Turbomachinery aerodynamic and aeromechanic design optimization using the adjoint method

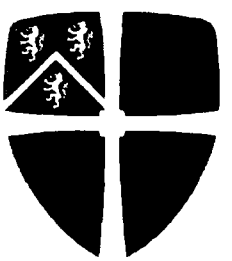
Dingxi Wang

TURBOMACHINERY AERODYNAMIC AND AEROMECHANIC DESIGN OPTIMIZATION USING THE ADJOINT METHOD

Dingxi Wang

School of Engineering

University of Durham



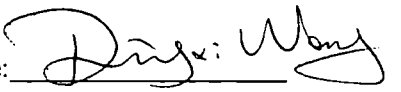
A thesis submitted to University of Durham

for the degree of Doctor of Philosophy

July 2008

Declaration

I declare that no material presented within this thesis has been submitted towards a degree or qualification at this or any other university

Signiture: 

date: 14/08/2008

Acknowledgements

I want to express my gratitude to my supervisor Prof. Li He for his offering me this studentship, his patient guidance, his generosity in financial and spiritual support, and his advice beyond academic issues. I would like to thank Prof. Roger Crouch for acting as my second supervisor during the last several months of this study, being my first year examiner and proofreading my thesis. Thanks go to Dr. Haidong Li for his help during the early stage of this research work and his advice via loads of emails during later stage of this research work. Dr. Xiuquan Huang deserves my heart-felt gratitude for his enormous help in Durham. Thanks also go to Mrs. Ying Dong for her friendship. Thank Mrs. Vera Menshikova and Mr. Orhan Aybay for their harmonious accompany as officemates. Efforts of proofreading the thesis by Mr. Jinguang Yang and Mr. Richard Williams are greatly appreciated. Siemens Industrial Turbomachinery Ltd. Lincoln has been providing funding for the work. Discussions with and suggestions from Dr. Yansheng Li, Dr. Tie Chen and Mr. Roger Wells at Siemens are constructive.

Last, but not least, thanks go to my parents and my wife for their unconditional love.

Abstract

The thesis documents the investigation of the application of the adjoint method to turbomachinery blading design optimization, with emphasis on blading aerodynamic design optimization in a multi-bladerow environment and concurrent blading aerodynamic and aeromechanic design optimization for a single bladerow.

Based on the nonlinear flow equations, a steady adjoint system has been developed using the continuous adjoint approach. The capability of the conventional adjoint system has been augmented by the introduction of an adjoint mixing-plane treatment. This treatment is a counterpart of the flow mixing-plane treatment, enabling the steady adjoint equations to be solved in multi-bladerow computational domains. This allows turbomachinery blades to be optimised to enhance their aerodynamic performance in a multi-bladerow environment with matching between adjacent bladerows dealt with in a timely manner.

The Nonlinear Harmonic Phase Solution method, a neat frequency domain method catered specifically for turbomachinery aeromechanics prediction, has been chosen to integrate with the adjoint method to calculate objective function sensitivities efficiently for concurrent aeromechanic and aerodynamic design optimization for single row turbomachinery blades. The Nonlinear Harmonic Phase Solution method, unlike the time-linearized methods, solves the unsteady flow equations at two or three carefully selected phases of a period of unsteadiness. This approach not only can conveniently turn a steady flow solver to one solving the unsteady flow equations efficiently, but also provides a good basis on which the corresponding adjoint system can be formulated and solved in a similar manner by extending a steady adjoint system.

In order to resolve the issue of having a good blading performance over a whole operating range at a given operation speed, a multi-operating-point design optimization is implemented by formulating an objective function of a weighted sum of performance at more than one operating point.

Keywords: design optimization, aerodynamics, aeromechanics, adjoint method, gradient, objective function, blade shape

Contents

List of Figures

List of Tables

Nomenclature

Roman Symbols

$-$	denote averaged value
$\bar{P}^*$	mass averaged total pressure
$\bar{s}$	mass averaged entropy
$\Delta\bar{s}$	entropy generation rate
$\dot{m}$	mass flow rate
μ_l	laminar viscosity
μ_t	turbulent viscosity
C_p	pressure coefficient
c_p	specific heat capacity at constant pressure
c_v	specific heat capacity at constant volume
ds	area differential element
e	specific total energy
f	flow residual sensitivity due to a pure geometry perturbation $-\frac{\partial R}{\partial \alpha}$ / frequency
F, G, H	convective/inviscid flux vectors in the axial, circumferential and radial directions
h	specific total enthalpy
I	objective function / identity matrix
k	thermal conductivity
p	static pressure
p^*	total pressure
Pr_l	laminar Prandtl number
Pr_t	turbulent Prandtl number
R	flow residual / Gas constant / the radial coordinate
r	the radial coordinate / a reduced radial coordinate
S	the source term vector of the flow equation

s	entropy defined by $\frac{p}{\rho^\gamma}$
T	static temperature
U	conservative flow variable vector
u, v, w	flow velocity components in the axial, circumferential and radial directions
u_g, v_g, w_g	mesh velocity components in the axial, circumferential and radial directions
V_x, V_θ, V_r	viscous flux vectors in the axial, circumferential and radial directions
X	mesh coordinate vector / the axial coordinate
x	the axial coordinate / a reduced axial coordinate
A	the axial flux Jacobian matrix $\frac{\partial F}{\partial U}$
B	the tangential flux Jacobian matrix $\frac{\partial G}{\partial U}$
C	the radial flux Jacobian matrix $\frac{\partial H}{\partial U}$ / blade chord length
Cax	blade axial chord length
D	the source term Jacobian matrix $\frac{\partial S}{\partial U}$
W	velocity magnitude/worksum
Y	pitch length

Greek Symbols

α	a design variable/ design variable vector, first harmonic pressure phase angle
Δ	denote a change of a quantity
δ_x	perturbation to the x coordinate of a mesh point
$\delta_{r\theta}$	perturbation to the $r\theta$ coordinate of a mesh point
η	isentropic efficiency
γ	specific heat ratio
ω	angular frequency
ϕ	inter-blade phase angle
π	pressure ratio
ρ	fluid density
σ_i	weighting factor in an objective function/adjoint stress term
τ	temperature ratio / pseudo time
τ_{ij}	shear stress components, $i, j = x, \theta, r$
θ	the circumferential coordinate

Superscripts

^{L}	a lower periodic boundary

^{T}	transpose of a matrix or vector
^{U}	an upper periodic boundary

Subscripts

$_0$	denote initial value/phase $\omega t = 0$
$_{-1}$	phase $\omega t = -\frac{\pi}{2}$
$_1$	phase $\omega t = \frac{\pi}{2}$

Abbreviations

N_{b1}	number of blades of the first blade row
N_{bn}	number of blades of the last blade row
2D	two dimension
3D	three dimension
AD	Automatic/Algorithmic Differentiation
ADJ	Adjoint Method
ATC	Advanced Transonic Compressor
bs	blade surface
CFD	Computational Fluid Dynamics
DLR	German Aerospace Center
FDM	Finite Difference Method
IBPA	inter-blade phase angle
IGV	Inlet Guide Vane
le	leading edge
log-dec	logarithmic decrement
MCA	Multiple Circular Arc
NACA	National Advisory Committee for Aeronautics
NASA	National Aeronautics and Space Administration
RANS	Reynolds Averaged Navier-Stokes
ref	reference
te	trailing edge
TFI	transfinite interpolation method
URANS	Unsteady Reynolds Averaged Navier-Stokes

Chapter 1

Introduction

1.1 Background and Motivation

Fans, compressors and turbines are typical turbomachines. Fans and compressors are designed to impart mechanical energy into their working fluid, while turbines are designed to extract energy out of their working fluid. They are also key components of gas turbines, which have a wide range of applications in our daily life, e.g. power generation in power plants, propulsion systems in aero-engines and mechanical drivers in oil & gas industries. No matter what kind of applications they are employed for, gas turbines consume a great deal of fuel and produce a lot of emissions. This will definitely have a great impact on our life in the near and far future in terms of the energy issue and the environmental issue. Power plants, oil and gas industries and airlines want to operate gas turbines with as little fuel and as much output as possible. This points to the development of turbomachines of higher efficiency. As a consequence, a higher efficiency turbomachine will accordingly generate less emission relieving its environmental impact. Therefore the benefit of developing higher efficiency turbomachines is at least twofold: less fuel consumption and lower emission. For gas turbines as propulsion systems, weight and size are also important issues. They are expected to be as light and small as possible so that an airplane can take more payload or an airplane will consume less fuel with the same payload but less weight.

The requirements of higher efficiency, lower weight and smaller size translate to the development of turbomachines of higher loading and more compact configurations with two types of intensified interactions.

The first is the intensified flow field interaction between adjacent blade rows, requiring matching between adjacent blade rows be checked simultaneously in a design optimization process. As one knows, turbomachines usually consist of more than one blade row, so that they can do more work than a single row can achieve. If each blade row is optimised separately, mismatch may be an issue once they are put together to operate. A costly iterative approach will normally be required to adjust the matching between adjacent rows in an explicit way. This will of course cause delay in a product development. It is ideal to optimise blades in a multi-bladerow environment, where matching between adjacent blade rows can be taken into account timely and implicitly during an optimization process. This intensified flow field makes it more urgent and beneficial to perform design optimization in a multi-bladerow environment.

The second is the intensified interaction between the working fluid inside a turbomachine and the blades immersed in the working fluid. The intensified fluid-structure interaction makes blades more vulnerable to flow-induced vibration problems. Therefore aeroelasticity/aeromechanics of a blade is brought into the foreground in a design optimization process, requiring aerodynamics and aeromechanics be checked simultaneously.

Another important issue in turbomachinery blading design optimization is the performance of optimised designs at their off-design conditions. If a blade is optimised at one particular operating point, in principle, there is no insurance about a good performance of the optimised design at other operating points. In order not to deteriorate the performance of a blade at its off-design points unacceptably, performance of a blade at several operating points needs to be looked at simultaneously in a design optimization process, pointing to the development of a multi-operating-point design optimization.

1.2 Turbomachinery Blading Design Optimization

In an enclosed turbomachine, the working fluid is enclosed by structures like casing, hub and blades. The change of any of these structures in any form can affect the performance of the machine composed of these structures. This kind of change has long been exploited as a means to improve the performance of turbomachines. Among

these structures, blades are the most crucial components of turbomachines. Blading aerodynamic and aeromechanic performance largely defines the performance of turbomachines. The focus here is placed on the change of a blade shape to pursue performance gain. Turbomachinery blades are immersed in their working fluid. The energy transfer occurs between rotating blades and the working fluid. Stationary blades tailor the flow field by adding or removing swirl for rotating blades following them. The shape of a blade has a great deal to do with blading aerodynamic and aeromechanic performance. Hence, a great potential exists in changing the shape of blades to achieve blading performance improvement.

Blading performance assessment is the starting point towards blading design optimization. It provides the criteria according to which alternative blading designs can be ranked, with the best one selected.

In early years, wind tunnel testing was the main, if not only, approach to the assessment of aerodynamic performance of aerodynamic configurations. Although nowadays it is a standard procedure for each gas turbine to undergo the rig test before its delivery, it is far from cost-effective to perform blading design optimization based on the experimental performance assessment. For example, the main feature of a flow field inside a turbomachine needs to be obtained via experiments. Once the main flow field features are obtained, they will be analyzed by engineers with appropriate expertise to decide how to change the shape of a blade so that its performance can be improved. Once the change is determined, new blades will be manufactured and assembled to form a new machine. The performance of the new machine will be tested by experiments to see whether an expected improvement has been achieved or not.

With the advance of computer technologies and numerical methods for fluid dynamics, computational fluid dynamics(CFD) has emerged as a new and dominant approach to the investigation of fluid motion inside turbomachines. This approach obtains the flow field information by solving the partial differential equations governing the fluid motion with appropriate boundary conditions. Over decades of extensive development, CFD has matured to the point of being able to simulate turbomachinery flow fields with desirable engineering accuracy. Modern turbomachinery blade design relies almost completely on CFD to explore the three-dimensional features of blades (Horlock and

Denton, 2005). Compared with the rig test approach, well-calibrated CFD is able to assess the aerodynamic performance of an aerodynamic configuration with much shorter turnaround and much lower cost and more details. Any change to the original configuration can be established quickly in a computerized model level together with the corresponding performance change by CFD.

In a gradient-based design optimization method, the sensitivities of blading performance to parameters describing a blade geometry are required to determine the change of a blade shape in a way to improve the blading performance. Modern turbomachinery blades have rich three dimensional features, such as twist, lean, bend, skew and so on. Many different parameters can be involved in the description of a blade. A change to any of these parameters results in a corresponding change to the blading performance. If this blading performance change is evaluated by solving the flow governing equations or its linearized version, these equations need to be solved for each design variable. The time cost of gradient evaluation can be prohibitive at a situation where there are hundreds of or even more design variables. In order to apply a design optimization system to routine design activities in an industrial environment, the method for gradient evaluation should be efficient enough.

1.3 Essential Elements for Blading Design Optimization

For a gradient-based design optimization, there are normally five essential elements: a flow solver, a gradient evaluation scheme, a shape parameterization scheme, a mesh deformation approach and an optimizer.

• Flow Solver

In a CFD-based design optimization, the flow solver provides the basis on which the whole design system is built. A flow solver serves two roles here. One is to provide solution information according to which further information (the value of an objective function and its gradients) will be obtained to determine how a blade profile will be changed. The other is to verify that a change in the blade profile will deliver an expected performance gain. If the flow solver can not predict the performance of interest with

sufficient engineering accuracy, then an optimised design can be useless.

For an aerodynamic design optimization, the Reynolds Averaged Navier-Stokes (RANS) equations are solved in a cell-centered finite volume framework. Details about the solution methods for the RANS equations are documented in Chapter 3. For a concurrent aeromechanic and aerodynamic design optimization, the Unsteady Reynolds Averaged Navier-Stokes (URANS) equations are solved using the frequency domain method of the Nonlinear Harmonic Phase Solution method to greatly reduce time cost. This method is explained in Chapter 7 with a comparison with the conventional time-linearized linear harmonic method in Appendix D.

• Gradient Evaluation Scheme

In terms of gradient calculation, there are the finite difference method, the linearized method, the complex variable method and the adjoint method. The first three methods have the same drawback of time cost directly proportional to the number of design variables, making them unfavorable at situations with a large number of design variables. More design variables give a design system more freedom and more choices to get a better design. The adjoint method has an attractive feature of independence of time cost for a single objective function gradient evaluation on the number of design variables. It is this feature that makes this method be our choice here for efficient gradient evaluation. The principle of the adjoint method is explained in Chapter 4 with the detailed derivation based on the partial differential equations in Chapter 4 for an adjoint Euler system and in Appendix A for an adjoint Navier-Stokes system.

• Shape Parameterization Scheme

Shape parameterization is vitally important in turbomachinery blading design optimization. Different shape parameterizations can lead to different final designs as revealed by Wu et al. (2003). A good shape parameterization approach is capable of producing realistic blades of a wide range of characteristics with the change of its parameters. In this work, the perturbation to a blade shape, instead of the blade shape itself, is parameterized. The shape perturbation parameterization is a popular scheme

for airfoil and wing shape design optimizations and is also used to the application of turbomachinery blading design optimization. The shape perturbation is usually parameterized using some mathematically smooth hump functions, based on the assumption that the original smooth blade shape plus smooth perturbations is still smooth. As a matter of fact, a mathematically smooth blade shape may not be a realistically smooth blade. Therefore some remedies are needed to have a realistically smooth blade. The details of the shape perturbation parameterization are expounded in Chapter 5.

• Mesh Perturbation Approach

Design optimization, no matter in what approach it is conducted, is normally performed in an iterative process. At each iteration, the computational mesh will be perturbed for the gradient calculation for each design variable. If the mesh is generated in an iterative process, it is usually quite costly if the mesh corresponding to the changed geometry needs to be regenerated in the same iterative way. It is quite preferable to change the mesh due to a geometry change in a much faster way independent of the original mesh generation approach. It seems that this kind of mesh perturbation approach has been emerging as a separate research topic (Duong et al., 2007). Representative approaches include the spring analogy method (Blom, 2000; Kim and Nakahashi, 2005; Nielsen and Anderson, 1999) and the transfinite interpolation method (TFI) (Tsai et al., 2001). In this work, the original mesh is generated in a fast algebraic way. The mesh is regenerated in the same algebraic way after perturbations to a blade geometry are calculated. The associated time cost is quite small compared with the whole time cost of a design iteration. There is no need to seek other approaches to perturb a base-line mesh due to a blade geometry change.

• Optimizer

An optimizer is used to obtain the amount change of each design variable according to the gradients of an objective function to design variables. Some optimizers, such as Newton's methods, Quasi-Newton methods (Health, 2002), need further information, such as the Hessian matrix or its approximation, to calculate such change of design variables. The determination or approximation of a Hessian matrix usually needs extra

calculation resulting in extra time cost which can be considerable. Among all these optimizers, the steepest descent and the conjugate gradient method do not need a Hessian matrix. The steepest descent method is the simplest and the most straightforward. It takes a step size at the negative gradient direction. The conjugate gradient method incorporates gradients at previous design iterations/cycles, therefore it is expected to behave better than the steepest descent method in terms of reaching the optimum design with fewer design cycles/iterations. Nevertheless the conjugate method does not normally change the final design. The focus of the work is on the application of the adjoint method, hence not much attention is paid to the optimizer, once it is delivering the right solution. The steepest descent method is our choice here.

1.4 Key Methodologies

Two key methodologies are involved in this work. They are the adjoint method and the Nonlinear Harmonic Phase Solution method.

The adjoint method is very efficient in calculating gradients of an objective function to a large number of design variables. It requires one flow solution and one adjoint solution at each design iteration at a cost roughly equal to two flow solutions. The time cost is nearly independent of the number of design variables. This allows a lot of design variables to be used even in a routine design optimization. The principle of the adjoint formulation and how it can be so efficient are explained concisely in Chapter 4 with a detailed review of the development and application of the method in aerodynamic design optimization provided in the next chapter.

The Nonlinear Harmonic Phase Solution method enables the unsteady flow equations with temporally periodic unsteadiness to be solved very efficiently for turbomachinery aeromechanics predictions. Besides it also provides a good basis on which the corresponding adjoint system can be formulated in a straightforward way, and then the developed adjoint system can be solved by extending a steady adjoint solver without much extra effort. The Nonlinear Harmonic Phase Solution method is illustrated in Chapter 7 with Appendix D giving a comparison with the conventional time-linearized linear harmonic method.

1.5 Overview of the Thesis

The research work is aimed at addressing two main issues about the application of the adjoint method to turbomachinery blading design optimization: one is the use of the adjoint method for blading aerodynamic design optimization in a multi-bladerow environment; the other is the use of the adjoint method in a concurrent blading aeromechanic and aerodynamic design optimization. The first issue is successfully addressed with the formulation of an adjoint mixing-plane treatment bridging the solution information between two adjacent bladerow domains separated by an interface. The other issue is completed successfully and efficiently using the Nonlinear Harmonic Phase Solution method to solve the unsteady flow equations for turbomachinery blading aeromechanics, laying down a good basis on which an adjoint system can be formulated and solved in an efficient manner.

Centered around the two main issues, the thesis is broken down into ten chapters. The background and motivation of the research work and methodologies involved are briefed in Chapter 1. Chapter 2 reviews methods for turbomachinery blading aerodynamics and aeromechanics calculations and methods for turbomachinery blading design optimization with emphasis on the recent development and applications of the adjoint method and the methods for solving the unsteady flow equations for turbomachinery blading aeromechanics. The third chapter deals with the steady nonlinear flow model and solution methods. The fourth chapter illustrates the adjoint formulation principle, and develops the adjoint systems, including the adjoint equations, their boundary conditions and the adjoint mixing-plane treatment based upon the steady nonlinear flow equations. The fifth chapter establishes a turbomachinery blading aerodynamic design optimization system. The sixth chapter presents verification cases and four design optimization cases to confirm the developed theories in the previous chapters. The seventh chapter develops the unsteady flow solver using the Nonlinear Harmonic Phase Solution method for unsteady flow field analyses for turbomachinery blading aeromechanics. The eighth chapter formulates the adjoint system for design optimization concerning both aeromechanics and aerodynamics. The ninth chapter provides three design cases to demonstrate the validity and effectiveness of the developed method-

ologies for concurrent aeromechanic and aerodynamic design optimizations. The tenth chapter concludes the research work and gives some suggestions for further work.

Chapter 2

Literature Review

This chapter reviews the numerical methods for turbomachinery blading aerodynamics, aeromechanics and methods for turbomachinery blading design optimization. The emphasis is given to methods for blading aeromechanics calculations and the development and applications of the adjoint method.

2.1 Blading Aerodynamics Calculation

In modern turbomachinery the flow field is inherently unsteady, mainly attributable to the relative movement of adjacent blade rows. The flow field also has three dimensional features mainly due to the twist of the blades. However, at a preliminary design stage, blading aerodynamic design has long been employing steady flow methods (Larosiliere et al., 2002; Cumpsty and Greitzer, 2004). The flow field in a stator is considered in a stationary reference frame, while the flow field in a rotor is considered in a rotating reference frame attached to the rotor. With this consideration, the flow field within a blade passage is assumed steady in the corresponding reference frame with the effect from adjacent rows seen in an averaged manner. Even at a detailed design stage, the solution to the steady flow equations rather than the unsteady flow equations is sought ubiquitously for blading aerodynamics predictions. This may be due to the two facts. Firstly, the unsteadiness in the unsteady flow field inside a turbomachine, particularly at its design operating point, is usually small. As a consequence, the influence of the unsteadiness on the time-averaged flow solution is small. This can be evidenced by the current blade designs which have achieved high aerodynamic performance by largely

using the steady flow model. Secondly, a solution of the unsteady flow equations is much more costly than that of the steady flow equations (He, 2008), although there has been a great advancement of computing power and numerical methods for unsteady flows over the last two decades. The huge extra time cost incurred by an unsteady flow solution may not be easily justified for turbomachinery aerodynamics predictions, particularly within a multi-disciplinary design environment.

Historically the Euler flow equations were first investigated to represent a flow field (Lieber, 2003). The Euler flow equations describe the motion of ideal fluids, which actually do not exist in reality. But the Euler solutions could be effectively used to predict loading and pressure distributions, providing a basis for boundary layer analyses and loss predictions. At a blading preliminary design stage, the Euler equations are normally used in conjunction with some loss models to provide a quick assessment of overall blading characteristics. Most turbomachinery fluid flows are highly turbulent and viscous. A more complete flow field representation necessarily requires the employment of the RANS equations. Jameson and Martinelli (2000) also point out that it is vitally important to take viscous effect into account when performing design optimization. The RANS equations have become a core method of flow field predictions (Molinari and Dawes, 2006).

2.1.1 Flow Mixing-plane Treatment

Turbomachines usually consist of more than one blade row with rotors and stators arranged alternately. The steady flow analysis employs different reference frames for adjacent blades. Meanwhile at an interface, the flow field is not uniform in the circumferential direction. With a steady flow analysis, the instantaneous one-to-one correspondence of the flow solution at an interface is lost. This therefore provokes the issue of how to pass solution information across an interface.

As has been mentioned, at an interface between adjacent blade rows, the flow field at one blade row is presented to the other row in an averaged sense. Now the issue turns to one of how to average what solution information. The solution information at an interface is usually averaged circumferentially and required to be equal across

an interface. All these methods based upon such an averaging process are classified as averaging-plane methods. There are several variations of averaging-plane methods, for example, the mixing-out averaging-plane method (Denton and Singh, 1979; Denton, 1992) , and the kinetic energy averaging-plane method (Chima, 1998).

The mixing-out averaging-plane method, also known as the mixing-plane treatment, is the most popular one among all the averaging-plane methods. The treatment averages fluxes of mass, momentum and energy circumferentially representing far downstream or upstream uniform flow field. It will inevitably lose the instantaneous flow field interaction and at the same time introduces artificial mixing losses. So far numerous numerical investigations have proved the validity of the mixing-plane treatment in a multi-bladerow flow field analysis.

The conservation requirement of the mixing-plane treatment only sets a target to be achieved. How to achieve the conservation requirement is equally important. At the start of a flow solution, these averaged quantities are usually not equal across an interface. An implementation is needed to drive the difference in these averaged quantities across an interface to zero during a solution process. This can be favorably achieved by using the general nonreflective boundary condition implementation proposed by Giles (1991), which is highly desirable when considering the small intra-row gap between adjacent blade rows. The nonreflective implementation allows the flow characteristic disturbances to propagate through an interface correctly, guaranteeing the conservation requirement without corrupting the flow solution in the interior domains.

All in all, the mixing-plane treatment has two important features: conservation and nonreflectiveness. The conservation is dictated by the conservation laws, while the nonreflective implementation will ensure that the conservation across the rotor-stator interface is achieved without spurious reflections to spoil the solution in the interior domains. The mixing-plane treatment has been widely accepted and has become a standard industrial tool for steady multi-bladerow flow analyses. This obviously provides a natural opportunity and sound basis for design optimization to be carried out in such an environment.

