynamic constraints easily. Table 2 compares the pure inverse method, the adjoint based direct optimization method and the adjoint based inverse optimization method. It should be pointed out that, because of solution camber line generation equation and inverse boundary condition imposement for inverse flow solution and adjoint equation solution, IOM takes about 30% longer time for one solution pass than the DOM.

Table 2. Comparison between the pure inverse method (PIM), the adjoint based direct optimization method (DOM) and inverse optimization method (IOM)

	PIM	DOM	IOM
turn-around time	a little longer than a direct analysis	hundreds of cycles of a conventional direct analysis	hundreds of cycles of a conventional direct analysis
multi-row capability	difficult to handle multi-row problem	easy to handle multi-row problem	easy to handle multi-row problem
experience dependence	strong experience dependence	weak experience dependence	weak experience dependence
optimized results	improved result, but not guarantee to be optimized	yes	yes
geometric constraints	with limited ability of applying geometric constrains	easy to apply geometric constraints	with limited ability of applying geometric constraints
aerodynamic constraint	easily apply aerodynamic constraints	with limited ability of applying aerodynamic constraints	easy to apply aerodynamic constraints
automation	human intervention may be needed	full automation,	full automation

CONCLUSIONS

In present paper, an inverse method is first developed and validated using the IGV of the Aachen 1-1/2 turbine stage, then the inverse method is extended to multi-row environment by combining with the adjoint optimization technique. Flow and adjoint mixing plane treatment is adopted to transfer information between upstream and downstream of the mixing plane. Inverse boundary conditions imposed on the blade surfaces for adjoint equation are derived and implemented. The rotor and stator of the Aachen turbine stage are redesigned simultaneously to verify the developed multi-row inverse method. The entropy generation rate is selected as the objective function, and the pressure loading distribution perturbation is parameterized using Hicks-Henne function. Numerical results demonstrate that the current method possesses lots of merits, such as no experience dependence, giving optimized results and considering the aerodynamic matching between blade rows.

ACKNOWLEDGMENTS

The authors would like to thank the National Natural Science Foundation of China (No. 51076131) for supporting the present research.

REFERENCES

[1] Denton, J. D. and Daws, W. N., 1999, "Computational Fluid Dynamics for Turbomachinery Design", Proc Instn Mech Engrs, 213:107-124.
[2] Tiow, W. T. and Zangeneh, M., 2002, "Application of a Three-

Dimensional Viscous Transonic Inverse Method to NASA Rotor 67", Journal of Power and Energy, 216: 243-255.
[3] van Rooij, M. P. C., Dang, T. Q. and Larosiliere, L. M., 2007, "Improving Aerodynamic Matching of Axial Compressor Blading Using a Three-Dimensional Multistage Inverse Design Method", Journal of Turbomachinery, 129:108-118.
[4] Roidl, B. and Ghaly, W., 2010, "Redesign of a Low Speed Turbine Stage Using a New Viscous Inverse Design Method", Journal of Turbomachinery, 133:011009.
[5] Tiow, W. T., Yiu, K. F. C. and Zangeneh, 2002, "Application of Simulated Annealing to Inverse Design of Transonic Turbomachinery Cascades", Proc Instn Mech Engrs, 216:59-73.
[6] Bonaiuti, D. and Zangeneh, M., 2009, "On the Coupling of the Inverse Design and Optimization Techniques for the Multiobjective, Multipoint Design of Turbomachinery Blades", Journal of Turbomachinery, 131:021014.
[7] Jameson, A., 2003, "Aerodynamic Shape Optimization Using the Adjoint Method", Lectures at the Von Karman Institute, Brussels.
[8] Yang, S., Wu, H., Liu, F., and Tsai, H., 2003, "Aerodynamic Design of Cascades by Using an Adjoint Equation Method," AIAA Paper No. 03-1068.
[9] Wu, H., Liu, F., and Tsai, H., 2005, "Aerodynamic Design of Turbine Blades Using an Adjoint Equation Method," AIAA Paper No. 05-1006.
[10] Wang, D. X. and He, L., 2010, "Adjoint Aerodynamic Design Optimization for Blades in Multistage Turbomachines—Part I: Methodology and Verification", Journal of Turbomachinery, 132:021011.
[11] Wang, D. X. and He, L., 2010, "Adjoint Aerodynamic Design Optimization for Blades in Multistage Turbomachines—Part II: Validation and Application", Journal of Turbomachinery, 132:021012.
[12] Wang, D. X. and Li, Y. S., 2010, "3D Direct and Inverse Design Using NS Equations and the Adjoint Method for Turbine Blades", ASME paper GT 2010-22049.
[13] Iollo, A., Ferlauto, M. and Zannetti, L., 2001, "An Aerodynamic Optimization Method Based on the Inverse Problem Adjoint Equations", Journal of Computational Physics, 173:87-115.
[14] Dang, T., Damle, S. and Qiu X., 2000, "Euler-Based Inverse Method for Turbomachinery Blades, Part 2: Three-Dimensional Flows", AIAA Journal, 38(11):2007-2010.
[15] Jameson, A., Schmidt, W. and Turkel, E., 1981, "Numerical Solution of the Euler Equations by Finite Volume Method Using Runger-Kutta Time Stepping Schemes", AIAA paper 81-1259.
[16] Papadimitriou, D. I., and Giannakougou, K. C., 2007, "A Continuous Adjoint Method with Objective Function Derivatives Based on Boundary Integrals, for Inviscid and Viscous Flows", Computers & Fluids, 36:325-341.
[17] Li, Y. C. and Feng, Z. P., 2008, "Three-Dimensional Aerodynamic Design of Turbine Blades Using the Adjoint Method", ASME GT2008-51225.
[18] Luo, J. Q., Xiong, J. T. and Liu, F., 2010, "Three-Dimensional Aerodynamic Design Optimization of a Turbine Blade by Using an adjoint Method", Journal of Turbomachinery, 133: 011026.
[19] Giles, M. B. and Pierce, N. A., 1997, "Adjoint Equations in CFD Duality, Boundary Conditions and Solution Behavior", AIAA 97-1850.