2.2 Blading Aeromechanics Calculation

Aeromechanics is the branch of mechanics that deals with the motion of gases (especially air) and the structural/mechanical characteristics of the bodies in the flow. In the context of turbomachinery blades, aeromechanics are normally referred to blade flutter and forced response. In this work, only blade flutter is considered. Blade flutter is a self-excited aeroelasticity instability phenomenon, usually starting from an infinitesimal disturbance which can be either aerodynamically or mechanically excited. Once a blade starts to vibrate, it tends to induce unsteady aerodynamic forces which in return will act on the vibrating blade. In each period of the blade vibration, there is a net energy exchange between the vibrating blade and the working fluid in which the blade is immersed. If energy is absorbed by the blade, then aeroelasticity instability will occur, leading to amplified blade vibration with excessive vibration stress causing blade failure eventually. This is actually a dynamic coupling process between the blade structure dynamics and the working fluid aerodynamics.

The prediction of blade flutter naturally requires the coupling solution of the governing equations of aerodynamics and structural dynamics (Wu et al., 2005b; Sadeghi and Liu, 2005). The solution methods for aerodynamics and structural dynamics have been developed largely independently. The difference embedded in these methods implies that it is more convenient and realistic to model aeromechanics using two distinct solvers of different methods with appropriate information exchange between them. Unfortunately this kind of modeling is extremely time-consuming with time cost dominated by the solution of the unsteady flow solution for aerodynamics. It is a valuable tool for occasional use for the final verification of a design or for troubleshooting, but the coupling modeling is not suitable for frequent use in a design application. This makes it impractical to perform an aeromechanic related design optimization based upon such modeling techniques, considering that a design optimization is carried out in an iterative process.

It is noted that the coupling effect between aerodynamics and structural dynamics for turbomachinery blades is largely determined by the structure/fluid mass ratio,

measured by the mass coefficient defined by

$$C_{mass} = \frac{m}{\rho\pi(C/2)^2}$$

where

m is the blade mass per unit blade span
C is the blade chord length
ρ is the density of the working fluid

The mass coefficient is usually quite large indicating a small coupling effect for compressor blades. The change of blade vibration mode shapes and frequencies due to the small coupling effect is accordingly small and negligible. With this primary observation, the coupling dynamics process can be split into two uncoupled dynamics problems in two domains. A blade vibratory movement can be considered the same as it vibrates in a vacuum with its vibration mode shape and frequency obtained through a finite element (FE) analysis. The unsteady aerodynamic forces induced by a specified blade vibration will be obtained through solving the unsteady flow equations.

The prediction of blade flutter is then based upon the energy method, with the energy exchange between the working fluid and a blade determined through an unsteady flow solution. The validity of the uncoupled or loosely coupled method has been validated by many researchers and has been widely accepted for routine use for turbomachinery blading aeromechanics predictions (Huang, 2006; Srivastava et al., 2002; Reddy and Srivastava, 1999; Duta, 2002).

With the splitting of the aeromechanic problem, the focus is now turning to the efficient solution of the unsteady flow equations. It is well known that solving the URANS equations in a time domain for predicting aeromechanics/aeroelasticity of turbomachine blades is still quite time-consuming, though there has been significant research effort leading to considerable progresses or even breakthrough in this field. With this premise, it is desirable to choose a methodology which, on one hand is efficient for an unsteady flow solution for aeromechanics, and on the other hand provides a good basis so that other efficient methods for blading design optimization can be incorporated without much extra effort. The methods for an efficient prediction of an unsteady flow field for aeromechanics are reviewed accordingly.

2.2.1 Unsteady Flow Solution Methods for Aeromechanics

All the solution methods for the unsteady flow equations for turbomachinery blading aeromechanics can be generally grouped into two categories: the time domain methods and the frequency domain methods.

2.2.1.1 Time Domain Methods

The most straightforward way to obtain the solution to the URANS equations is to solve these unsteady equations using a time-marching method in the time domain (He, 1989; He and Denton, 1994). All the flow field features, including nonlinearity, can be included in such an unsteady flow solution. Conceptually it is also very straightforward to formulate the corresponding unsteady adjoint system and solve it in the same manner. The drawback of this method is its huge turnaround. Even for the same mesh, an unsteady flow solution using this method usually consumes 15-20 times more CPU time than a steady flow solution. To model proper circumferential length scales will normally require the use of a whole-annulus computational domain, resulting in another factor between 20 and 100. Hence, a direct time-domain nonlinear unsteady flow solution would typically be by two to three orders of magnitudes more time consuming than a steady flow solution (He, 2008). The unsteady adjoint equation requires backward time-marching: from the null adjoint field at the final time to the unknown adjoint field at the initial time. The backward time-marching demands the history of the unsteady flow solution be stored, resulting in considerable memory or disk usage (Nadarajah and Jameson, 2002; Mavriplis, 2008). The high computational demand from this method makes it highly unfavorable in a design optimization.

The unsteady flow field in two adjacent blade passages of a blade row is not identical as the steady flow field is. However there is a fixed phase difference in the unsteady flow field between two adjacent blade passages. In many situations, this phase difference can be known *a priori*. A phase-shift boundary condition has been employed in computations in conjunction with the use of a single blade passage in a blade row to significantly reduce time cost. The first effort of this application might attribute to Erdos et al. (1977) for the Direct Store method. This method has to store the

time history of the flow solution over at least one period at the periodic boundaries, which is then recalled to update the flow solution at periodic boundaries taking into account the phase shift. The memory usage overhead by this method can become prohibitive for 3D calculations. The second method is the time-inclined method proposed by Giles (1991). This method employs a coordinate transformation to transfer the unsteady flow equation to one in a computational time level determined according to the phase difference of interest. With this coordinate transformation, a direct periodic boundary condition as for the steady flow equations can be applied. The application of this method needs extra coding development because the coordinate transformation changes the conservative solution variables. Its application is restricted by the range of time lag divided by a blade pitch. The third approach of its kind is the Shape Correction method proposed by He (1989). Instead of storing the time history of the flow solution over a period at periodic boundaries, the Shape Correction method represents the unsteady flow solution at periodic boundaries in terms of Fourier series. Therefore the Fourier coefficients only are stored, leading to a considerable reduction in memory usage.

Even if a single passage solution can significantly reduce the computational cost over a whole-annulus solution, the cost is still considerably high. The Shape Correction method based on a single passage has a time cost roughly equal to that of a whole-annulus calculation with 2-2.5 blade passages (Huang, 2006). For the same mesh, an unsteady flow solution costs accordingly about 30-50 times more CPU than a steady flow solution. With the adjoint solution requiring a backward time-integration, the memory or disk usage can still be significant. The high demand of both CPU time and memory or disk usage is the biggest obstacle to build up a design optimization system based upon even the Shape Correction method employing a single blade passage for each blade row.

2.2.1.2 Frequency Domain Methods

The frequency domain methods include the time-linearized linear/nonlinear harmonic method, the Harmonic Balance Method, the Nonlinear Frequency Domain method and the Nonlinear Harmonic Phase Solution method. All these methods are making use of

the Fourier series to represent a part or the whole of an unsteady flow solution. The time-linearized linear/nonlinear harmonic method splits the unsteady flow equation into one time-averaged equation and one time-linearized equation, with the unsteady perturbation of the flow solution represented in a Fourier series. While the other three methods express the whole flow solution in a Fourier series, casting the unsteady flow equations into a set of steady-like equations at a series of phases of a period of unsteadiness.

• Time-Linearized Methods

The unsteadiness of interest in a turbomachinery flow field for aeromechanics can normally be attributed to either time-periodic aerodynamic disturbances (e.g. upstream wake, inlet/exit distortion) or time-periodic mechanical disturbances (e.g. blade vibration). As a consequence the induced unsteadiness in the flow field also shows strong temporal periodicity. The periodicity feature inside turbomachinery flow fields is very important information which can be made use of in the solution of the unsteady flow equations to greatly reduce time cost.

Time-averaging the nonlinear flow equations gives the time-averaged equations containing deterministic stress terms (Adamczyk, 1985). Subtracting the time-averaged equations from the original unsteady flow equations and neglecting the nonlinear terms results in the time-linearized equations.

With the assumption that the unsteadiness has a magnitude which is much smaller than that of the time averaged flow, the deterministic stress terms in the time-averaged equations can be dropped. The resultant time-averaged equations are effectively steady flow equations. They can be solved without any consideration of the time-linearized equations. The steady flow solution forms the variable coefficients of the time-linearized equations. The time-linearized equations are still time-dependent. Solving this system in the time domain is still nontrivial. The dominant period or frequency of the unsteadiness in the unsteady flow field can be known *a priori* in many situations. With this preliminary information, the solution to this unsteady time-linearized equations can be expressed as one harmonic. With the harmonic formulation, the time-domain unsteady time-linearized equations can be converted into steady like equations about

the harmonic amplitude in the frequency domain. The resulting equations can be solved very efficiently, costing about twice the time of a steady flow solution. This methodology is called the time-linearized linear harmonic method (Hall and Lorence, 1993).

This method is much more efficient in terms of time cost compared with the nonlinear time-domain method. However, this method is built upon the small disturbance or linear assumption. Consequently nonlinear effects can not be captured by this method. The nonlinear effect plays an important role if there is shock or flow separation in the flow field. And this is usually the situation for modern high loading and high speed turbomachines.

The problem arising with the linear harmonic method in the context of design optimization using the adjoint method is the great difficulty in formulating and solving the corresponding adjoint system based upon the split equations: the linear harmonic equation, which has already been linearized in time, also needs linearizing with respect to a design variable in order to formulate the corresponding adjoint system. The resultant adjoint systems include two sets of equations: one corresponding to the time-averaged flow equation and the other corresponding to the linear harmonic flow equation. The adjoint system, corresponding to the time-averaged flow equation, depends on the adjoint system, corresponding to the linear harmonic flow equation. This makes it necessary to solve the linear harmonic adjoint system followed by solving the adjoint system corresponding to the time-averaged flow equation, which is consistent with the backward integration property of an unsteady adjoint system. Though all these are mathematically consistent, it is quite counter-intuitive, making the problem extremely complicated.

In order to capture the nonlinear unsteady effect in the flow field, He and Ning (He and Ning, 1998; Ning, 1998) proposed the time-linearized nonlinear harmonic method. As one knows, there are deterministic terms resulting from the time-averaging process of the nonlinear time-domain equations and contained in the time-averaged equations. With the harmonic formulation of the unsteadiness, the deterministic stress terms can be approximated efficiently with the coupling solution of the time-averaged equations and the linear harmonic equations. Although this coupling process consumes more

computing resources than that of the linear harmonic method, it is still much more efficient than the time domain method. However, it becomes more difficult if the number of harmonics is increasing due to the difficulty in calculating the deterministic stress terms. The same difficulty as that of the linear harmonic method will arise when an adjoint system is built upon the time-linearized nonlinear harmonic method.

•Fourier Modeling Methods

Another way to circumvent the huge time cost by the time-domain methods is the use of the Fourier transformation technique over the whole computational domain. The use of the Fourier transformation technique in solving the nonlinear unsteady flow equations for unsteady turbomachinery flow fields was pioneered by He (1989). He proposed the Shape Correction method to transform a time domain flow solution at the periodic boundaries of a blade passage to a frequency domain, with some Fourier coefficients stored for updating the flow solution to ease the huge memory requirement from the Direct Store method by Erdos et al. (1977). The use of the Fourier transformation technique was further exploited by Hall et al. (2002) and McMullen (2003) to extend the use of the technique from periodic boundaries to a whole computational domain. The Harmonic Balance Method, which was coined by Hall et al. (2002), expresses an unsteady flow solution as a whole in Fourier series, giving rise to work out the time derivative in the unsteady flow equations using the flow solution at equally spaced phases in a period of unsteadiness. The solution variables are conservative flow variables at those equally spaced phases. Fourier transformation is utilized to get the time derivatives of the conservative flow variables in terms of the flow solution variables at those discrete phases. While in the Nonlinear Frequency Domain method coined by McMullen (2003), both the conservative flow variables and the flow residuals of the flow governing equations are expanded in Fourier series. The solution variables are the Fourier coefficients of the conservative flow variables. Fourier transformation and its inverse transformation are utilized to transfer both the flow solution and the flow residual between the time domain and the frequency domain. In a CFD context, the cost of doing these transformations can be considerable. Therefore the Nonlinear Frequency Domain method generally costs more CPU time and memory than the

Harmonic Balance Method.

The Nonlinear Harmonic Phase Solution method, which is a variant of the Harmonic Balance Method (Hall et al., 2002), was proposed by He (2008). Unlike the Harmonic Balance Method which uses equally spaced phases, He's harmonic formulation chooses the two particular phases $(0, \frac{\pi}{2})$ for situations of largely linear unsteadiness, or the three particular phases: $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$ for including nonlinear unsteadiness. This results in further cost reduction (the two-phase scenario), considerable convenience and simplicity when working out the time derivative term in terms of the flow solution at the two or three phases. The method is a very neat formulation making it highly attractive for turbomachinery aeromechanics. Both the Harmonic Balance Method and the Nonlinear Harmonic Phase Solution method are as efficient as the time-linearized linear harmonic method. However they offer a basis on which the corresponding adjoint systems can be formulated and solved with greater ease, compared with the conventional time-linearized linear harmonic method. The Nonlinear Harmonic Phase Solution method is explained in Chapter 7 with Appendix D providing a comparison between the Nonlinear Harmonic Phase Solution method and the time-linearized linear harmonic method. The corresponding adjoint solver can be extended from a steady adjoint solver without much extra code development, as described in Chapter 8.

2.3 Methods for Turbomachinery Blading Design Optimization

There have been many different methods used historically for turbomachinery blading design optimization. These methods can also be categorized in many different ways. For example, they can be categorized into automated methods and trial methods, inverse design methods and direct design methods, deterministic methods and non-deterministic methods, gradient-based methods and non-gradient based methods, CFD-based methods and non-CFD based methods, high-fidelity methods and low-fidelity methods, ... One particular method can belong to more than one of the above categories.

Turbomachinery blading aerodynamic design can be divided into two stages: preliminary design and detailed design. The preliminary design defines the main features of

blades, while the detailed design verifies the output of the preliminary design and furthermore refines the preliminary design to explore potential performance gain. Blading design optimization falls on both the preliminary design and the detailed design. At the preliminary design stage, it is now a common practice to use customized airfoils obtained by design optimization (Köller et al., 2000; Küsters et al., 2000; Sieverding et al., 2004) instead of airfoil families developed during early years, such as NACA-65 airfoils, British C4 airfoils, and so on.

In the early years, design optimization was mainly performed in a trial and error fashion by engineers making use of their expertise and experience. The blade surface curvature is directly linked to the local flow velocity/Mach number (therefore local loading) and can be tailored to obtain desirable local flow fields. Based on a similar principle, Li and Wells (1999) analyzed the flow field of an existing compressor and redesigned it by redistributing the chordwise loading to improve its performance successfully. The effect of blade lean, sweep and twist on the performance of a blade has been investigated systematically by many researchers to explore potential performance gain (Hah, 1999; Wang, 1999; Denton and Xu, 2002). Blade forward sweep tends to broaden the stall margin of a blade, but it has little contribution to peak efficiency; blade backward sweep tends to enhance a blade's peak efficiency, but narrows its stall margin. Blade positive lean normally increases peak efficiency.

In the present work, the attention is focused on automated design optimization methods making use of high fidelity flow field simulations by CFD. In a CFD-based blade shape optimization process, CFD is an indispensable part and also the basis part. Other elements for design optimization are built upon the reliable results by CFD. All the CFD-based design optimization methods can be generally categorized into two groups: inverse design methods and direct design methods.

The inverse design methods still have their market nowadays (van Rooij et al., 2007; Hu et al., 2006). An inverse method was recently extended to redesign blades in a multi-bladerow environment handling blade row matching timely (van Rooij et al., 2007). The method requires the specification of a target pressure or velocity distribution on a blade surface, providing experienced engineers with a direct means to make use of their developed experience and insight. Then a computer code is used to calculate the blade

geometry which is expected to produce the required performance by the transpiration boundary condition. The transpiration boundary condition can be easily incorporated with a slip wall boundary condition, however it is problematic with a non-slip wall boundary condition, where the flow velocity is zero relative to the non-slip wall. There has been effort circumventing the difficulty and extending this method to applications with RANS equations (de Vito et al., 2003). This method, as it is claimed, has a very short turnaround. The major disadvantage of this method is that it requires the specification of target flow parameters in terms of pressure or velocity distributions on a blade surface; hence its success depends largely on a designer's experience and insight. Apart from that, it is usually difficult to apply constraints in an inverse design (Büche et al., 2003), and it is not possible to consider performance of a blade at off-design points in an inverse design (Köller et al., 2000).

The direct design methods, on the contrary, do not require the specification of target pressure or velocity distributions on a blade surface. In a direct design method, an initial design is normally changed to a better one little by little in an iterative process until no improvement can be made any more. The direct design methods are a very big family and can be further classified into two groups: non-deterministic methods and deterministic methods.

Non-deterministic methods, such as the genetic algorithm and the simulated annealing, do not need gradients of an objective function, but values of an objective function only. In principle, these methods are able to find the global optimum in a design space and have long been researched for turbomachinery blading aerodynamic design optimization (Benini, 2004; Keskin et al., 2006; Lotfi et al., 2006; Oyama and Liou, 2004). But their applications in routine designs are usually restricted due to their huge time costs. The response surface method and the evolutionary method with approximate models have been gaining increasing attention over last several years mainly due to their easy implementation and affordable time cost with a few design variables (Yi et al., 2006; Jang et al., 2006).

Among deterministic methods, some need gradients of an objective function to design variables. According to the way in which gradients of an objective function to design variables are calculated, there are: the finite difference method, the linearized method,

the complex variable method and the adjoint method. The finite difference method requires the original nonlinear flow governing equations to be solved after a design variable is perturbed a small amount. Gradient results obtained by the finite difference method normally suffer from step size dependence. Regardless of its shortcoming, this method is used in gradient calculation for design optimization due to its simplicity (Burguburu et al., 2004; Harvey et al., 2000). The complex variable method is the complex variable implementation of the finite difference method, but it does not share the shortcoming of the finite difference method (Vatsa, 1999). The linearized method requires the linearized flow governing equations (with respect to a design variable) to be solved and does not have the shortcoming of the finite difference method. The time cost for evaluating gradients by the finite difference method or the complex variable method or the linearized method is directly proportional to the number of design variables, preventing the use of a large number of design variables. However it is usually desirable to have sufficient many design variables so that a reasonably good design can be obtained within the design space spanned by the design variables. The adjoint method offers a means to calculate gradients of an objective function to a large number of design variables at a cost almost independent of the number of design variables. It is this property of the adjoint method that makes it very attractive for design optimization with a large number of design variables.

2.4 Development and Application of the Adjoint Method

The adjoint method originated from the field of optimum control theory (Lions, 1971). Pironneau (1974) is probably the first researcher who introduced the idea of using the adjoint method in the context of fluid mechanics. However, the application of the adjoint method in aerodynamic design optimization based on CFD was pioneered by Jameson (1988, 1989). The adjoint method has been established as an effective method of calculating gradients of an objective function to a large number of design variables in CFD-based design optimizations, and it has been widely accepted in design optimization in many disciplines (Mohammadi and Pironneau, 2004). However it did not gain enough attention till a decade later since its first introduction to the CFD community. As Giles and Pierce (2001) point out, this can be partly due to the complexity

of the method in its formulation and implementation. In an aerodynamic design optimization, the shape of an aerodynamic configuration, which normally corresponds to a viscous solid wall boundary in a CFD calculation, is the control element which can be changed in a favorable way to optimise aerodynamic parameters of engineering merit.

Based on the way in which a final discrete adjoint system is formed, there are two variations: the continuous adjoint method and the discrete adjoint method. In the continuous adjoint method, the nonlinear flow equations in a partial differential form are linearized first with respect to a design variable. Then an adjoint system will be derived from the linearized flow equations, followed by discretization. In the discrete adjoint method, the flow equations in a partial differential form are discretized first, followed by the linearization and adjoint formulation. This difference can also be summarized in this way: the linearization and adjoint formulation are performed in the partial differential equation level for the continuous adjoint method and in the discrete algebraic equation level for the discrete adjoint method.

The discrete adjoint method can produce exact gradients of a discrete objective function with respect to design variables, which will ensure a design process converges quickly and fully. However it is very difficult to develop discrete adjoint codes particularly by hand and with high-order upwind schemes and sophisticated turbulence models implemented in a flow solver. There have been efforts to reduce the complexity of developing discrete adjoint systems using complex variables (Nielsen and Kleb, 2005) and Automatic/Algorithmic Differentiation (AD) (Mohammadi, 1998). The discrete adjoint codes usually require more memory and CPU time compared with its continuous counterpart (Nadarajah and Jameson, 2000). In particular, the discrete adjoint codes obtained with AD would typically consume three times the CPU time compared to that by a flow solver (Duta et al., 2007; Thomas et al., 2003), while the time cost of a continuous adjoint solver is usually the same as that of its flow solver. To apply AD as a black box to a large scale CFD code may generate useless codes consuming enormous memory and CPU. It is advocated by Duta et al. (2007) that making selective use of AD can improve the performance of AD generated adjoint codes dramatically. This implementation however normally needs a series of code reconstruction or preparation, which can require a considerable work depending on the original code structure. In

the continuous adjoint method, one is free to discretize the adjoint equations in any consistent way, although it is always better to consult the flow solver schemes. The gradients by a continuous adjoint solver are subject to two levels of discretization errors for both the flow equations and the adjoint equations, which vanish with mesh refinement.

Giles and Pierce (Giles et al., 2006; Giles and Pierce, 2000) point out that there is no fundamental reason to prefer one approach over the other and the continuous adjoint method can produce the same good design as the discrete one if gradients by the continuous one is preconditioned. Nadarajah and Jameson (2000) make an exhaustive comparison of the two approaches in the derivation level, pointing out that with infinite fine mesh the continuous adjoint equations can be recovered from the discrete adjoint equations by the discrete adjoint method. This reveals the consistency of the two adjoint approaches. Nadarajah and Jameson (2000) further suggest that the best practice in terms of accuracy and simplicity of formulating an adjoint system is to make use of the continuous adjoint method in the interior domain and use the discrete adjoint method along the boundary of a computational domain. More details about the advantages and disadvantages of the two approaches can be found in (Giles and Pierce, 2000) and (Nadarajah and Jameson, 2000).

When deriving an adjoint system based upon the RANS equations, some choose to freeze the eddy viscosity to reduce complexity. The influence of freezing the eddy viscosity on gradient accuracy was investigated by some researchers (Kim, 2007; Kim and Nakahashi, 2005; Nielsen and Anderson, 1999). The conclusion seems to suggest that freezing the eddy viscosity will normally change gradient results obviously if the relationship between an objective function and a design variable is highly nonlinear. The inaccuracy in gradient results due to a frozen eddy viscosity will generally deter the convergence of an optimization process, but not change the final design (Kim, 2007). Nevertheless, it has been successfully applied to design optimization by some researchers (Nadarajah, 2003; Kim, 2001). Freezing the eddy viscosity is somewhat a good compromise between accuracy and reduced complexity in developing an adjoint system.

2.4.1 External Steady Flow Application

In the context of a CFD based design optimization, the adjoint method (both the continuous one and the discrete one) first gained attention in applications to external flows (e.g. for airfoil designs). This might be partially due to the fact that the adjoint method was first introduced to the external flow community. It might be partially due to the first maturity of CFD in external flow applications which is less complicated and more easily simulated with engineering accuracy than internal flow applications.

The development of the adjoint method has progressed through adjoint potential flow equations, adjoint Euler flow equations (Jameson, 1988) to adjoint Navier-Stokes flow equations (Jameson, 2003). Its application has advanced from 2D airfoils, 3D wings to complete aircraft configurations (Jameson, 1988, 1989; Jameson and Kim, 2003; Alonso et al., 2002; Jameson, 2003; Jameson and Alonso, 1996; Reuther et al., 1996, 1999a,b). Jameson's group also investigated the discrete adjoint method for design optimization (Nadarajah and Jameson, 2000).

There are also a number of other researchers using the adjoint method for design optimization (Soto and Löhner, 2000; Nielsen and Kleb, 2005; Kim, 2001, 2007; Anderson and Venkatakrishnan, 1997; Mohammadi, 1998). Kim and Nakahashi (2005) and Nielsen and Anderson (1999) developed hand-coded discrete adjoint solvers based on flow solvers using unstructured meshes for the ONERA M6 wing design optimization. Kim (2007) developed hand-coded discrete adjoint solvers based on flow solvers using structured and overset grids for various design optimizations. Mohammadi (1998) used the AD software Odyssée to generate discrete adjoint codes for 2D airfoil and 3D aircraft design optimizations.

In the interior of a domain, both the magnitudes of adjoint variables and the mesh perturbation due to a boundary movement decrease quickly with the distance to a boundary. With this observation, an incomplete gradient approximated through the adjoint boundary conditions without solving the adjoint equations has been used to further reduce time cost (Soto and Löhner, 2001; Mohammadi and Pironneau, 2004). The successful application of incomplete gradients in design optimization also justifies freezing eddy viscosity in an adjoint formulation.

2.4.2 External Unsteady Flow Application

Recently the adjoint method has been extended to airfoil designs with an unsteady flow field involved using the unsteady time domain flow equations (Nadarajah and Jameson, 2002; Mavriplis, 2008) to control the time-averaged aerodynamic performance, unsteady aerodynamics or aeroelasticity. The unsteady adjoint equation requires integration backwards in time from the null adjoint field at the final time to the unknown adjoint field at the initial time. This makes it necessary to record the solution history of the unsteady flow solution for later use during the integration of the unsteady adjoint equation, which consumes a lot of memory and the memory usage can be prohibitive for 3D cases if all data is held in memory. In order to reduce the memory usage overhead, in (Mavriplis, 2008) the flow solution history is written to disk during the flow solution process and read back in the reverse time integration of the adjoint solution process. The time periodicity of the unsteady flow field was exploited by Nadarajah and Jameson (2006) to solve both the unsteady flow equations and the corresponding adjoint equations using the Nonlinear Frequency Domain method. In this application, the unsteady adjoint equations are derived from the unsteady flow equations by the continuous adjoint approach. The two sets of partial differential equations are solved in the same way using the Nonlinear Frequency Domain method. This is a full continuous adjoint approach.

2.4.3 Internal Steady Flow Application

The applications of the adjoint method in design optimization to internal flows have been lagging far behind. The status might be indicated by that an adjoint solution with the 2D Euler flow equations for turbomachinery blading aerodynamics was still a research topic very recently, e.g. (Li et al., 2006b; Arens et al., 2005). There have been increasing efforts made in the applications of the adjoint method in turbomachinery blading aerodynamics. Yang et al. (2003) and Wu et al. (2003) applied the continuous adjoint method to 2D turbomachinery blading aerodynamic design optimization. Wu et al. (2005a) and Papadimitriou and Giannakoglou (2006) developed the continuous adjoint solvers for 3D turbomachinery blading aerodynamic design optimization.

Corral and Gisbert (2006) developed a hand-coded discrete adjoint solver for turbine endwall profiling to reduce the secondary flow losses. The AD tool-TAPENADE was exploited by Duta and Giles (Duta and Giles, 2006; Duta et al., 2007) to generate discrete adjoint solvers to save human effort in adjoint coding development for turbomachinery aerodynamic design optimization.

It is noted that all the reported design optimizations of turbomachinery blades by using the adjoint method have been carried out in a single blade row computational domain. For multi-bladerow turbomachines, the matching between adjacent blade rows has important effects. There are different levels of computational analysis tools for including these bladerow matching/interaction effects, ranging from a fully time-domain URANS to a relatively simple circumferentially mixing-out steady flow treatment (the mixing-plane treatment, first proposed by Denton (1992)). The mixing-plane multi-bladerow steady flow analysis has become a standard industrial tool used in a design environment. Given the importance of the multistage matching, it seems natural to argue that an aerodynamic design optimization should be carried out in a multi-bladerow environment, in order to make an effective impact on practical design processes. Recognizing this need and the current status of the multi-bladerow analysis in practical blading designs, the present work is aimed at developing an adjoint mixing-plane method for aerodynamic design optimization of blades in a multi-bladerow environment using the adjoint method.

2.4.4 Internal Unsteady Flow Application

Florea and Hall (2001) are probably among the first ones to make use of the adjoint method for efficient sensitivity analysis of an unsteady inviscid flow through turbomachinery cascades. The adjoint system is formulated upon the time averaged Euler equations and the corresponding linear harmonic equations. The linearization of the linear harmonic equations to a design variable is quite complicated because the system is already a linearized system. The complexity is dramatically increased if the Navier-Stokes equations are involved. Thomas et al. (2003) used the AD tool-TAF to develop discrete adjoint codes based upon the unsteady flow equations solved by the Harmonic Balance Method (Hall et al., 2002) to model unsteady aerodynamic design sensitivities.

The linear harmonic adjoint approach was developed by Duta (Duta, 2002; Duta et al., 2002) for an efficient blade forced response minimization. In Duta's work, the linear harmonic adjoint equation is used to replace the linear harmonic flow equation to calculate worksum efficiently for a blade undergoing many possible unsteady disturbances of the same frequency and inter-blade phase angle. Therefore the application of the linear harmonic adjoint equation is playing an analysis role.

In this work, an adjoint system is formulated based upon the unsteady flow equations discretized on three particular phases of a period of a time-periodic unsteadiness by the Nonlinear Harmonic Phase Solution method (He, 2008). The adjoint system is used to replace a linearized unsteady flow equation with respect to a design variable. The sensitivities of an objective function representing both aeromechanic and aerodynamic performance to arbitrary many geometry changes can be efficiently evaluated by the adjoint solution. Different from the linear harmonic adjoint approach, the developed adjoint system is able to handle aerodynamics and aeromechanics simultaneously in a design role.

2.5 Summary

In a practical design optimization, it is necessary to consider the viscous effect in a flow field by using the RANS equations for steady flows and the URANS equations for unsteady flows. The adjoint method has been established as a popular and efficient method for gradient calculation at situations where there are a large number of variables and a single objective function. It is noted that no effort has been reported on the use of the adjoint method for blading design optimization in a multi-bladerow environment. The adjoint method is exploited in this work for turbomachinery blading design optimization in a multi-bladerow environment.

Among the various efficient solution methods for the solution of the URANS equations for turbomachinery blading aeromechanics, the Nonlinear Harmonic Phase Solution method is selected to integrate with the adjoint method for efficient gradient calculation for turbomachinery blading aeromechanics related design optimizations. Different from the linear harmonic adjoint formulation by Duta et al. (Duta, 2002; Duta

et al., 2002), this developed adjoint formulation can handle both blading aerodynamics and aeromechanics simultaneously. Compared with the unsteady adjoint approach by Nadarajah and Jameson (2006), the adjoint system here is formulated from the unsteady flow equations at three phases of a period of unsteadiness rather than is derived from the unsteady flow equations first and then discretized to the three phases.

Chapter 3

Steady Nonlinear Flow Model and Solution Methods

The steady flow model and the solution methods are presented in this chapter. The flow model includes the Reynolds Averaged Navier-Stokes equations and the turbulence models for closing the equations. The solution methods include spatial discretization, time-integration and boundary condition specifications.

3.1 Reynolds Averaged Navier-Stokes Equations

The flow solver was developed at Durham University (He et al., 2002; He and Denton, 1994). It has been under extensive development for steady and unsteady flow field calculations and has been validated against experimental data, analytic solutions or benchmark results for its capability of producing high quality steady and unsteady flow fields (He and Ning, 1998; Chen et al., 2001; He et al., 2002; Li and He, 2003; Moffatt et al., 2005; Li and He, 2005). As we all know, turbomachines have rotary components rotating around a shaft. Therefore it is natural and convenient to utilize the flow governing equations in a cylindrical coordinate system. A steady flow field analysis for turbomachinery employs meshes attached to the corresponding blades with all unsteady disturbances from adjacent rows truncated. The steady flow solution inside the whole annulus domain of a turbomachine usually exhibits spatial periodicity if identical blades are equally spaced in the circumference of a turbomachine. It saves considerable CPU time and memory usage when one blade passage domain is used in a computation in conjunction with a periodic boundary condition applied along

the periodic boundaries. Consequently the use of a cylindrical coordinate system will facilitate an easy implementation of the periodic boundary condition (see section 3.3 for details). The RANS equations with velocity components measured in an absolute reference frame in a cylindrical coordinate system are given by

$$\frac{\partial(F - V_x)}{\partial x} + \frac{\partial(G - Uv_g - V_\theta)}{r\partial\theta} + \frac{\partial r\,(H - V_r)}{r\partial r} = S \tag{3.1}$$

where

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v r \\ \rho w \\ \rho e \end{pmatrix}; F = \begin{pmatrix} \rho u \\ \rho u u + p \\ \rho v r u \\ \rho w u \\ \rho h u \end{pmatrix}; G = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v r v + p r \\ \rho w v \\ \rho h v \end{pmatrix}$$

$$H = \begin{pmatrix} \rho w \\ \rho u w \\ \rho v r w \\ \rho w w + p \\ \rho h w \end{pmatrix}; S = \begin{pmatrix} 0 \\ 0 \\ 0 \\ (\rho v v + p)/r \\ 0 \end{pmatrix}; V_x = \begin{pmatrix} 0 \\ \tau_{xx} \\ r\tau_{\theta x} \\ \tau_{rx} \\ u\tau_{xx} + v\tau_{\theta x} + w\tau_{rx} - q_x \end{pmatrix}$$

$$V_\theta = \begin{pmatrix} 0 \\ \tau_{x\theta} \\ r\tau_{\theta\theta} \\ \tau_{r\theta} \\ u\tau_{x\theta} + v\tau_{\theta\theta} + w\tau_{r\theta} - q_\theta \end{pmatrix}; V_r = \begin{pmatrix} 0 \\ \tau_{xr} \\ r\tau_{\theta r} \\ \tau_{rr} \\ u\tau_{xr} + v\tau_{\theta r} + w\tau_{rr} - q_r \end{pmatrix}$$

$$\begin{aligned}
\tau_{xx} &= \frac{2}{3}\mu\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{r\partial\theta} - \frac{\partial rw}{r\partial r}\right) \\
\tau_{\theta\theta} &= \frac{2}{3}\mu\left(2\frac{\partial v}{r\partial\theta} - \frac{\partial u}{\partial x} - \frac{\partial rw}{r\partial r}\right) \\
\tau_{rr} &= \frac{2}{3}\mu\left(2\frac{\partial rw}{r\partial r} - \frac{\partial v}{r\partial\theta} - \frac{\partial u}{\partial x}\right) \\
\tau_{x\theta} &= \tau_{\theta x} = \mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{r\partial\theta}\right) \\
\tau_{xr} &= \tau_{rx} = \mu\left(\frac{\partial w}{\partial x} + \frac{\partial ru}{r\partial r}\right) \\
\frac{\partial q}{\partial x} &= -k\frac{\partial T}{\partial x} \\
\frac{\partial q}{\partial\theta} &= -k\frac{\partial T}{r\partial\theta} \\
\frac{\partial q}{\partial r} &= -k\frac{\partial rT}{r\partial r}
\end{aligned}$$

where μ is the effective viscosity, namely, $\mu = \mu_l + \mu_t$. μ_l is the laminar/molecular viscosity, μ_t is the turbulent/eddy viscosity. In a compressor temperature does not

normally vary much. Hence μ_l is held constant everywhere in the flow field and is calculated via the following formulation:

$$\mu_l = \frac{l_{ref} \cdot \rho_{ref} \cdot V_{ref}}{Re} \tag{3.2}$$

l_{ref} is the reference length for the determination of Reynolds number (Re). It is usually assigned a value of the axial chord length of a blade at its mid span. ρ_{ref} is the reference density, corresponding to the density of the flow field at the mid span region of the inlet plane of a computational domain. V_{ref} is the reference velocity, which is usually set to the relative velocity of the flow field at the mid span region of the inlet plane of a computational domain. Re is the Reynolds number, which needs to be specified as an input. The eddy viscosity μ_t is calculated either by the Baldwin-Lomax turbulence model (Baldwin and Lomax, 1978) or by the Spalart-Allmaras turbulence model (Spalart and Allmaras, 1992). k is the effective heat transfer coefficient, namely, $k = c_p \left(\frac{\mu_l}{Pr_l} + \frac{\mu_t}{Pr_t} \right)$. Pr_l and Pr_t are laminar and turbulent Prandtl numbers and take values of 0.7 and 0.9 respectively. v_g is the grid velocity due to the rotation of a computational domain with the corresponding blade. The definitions of total specific energy, enthalpy and specific heat ratio and the gas state equation are given as follows to close the system of equations:

$$\begin{aligned} e &= c_v T + \frac{1}{2}\left(u^2 + v^2 + w^2\right) \\ h &= c_p T + \frac{1}{2}\left(u^2 + v^2 + w^2\right) = e + \frac{p}{\rho} \\ \gamma &= \frac{c_p}{c_v} \\ p &= \rho R T \end{aligned}$$

3.2 Turbulence Models

There are two turbulence models available in the flow solver. One is the algebraic turbulence model - Baldwin Lomax model (Baldwin and Lomax, 1978), the other is the one equation turbulence model - Spalart Allmaras model (Spalart and Allmaras, 1992).

3.2.1 Baldwin Lomax Model

The Baldwin Lomax model is a mixing length model using a two-layer approach to calculate the eddy viscosity:

$$\mu_t = \begin{cases} \mu_{t_{inner}} & \text{if } y \leq y_{crossover} \\ \mu_{t_{outer}} & \text{if } y > y_{crossover} \end{cases} \tag{3.3}$$

where $y_{crossover}$ is the smallest value of y at which $\mu_{t_{inner}} = \mu_{t_{outer}}$.

The model for the inner region is given by the Prandtl-Van Driest formulation:

$$\mu_t = \rho l^2 \left| \nabla \times \vec{u} \right|$$

with l being the mixing length. The mixing length normally uses an expression in the Van Driest damping function:

$$l = K \left(1 - e^{-\frac{y^+}{A^+}} \right) y$$

where $K = 0.41$ is the Von Karman constant; the parameter A^+ is suggested to be 26.0 for zero pressure gradient flow. The non-dimensional space coordinate y^+ is defined as follows:

$$y^+ = \frac{y}{\mu_l} \sqrt{\rho \tau_w}$$

The outer layer turbulent viscosity is defined by

$$\mu_t = 0.0168 \beta \rho I_0 y_{max} \Gamma_{max}$$

The intermittency function I_0 is expressed by

$$I_0 = \frac{1}{1 + 5.5(0.3y/y_{max})^6}$$

The function Γ is defined by

$$\Gamma = y \left(1 - e^{-\frac{y^+}{A^+}} \right) \left| \nabla \times \vec{u} \right|$$

where y_{max} is the value of y at which Γ achieves its maximum value Γ_{max}. β takes a value of 1.6.

3.2.2 Spalart Allmaras Model

The equation governing the transportation, dissipation and generation of the eddy viscosity is given by

$$\begin{aligned}\frac{\partial \rho\tilde{\nu}u}{\partial x}+\frac{\partial \rho\tilde{\nu}v}{r\partial\theta}+\frac{\partial r\rho\tilde{\nu}w}{r\partial r} &= C_{b1}(1-f_{t2})\tilde{S}\tilde{\nu} \\ &+1.5\{\nabla\cdot[(1+\rho\tilde{\nu})\mu_l]\nabla\tilde{\nu}+C_{b2}\rho\mu_l\nabla\tilde{\nu}\cdot\triangle\tilde{\nu}\} \\ &-\frac{\rho}{\mu_l}\left(C_{w1}f_w-\frac{C_{b1}}{k^2}f_{t2}\right)\left(\frac{\tilde{\nu}\mu_l}{d}\right)^2+f_{t1}\rho\Delta U^2\end{aligned} \tag{3.4}$$

where

$$C_{b1}=0.1355,\ f_{t2}=c_{t3}exp(-c_{t4}\chi^2),\ \tilde{S}=\omega+\frac{\tilde{\nu}\mu_l}{(k\cdot d)^2}f_{v2},\ C_{b2}=0.622$$

$$C_{w1}=\frac{C_{b1}}{k^2}+1.5(1+C_{b2}),\ f_w=g\left(\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right)^{\frac{1}{6}},\ k=0.4187$$

$$f_{t1}=c_{t1}g_t exp\left[-c_{t2}\frac{\omega_t^2}{\Delta U^2}\left(d^2+g_t^2d_t^2\right)\right],\ c_{t1}=1.0,\ c_{t2}=2.0$$

$$c_{t3}=1.2,\ c_{t4}=0.5,\ \chi=\rho\tilde{\nu},\ f_{v2}=1-\frac{\chi}{1+\chi f_{v1}},\ c_{w2}=0.3,\ r=\frac{\tilde{\nu}\mu_l}{\tilde{S}(k\cdot d)^2}$$

$$g_t=min(0.1,\frac{\Delta U}{\omega_t\Delta x_t}),\ \omega=|\nabla\times\vec{u}|,\ c_{w3}=2.0,\ g=r\left[1+c_{w2}(r^5-1)\right]$$

d is the shortest distance from the wall, ω_t is the magnitude of vorticity at the boundary layer trip point, ΔU is the norm of the difference between the velocity at a field point and the velocity at the trip point, and Δx_t is the grid spacing along the wall at the trip point.

$\tilde{\nu}$ is the solution variable in the turbulence model equation. The eddy viscosity is calculated in the following way:

$$\mu_t=\rho\tilde{\nu}\mu_l f_{v1}$$

with

$$f_{v1}=\frac{\chi^3}{\chi^3+C_{v1}^3},\ C_{v1}=7.1,$$

This equation is incorporated into the flow governing equations and discretized and solved in the same way as other equations.

3.3 Boundary Conditions

There are four types of boundary conditions in use. They are: the subsonic inlet boundary condition, the subsonic exit boundary condition, the viscous solid wall boundary condition and the periodic boundary condition.

The RANS equations are normally solved by adding a pseudo time derivative term to form a hyperbolic system. The specification of the boundary conditions at an inlet/exit boundary is closely related to the characteristics of the hyperbolic equations. The number of the boundary conditions to be specified should be equal to the number of characteristics running into a computational domain. This is the basic compatibility requirement. The implementation of the inlet/exit boundary conditions in the present flow solver is not based upon the flow characteristics and is accordingly reflective, though the compatibility requirement is satisfied. This should not be a big issue since the inlet/exit can be normally placed sufficiently far away from the blade leading and trailing edge.

Inlet Boundary Condition

The flow field at an inlet boundary is usually subsonic, with four incoming flow characteristics and one outgoing flow characteristic running across the boundary. The four incoming flow characteristics are: 1) entropy disturbance, convected downstream at a local flow velocity; 2) two vorticity disturbances, also convected downstream at a local flow velocity; 3) an acoustic disturbance, convected downstream at a speed of the local flow velocity plus the local sound speed. The one outgoing flow characteristic is an acoustic disturbance running upstream at a speed of the local flow velocity minus the local sound speed. According to the compatibility requirements, the four incoming flow characteristics require four boundary conditions to be specified at such an inlet. In this present flow solver, total pressure, total temperature and flow angles (in terms of swirl angles and pitch angles) are required. Static pressure at the boundary is extrapolated from the interior domain, then the specified quantities together with the isentropic relation are used to work out other flow variables at the boundary.

Exit Boundary Condition

At an exit boundary, there are four outgoing flow characteristics and one incoming flow characteristic. The four outgoing flow characteristics are the entropy disturbance, two vorticity disturbances and an acoustic disturbance. The one incoming flow characteristic is an acoustic disturbance traveling upstream. One boundary condition needs to be specified to satisfy the compatibility requirement. In the present flow solver, there are three options to specify a static pressure at an exit. One is to specify the static pressure at a hub or a tip with static pressure at other radial locations worked out according to the simple equilibrium equation (Eq.3.5). Another is to specify the static pressure at both a tip and a hub, the static pressure in between is worked out by a linear interpolation. The third is to specify a static pressure profile in the radial direction.

$$\frac{\partial p}{\partial r} = \frac{\rho v^2}{r} \tag{3.5}$$

Solid Wall Boundary Condition

Along a solid wall, whether it is viscous or inviscid, the relative velocity normal to a wall is zero:

$$u \cdot n_x + (v - v_g) \cdot n_\theta + w \cdot n_r = 0$$

This is realized in the code development by setting the convective fluxes to zero for all the five equations, namely,

$$U \cdot [u \cdot n_x + (v - v_g) \cdot n_\theta + w \cdot n_r] = 0 \tag{3.6}$$

If the wall is viscous, the log-law is used to calculate the wall shear stress τ_w so that a slip boundary condition can be applied with reduced mesh density in the near wall region to save computational resources:

$$\tau_w = \frac{1}{2} c_f \rho W^2 \tag{3.7}$$

where

$$c_f = \begin{cases} \frac{2}{\mathrm{Re}} & \mathrm{Re} < 125 \\ -0.001767 + \frac{0.03177}{\ln \mathrm{Re}} + \frac{0.25614}{(\ln \mathrm{Re})^2} & \mathrm{Re} \geq 125 \end{cases}$$

$$\mathrm{Re} = \frac{\rho W \Delta y}{\mu_l}$$

in which W is the velocity magnitude and ρ the fluid density at the first mesh point away from the wall. Δy approximates the normal distance Δn between the first mesh point and the wall.

If the wall is adiabatic, then the following condition for temperature is applied:

$$\frac{\partial T}{\partial n} = 0$$

otherwise temperature is specified. For pressure, it is always assumed that pressure gradient normal to the wall is zero, namely,

$$\frac{\partial p}{\partial n} = 0$$

Periodic Boundary Condition

The periodic boundary condition is the easiest one to be implemented among the four types of boundary conditions for a steady flow solution. As Fig.3.1 shows, the mesh employed in the flow solver assumes periodicity at the periodic boundaries *ab* and *dc*. In the code implementation, a dummy cell is employed for each physical cell along a periodic boundary. The periodic boundary condition can be implemented, for example, by setting the flow variables at dummy cells along the periodic boundaries *ab* or *dc* to flow variables at corresponding physical cells along the periodic boundaries *dc* or *ab*.

3.4 Flow Mixing-plane Treatment

The flow mixing-plane treatment was initially proposed by Denton and Singh (1979) and supplemented by other researchers, like Giles (1991). The flow mixing-plane treatment features two important properties: conservation, which is fundamental and dictated by physical conservation laws, and non-reflectiveness, which is highly desirable due to the small intra-row gap between adjacent blade rows. The conservative property of the treatment requires that the total mass, momentum and energy fluxes be equal across an interface. The non-reflective property guides the right propagation of the flow solution information across an interface, ensuring the conservation of mass, momentum and energy fluxes will be achieved without artificial reflections to corrupt the flow solution in the interior domains.

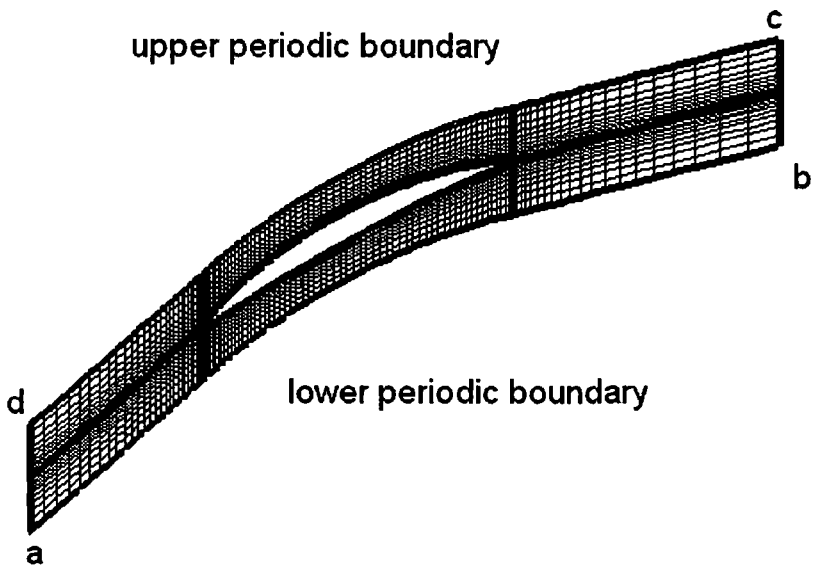


Figure 3.1: Mesh for the illustration of periodic boundary conditions

Fig.3.2 is a schematic of an interface between a rotor domain and a stator domain. The x and y coordinates correspond to the axial direction and the circumferential direction respectively. It is assumed that an interface is normal to the axial direction resulting in the axial-directional flux only through the interface. The mathematical illustration of the conservation property is written as

$$\begin{aligned}
\frac{1}{Y_1}\int_{y_1} \rho u dy &= \frac{1}{Y_2}\int_{y_2} \rho u dy \\
\frac{1}{Y_1}\int_{y_1} (\rho u^2 + p) dy &= \frac{1}{Y_2}\int_{y_2} (\rho u^2 + p) dy \\
\frac{1}{Y_1}\int_{y_1} \rho v u dy &= \frac{1}{Y_2}\int_{y_2} \rho v u dy \\
\frac{1}{Y_1}\int_{y_1} \rho w u dy &= \frac{1}{Y_2}\int_{y_2} \rho w u dy \\
\frac{1}{Y_1}\int_{y_1} \rho h u dy &= \frac{1}{Y_2}\int_{y_2} \rho h u dy
\end{aligned} \tag{3.8}$$

where Y_1 and Y_2 are pitch lengthes of domain 1 and domain 2 respectively. The above

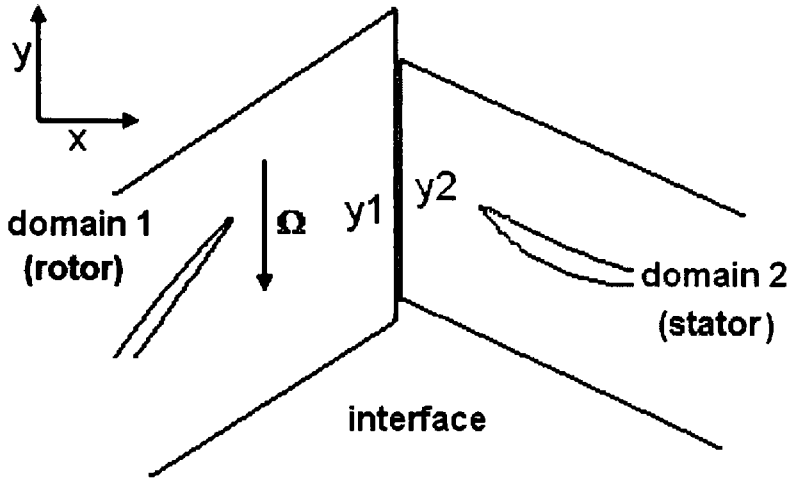


Figure 3.2: An interface between a rotor and a stator computational domains

five equalities can be written in the following shorthand:

$$\begin{aligned} F_{1,1} &= F_{1,2} \\ F_{2,1} &= F_{2,2} \\ F_{3,1} &= F_{3,2} \\ F_{4,1} &= F_{4,2} \\ F_{5,1} &= F_{5,2} \end{aligned} \tag{3.9}$$

The second index denotes the side of an interface. On each side of an interface, once the five circumferentially averaged fluxes are obtained, other intermediate/mean flow variables can be worked out uniquely:

$$\begin{aligned} \bar{v} &= \frac{F_3}{F_1} \\ \bar{w} &= \frac{F_4}{F_1} \\ \bar{h} &= \frac{F_5}{F_1} \\ \bar{p} &= \frac{F_2 + \sqrt{F_2^2 + (\gamma^2 - 1)(F_2^2 + F_3^2 + F_4^2 - 2F_1F_5)}}{\gamma + 1} \\ \bar{u} &= \frac{F_2 - \bar{p}}{F_1} \end{aligned} \tag{3.10}$$

If the five fluxes are equal across an interface, then these intermediate flow variables should also be equal. This is due to the unique mapping between the fluxes and the intermediate flow variables.