APPENDIX

NUMERICAL IMPLEMENTATION OF INVERSE BOUNDARY CONDITION FOR ADJOINT EQUATION

The normal flux is defined by

$$F_n = \begin{pmatrix} \rho Q \\ \rho u Q + p n_x \\ r(\rho v Q + p n_\theta) \\ \rho w Q + p n_r \\ \rho H Q \end{pmatrix} - v_g U n_\theta \tag{A.1}$$

According to

$$\frac{\partial F_n}{\partial W} = \begin{pmatrix} Q & \rho n_x & \rho n_\theta & \rho n_r & 0 \\ uQ & \rho(Q + u n_x) & \rho u n_\theta & \rho u n_r & n_x \\ rvQ & r\rho v n_x & r\rho(Q + v n_\theta) & r\rho v n_r & r n_\theta \\ wQ & \rho w n_x & \rho w n_\theta & \rho(Q + w n_r) & n_r \\ qQ & \rho H n_x + \rho u Q & \rho H n_\theta + \rho v Q & \rho H n_r + \rho w Q & \frac{\gamma}{\gamma - 1} Q \end{pmatrix} - v_g n_\theta M \tag{A.2}$$

where

$$Q = u n_x + v n_\theta + w n_r \tag{A.3}$$

and M is defined by

$$M = \frac{\partial U}{\partial W} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ u & \rho & 0 & 0 & 0 \\ vr & 0 & \rho r & 0 & 0 \\ w & 0 & 0 & \rho & 0 \\ q & \rho u & \rho v & \rho w & \frac{1}{\gamma - 1} \end{pmatrix} \tag{A.4}$$

where $q = \frac{1}{2}\left(u^2 + v^2 + w^2\right)$. Substituting Eq. (A.2) and (A.3) into Eq. (41) will give

$$\begin{aligned} X_1 &= (Q - v_g n_\theta)(\lambda_1 + u\lambda_2 + rv\lambda_3 + w\lambda_4 + q\lambda_5) \\ X_2^\pm &= \rho n_x(\lambda_1 + u\lambda_2 + rv\lambda_3 + w\lambda_4 + H\lambda_5) \\ &+ \rho(Q - v_g n_\theta)(\lambda_2 + u\lambda_5) - \left(\frac{1}{2} f_x \pm \frac{1}{4}\Delta u t_x\right) n_\theta \phi \\ X_3 &= \rho n_\theta(\lambda_1 + u\lambda_2 + rv\lambda_3 + w\lambda_4 + H\lambda_5) \\ &+ \rho(Q - v_g n_\theta)(r\lambda_3 + v\lambda_5) - \frac{1}{2r} n_\theta \phi \\ X_4^\pm &= \rho n_r(\lambda_1 + u\lambda_2 + rv\lambda_3 + w\lambda_4 + H\lambda_5) \\ &+ \rho(Q - v_g n_\theta)(\lambda_4 + w\lambda_5) - \left(\frac{1}{2} f_r \pm \frac{1}{4} t_r\right) n_\theta \phi \\ X_5 &= n_x\lambda_2 + r n_\theta \lambda_3 + n_r \lambda_4 + \frac{1}{\gamma - 1}(\gamma Q - v_g n_\theta)\lambda_5 \end{aligned} \tag{A.5}$$

For the inverse adjoint boundary conditions implementation, λ_5 is extrapolated for both upper and lower surfaces, then Eq. (47)-(54) construct a linear system for the unknowns ($\lambda_1^\pm$, $\lambda_2^\pm$, $\lambda_3^\pm$ and $\lambda_4^\pm$), which can be solved by direct matrix inversion method.