At the beginning of a solution process, the fluxes are normally not equal across an interface. Accordingly the intermediate flow variables are unequal across an interface. The difference in these intermediate flow variables can be used to define the corresponding difference in flow characteristics (called characteristic jumps) traveling into a domain. The flow characteristic jumps are defined by

$$\begin{aligned} \phi_1 &= -\rho c \delta \bar{u} + \delta \bar{p} \\ \phi_2 &= \rho c \delta \bar{u} + \delta \bar{p} \\ \phi_3 &= \rho c \delta \bar{v} \\ \phi_4 &= \rho c \delta \bar{w} \\ \phi_5 &= -c^2 \delta \bar{\rho} + \delta \bar{p} \end{aligned} \tag{3.11}$$

where $\delta \bar{q} = \bar{q}_2 - \bar{q}_1$. $\bar{q}$ is an intermediate flow variable as defined in Eq.3.10. $c = \sqrt{\gamma \frac{p}{\rho}}$ is the speed of sound. ϕ_1 represents an acoustic disturbance traveling upstream at a speed of $u - c$, ϕ_2 represents an acoustic disturbance traveling downstream at a speed of $u + c$, ϕ_3 and ϕ_4 both represent vorticity disturbances traveling downstream at a speed of u, ϕ_5 represents an entropy disturbance traveling downstream at a speed of u.

In Fig.3.2, across the interface, one flow characteristic (ϕ_1) travels into domain 1, while four (ϕ_2, ϕ_3, ϕ_4 and ϕ_5) travel into domain 2. For domain 1, the one incoming flow characteristic (ϕ_1) is calculated using the definition in Eq.3.12, while the other four outgoing flow characteristics are extrapolated from the interior domain. For domain 2, the situation is exactly reversed. The four incoming flow characteristics (ϕ_2, ϕ_3, ϕ_4 and ϕ_5) are calculated using the definition in Eq.3.12, while the one outgoing flow characteristic is extrapolated from the interior domain.

Once the five flow characteristic jumps are obtained for one domain, they can be converted to the primitive flow variable perturbations which can be used to update the

local flow solution directly:

$$\begin{aligned}
\delta\rho &= \frac{1}{c^2}\left(-\phi_5 + \frac{1}{2}\phi_1 + \frac{1}{2}\phi_2\right) \\
\delta u &= \frac{1}{2\rho c}\left(\phi_2 - \phi_1\right) \\
\delta v &= \frac{1}{\rho c}\phi_3 \\
\delta w &= \frac{1}{\rho c}\phi_4 \\
\delta p &= \frac{1}{2}\left(\phi_1 + \phi_2\right)
\end{aligned} \tag{3.12}$$

3.5 Discretization of the RANS Equations

A whole-annulus computational domain of a single blade row can be considered as a multiple of identical sub-domains between two adjacent blades. A single passage computational domain is decomposed into a set of non-overlapped hexahedra in 3D as shown in Fig.3.3. A cell-centred finite volume method is employed to solve the flow governing equations. Therefore the control volumes are these hexahedra bounded by surfaces formed by mesh lines. The flow governing equations (Eq.3.1) are integrated over a control volume V_i as follows:

$$\int_{\partial V_i} \left[(F - V_x)\cdot n_x + (G - Uv_g - V_\theta)\cdot n_\theta + (H - V_r)\cdot n_r\right] ds - \int_{V_i} S dv = 0 \tag{3.13}$$

The control volume is a hexahedral in 3D, bounded by the surface of ∂V_i and having a volume of ΔV_i. A discrete form of the above is given by

$$\sum_{j=1}^{ND}\left(\text{Flux}_{j+\frac{1}{2}} - \text{Flux}_{j-\frac{1}{2}}\right) - \Delta V_i S_i = 0 \tag{3.14}$$

in which ND is the dimension of a control volume with $ND = 3$ for 3D control volumes, $\text{Flux}_{j+\frac{1}{2}}$ is the flux through the bounding surface indexed $j + \frac{1}{2}$.

3.5.1 Spatial Discretization

The inviscid flux is calculated using the central difference method with a second order and fourth order blended artificial viscosity. The second order artificial viscosity removes non-physical oscillations around shock regions in a flow field, while the fourth

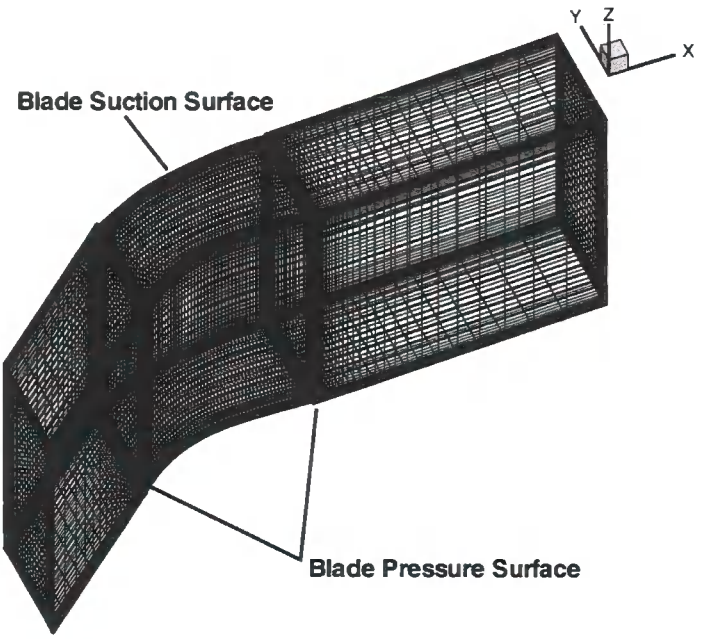


Figure 3.3: A 3D mesh discretizing a blade passage

order artificial viscosity ensures stability in a smooth region. The formulation of the artificial viscosity follows the one introduced by Jameson et al. (1981), but is specially catered for turbomachinery flow field simulations, therefore it is slightly different from the original one in terms of some adjustable factors.

For clarity and simplicity, a 2D mesh is sketched in Fig.3.4 to illustrate the calculation of the convective flux through a bounding surface. The algorithm is equally applicable to 3D control volumes. Along the bounding surface indexed $(i+\frac{1}{2},j)$ of the control volume indexed (i,j), the inviscid flux is expressed as follows:

$$\text{Flux}_{i+\frac{1}{2},j} = F_{i+\frac{1}{2},j} \cdot S_x + (G - Uv_g)_{i+\frac{1}{2},j} \cdot S_\theta + H_{i+\frac{1}{2},j} \cdot S_r + d_{i+\frac{1}{2},j} \tag{3.15}$$

in which S_x, S_θ and S_r are projections of the area of the bounding surface indexed $(i+\frac{1}{2},j)$ onto the x, θ and r directions respectively.

$$\begin{aligned} F_{i+\frac{1}{2},j} &= \frac{1}{2}\left[F(U_{i,j}) + F(U_{i+1,j})\right] \\ (G - Uv_g)_{i+\frac{1}{2},j} &= \frac{1}{2}\left[G(U_{i,j}) + G(U_{i+1,j}) - v_g(U_{i,j} + U_{i+1,j})\right] \\ H_{i+\frac{1}{2},j} &= \frac{1}{2}\left[H(U_{i,j}) + H(U_{i+1,j})\right] \\ d_{i+\frac{1}{2},j} &= \epsilon^{(2)}_{i+\frac{1}{2},j}\Delta_{i+\frac{1}{2},j} - \epsilon^{(4)}_{i+\frac{1}{2},j}\left(\Delta_{i+\frac{3}{2},j} - 2\Delta_{i+\frac{1}{2},j} + \Delta_{i-\frac{1}{2},j}\right) \end{aligned} \tag{3.16}$$

where

$$\Delta_{i+\frac{1}{2},j} = \Delta Vt_{i+\frac{1}{2},j}\left(U_{i+1,j} - U_{i,j}\right)$$

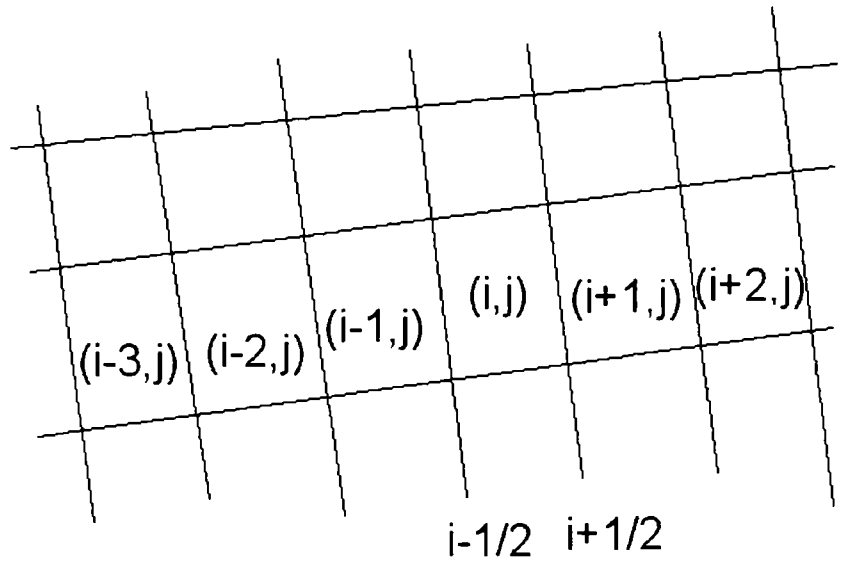


Figure 3.4: A 2D mesh for the illustration of inviscid flux calculations

$$\Delta Vt_{i+\frac{1}{2},j} = \frac{1}{2}\left(\frac{\Delta V_{i+1,j}}{\Delta t_{i+1,j}} + \frac{\Delta V_{i,j}}{\Delta t_{i,j}}\right)$$

$$\epsilon^{(2)}_{i+\frac{1}{2},j} = max(min(0.05, c_2 v_{i+\frac{1}{2},j}), 0.0075)$$

$$\epsilon^{(4)}_{i+\frac{1}{2},j} = max(0, c_4 - \frac{c}{c_2}\epsilon^{(2)}_{i+\frac{1}{2},j})$$

where c is a factor to bend the second and fourth order smoothing effects, c_2 is the second order smoothing factor with a typical value of 1.0, c_4 is the fourth order smoothing factor with a typical value of $\frac{1}{32}$.

$$v_{i+\frac{1}{2},j} = max(v_{i-1,j}, v_{i,j}, v_{i+1,j}, v_{i+2,j})$$

$$v_{i,j} = \left|\frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{p_{i+1,j} + 2p_{i,j} + p_{i-1,j}}\right|$$

An auxiliary staggered mesh is used to calculate the viscous fluxes. An auxiliary mesh line lies exactly in the middle of two primary mesh lines. In Fig.3.5, the solid lines constitute the primary mesh, while dash-dotted lines constitute the auxiliary mesh. Viscous fluxes are functions of flow variable spatial derivatives. These flow variable spatial derivatives are calculated using Green's theorem. Inside the control volume formed by bounding surfaces ef, fg, gh and he, its averaged flow variable spatial

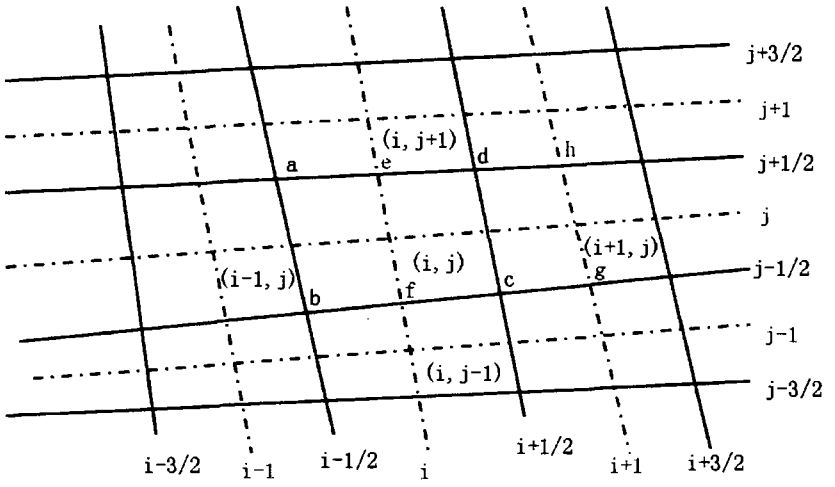


Figure 3.5: A primary mesh (solid lines) and an auxiliary mesh (dash-dotted lines) for the illustration of viscous flux calculations

derivatives can be obtained in the following way:

$$\frac{\partial u}{\partial x} = \frac{1}{\Delta V_{efghe}} \int_{V_{efghe}} \frac{\partial u}{\partial x} dv = \frac{1}{\Delta V_{efghe}} \oint_{efghe} u \cdot n_x ds \tag{3.17}$$

where

$$\Delta V_{efghe} = \Delta V_{i+\frac{1}{2},j} = \frac{\Delta V_{i+1,j} + \Delta V_{i,j}}{2}$$

Along the bounding surfaces ef and gh, the flow variable u takes the value of $u_{i,j}$ and $u_{i+1,j}$ respectively. While along the bounding surfaces of fg and he the flow variable takes the arithmetic average of the flow solution at the four control volumes sharing the vertex c and d respectively. Replacing u with other flow variables and x with other coordinates, other flow variable spatial derivatives can be obtained in the same way. Once those flow variable spatial derivatives are calculated appropriately, it is straightforward to substitute them to those viscous fluxes V_x, V_θ and V_r as defined in Eq.3.1.

3.5.2 Temporal Discretization

In order to solve Eqs.3.14 or 3.1 using the time-marching method, a pseudo time derivative term is added to these equations:

$$\frac{\partial (U \Delta V)}{\partial \tau} + R(U) = 0$$

where $R(U)$ represents the left hand side of Eq.3.13 and is called the flow residual. The flow residual $R(U)$ is the summation of the net convective/inviscid flux, the net viscous flux, the net artificial viscous flux and the source term, namely,

$$R(U) = R_i(U) + R_v(U) + R_a(U) - S(U)$$

Time integration of the above equation is achieved using the four stage Runge-Kutta method. In order to reduce time cost, the viscous and artificial viscous residuals are evaluated at the first stage only but used at all four stages of a complete Runge-Kutta step.

$$\begin{aligned}
U^{n+\frac{1}{4}} &= U^n - \frac{1}{4}\frac{\Delta\tau}{\Delta V}[R_i(U^n) + R_v(U^n) + R_a(U^n) - S(U^n)] \\
U^{n+\frac{1}{3}} &= U^n - \frac{1}{3}\frac{\Delta\tau}{\Delta V}[R_i(U^{n+\frac{1}{4}}) + R_v(U^n) + R_a(U^n) - S(U^{n+\frac{1}{4}})] \\
U^{n+\frac{1}{2}} &= U^n - \frac{1}{2}\frac{\Delta\tau}{\Delta V}[R_i(U^{n+\frac{1}{3}}) + R_v(U^n) + R_a(U^n) - S(U^{n+\frac{1}{3}})] \\
U^{n+1} &= U^n - \frac{\Delta\tau}{\Delta V}[R_i(U^{n+\frac{1}{2}}) + R_v(U^n) + R_a(U^n) - S(U^{n+\frac{1}{2}})] \qquad (3.18)
\end{aligned}$$

Techniques, like the local time-stepping and the multi-grid (He and Denton, 1994) are implemented in the flow solver to accelerate the solution process.

3.6 Summary

The RANS equations in a cylindrical coordinate system are used in turbomachinery flow field simulations. The RANS equations are closed by either the Baldwin-Lomax turbulence model or the Spalart-Allmaras turbulence model. The flow mixing-plane treatment is also implemented in the flow solver to allow a steady flow field analysis to be carried out in a multi-bladerow environment. The flow mixing-plane treatment features two important properties: conservation and non-reflectiveness. The equations are discretized in a cell-centered finite volume framework. The inviscid flux is calculated by the central difference method together with a blended second order and forth order artificial viscosity. Time-marching in the pseudo time is achieved by the four-stage Runge-Kutta method with the acceleration approaches of the local time-stepping and multi-grid.

Chapter 4

Development of Steady Flow Adjoint System

The chapter starts with a concise illustration of the adjoint formulation principle. Then a steady adjoint system based on the Euler flow equations is derived by the continuous adjoint approach. The derivation of the adjoint equations follows the same way as that by Jameson (2003), Giles and Pierce (2000) and Arens et al. (2005). The eddy viscosity is frozen in the adjoint formulation for simplicity. Gradient calculations are conducted at a frozen eddy viscosity, whilst it will be updated in the direct flow solution once in every design cycle. An adjoint mixing-plane treatment is introduced corresponding to the flow mixing-plane treatment to enable the steady adjoint system to be solved in a multi-bladerow environment. The solution methods for the derived adjoint system are also presented.

4.1 Adjoint Formulation Principle

In a CFD-based design optimization, there is normally a scalar objective function I, which is an explicit function of the mesh coordinate vector X and the flow variable vector U:

$$I = I(X, U) \tag{4.1}$$

The mesh coordinate vector X is usually determined by design variables parameterizing a blade profile. Let α be a design variable. The total derivative of the objective function (Eq.4.1) to the design variable α can be expressed as follows using the chain rule:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial X}\frac{\partial X}{\partial \alpha} + \frac{\partial I}{\partial U}\frac{\partial U}{\partial X}\frac{\partial X}{\partial \alpha} \tag{4.2}$$

The relation between U and X is determined by the nonlinear flow equations which can be symbolized as

$$R(U, X) = 0 \tag{4.3}$$

If a mesh is generated in an algebraic way, then $\frac{\partial X}{\partial \alpha}$ can be obtained efficiently without solving any equations. Even if a mesh is generated in an iterative way, there are ways, as pointed out in Chapter 1, to obtain this gradient $\frac{\partial X}{\partial \alpha}$ efficiently (through an efficient mesh movement/perturbation strategy without regenerating the new mesh in an iterative way). I is an explicit function of X and U, therefore $\frac{\partial I}{\partial X}$ and $\frac{\partial I}{\partial U}$ can normally be obtained in an analytic way. The only term left in the gradient expression of $\frac{dI}{d\alpha}$ is $\frac{\partial U}{\partial X}$. The solution of U is normally achieved through solving Eq.4.3 in a costly iterative process, implying an equivalent costly iterative process is required to obtain $\frac{\partial U}{\partial X}$. Now U and X are vectors, therefore $\frac{\partial U}{\partial X}$ is a matrix of a very big size, i.e. having millions of elements for a common three dimensional case. The calculation of every element of the matrix requires one separate solution of the flow equations. It is quite normal that in a CFD calculation the number of design variables is much fewer than the number of all the mesh points. It is therefore much more efficient to calculate the product of $\frac{\partial U}{\partial X}\frac{\partial X}{\partial \alpha}$ in terms of $\frac{\partial U}{\partial \alpha}$ with the number of calculations reduced from the magnitude of the square of the number of mesh points to the number of design variables. This is also the approach in which the flow variable sensitivity is usually calculated by the finite difference method or the linearized method. The gradient expression (Eq.4.2) is rewritten as

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} + \frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha} \tag{4.4}$$

Similar to the definition of $\frac{\partial U}{\partial \alpha}$, $\frac{\partial I}{\partial \alpha} = \frac{\partial I}{\partial X}\frac{\partial X}{\partial \alpha}$. The gradient $\frac{\partial U}{\partial \alpha}$, usually called the flow variable sensitivity, can be either approximated using the finite difference method by solving the nonlinear flow equation after a design variable is perturbed a small amount of $\Delta\alpha$:

$$\frac{\partial U}{\partial \alpha} \doteq \frac{U(\alpha + \Delta\alpha) - U(\alpha)}{\Delta\alpha} \tag{4.5}$$

or approximated using the complex variable method (Vatsa, 1999) by solving the nonlinear flow equation in a complex variable manner after a design variable is perturbed a small amount of $i\Delta\alpha$:

$$\frac{\partial U}{\partial \alpha} \doteq Imag(\frac{U(\alpha + i\Delta\alpha) - U(\alpha)}{\Delta\alpha}) \tag{4.6}$$

with *Imag* denoting the imaginary part of a complex variable. The gradient can also be obtained using the linearized method by solving the linearized flow equation [1]:

$$\frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} + \frac{\partial R}{\partial \alpha} = 0 \tag{4.7}$$

Among the above three approaches, the finite difference method suffers from step size dependence; the complex variable method normally costs twice as the other two. In addition, the cost of gradient evaluation by any of the three approaches is directly proportional to the number of design variables making it prohibitive to use these methods in a situation with a large number of design variables, because both the nonlinear flow equations and the linearized flow equations are directly related to design variables. For each design variable, these equations need to be solved once to get the flow variable sensitivity to that particular design variable.

To start with the adjoint method, multiplying the left hand side of the linearized flow equation (Eq.4.7) with a vector variable λ called the adjoint variable or the Lagrange multiplier, and then subtracting this product from the gradient expression (Eq.4.4) yields an augmented gradient expression:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha} + \frac{\partial I}{\partial \alpha} - \lambda^T\left(\frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} + \frac{\partial R}{\partial \alpha}\right) \tag{4.8}$$

No matter what value the variable λ holds, the product $\lambda^T\left(\frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} + \frac{\partial R}{\partial \alpha}\right)$ is always zero because of the vanishing of the other term. Regrouping the augmented gradient expression by collecting like terms with $\frac{\partial U}{\partial \alpha}$ yields

$$\frac{dI}{d\alpha} = \left(\frac{\partial I}{\partial U} - \lambda^T\frac{\partial R}{\partial U}\right)\frac{\partial U}{\partial \alpha} + \frac{\partial I}{\partial \alpha} - \lambda^T\frac{\partial R}{\partial \alpha} \tag{4.9}$$

One has the freedom to choose the value of the adjoint variable, it is therefore chosen in such a way that the following equation will be satisfied:

$$\frac{\partial I}{\partial U} - \lambda^T\frac{\partial R}{\partial U} = 0 \tag{4.10}$$

This Equation is called the adjoint equation. With the adjoint variable satisfying the adjoint equation (Eq.4.10), the gradient expression (Eq.4.9) is reduced to one independent of the flow variable sensitivity:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} - \lambda^T\frac{\partial R}{\partial \alpha} \tag{4.11}$$

[1] In contrast to the time linearized equation, where the linearization is conducted with respect to time, the linearization in this context is done with respect to a design variable.

Like the linearized flow equation (Eq.4.7), the adjoint equation (Eq.4.10) is also a set of linear system with the flow solution to Eq.4.3 forming the variable coefficients of the equation. Different from Eq.4.7 which is related to a particular design variable, Eq.4.10 is free from any design variable. This means only one set of the adjoint equation of the same size of the flow equation will be solved for all design variables. Though the new gradient expression (Eq.4.11) is related to design variables, the calculation of $\frac{\partial R}{\partial \alpha}$ for hundreds of design variables does not normally exceed one flow/adjoint solution.

The adjoint calculation of objective function gradients is conducted in the following sequence: the flow equation is solved first, then the adjoint equation is solved accordingly, finally the gradients of an objective function can be obtained by some post-processing with the known flow and adjoint solutions. In a continuous adjoint method, the cost of gradient evaluation is dominated by solving the two sets of equations at a cost roughly equal to two flow solutions.

It should be noted both the flow equation (Eq.4.3) and the adjoint equation (Eq.4.10) contain both the field equation and its corresponding boundary condition. In practice, the field equation and its boundary condition are normally treated separately in different ways. Nevertheless the principle is the same: eliminating the dependence of relevant terms on the flow variable sensitivity.

4.2 Steady Adjoint Equations

This section presents the detailed derivation of the adjoint equations from the flow equations in a partial differential equation form. The derivation of the adjoint equations based on the RANS equations (Eq.3.1) is given in Appendix A. The following derivation is based on the Euler flow equations, aimed at a simpler illustration of the principal procedure of a continuous adjoint formulation.

In the present work, the objective function is a weighted sum of mass flow rate, pressure ratio and entropy generation rate, which can be expressed as a boundary integral in the following general formulation:

$$I = \int_{\partial D} M(U, X) ds \tag{4.12}$$

where M is a function of the flow variable vector U and the mesh coordinate vector X. The gradient of the objective function to a design variable is given by

$$\frac{dI}{d\alpha} = \int_{\partial D} \left(\frac{\partial M}{\partial U} \tilde{U} + \frac{\partial M}{\partial \alpha} \right) ds + \int_{\partial D} M \tilde{ds} \tag{4.13}$$

in which α is a design variable, $\tilde{U} = \frac{\partial U}{\partial \alpha} = \frac{\partial U}{\partial X}\frac{\partial X}{\partial \alpha}$ and $\frac{\partial M}{\partial \alpha} = \frac{\partial M}{\partial X}\frac{\partial X}{\partial \alpha}$, $\tilde{ds} = \frac{d(ds)}{d\alpha}$ accounts for the boundary geometry perturbation due to a perturbation in the design variable. In the above gradient expression, $\frac{\partial M}{\partial U}$ and $\frac{\partial M}{\partial \alpha}$ can be obtained without solving any equations, thus their corresponding time cost is very small. However the calculation of $\tilde{U}$, which is called the flow variable sensitivity, has to involve solving equations. $\tilde{U}$ can be either approximated using the finite difference method [1]:

$$\frac{\partial U}{\partial \alpha} = \frac{U(\alpha + \Delta\alpha) - U(\alpha)}{\Delta\alpha}$$

by solving the nonlinear flow governing equations after the design variable α is perturbed $\Delta\alpha$, or obtained by solving the linearized flow equations:

$$\frac{\partial\left(A\tilde{U}\right)}{\partial x} + \frac{\partial\left(B\tilde{U} - v_g\tilde{U}\right)}{r\partial\theta} + \frac{\partial r\left(C\tilde{U}\right)}{r\partial r} - D\tilde{U} = f \tag{4.14}$$

where $A = \frac{\partial F}{\partial U}$, $B = \frac{\partial G}{\partial U}$, $C = \frac{\partial H}{\partial U}$ and $D = \frac{\partial S}{\partial U}$ are Jacobian matrices. f contains the linearization of geometric terms in the flow governing equations to the design variable α, corresponding to $-\frac{\partial R}{\partial \alpha}$ in Eq.4.7. Both the finite difference method and the linearized method have the drawback of a proportional increase of time cost to the number of design variables. This is to say, either the nonlinear flow equations or the linearized flow equations have to be solved at least n times for n design variables to get flow variable sensitivities to all the n design variables.

In order to avoid solving Eq.4.14 n times for n design variables, we resort to the adjoint method, which offers a means to calculate the objective function sensitivity at a low time cost independent of the number of design variables. Multiplying two sides of Eq.4.14 with the adjoint variable λ, and rearranging yields

$$\lambda^T \left[\frac{\partial\left(A\tilde{U}\right)}{\partial x} + \frac{\partial\left(B\tilde{U} - v_g\tilde{U}\right)}{r\partial\theta} + \frac{\partial r\left(C\tilde{U}\right)}{r\partial r} - D\tilde{U} - f \right] = 0 \tag{4.15}$$

[1]The gradient can also be calculated using the central or forward scheme, moreover it can be calculated using the complex variable method. All these methods share the same principle.

where $\lambda = (\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5)^T$. $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ and λ_5 correspond to the continuity equation, the axial momentum equation, the moment of momentum equation, the radial momentum equation and the energy equation, respectively. The transpose of the adjoint variable λ is used because the linearized equation and the adjoint variable are both column vectors. The above product yields a scalar quantity. Eq.4.15 is valid everywhere in a computational domain, no matter what value the adjoint variable will take. Integrating the above equation over the whole computational domain, one has

$$\int_D \lambda^T \left[\frac{\partial\left(A\tilde{U}\right)}{\partial x} + \frac{\partial\left(B\tilde{U} - v_g\tilde{U}\right)}{r\partial\theta} + \frac{\partial r\left(C\tilde{U}\right)}{r\partial r} - D\tilde{U} - f \right] dv = 0 \quad (4.16)$$

Performing integration by parts once yields

$$\begin{aligned} &\int_{\partial D} \lambda^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right] \tilde{U} ds \\ &- \int_D \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial\theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv - \int_D \lambda^T f dv = 0 \end{aligned} \quad (4.17)$$

where I is a 5×5 identity matrix. Subtracting the left hand side of Eq.4.17 from the gradient expression (Eq.4.13) will not change the gradient value, namely,

$$\begin{aligned} \frac{dI}{d\alpha} = &\int_{\partial D} \left(\frac{\partial M}{\partial U}\tilde{U} + \frac{\partial M}{\partial \alpha} \right) ds + \int_{\partial D} M\tilde{ds} \\ &- \int_{\partial D} \lambda^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right] \tilde{U} ds \\ &+ \int_D \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial\theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv + \int_D \lambda^T f dv \end{aligned} \quad (4.18)$$

Grouping the domain integral terms with $\tilde{U}$ and boundary integral terms with $\tilde{U}$, one has

$$\begin{aligned} \frac{dI}{d\alpha} = &\int_D \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial\theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv \\ &- \int_{\partial D} \left[\lambda^T A \cdot n_x + \lambda^T (B - v_g I) \cdot n_\theta + \lambda^T C \cdot n_r - \frac{\partial M}{\partial U} \right] \tilde{U} ds \\ &+ \int_{\partial D} \frac{\partial M}{\partial \alpha} ds + \int_{\partial D} M\tilde{ds} + \int_D \lambda^T f dv \end{aligned} \quad (4.19)$$

Returning to the original objective of developing an adjoint system, the gradient for n design variables can be efficiently evaluated if we do not have to solve the flow field n times. This can be achieved if the explicit dependence of the objective function gradient on the flow variable sensitivity can be eliminated. In order to achieve this goal, the

first two integral terms on the right hand side of Eq.4.19 should vanish, leading to the following two equations:

$$\int_D \left[\frac{\partial \lambda^T}{\partial x}A + \frac{\partial \lambda^T}{r\partial \theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r}C + \lambda^T D\right]\tilde{U}dv = 0 \tag{4.20}$$

$$\int_{\partial D} \left[\lambda^T A \cdot n_x + \lambda^T (B - v_g I) \cdot n_\theta + \lambda^T C \cdot n_r - \frac{\partial M}{\partial U}\right]\tilde{U}ds = 0 \tag{4.21}$$

Inside a computational domain, at every point, $\tilde{U}$ cannot always be zero, the necessary and sufficient condition to satisfy Eq.4.20 is to vanish the term multiplying $\tilde{U}$, leading to the adjoint Euler equations:

$$\frac{\partial \lambda^T}{\partial x}A + \frac{\partial \lambda^T}{r\partial \theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r}C + \lambda^T D = 0 \tag{4.22}$$

Along the boundary of a computational domain, however $\tilde{U}$ can always be zero somewhere, e.g. on a viscous solid wall, the flow velocity is always zero. Therefore it is not necessary, though sufficient, to vanish the term multiplying $\tilde{U}$ in Eq.4.21. The flow boundary conditions need to be taken into account to derive the necessary and sufficient conditions: adjoint boundary conditions, to satisfy Eq.4.21. These boundary conditions will be detailed in the following adjoint boundary condition section. The remaining terms including both domain integral terms and boundary integral terms in Eq.4.19 are independent of the flow variable sensitivity $\tilde{U}$. These terms give the final expression of the objective function gradient in an intended form, independent of the flow variable sensitivity:

$$\frac{dI}{d\alpha} = \int_{\partial D} \frac{\partial M}{\partial \alpha}ds + \int_{\partial D} M\tilde{ds} + \int_D \lambda^T f dv \tag{4.23}$$

The values of the adjoint variables are determined through a solution to the adjoint equation (Eq.4.22). No matter how many design variables, there are now only two sets of equations to be solved within each design cycle: the non-linear flow equations and the linear adjoint equations. The time cost of solving the adjoint equations is roughly equal to that for the baseline flow equations. Once the adjoint solution is obtained, it can be substituted to the above gradient expression to calculate the gradients efficiently.

4.3 Boundary Conditions

It should be remembered that the symbolized flow equation (Eq.4.3) contains both the field equation inside a domain and its boundary condition. Accordingly the adjoint equation (Eq.4.10) also contains both the field equation and its boundary condition. In practice, a field equation and its boundary condition are normally treated differently. This is why this separate section for the adjoint boundary condition of the adjoint system based on the Euler flow equation comes to exist. Nevertheless, the principle of the derivation of both the adjoint field equation and its boundary condition is the same: eliminating the dependence of relevant terms on the flow variable sensitivity.

As pointed out in the previous section, the derivation of the adjoint boundary conditions is to find the necessary and sufficient conditions which satisfy Eq.4.21 by taking into account the relevant flow boundary conditions at a boundary. Corresponding to the four sets of flow boundary conditions, there are also four sets of adjoint boundary conditions: the subsonic inlet boundary condition, the subsonic exit boundary condition, the solid wall boundary condition and the periodic boundary condition. These four sets of boundaries will be derived separately in the following.

An important element in the boundary condition treatment is the equation and its characteristics. Eq.4.22 is usually solved in a time-marching manner with a pseudo time derivative term added to the equation to form a hyperbolic system. For a hyperbolic system, the specification of boundary conditions is based on the propagation of its characteristic information. The basic compatibility requirement is that the number of boundary conditions specified at a boundary must be the same as the number of the characteristics running into a computational domain through this boundary. Hence the directions of the characteristics matter. As Giles and Pierce (Giles and Pierce, 1997, 1998) point out, the sign of each adjoint characteristic velocity is opposite to that of its corresponding flow characteristic velocity. It can also be put in this way: for each flow characteristic, there is a corresponding adjoint flow characteristic traveling at a speed of the same magnitude but in an opposite direction. Overall it is useful to note that in a simple but consistent sense, the adjoint information propagates in exactly the opposite way to the physical flow information. This 'anti-physics' path of

information propagation in fact governs all the boundary condition treatments for the adjoint equations, described in this section and the next section on the adjoint mixing-plane treatment. This will also be illustrated further in the numerical examples, to be presented in Chapter 6.

Inlet Boundary Condition

At a subsonic inlet boundary for the flow equations, there are four incoming characteristics and one outgoing characteristic. Typically the static pressure at the boundary is extrapolated from the interior domain, and other flow variables are determined through the isentropic relation and specified quantities. Thus at the inlet boundary, pressure is the only independent flow variable. The perturbation of any quantity at this boundary can be expressed as the derivative of the quantity to the pressure times the pressure perturbation. Eq.4.21 can be rewritten as follows at an inlet boundary:

$$\int_{\partial D} \left(\lambda^T \frac{\partial F}{\partial p} \cdot n_x - \frac{\partial M}{\partial p} \right) \tilde{p} ds = 0 \tag{4.24}$$

in which an inlet plane is assumed to be normal to the x axis, resulting in the x-directional flux only through the boundary. The corresponding sufficient and necessary condition to satisfy the above is applied as follows ($\tilde{p}$ can not always be zero):

$$\lambda^T \frac{\partial F}{\partial p} \cdot n_x - \frac{\partial M}{\partial p} = 0 \tag{4.25}$$

Eq.4.25 is a scalar equation, corresponding to one boundary condition. This is consistent with one incoming and four outgoing characteristics for the adjoint equations at a subsonic inlet boundary. The adjoint variables $\lambda_{1,2,3,4}$ are extrapolated from the interior domain, while λ_5 is worked out according to Eq.4.25. The derivative of the flux vector to the pressure ($\frac{\partial F}{\partial p}$) can be obtained via the complex variable method (Vatsa, 1999), while $\frac{\partial M}{\partial p}$ can be normally obtained in an analytic way.

Exit Boundary Condition

At a subsonic exit boundary, there is one incoming and four outgoing characteristics for the flow equations. In the present flow solver, the density and three velocity components are extrapolated from the interior domain, and static pressure is either specified or

determined through the static pressure specified at a hub or tip with the simple radial equilibrium distribution or by a linear interpolation. This means at a flow exit, density and velocity components are independent flow variables. The flux perturbation at an exit boundary can be expressed as follows:

$$\tilde{F} = \frac{\partial F}{\partial \rho}\tilde{\rho} + \frac{\partial F}{\partial u}\tilde{u} + \frac{\partial F}{\partial v}\tilde{v} + \frac{\partial F}{\partial w}\tilde{w}$$

At an exit boundary Eq.4.21 is reduced to the following one:

$$\begin{aligned}\int_{\partial D}\Big[&(\lambda^T\frac{\partial F}{\partial \rho}\cdot n_x - \frac{\partial M}{\partial \rho})\tilde{\rho} + (\lambda^T\frac{\partial F}{\partial u}\cdot n_x - \frac{\partial M}{\partial u})\tilde{u} \\ &+(\lambda^T\frac{\partial F}{\partial v}\cdot n_x - \frac{\partial M}{\partial v})\tilde{v} + (\lambda^T\frac{\partial F}{\partial w}\cdot n_x - \frac{\partial M}{\partial w})\tilde{w}\Big]\, ds = 0\end{aligned} \tag{4.26}$$

The corresponding sufficient and necessary condition takes the following form:

$$\lambda^T\frac{\partial F}{\partial q}\cdot n_x - \frac{\partial M}{\partial q} = 0 \qquad q = \rho, u, v, w \tag{4.27}$$

where again an exit plane is assumed to be normal to the x axis. Eq.4.27 represents 4 scalar equations, corresponding to four boundary conditions. This is in line with the four incoming and one outgoing adjoint characteristics at a subsonic exit boundary. The adjoint variable λ_5 is extrapolated from the interior domain, other four adjoint variables are determined through Eq.4.27.

Solid Wall Boundary Condition

At a solid wall boundary, whether it is slip or non-slip, it is impermeable. Eq.3.6 can also be written in the following shorthand:

$$U \cdot v_n = 0 \tag{4.28}$$

where $v_n = u \cdot n_x + (v - v_g) \cdot n_\theta + w \cdot n_r$ is the relative velocity normal to a wall surface. The flux through a control face can be split by

$$F \cdot n_x + (G - v_g U) \cdot n_\theta + H \cdot n_r \;=\; U \cdot v_n + \begin{pmatrix} 0 \\ p \cdot n_x \\ p \cdot r \cdot n_\theta \\ p \cdot n_r \\ p \cdot v_g \cdot n_\theta \end{pmatrix} \tag{4.29}$$

At a solid wall boundary, the above flux is reduced as follows:

$$F \cdot n_x + (G - v_g U) \cdot n_\theta + H \cdot n_r = \begin{pmatrix} 0 \\ p \cdot n_x \\ p \cdot r \cdot n_\theta \\ p \cdot n_r \\ p \cdot v_g \cdot n_\theta \end{pmatrix} \tag{4.30}$$

With the above substituted into Eq.4.21, one has

$$\int_{\partial D} \left(\lambda_2 n_x + r\lambda_3 n_\theta + \lambda_4 n_r + \lambda_5 v_g n_\theta - \frac{\partial M}{\partial p} \right) \tilde{p} ds = 0 \tag{4.31}$$

The corresponding sufficient and necessary condition of the above equality is to vanish the term before $\tilde{p}$:

$$\lambda_2 n_x + r\lambda_3 n_\theta + \lambda_4 n_r + \lambda_5 v_g n_\theta = \frac{\partial M}{\partial p} \tag{4.32}$$

Corresponding to a slip boundary condition with the log-law for the wall shear stress calculation at a viscous solid wall boundary, the following extra conditions need to be applied:

$$\begin{aligned} \sigma_x &= \frac{\tau_w}{W}\lambda_2 \\ \sigma_\theta &= \frac{\tau_w}{W}(\lambda_3 + \Omega \cdot \lambda_5) \\ \sigma_r &= \frac{\tau_w}{W}\lambda_4 \\ \frac{\partial \lambda_5}{\partial n} &= 0 \quad \text{for an adiabatic wall} \\ \lambda_5 &= 0 \quad \text{for a non-adiabatic wall} \end{aligned} \tag{4.33}$$

Ω is the angular speed of the rotating mesh associated with a blade. τ_w is the wall shear stress calculated by the log-law. W is the magnitude of the velocity along the slip wall. Definition of σ_x, σ_θ and σ_r is given by

$$\sigma_i = \sigma_{xi} \cdot n_x + \sigma_{\theta i} n_\theta + \sigma_{ri} n_r \qquad i = x, \theta, r \tag{4.34}$$

Definition of σ_{ij} can be found in Appendix A. The derivation of this part should be based upon the Navier-Stokes equation and is detailed in Appendix B.

Periodic Boundary Condition

Periodic boundaries come in pair. They are not physical boundaries, but artificial ones employed to reduce the size of a computational problem. Therefore an objective

function does not explicitly depend on the flow variables along periodic boundaries. That is to say, $M = 0$. Eq.4.21 is reduced as

$$\int_{\partial D} \lambda^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right] \tilde{U} ds = 0 \tag{4.35}$$

At periodic boundaries, the pair of boundaries should be considered together rather than be considered separately. Referring to Fig.3.1, the terms appearing on the boundaries ab and dc are written by

$$\begin{aligned} &\int_{ab} \lambda^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right] \tilde{U} ds \\ &+ \int_{cd} \lambda^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right] \tilde{U} ds = 0 \end{aligned} \tag{4.36}$$

Along a pair of periodic boundaries, the flow variables are equal. The only difference is that the outward norm of one periodic boundary is just opposite to its corresponding one, therefore the above is equivalent to

$$\begin{aligned} &\int_{ab} \lambda_{ab}^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right]_{ab} \tilde{U}_{ab} ds \\ &= \int_{ab} \lambda_{dc}^T \left[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r\right]_{ab} \tilde{U}_{ab} ds \end{aligned} \tag{4.37}$$

It is apparent that the periodic boundary condition is given by

$$\lambda_{ab} = \lambda_{dc}$$

The implementation of the adjoint periodic boundary condition is identical to the flow periodic boundary condition.

4.4 Adjoint Mixing-plane Treatment

One of the main novel points of the present work is to enable the adjoint method based design optimization to be carried out in a multi-bladerow environment. The basis of the present development is the mixing-plane treatment for flow solutions after Denton and Singh (1979). There are two basic features in the mixing-plane approach. The first is to satisfy the conservation of the total mass, momentum and energy, which is fundamentally required. The second is to be non-reflective, which is highly desirable when considering the small intra-row gaps. The present adjoint mixing-plane treatment is also aimed at having these two features.

Along an interface, the boundary integral in Eq.4.21 appears in a pair and needs to be eliminated to reach the objective of the final gradient expression free of the flow variable sensitivity. It is quite usual in a turbomachinery flow field calculation that an interface is normal to the rotation axis of a turbomachine as sketched in Fig.3.2. This leaves x-directional flux only through such an interface. It is also assumed that an objective function of a design optimization does not explicitly depend on the flow solution at an interface, implying $M = 0$. The boundary integral in Eq.4.21 can be accordingly reduced to

$$\frac{1}{Y_1}\int_{y1} \lambda^T A\tilde{U}dy - \frac{1}{Y_2}\int_{y2} \lambda^T A\tilde{U}dy = 0 \tag{4.38}$$

The appearance of the pitch length accounts for the whole annulus effect. The first term in the above belongs to domain 1 in Fig.3.2, and the second term belongs to domain 2.

At a first glance, Eq.4.38 can be satisfied in two ways. One is to make each of them vanish. This requires setting $\lambda^T A = 0$ at every point along an interface, as the flow variable sensitivity $\tilde{U}$ is not always zero along an interface. Because the Jacobian matrix A usually has a full rank, setting $\lambda^T A = 0$ is equivalent to setting $\lambda = 0$, which is of course sufficient but not necessary. The other way is to satisfy Eq.4.38 as a whole across an interface. This means each of the two terms from the two sides of an interface may not be zero, but they have the same value, hence cancel out each other. Eq.4.38 can be written in the following equivalent form:

$$\frac{1}{Y_1}\int_{y1} \lambda^T \tilde{F}dy - \frac{1}{Y_2}\int_{y2} \lambda^T \tilde{F}dy = 0 \tag{4.39}$$

The adjoint variable vector is independent of a design variable, therefore, one has

$$\lambda^T \tilde{F} = \widetilde{\lambda^T F} \tag{4.40}$$

Substituting the above into Eq.4.39 yields

$$\frac{1}{Y_1}\int_{s1} \widetilde{\lambda^T F}ds_x - \frac{1}{Y_2}\int_{s2} \widetilde{\lambda^T F}ds_x = 0 \tag{4.41}$$

The interface geometry can be kept intact, when a design variable is perturbed. Hence the differentiation operation and integration operation can exchange their sequences,

resulting in the following equivalent equation:

$$\frac{1}{Y_1}\widetilde{\int_{y1}\lambda^T F dy} - \frac{1}{Y_2}\widetilde{\int_{y2}\lambda^T F dy} = 0 \tag{4.42}$$

The above equation can be further rearranged as follows by removing the differentiation operation:

$$\frac{1}{Y_1}\int_{y1}\lambda^T F dy - \frac{1}{Y_2}\int_{y2}\lambda^T F dy = 0 \tag{4.43}$$

The above equation is a scalar equation with each side being a sum of five independent terms. It can be satisfied by balancing all the corresponding terms separately across an interface, leading to the following five equations, with each one corresponding to one adjoint variable:

$$\begin{aligned}
\frac{1}{Y_1}\int_{y1}\lambda_1(\rho u)dy - \frac{1}{Y_2}\int_{y2}\lambda_1(\rho u)dy &= 0 \\
\frac{1}{Y_1}\int_{y1}\lambda_2(\rho u^2+p)dy - \frac{1}{Y_2}\int_{y2}\lambda_2(\rho u^2+p)dy &= 0 \\
\frac{1}{Y_1}\int_{y1}\lambda_3(\rho uvr)dy - \frac{1}{Y_2}\int_{y2}\lambda_3(\rho uvr)dy &= 0 \\
\frac{1}{Y_1}\int_{y1}\lambda_4(\rho uw)dy - \frac{1}{Y_2}\int_{y2}\lambda_4(\rho uw)dy &= 0 \\
\frac{1}{Y_1}\int_{y1}\lambda_5(\rho Hu)dy - \frac{1}{Y_2}\int_{y2}\lambda_5(\rho Hu)dy &= 0
\end{aligned} \tag{4.44}$$

The five equations state that the adjoint variable weighted and circumferentially averaged fluxes of mass, momentum and energy are conserved across an interface. Note that along an interface at a radial mesh section, the radial coordinate r is constant. This allows the r in the third equation in the above can be taken out of the spatial integral and cancel out, leading to the five equivalent equations:

$$\begin{aligned}
\frac{1}{Y_1}\int_{y_1}\lambda_1\rho u dy &= \frac{1}{Y_2}\int_{y_2}\lambda_1\rho u dy \\
\frac{1}{Y_1}\int_{y_1}\lambda_2(\rho u^2+p)dy &= \frac{1}{Y_2}\int_{y_2}\lambda_2(\rho u^2+p)dy \\
\frac{1}{Y_1}\int_{y_1}\lambda_3\rho uv dy &= \frac{1}{Y_2}\int_{y_2}\lambda_3\rho uv dy \\
\frac{1}{Y_1}\int_{y_1}\lambda_4\rho uw dy &= \frac{1}{Y_2}\int_{y_2}\lambda_4\rho uw dy \\
\frac{1}{Y_1}\int_{y_1}\lambda_5\rho hu dy &= \frac{1}{Y_2}\int_{y_2}\lambda_5\rho hu dy
\end{aligned} \tag{4.45}$$

Eq.4.45 only sets the a target of conservation to be achieved. How to achieve this target in an appropriate way is also important. When considering the small distance between an interface and a blade leading/trailing edge, it is highly desirable to employ a similar implementation to that for non-reflective boundary conditions for the flow equations (Giles, 1991) to avoid nonphysical reflections corrupting flow solutions in the interior domain.

Applying the mean value theorem for integration to either side of Eq.4.45, the following mixed-out variables can be obtained:

$$\begin{aligned}
\widehat{\lambda_{1,i}} &= \frac{\int_{y_i} \lambda_1 \rho u dy}{\int_{y_i} \rho u dy} \\
\widehat{\lambda_{2,i}} &= \frac{\int_{y_i} \lambda_2 (\rho u^2 + p) dy}{\int_{y_i} (\rho u^2 + p) dy} \\
\widehat{(\lambda_3 v)_i} &= \frac{\int_{y_i} \lambda_3 \rho u v dy}{\int_{y_i} \rho u dy} \\
\widehat{(\lambda_4 w)_i} &= \frac{\int_{y_i} \lambda_4 \rho u w dy}{\int_{y_i} \rho u dy} \\
\widehat{\lambda_{5,i}} &= \frac{\int_{y_i} \lambda_5 \rho u h dy}{\int_{y_i} \rho u h dy}
\end{aligned} \tag{4.46}$$

In the third and fourth equations of the above equations, the mass flux is used in the denominator instead of corresponding momentum fluxes. This is because the circumferential and radial momentum fluxes may change their signs somewhere along an interface, which violates the premise of the mean value theorem for integration. However the mass flux, axial momentum flux and energy flux do not normally change their signs along an interface (no reverse flow). According to the conservation in Eq.4.45, these mixed-out variables in Eq.4.46 should also be equal across an interface. Once there is a difference in these mixed-out variables across the interface, this difference is taken as a characteristic jump disturbance. The present non-reflective implementation will pass the disturbance across the interface correctly. The local adjoint solution variables will be corrected according to the adjoint characteristic propagation to drive the difference to zero.

The adjoint characteristic variables in term of primitive adjoint variable perturbations are expressed as follows:

$$\tilde{\lambda}_c = S^T \tilde{\lambda} \tag{4.47}$$

where S is the matrix which diagonalizes the Jacobian matrix $A = \frac{\partial F}{\partial U} = S\Lambda S^{-1}$, with Λ being a diagonal matrix with diagonal elements as eigenvalues of the matrix A. Expanding the vector form of Eq.4.47 gives

$$\begin{aligned}
\tilde{\lambda}_{c1} &= \tilde{\lambda}_1 + (u-c)\tilde{\lambda}_2 + vr\tilde{\lambda}_3 + w\tilde{\lambda}_4 + (h-uc)\tilde{\lambda}_5 \\
\tilde{\lambda}_{c2} &= \tilde{\lambda}_1 + (u+c)\tilde{\lambda}_2 + vr\tilde{\lambda}_3 + w\tilde{\lambda}_4 + (h+uc)\tilde{\lambda}_5 \\
\tilde{\lambda}_{c3} &= \tilde{\lambda}_3 + \frac{v}{r}\tilde{\lambda}_5 \\
\tilde{\lambda}_{c4} &= \tilde{\lambda}_4 + w\tilde{\lambda}_5 \\
\tilde{\lambda}_{c5} &= \frac{\lambda_1}{u} + \tilde{\lambda}_2 + (u - \frac{W^2}{2u})\tilde{\lambda}_5
\end{aligned} \tag{4.48}$$

where h is the total enthalpy, c is the speed of sound, u, v, w are velocity components in the axial, circumferential and radial directions respectively, $W^2 = u^2 + v^2 + w^2$ is square of the velocity magnitude, r is the radial coordinate.

$\tilde{\lambda}_{c1}$ represents an adjoint acoustic disturbance propagating downstream at a speed of $-u+c$; $\tilde{\lambda}_{c2}$ represents an adjoint acoustic disturbance propagating upstream at a speed of $-u-c$; $\tilde{\lambda}_{c3}$ and $\tilde{\lambda}_{c4}$ represent two adjoint vorticity disturbances, $\tilde{\lambda}_{c5}$ represents an adjoint entropy disturbance, the three all propagate upstream at the same speed of $-u$. For both incoming and outgoing adjoint characteristic variables, flow variables take their local values. For incoming adjoint characteristic variables, the primitive adjoint perturbation takes values through the difference in the mixed-out quantities in Eq.4.46, e.g.

$$\begin{aligned}
\tilde{\lambda}_1 &= \widehat{\lambda_{11}} - \widehat{\lambda_{21}} \\
\tilde{\lambda}_2 &= \widehat{\lambda_{12}} - \widehat{\lambda_{22}} \\
\tilde{\lambda}_3 &= \frac{\widehat{\lambda_{13}v} - \widehat{\lambda_{23}v}}{v} \\
\tilde{\lambda}_4 &= \frac{\widehat{\lambda_{14}w} - \widehat{\lambda_{24}w}}{w} \\
\tilde{\lambda}_5 &= \widehat{\lambda_{15}} - \widehat{\lambda_{25}}
\end{aligned} \tag{4.49}$$

$\tilde{\lambda}_3$ or $\tilde{\lambda}_4$ is set to zero if the magnitude of v or w is smaller than some threshold values (say 0.1). For outgoing adjoint characteristic variables, the primitive adjoint perturbations take local values.

In Fig.3.2, the interface separates two boundaries, an exit for domain 1, and an inlet for domain 2. The exit of domain 1 has four incoming adjoint characteristics

$(\tilde{\lambda}_{c2}, \tilde{\lambda}_{c3}, \tilde{\lambda}_{c4}, \tilde{\lambda}_{c5})$ of which the primitive adjoint variable perturbations will be calculated through Eqs.4.49. The inlet of domain 2 has one incoming adjoint characteristic $(\tilde{\lambda}_{c1})$ of which the primitive adjoint variable perturbations will be also obtained through Eqs.4.49. Outgoing adjoint characteristics are calculated according to local primitive adjoint perturbations.

Once all the adjoint characteristic variables are obtained for each side of an interface, then they can be converted to primitive adjoint variables via the following inverse transformation:

$$\tilde{\lambda} = \left(S^T\right)^{-1} \tilde{\lambda}_c$$

The scalar form of the above is given as follows:

$$\begin{aligned}
\tilde{\lambda}_5 &= \frac{k-1}{c^2}\left(\frac{\tilde{\lambda}_{c1}}{2} + \frac{\tilde{\lambda}_{c2}}{2} - u\tilde{\lambda}_{c5} - vr\tilde{\lambda}_{c3} - w\tilde{\lambda}_{c4}\right) \\
\tilde{\lambda}_4 &= -w\tilde{\lambda}_5 + \tilde{\lambda}_{c4} \\
\tilde{\lambda}_3 &= \frac{-v}{r}\tilde{\lambda}_5 + \tilde{\lambda}_{c3} \\
\tilde{\lambda}_2 &= -u\tilde{\lambda}_5 + \frac{-\tilde{\lambda}_{c1} + \tilde{\lambda}_{c2}}{2c} \\
\tilde{\lambda}_1 &= \frac{W^2}{2}\tilde{\lambda}_5 + \frac{u}{2c}(\tilde{\lambda}_{c1} + \tilde{\lambda}_{c2} + 2c\tilde{\lambda}_{c5})
\end{aligned} \tag{4.50}$$

The primitive adjoint variable perturbations can therefore be directly used to update the local adjoint solution.

4.5 Solution of the Adjoint Equations

In order to solve Eq.4.22 in a finite volume framework, it is first written in a strong conservative form with extra terms taken as source terms. Then a pseudo time derivative term is added to the adjoint equations as follows to form a hyperbolic system so that the Runge-Kutta time integration method can be made use of:

$$\begin{aligned}
&\frac{\partial \lambda}{\partial \tau} - \frac{\partial A^T \lambda}{\partial x} - \frac{\partial (B - v_g I)^T \lambda}{r\partial\theta} - \frac{\partial r C^T \lambda}{r\partial r} = \\
&- \left[\frac{\partial A^T}{\partial x} + \frac{\partial (B - v_g I)^T}{r\partial\theta} + \frac{\partial r C^T}{r\partial r} - D^T\right]\lambda
\end{aligned} \tag{4.51}$$

As can be seen from the above, compared with Eq.4.22, all terms change their signs into their opposite ones when the pseudo time derivative term is added. This is again in line

with the property of adjoint characteristics (Giles and Pierce, 1997, 1998) propagating in exactly opposite directions to the flow characteristics.

Integrating the above equation over a control volume yields

$$\int_{V_i} \frac{\partial \lambda}{\partial \tau} dv - \int_{\partial V_i} \left[A^T \lambda \cdot n_x + (B - v_g I)^T \lambda \cdot n_\theta + C^T \lambda \cdot n_r \right] ds \qquad (4.52)$$
$$= - \int_{V_i} \left[\frac{\partial A^T}{\partial x} + \frac{\partial (B - v_g I)^T}{r \partial \theta} + \frac{\partial r C^T}{r \partial r} - D^T \right] \lambda dv$$

The right hand side term is approximated by

$$\int_{V_i} \left[\frac{\partial A^T}{\partial x} + \frac{\partial (B - v_g I)^T}{r \partial \theta} + \frac{\partial r C^T}{r \partial r} - D^T \right] \lambda dv \qquad (4.53)$$
$$\doteq \int_{V_i} \left[\frac{\partial A^T}{\partial x} + \frac{\partial (B - v_g I)^T}{r \partial \theta} + \frac{\partial r C^T}{r \partial r} \right] dv \cdot \lambda_i - \int_{V_i} D^T \lambda dv$$
$$= \int_{\partial V_i} \left[A^T \cdot n_x + (B - v_g I)^T \cdot n_\theta + C^T \cdot n_r \right] ds \cdot \lambda_i - \int_{V_i} D^T \lambda dv$$

where λ_i is the adjoint solution at the i^{th} cell.

The following term is called the adjoint inviscid flux:

$$A^T \lambda \cdot n_x + (B - v_g I)^T \lambda \cdot n_\theta + C^T \lambda \cdot n_r$$

Referring to Fig.3.4, along the bounding surface indexed $(i + \frac{1}{2}, j)$ of the control volume indexed (i, j), the inviscid adjoint flux is simply the average of the adjoint fluxes calculated using the flow and adjoint solution at the two adjacent cells indexed (i, j) and $(i + 1, j)$. This is the well-known central difference scheme which requires numerical dissipation to be stable. The same artificial dissipation scheme as used in the flow equation discretization is utilized here.

Time integration is achieved using the Runge-Kutta method as detailed in Chapter 3 for the solution of the flow equations. The multi-grid and local-time stepping techniques are also used to speed up the convergence of the adjoint equations. Subroutines in the flow solver, such as those for time-marching, multi-grid, and metrics calculation, can be reused in developing the adjoint solver without any major modification. As for the initialization of the adjoint solution, our practice shows that the adjoint solution can be initialized with zeros or real numbers of small magnitudes.

4.6 Summary

The adjoint formulation principle is first elaborated in an algebraic way. An adjoint Euler flow system, including the field equations and corresponding boundary conditions, is derived in detail by the continuous adjoint approach. The developed adjoint mixing-plane treatment, like the flow mixing-plane treatment, has two important features: conservation and nonreflectiveness. With this adjoint mixing-plane treatment, the sensitivity calculation can be conducted in a multi-bladerow environment. This provides an opportunity to carry out blading aerodynamic design optimization in a multi-bladerow environment. The derived adjoint system is solved in a cell-centered finite volume framework as the flow system is solved. The inviscid adjoint fluxes are computed using the central difference scheme with a blended second and fourth order artificial viscosity. Time integration is accomplished using the four stage Runge-Kutta method.

Chapter 5

Blading Aerodynamic Design and Optimization System

This chapter supplements the necessary elements for establishing a system for turbomachinery blading aerodynamic design optimization. This includes the gradient calculation, the objective function to be optimised, the shape parameterization approach, the single-operating-point design optimization process and the multi-operating-point design optimization process.

5.1 Gradient Calculation

For a convenient reference, the gradient expression (Eq.4.23) is repeated here:

$$\frac{dI}{d\alpha} = \int_{\partial D} \frac{\partial M}{\partial \alpha} ds + \int_{\partial D} M \tilde{ds} + \int_{D} \lambda^T f dv \tag{5.1}$$

The gradient expression is expressed in terms of the adjoint variable, the flow variable and the geometric gradient. The geometric gradient is calculated using the complex variable method (Vatsa, 1999). The term f is the opposite of the flow residual perturbation due to a unit perturbation in a design variable, with flow variables fixed at all control volumes, defined as follows:

$$f = -\frac{\partial R}{\partial \alpha} \tag{5.2}$$

The calculation of f requires perturbing the mesh followed by the flow residual perturbation calculation. The procedures of the gradient calculation are summarized as follows:

1) perturbing a design variable and calculating the mesh perturbation.

2) calculating $\int_{\partial D} \frac{\partial M}{\partial \alpha} ds$ and $\int_{\partial D} M\widetilde{ds}$.

3) calculating flux perturbations for each bounding surface, then assembling flux perturbations for each control volume to form the flow residual perturbation $\frac{\partial R_i}{\partial \alpha}$.

4) multiplying the adjoint variable with the flow residual perturbation at each control volume and summing up:

$$\sum_{i}^{N} \lambda_i^T \frac{\partial R_i}{\partial \alpha}$$

with N as the number of control volumes.

5) assembling all the three parts to form a complete gradient (Eq.5.1).

For each design variable, the above five steps need to be completed once. For thousands of design variables, the time cost for all the gradient evaluations is not negligible, but it will usually not exceed that for solving the flow equations or the adjoint equations once. For example, the mesh needs to be perturbed once for each design variable to calculate the corresponding flow residual perturbation. Such a residual perturbation calculation has a cost which is more or less one-fourth of one full four-stage Runge-Kutta iteration. With a complex variable implementation, the cost of a flow residual perturbation is doubled. Therefore for one hundred design variables, the time cost will be 50 full four-stage Runge-Kutta iterations, which is only a fraction of thousands of iterations required to solve either the flow equations or the adjoint equations.

Though there are ways to avoid perturbing the mesh by converting the domain integral in Eq.5.1 into a boundary integral (Jameson, 2003; Papadimitriou and Giannakoglou, 2006; Soto and Löhner, 2004), it involves calculation of spatial derivatives of flow variables along the boundary of a computational domain, which is normally not easy to be obtained with desired accuracy.

5.2 Objective Function

One design optimization option of the present design system is to increase a blade isentropic efficiency whilst applying constraints to the mass flow rate and stagnation

pressure ratio. The constraints are set to limit the mass flow rate and stagnation pressure ratio changes to some specified values, say 0.5%. Although the design target is to increase isentropic efficiency, this is achieved by minimizing the entropy generation rate, hence flow losses. The entropy generation rate, as used by Oyama and Liou (2004), has a much simpler expression than that of isentropic efficiency. This reduces complexity in deriving objective function related terms ($\frac{\partial M}{\partial U}$ in Eq.4.13), which are needed for specifying adjoint boundary conditions. Besides, the entropy generation rate can also be used as an objective function to be minimised to optimise the performance of turbine blades for which the efficiency definition for compressor blades can not be applied any more. This makes the design system more general for turbomachinery blading design optimizations.

There are at least two strategies available for a constrained design optimization. One approach is to consider a single objective function that is a weighted sum of all objectives and constraints (Elliott and Peraire, 1998; Pierret, 2005). This strategy requires one adjoint solution only for gradients of the single objective function to all design variables. The other approach is to calculate the gradients of objectives and constraints separately, asking for a separate adjoint solution for each functional. The first strategy is adopted here in the light of using a single adjoint solution reducing computational cost. The objective function for design optimization is a weighted sum of the two aforementioned constraints and the entropy generation rate, taking the following formulation:

$$I = \frac{\Delta \bar{s}}{\Delta \bar{s}_0} + \sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right)^2 + \sigma_2 \left(\frac{\pi}{\pi_0} - 1 \right)^2 \tag{5.3}$$

where Δs is called entropy generation rate and defined as

$$\Delta \bar{s} = \bar{s}_e - \bar{s}_i$$

in which e denotes an exit and i denotes an inlet. $\bar{s}$ is the mass averaged entropy and defined by

$$\bar{s} = \frac{\int s \rho v_n ds}{\int \rho v_n ds}$$

with $s = \frac{p}{\rho^\gamma}$. v_n is the velocity normal to the plane at which the integration is conducted. ds is a differential area element. The mass flow rate $\dot{m}$ is defined by

$$\dot{m} = \frac{1}{2} \left(\int_i \rho v_n ds \cdot N_{b1} + \int_e \rho v_n ds \cdot N_{bn} \right)$$

where N_{b1} is the number of blade passages of the first blade row, while N_{bn} is the number of blade passages of the last blade row. The stagnation pressure ratio is defined by

$$\pi = \frac{\bar{P}_e^*}{\bar{P}_i^*}$$

with $\bar{P}^*$ defined as the mass averaged total pressure at an inlet or exit plane:

$$\bar{P}^* = \frac{\int p^* \rho v_n ds}{\int \rho v_n ds}$$

in which p^* is the local total pressure calculated by

$$p^* = p\left(1 + \frac{W^2}{2c_p T}\right)^{\gamma/(\gamma-1)}$$

The subscript 0 in Eq.5.3 denotes initial values. σ_1 and σ_2 are weighting factors. Different values of σ_1 and σ_2 will usually lead to different designs as also revealed by Wu et al. (2005a).

A good choice of these values usually needs some trial and error. If the curve of the objective function vs. design iteration has a lot of zigzags, and the curve of a constraint vs. design iteration also has zigzags, this usually indicates the corresponding weighting factor is too big. If a constraint is seriously violated, even if the objective function converges, the corresponding weighting factor needs to be increased.

The objective function gradient with respect to flow variables $\frac{\partial M}{\partial U}$ for an adjoint boundary specification can be found in Appendix C.

5.3 Shape Parameterization

Shape parameterization plays a very important role in turbomachinery blading aerodynamic design optimization. Different shape parameterization generally leads to different optimised designs (Wu et al., 2003). Shape parameterization for turbomachinery blades has long been a separate research topic (Miller IV et al., 1996; Samareh, 1999). With many different approaches for blade shape parameterization appearing in the open literature, what is the best way to parameterize turbomachinery blades remains an open question. A good shape parameterization should be able to accommodate sufficiently many blade shapes of different features, meanwhile it should be also robust

enough so that any arbitrary perturbation to its parameters will generate a realistically different blade shape. It is also highly desirable that a shape parameterization can approximate the original blades to be optimised with sufficient accuracy.

It is quite difficult to implement such a good shape parameterization easily for turbomachinery blades, which normally have quite rich two dimensional features for 2D blade sections and rich three dimensional features for 3D blades. All the shape parameterization methods in the open literature can be divided into two kinds: blade shape parameterization and blade shape perturbation/deformation parameterization.

NACA airfoils are described by four digits which can be entered into equations to get a required airfoil geometry. Bézier polynomials were used to parameterize a blade shape by Arens et al. (2005), Büche et al. (2003), Benini (2004), Sieverding et al. (2004) and Lotfi et al. (2006). B splines were adopted to describe a blade shape by Anderson and Venkatakrishnan (1997) and Oyama and Liou (2004). Aerodynamic modes like sweep, lean (Jang et al., 2006; Li et al., 2006b; Yi et al., 2006) are a popular choice to parameterize perturbations to a blade shape, reducing the number of design variables. The Hicks-Henne functions (Hicks and Henne, 1978) were used by Yang et al. (2003) and Kim and Nakahashi (2005) and Bézier curves/surfaces were utilized by Burguburu et al. (2004) to parameterize perturbations to a blade shape.

This shape perturbation parameterization approach has been adopted in this research work using the Hicks-Henne functions and its variants. The major advantage of the shape perturbation parameterization approach is that the original blade shape can be recovered by simply setting zero perturbation. Two sets of hump functions are used to parameterize perturbations to tangential coordinates of mesh points on a blade surface. One is the 2D Hicks-Henne function expressed in 5.4 which is a variant of the original Hicks-Henne function(Hicks and Henne, 1978). The other is given in 5.5. The Hicks-Henne functions (5.4) are applied to mesh points which are not at the leading edge or the trailing edge or the casing or the hub of a blade, while the other ones (5.5) are used for those mesh points where (the leading edge, trailing edge, casing section and hub section mesh points) Hicks-Henne functions fail (x_d or r_d can not be one or zero

in 5.4).

$$\delta_{r\theta} = a \cdot \sin^{b_r}\left(\pi r^{\frac{\ln 0.5}{\ln r_d}}\right) \cdot \sin^{b_x}\left(\pi x^{\frac{\ln 0.5}{\ln x_d}}\right) \tag{5.4}$$

$$\delta_{r\theta} = a \cdot \sin^{b_r}\left[\frac{\pi}{2}\left(1 - |r_d - r|\right)\right] \cdot \sin^{b_x}\left[\frac{\pi}{2}\left(1 - |x_d - x|\right)\right] \tag{5.5}$$

where x is the reduced axial coordinate of a mesh point along the axially running grid line on which the mesh point resides, and r is the reduced radial coordinate of a mesh point along the radially running grid line on which the mesh point resides. The value of x is 0 and 1 at the leading edge and trailing edge respectively. The value of r is 0 and 1 at the hub and casing section respectively. x_d and r_d are reduced axial and radial coordinates of a design mesh point.

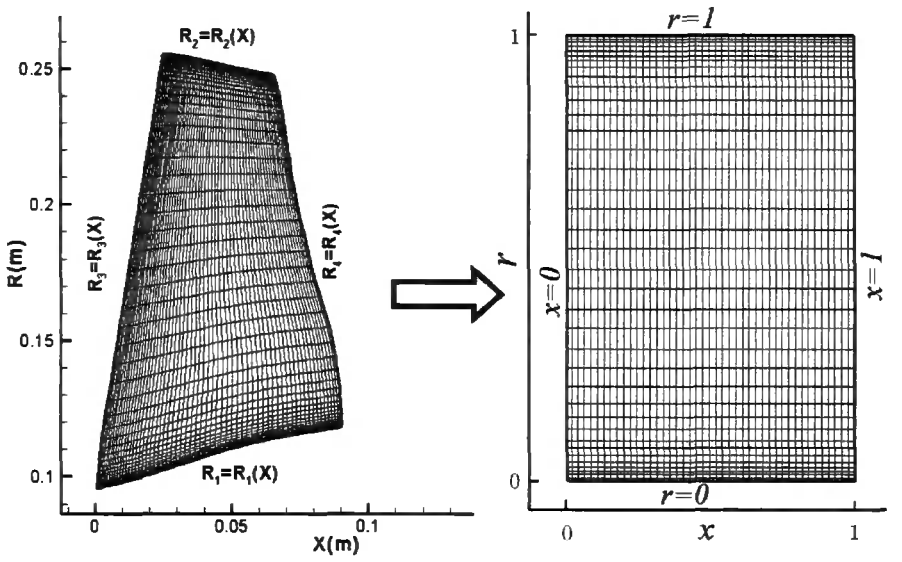


Figure 5.1: Mapping from a curved blade surface to a normalized region

These reduced coordinates can be better explained by mapping a blade surface, which is formed by four curved edges, to a rectangle as shown in Fig.5.1. The blade surface is composed of four curved edges: $R_1 = R_1(X)$ forms the hub contour of the blade, $R_2 = R_2(X)$ forms the tip contour of the blade, $R_3 = R_3(X)$ forms the leading edge contour and $R_4 = R_4(X)$ forms the trailing edge contour. The blade surface is mapped to the rectangle with $R_1 \rightarrow r = 0$, $R_2 \rightarrow r = 1$, $R_3 \rightarrow x = 0$ and $R_4 \rightarrow x = 1$. For a mesh point (j, k), with j indexing mesh points on an axially running mesh line and k indexing mesh points on a radially running mesh line, its reduced coordinates are

calculated by

$$\begin{aligned} x &= \frac{X(j,k) - X(j_{le},k)}{X(j_{te},k) - X(j_{le},k)} \\ r &= \frac{R(j,k) - R(j_{hub},k)}{R(j_{tip},k) - R(j_{hub},k)} \end{aligned} \tag{5.6}$$

where *le* denotes a leading edge, *te* a trailing edge, *hub* and *tip* are self-explanatory.

In the hump functions (5.4 and 5.5), b_x and b_r control the influence of the perturbation to the tangential coordinate of a design mesh point on other mesh points. The bigger the value of b_x or b_r, the more local the hump functions will be. a is a design variable and determines the amount of perturbation to the tangential coordinate of a design mesh point. On a blade surface, if it is densely populated with design mesh points, then each hump function can be very local by choosing a bigger value of b. Otherwise, the hump functions need a smaller value of b to be more globally smooth. As a geometrical constraint, the blade tangential thickness is kept constant during a design process to avoid producing unrealistically thin blades. With this geometric change, the blade axial chord length will not be changed.

The mesh points on a blade span section are also allowed to move along axially running grid lines to accommodate blade sweep effect. When one radial mesh section is perturbed in the axial direction along the axially running grid line, the perturbations to other radial sections are determined according to the following Hicks-Henne function:

$$\delta_x = a \cdot \sin^{b_r}\left(\pi r^{\frac{\ln 0.5}{\ln r_d}}\right) \tag{5.7}$$

The resultant perturbation to radial coordinates of mesh points is determined through ensuring the perturbed blade span section moves along the axially running grid line on which this span section resides.

Finally if a blade has a circle fitted around its leading edge, mesh points on the leading edge circle are not allowed to be taken as design mesh points. However the coordinates of these mesh points will still be changed due to perturbations to other mesh points to preserve the round blade leading edge shape.

5.4 Single-Operating-Point Design Optimization Process

The search procedure used in this work is the steepest descent method in which a small step is taken in the negative gradient direction.

$$\Delta\alpha = -s\frac{dI}{d\alpha} \tag{5.8}$$

where s is the step size to be prescribed. If $\Delta\alpha$ is small enough, then the change in the objective function can be approximated by

$$\Delta I = \frac{dI}{d\alpha} \cdot \Delta\alpha \tag{5.9}$$

Substituting $\Delta\alpha$ in 5.8 into the above gives

$$\Delta I = -s\frac{dI}{d\alpha} \cdot \frac{dI}{d\alpha} \tag{5.10}$$

It is evident that $\Delta I < 0$, if the gradient is not zero. In practice, the optimum value of the step size s is normally determined through a line search along the negative gradient direction. This kind of search usually requires extra time cost for solving flow equations. Here the step size s is given by an empirical rule. Based on gradients of an objective function to all design variables, an arbitrary step size (big enough to avoid excessive rounding errors) is used first to calculate perturbations to the coordinates of all the mesh points. Then the maximum perturbation is found and all the perturbation will be scaled to such an extent that the maximum perturbation will be equal to a specified value, say 0.1 mm. In this way, a big perturbation which may result in a drastic change in a blade profile in one design cycle will be avoided.

A design optimization process is broken down into seven procedures as shown schematically in Fig.5.2.

Step 1: Problem setup This is the starting point of a design optimization. It includes specification of the initial blade geometry, boundary condition, objectives and constraints, design mesh points, etc.

Step 2: Mesh generation A mesh is generated in an algebraic way according to a given blade geometry.

Step 3: Solve flow equation The flow equation is solved to obtain the flow solution, which is used to evaluate the performance of the current design and provides information for following steps.

Step 4: Solve adjoint equation Based on the flow solution, the adjoint equation is solved. The adjoint solution together with the flow solution provides information for the evaluation of gradients of the objective function to design variables.

Step 5: Calculate gradient Gradients of an objective function to design variables can be obtained with mesh perturbation (see details in section 5.1).

Step 6: Calculate the step size The step size is calculated according to the empirical rule described at the beginning of this section. The step size determines the amount of change made to the current blade geometry.

Step 7: Update blade geometry Based upon the step size and the steepest descent method, the perturbation to a blade geometry is obtained first, then it is used to update the blade geometry.

The above steps 2-7, which constitute one design cycle, will repeat until a certain stopping criterion is reached. In one design cycle, the main time cost is consumed by the flow solver and the adjoint solver. In a design process, the flow variables and adjoint variables in a successive design cycle can be initialized using the flow solution and the adjoint solution in the previous design cycle respectively. Between two successive design cycles, the change in a blade profile is normally small. Therefore both the flow solution and the adjoint solution between two successive design cycles are quite close to each other. The use of the flow solution and the adjoint solution in the previous design cycle to initialize the flow variables and the adjoint variables in the successive design cycle can lead to a fast convergence.

5.5 Multi-Operating-Point Design Optimization

Gas-turbines, especially those used in aero-engines, operate in a wide range of operating conditions. High isentropic efficiency at design point is only one desirable factor among all characterizing a high performance machine. Performance of turbomachines at off-

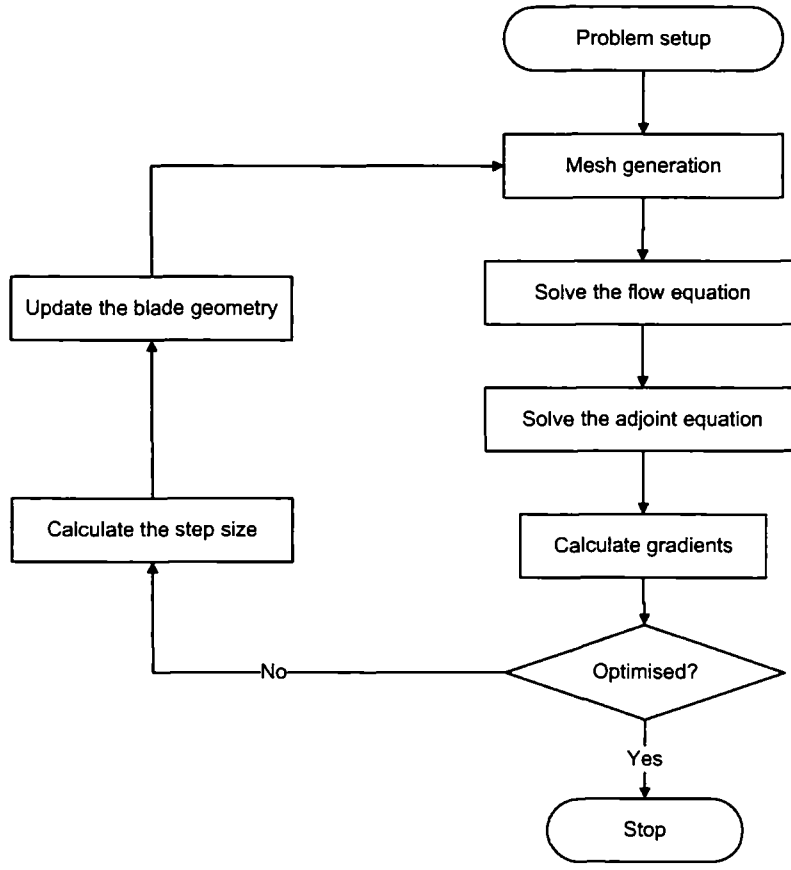


Figure 5.2: Flow chart of aerodynamic design optimization

design conditions are also quite important. Design optimization of a turbomachine at one operating point can normally not guarantee a good performance at other operating conditions. In order to ensure a better performance of an optimised design over the whole operating conditions, there is a need to take into account the blading performance at different operating conditions, leading to the multi-operating-point design optimization. The simplest and most straightforward approach to conduct a multi-operating-point design optimization might be to form an objective function which is a weighted sum of objective functions at multiple operating conditions:

$$I = \sum_{i}^{N} \sigma_i \cdot I_i \tag{5.11}$$

where σ_i and I_i are the weighting factor and the sub-objective function for the i^{th} operating condition respectively; N is the number of operating points to be considered in a design optimization process. The weighting factor is specified *a priori* and independent of the flow solution at any operating condition considered.

The gradient of the objective function I in 5.11 to a design variable is the weighted sum of the gradients of the sub-objective functions to a design variable:

$$\frac{dI}{d\alpha} = \sum_{i}^{N} \sigma_i \cdot \frac{dI_i}{d\alpha} \tag{5.12}$$

The calculation of the gradient of a sub-objective function to a design variable at a particular operating point needs the nonlinear flow solution and the corresponding adjoint solution at that particular operating point only, because of the linear combination of sub-objective functions. Thus these gradients of sub-objective functions to a design variable at different operating points can be obtained in any order or simultaneously, then these gradients are combined linearly with their corresponding weighting factors to form the total gradient. This gives rise to parallelizing the multi-operating-point design optimization process. On one hand, it is quite straightforward to implement the parallelization using the Message Passing Interface(MPI) standard, because the calculation at different operating points provides a natural task decomposition; on the other hand, the parallelization will result in a considerable real time saving.

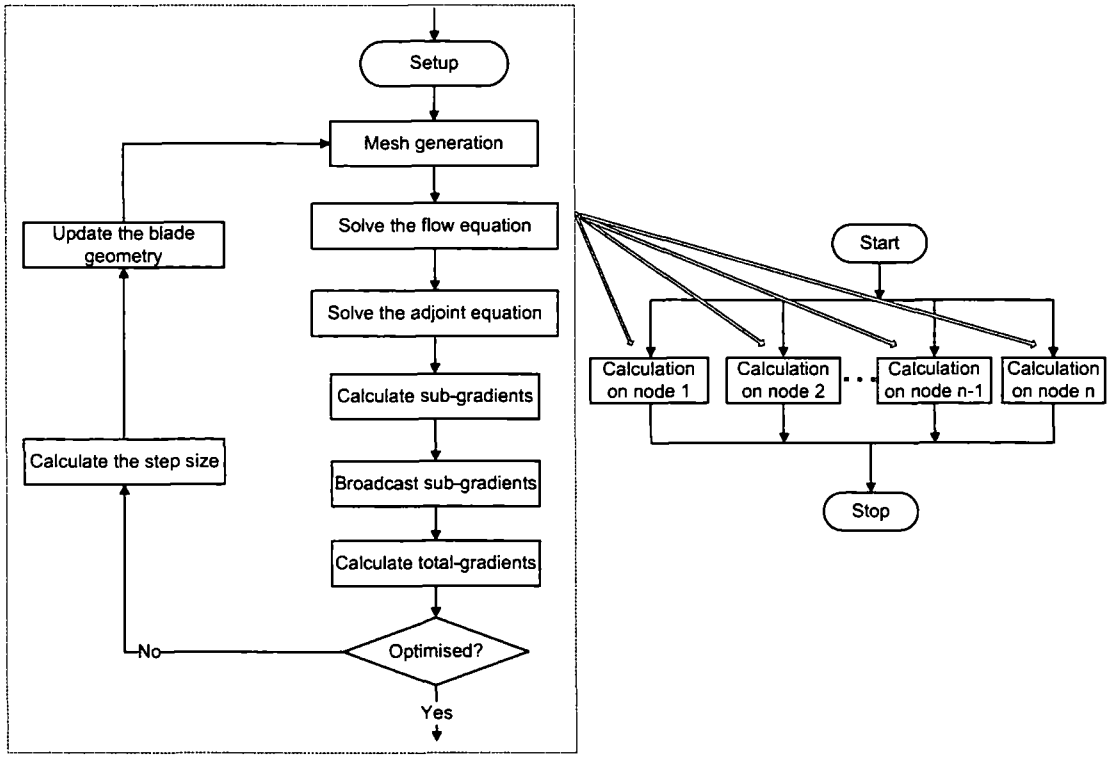


Figure 5.3: Flow chart of a multi-operating-point aerodynamic design optimization

The serial code for aerodynamic design optimization is therefore parallelized with only a little code development forming the framework of a multi-operating-point design optimization. The flow chart of a multi-operating-point design optimization is schematically depicted in Fig.5.3. Comparing the calculation at each computer node with the flow chart of the single-operating-point design optimization (Fig.5.2), the only difference is that sub-gradients need broadcasting to other nodes followed by forming the total gradients in a linear combination way. The procedures of forming the total gradients, calculating the step size, updating a blade geometry and mesh generation are identical at all the computer nodes involved. These procedures can be done in one host node with results broadcast to other nodes. As a matter of fact, this approach will increase communication overhead to some extent and also needs extra code development. It is therefore better to perform these calculations by each computer node.

5.6 Summary

A design optimization system has been established and developed for turbomachinery blading aerodynamic design optimization. The steepest descent method, with the gradient calculated by an adjoint solution with mesh perturbation, is used to guide the original design to an optimum design. The change of blade shape is realized by parameterizing perturbations to a blade using Hicks-Henne like functions. The design system is able to perform design optimization increasing isentropic efficiency while maintaining the change of mass flow rate and pressure ratio within a certain range in a multi-bladerow environment. A multi-operating-point design system is also established to make sure an optimised design will have a better performance than the original one over the whole operating range at a given operation speed. The multi-operating-point design optimization process is parallelized in such a way that the computation for each operating point can be conducted in a separate computer processor simultaneously, leading to significant reduction in real time cost.

Chapter 6

Verification and Application of Aerodynamic Design Optimization System

Verifications of the basic adjoint solver and the proposed adjoint mixing-plane treatment are carried out in this chapter, followed by four design optimization case studies.

6.1 Verification of Gradient Calculation

This verification section is composed of two parts. The first part is to demonstrate the validity and the appropriate implementation of the adjoint method for gradient calculation in a single row computational domain. This part lays down the basis for the second part. The second part is to demonstrate the validity and the appropriate implementation of the adjoint mixing-plane treatment, allowing an adjoint gradient calculation to be carried out in a multi-bladerow environment.

6.1.1 Verifying the Adjoint Solver

Two cases are used to verify the basic adjoint solver for gradient calculations for a single bladerow configuration. The lift gradient of an airfoil to its angle of attack is computed by the adjoint method and compared with the result by the finite difference method. Then an inverse design is carried out to show the validity and the benefit of an adjoint aerodynamic design optimization.

• Lift Coefficient Gradient

The adjoint solution provides information with which gradients of an objective function to a large number of design variables can be obtained efficiently. Thus the adjoint solution can be verified by comparing gradients using the adjoint solution with those calculated by other well established methods. A simple and straightforward way to obtain gradients of an objective function to design variables is the finite difference method. However it should be reminded that this approach has the shortcoming of subtraction cancelation when a step size is too small. While a big step size may lead to unreliable gradient results, if the relation between an objective function and design variables is highly nonlinear. The complex variable method or linearized method is a better alternative to the finite difference method, however its implementation in a large-scale nonlinear flow solver requires a lot of extra work.

As we know, within a certain range of angle of attack, the lift coefficient of an airfoil has a linear relationship with its angle of attack. This provides a good case to calculate the gradient of the lift coefficient of an airfoil to its angle of attack using the finite difference method by choosing a step size which is big enough to avoid significant subtraction cancellation errors.

The lift coefficient of NACA0012 with a chord length of 0.15 m is calculated for angle of attack between 0 and 2 degrees at a Mach number of 0.62 in the far field. The computational domain is extended about 10 times chord away from the airfoil surface and divided into 4 blocks with about 15,000 mesh points (Fig.6.1).

The Mach number distribution at zero angle of attack is shown in Fig.6.2(a) with Fig.6.2(b) showing the corresponding adjoint solution (the adjoint variable corresponding to the continuity equation). Comparing the adjoint solution contours and those of Mach number, one can see clearly that the upstream of the flow field corresponds to the downstream of the adjoint field, and vice versa. This is in consistency with the property of the adjoint characteristics which have an opposite direction to the flow characteristics.

The linear relationship between the lift coefficient of NACA0012 and its angle of attack is presented in Fig.6.3. The gradient by the finite difference method and that by the

adjoint method are in a very good agreement at different angles of attack. The biggest relative difference between the gradients by the two methods is within 5%.

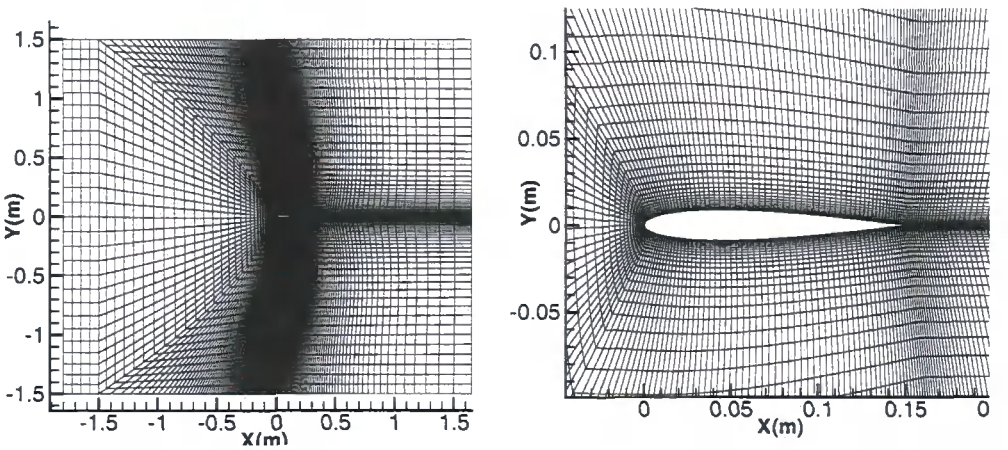
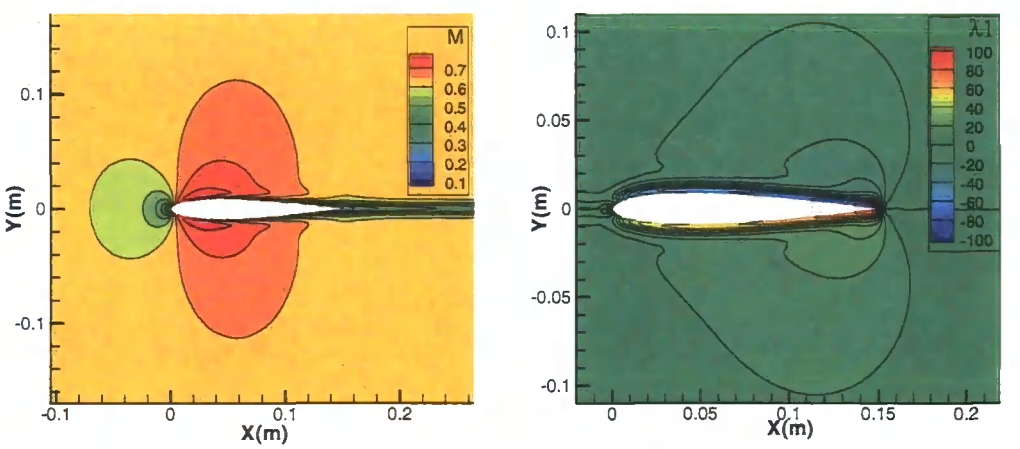


Figure 6.1: Mesh overview (left) and close-up (right) for NACA0012 at zero angle of attack



(a) Mach number contours

(b) Adjoint field contours (λ_1)

Figure 6.2: Flow field and adjoint field contours around NACA0012 at zero angle of attack

• Inverse Design

Inverse design is a popular case for adjoint method researchers to demonstrate their proper implementation of the adjoint method (Yang et al., 2003; Nadarajah and Jameson, 2000). As a conventional inverse design, it requires the specification of a target pressure or velocity distribution on a blade surface. The pressure distribution on the blade surface of NACA0015 with a chord length of 0.15 m is obtained by the present

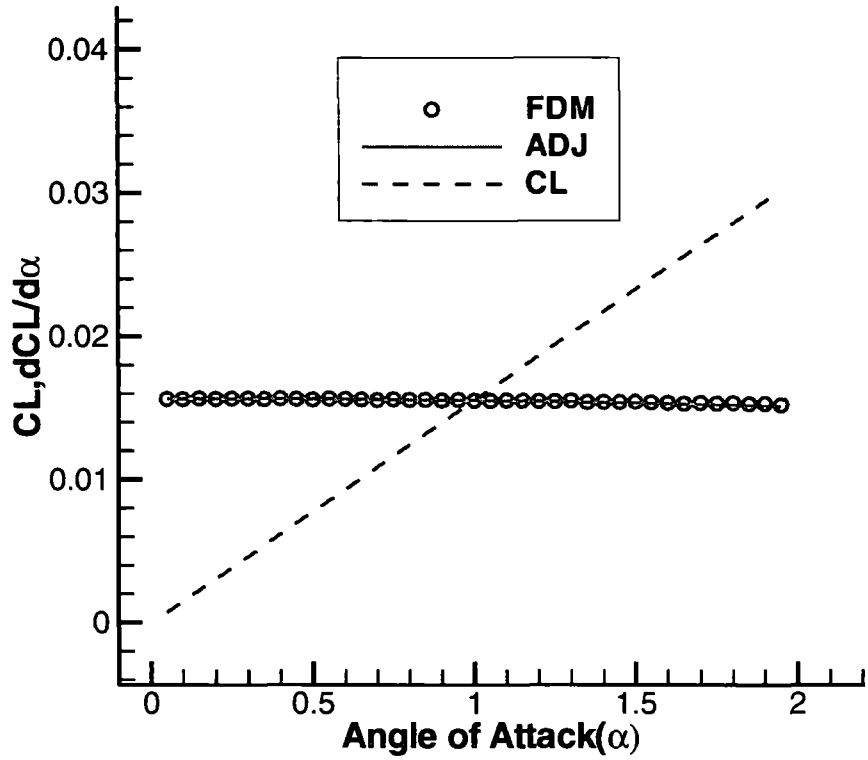


Figure 6.3: Lift coefficient and its gradient versus angle of attack for NACA0012 (FDM: Finite Difference Method; ADJ: Adjoint Method; CL: calculated Lift Coefficient)

flow solver at a flow condition which is identical to that for the previous case. The calculated pressure distribution is taken as the target pressure distribution. The NACA0012 profile of the same chord length is used as an initial airfoil by the design system to approach the target pressure distribution.

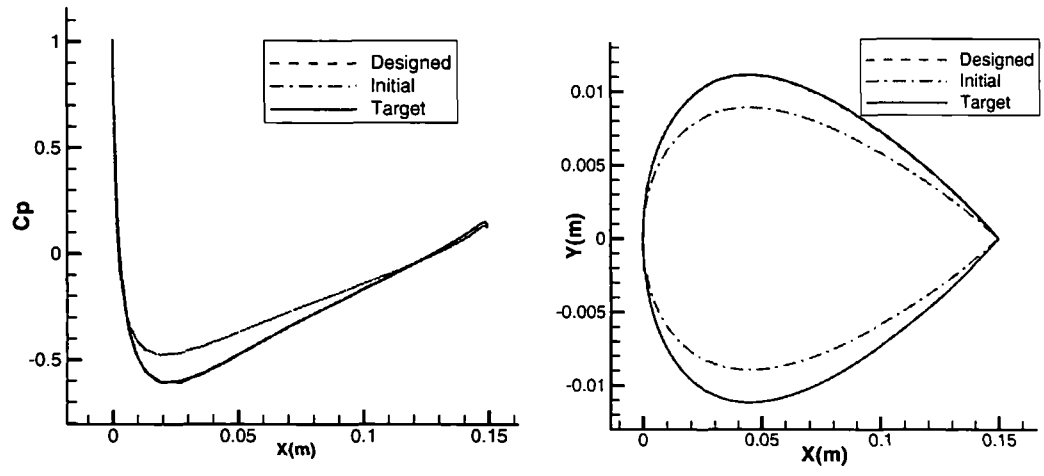
The objective function for this inverse design is given by

$$I = \frac{\int_{bs} (p - p_t)^2 \, ds}{\int_{bs} (p_0 - p_t)^2 \, ds} \tag{6.1}$$

where $_{bs}$ denotes the blade surface, p_t is the target pressure distribution and p_0 is the pressure distribution on the NACA0012 blade surface.

During the design process, the chord length of NACA0012 airfoil is fixed by fixing its leading and trailing edge mesh points. Hicks-Henne function (Hicks and Henne, 1978) is used to parameterize perturbations of y coordinates of all mesh points (except for the leading and trailing edge points) on the blade surface resulting in 155 design variables. With the adjoint method, only two sets of equations (the flow equation and the adjoint equation) are solved together with some post-processing at each design cycle to enable

the gradients of the objective function to all design variables to be calculated. However the finite different method requires the flow equation to be solved at least 156 times at each design cycle (one sided finite difference method). The great time saving by the adjoint method over the finite difference method is apparent.



(a) Pressure coefficients (b) Blade profiles

Figure 6.4: Comparison of pressure coefficients and blade profiles between the initial, designed and target blades for the inverse design

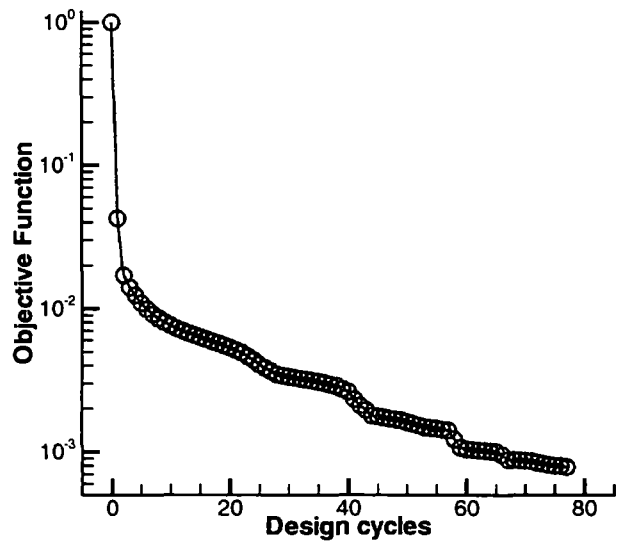


Figure 6.5: History of the objective function with design cycles for the inverse design

The initial, target and designed pressure coefficient distributions are compared in Fig.6.4(a). The target pressure is achieved with a very good accuracy in the inverse design. As Fig.6.4(b) reveals, the target airfoil profile is well recovered. Fig.6.5 depicts

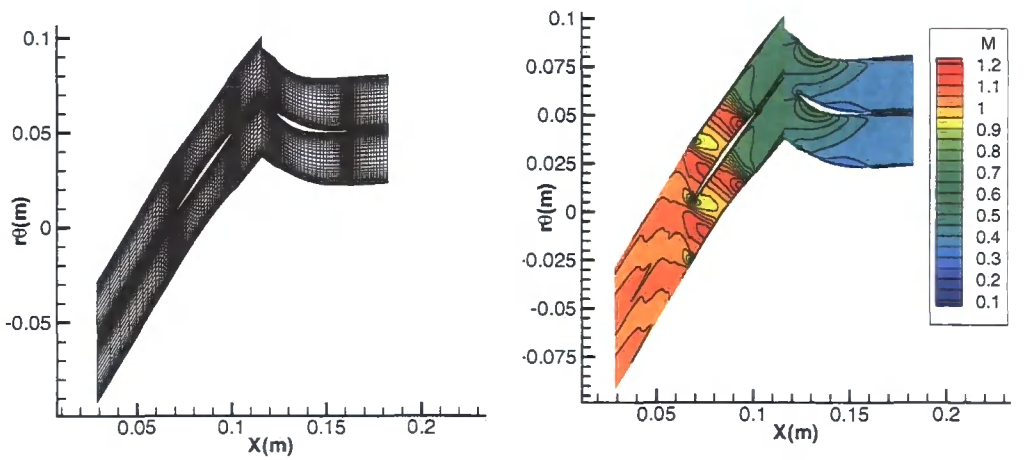
the change of the objective function with design cycles. The objective function in terms of the pressure difference is reduced by 3 orders of magnitude.

6.1.2 Verifying the Adjoint Mixing-plane Treatment

The adjoint mixing-plane treatment plays the role of passing the adjoint solution information correctly between two adjacent bladerow domains. This allows the steady adjoint equations to be solved in a multi-bladerow environment. One case is to show that the adjoint method with the aid of the adjoint mixing-plane treatment is able to calculate gradients correctly in a multi-bladerow environment. The other case is to demonstrate the advantage of carrying out aerodynamic design optimization in a multi-bladerow environment.

• Gradients of 2D Transonic Compressor Stage Configuration

The Euler flow equations together with the adjoint Euler equations are used for the verification of the adjoint mixing-plane treatment to avoid complexity introduced by the viscous effects and freezing the turbulent viscosity. The verification case uses a span



(a) Computational mesh (b) Relative Mach number distributions

Figure 6.6: Computational mesh and relative Mach number distributions for the 2D compressor stage configuration

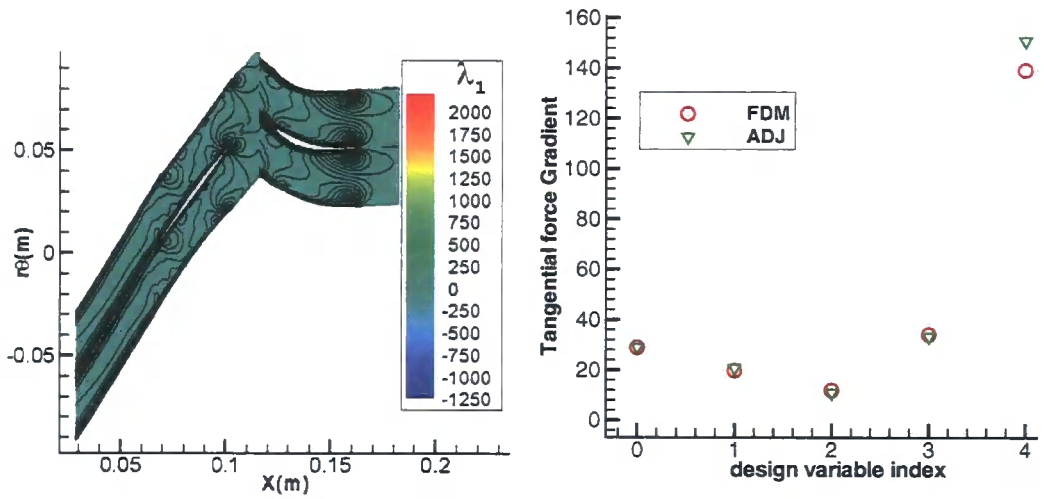
section of a 3D compressor stage - a transonic DLR compressor stage (Dunker et al., 1984). The mesh section is on a cylindrical surface at a constant radius, with a mesh

density of 110x37 for the rotor domain and 90x37 for the stator domain (Fig.6.6(a)). The flow field inside the 2D compressor section is transonic as revealed by the Mach number distribution (Fig.6.6(b)).

Two gradients are considered with this 2D compressor stage. One is the gradient of the stator blade tangential force to stator blade geometry perturbations. The other is the gradient of the stage mass flow rate to rotor blade geometry perturbations. Fig.6.7(a) shows the contours of the adjoint variable (λ_1) corresponding to the continuity equation with the stator blade tangential force as the objective function, while Fig.6.8(a) shows the contours of the adjoint variable (λ_3) corresponding to the moment equation with the stage mass flow rate as the objective function. The two adjoint solution contours consistently illustrate that the adjoint solution information propagates in a direction opposite to that of the physical flow disturbances. The flow upstream region becomes the downstream region of the adjoint solution, and vice versa. The two gradients by the adjoint method are compared with their counterparts by the finite difference method in Fig.6.7(b) and Fig.6.8(b). The gradients by the adjoint method are very close to those by the finite difference method.

It should be noted that the mass flow rate is calculated at the stator exit (also the stage exit). According to the subsonic adjoint inlet boundary condition in Eq.4.25, $\frac{\partial M}{\partial p}$ is zero ($M = 0$) at the inlet of the stage. This means the objective function information is applied at the stage exit only and needs to be passed to the rotor domain correctly through the interface using the adjoint mixing-plane treatment. Otherwise the adjoint solution in the rotor domain can not be right, thereby the gradients by the adjoint method can be wrong.

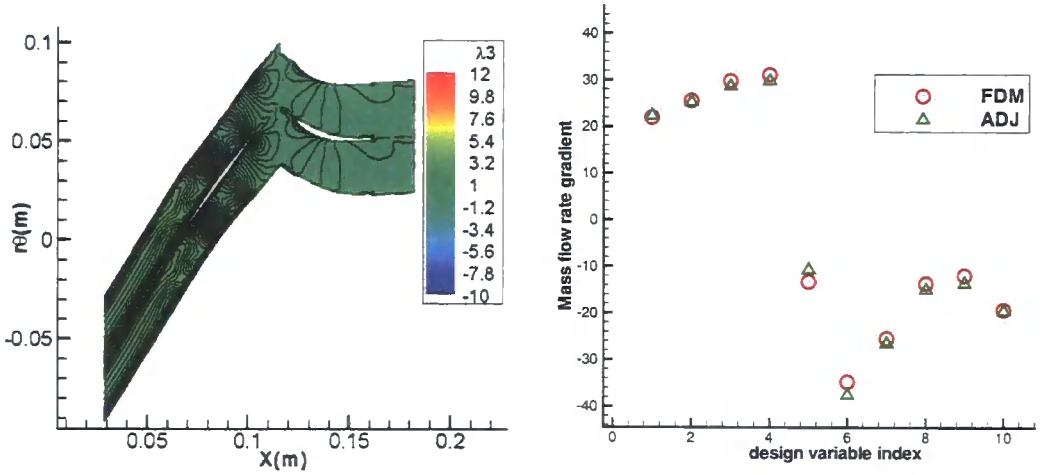
Figs.6.7(a) and 6.8(a) both show that there is a clear cut of contour lines on both sides of the interface, which indicates clearly that the adjoint mixing-plane treatment is non-reflective as intended.



(a) Adjoint field (λ_1)

(b) Tangential force gradients (FDM: Finite Difference Method; ADJ: Adjoint Method)

Figure 6.7: Adjoint field and gradients with the tangential force of the stator blade as the objective function for the 2D compressor stage configuration



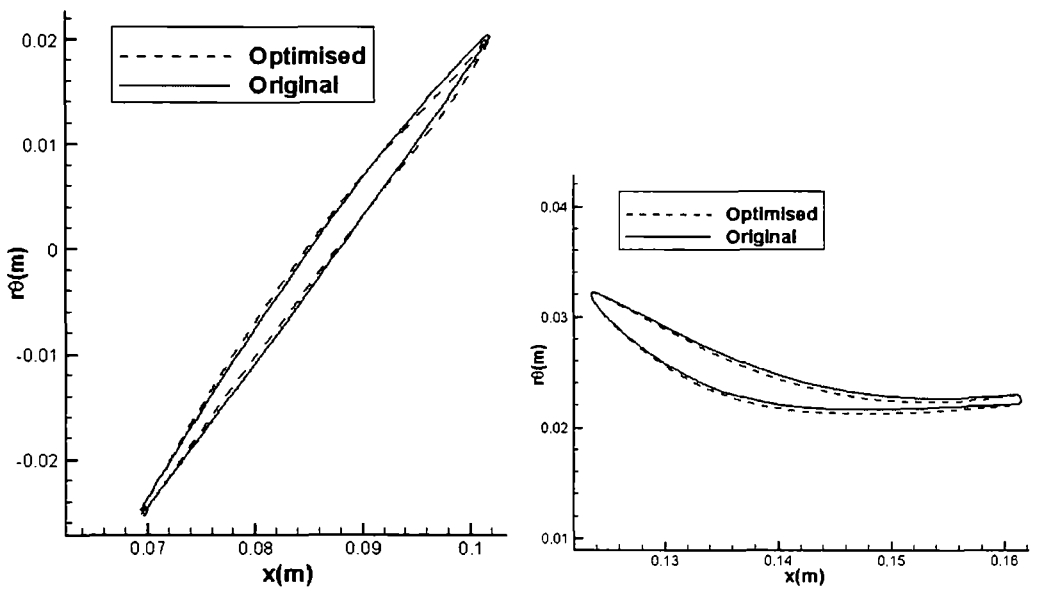
(a) Adjoint field (λ_3)

(b) Stage mass flow rate gradients (FDM: Finite Difference Method; ADJ: Adjoint Method)

Figure 6.8: Adjoint field and gradients with the stage mass flow rate as the objective function for the 2D compressor stage configuration

• Redesign of 2D Compressor Stage

The 2D section of the compressor stage used in the previous section is optimised to increase the isentropic efficiency while maintaining its mass flow rate and pressure ratio. The objective function is a weighted sum of the entropy generation rate and the two constraints of mass flow rate and stagnation pressure ratio. In this case, the RANS flow equations and adjoint RANS equations are solved. 10 design variables are used to parameterize perturbations to the rotor blade shape and 8 design variables to the stator blade shape. The isentropic efficiency of the stage is increased from 84.91% to 88.00% with negligible changes of mass flow rate and pressure ratio. As can be seen from Fig. 6.10, the performance enhancement is due to a weaker shock in the passage of the optimised rotor and the removal of the flow separation around the trailing edge of the optimised stator. It should be pointed out the base flow state is not a very efficient one and is chosen here mainly for validating the adjoint mixing-plane method and demonstrating the benefit of using it in a similar context.



(a) Rotor blades (b) Stator blades

Figure 6.9: Geometry comparison of the original and optimised blades for the 2D compressor stage configuration redesign

Fig.6.9(a) shows the difference of the optimised rotor blade from the original one. The optimised blade is curved towards its suction side between the leading edge and the mid-chord, leading to a weaker shock in the blade passage (Fig.6.10). The trailing edge

region is curved in an opposite direction-towards the pressure side, better matched with the stator. Fig.6.9(b) shows the geometry change of the stator blade with a increased camber to reduce the flow separation around the stator trailing edge (Fig.6.10). This case provides an illustration of the validity and advantage of the present adjoint mixing-plane treatment. Clearly it is far more effective to optimise both the rotor and stator at the same time than to deal with them separately and iteratively.

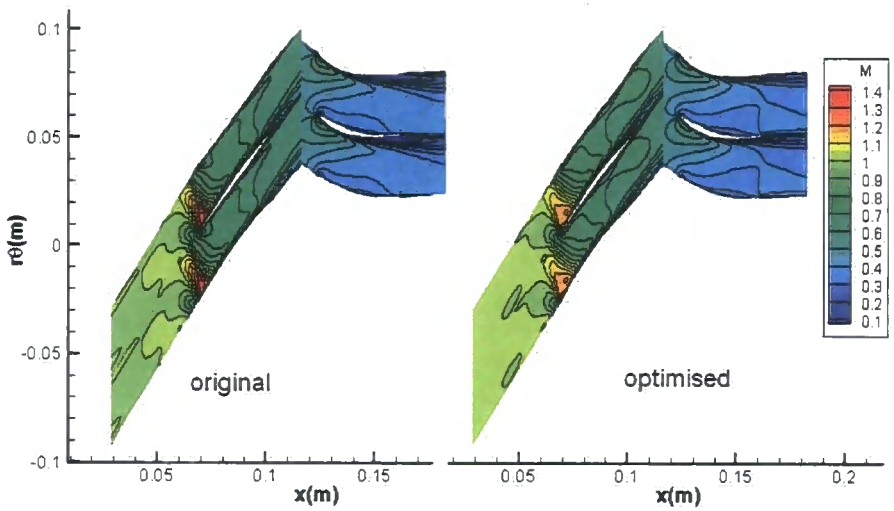


Figure 6.10: Mach number contours inside the original blade passage and the optimised blade passage for the 2D compressor stage configuration redesign

6.2 Application of Aerodynamic Design Optimization System

Four case studies are presented in this section. The first case is the redesign of a transonic rotor (NASA rotor 67) at a near choke operating point. The second case is a redesign of a transonic compressor stage originally designed by DLR. The redesign is carried out at the stage peak efficiency point. In the third and fourth cases, we examine the redesign of blade rows within a three-stage transonic test compressor that was originally designed by Siemens Industrial Turbomachinery Ltd known as the Advanced Transonic Compressor (ATC). Specifically the third case is a redesign of the IGV-Rotor-Stator configuration. Two single-operating-point design optimization scenarios are conducted initially: one redesign is at the stage peak efficiency point; the other is at a near choke flow point of a low stagnation pressure ratio. The redesign at the stage peak efficiency point produces considerable efficiency gain, but leads to a noticeably

reduced choke mass flow rate. The redesign at a near choke mass flow rate point, on the other hand, leads to considerable performance deterioration at operating points with lower mass flow rate, though the choke mass flow rate is even increased. Subsequently, a parallel two-operating-point approach has been carried out. Results show that the two-operating-point design optimization avoids unacceptable performance deterioration at off design conditions. In the fourth case a redesign is applied across all 7 blade rows of the ATC compressor at the compressor design point. All these case studies are aimed to increase the isentropic efficiency whilst meeting some specified constraints.

6.2.1 Redesign of NASA Rotor 67 at a Single Operating Point

NASA rotor 67 is a low-aspect-ratio design and is the first stage rotor of a two-stage fan. The rotor design pressure ratio is 1.63 at a mass flow rate of 33.25 kg/s. The design rotational speed is 16,043 rpm, which yields a tip speed of 429 m/s and an inlet tip relative Mach number of 1.38. The rotor has 22 blades and an aspect ratio of 1.65 (based on average span/root axial chord). The rotor solidity varies from 3.11 at the hub to 1.29 at the tip. The inlet and exit tip diameters are 51.4 cm and 48.5 cm, respectively, and the inlet and exit hub/tip radius ratios are 0.375 and 0.478 respectively. The rotor was experimentally investigated in NASA Lewis Research Center to produce data for validating numerical results (Strazisar et al., 1989) and has been a popular test case for design optimization (Li et al., 2006a; Lian and Liou, 2005; Oyama and Liou, 2004; Pierret, 2005).

A single passage computational domain with a mesh density of 110x49x37 in the axial, pitchwise and radial directions has been used to model the blade. Fig.6.11 shows the meridional view and blade to blade view of the mesh. The calculated performance map of NASA rotor 67 is compared with the experimental data in Fig.6.12. The mass flow rate is normalized by its choke mass flow rate as also shown in Arnone (1993). Both the calculated pressure ratio and efficiency show fairly good agreement with their experimental values.

Even though the design optimization is carried out at a near choke operating point[1], the machine operates at a very high isentropic efficiency with its performance listed in

[1]The normalized mass flow rate at this operating is 0.995.

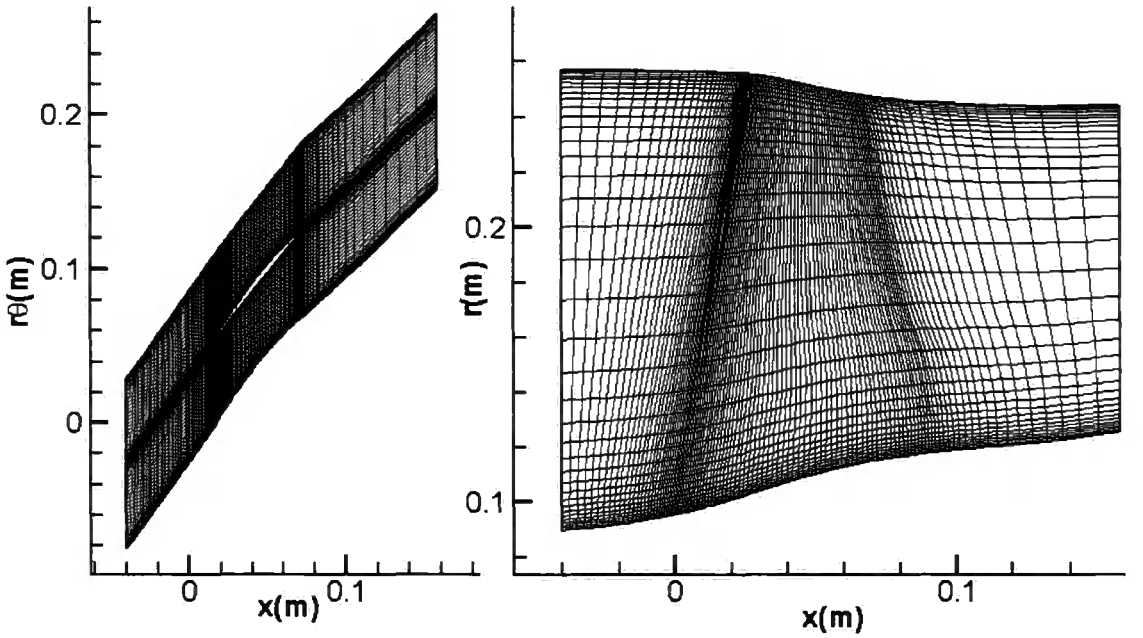


Figure 6.11: Blade to blade view and meridional view of the NASA rotor 67 mesh

Table 6.1. 143 design variables with 13 design variables for each of 11 spanwise sections are used to parameterize the blade shape perturbations. The 11 spanwise mesh sections divide the blade span into 10 approximately equal intervals (it is normally impossible to find 11 radial mesh sections that divide the blade span into 10 exactly equal intervals).

The design optimization is completed with 48 design cycles over 63 hours[1]. The evolution of the objective function and two constraints with design cycles is depicted in Fig.6.13. The objective function is reduced by 19% in total. The overshoot in the objective function curve at the 21^{st} design cycle corresponds to an increase of entropy generation rate and a severe violation of the pressure ratio constraint and much better satisfaction of the mass flow rate constraint relative to its preceding design cycle as revealed by the corresponding spikes in the two constraint curves.

Table 6.1: Performance comparison between the original NASA rotor 67 and the optimised blade

	mass flow rate (kg/s)	pressure ratio	isentropic efficiency (%)
original	34.32	1.6536	90.50
optimised	34.37	1.6523	92.27
change	+0.15%	-0.08%	+1.77

[1]All the single operating point design optimization was run on a 2.2 GHz Opteron processor. The exact time cost also depends on the loading of the computer system.

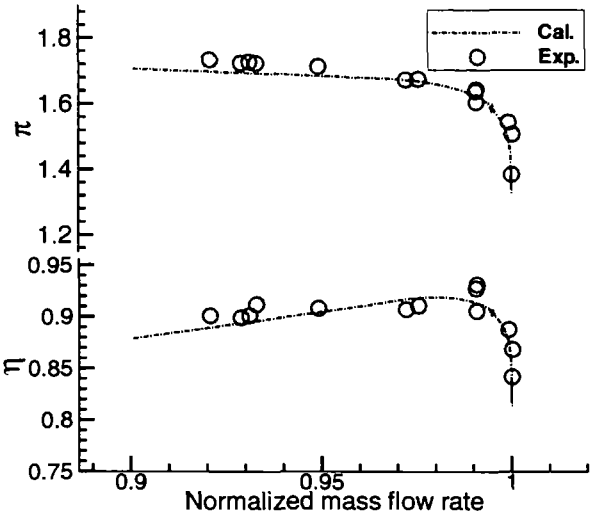


Figure 6.12: Comparison of computed and experimental performance map for the NASA rotor 67

A comparison of the overall performance between the optimised design and the original one at the near choke operating point is given in Table 6.1. The optimised blade increases efficiency by 1.77% point with 0.15% increase in mass flow rate and 0.08% reduction in pressure ratio. This can be considered as a fairly good redesign in terms of efficiency gain and the satisfaction of constraints.

A more detailed flow field comparison between the optimised design and the original one is revealed in Figs.6.14, 6.16 and 6.15. Fig.6.14 depicts the spanwise distribution of the pitchwise averaged efficiency, stagnation pressure ratio and stagnation temperature ratio at the rotor exit. The local efficiency is increased over the whole span with greater increase in the tip and hub regions. The local stagnation pressure ratio and stagnation temperature ratio are raised in the hub region and lowered in the tip region shifting blade loading from the tip region towards the hub region. In the original flow field, as Fig.6.16 reveals, there is a strong passage shock normal to the casing. However in the flow field for the optimised blade, on the blade pressure surface, there are two shocks which are much weaker than that in the original flow field. On the blade suction surface (Fig.6.15), compared with the original flow field, the optimised flow field has a shock which is slightly weaker and is pushed upstream a little. The reduction of the shock strength accounts for the main efficiency gain.

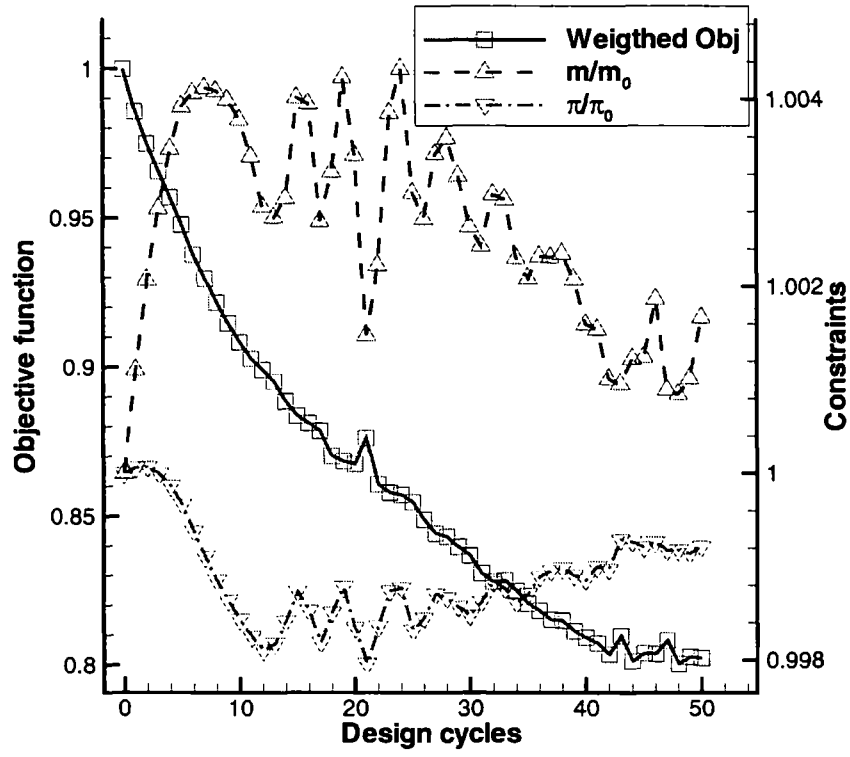


Figure 6.13: Evolution of the objective function and two constraints with design cycles for the NASA rotor 67 redesign

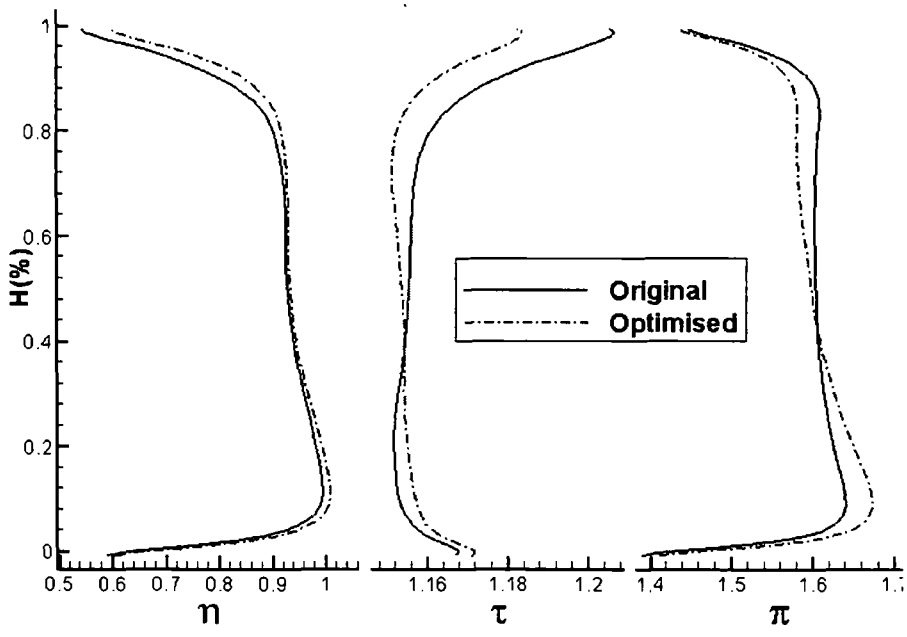


Figure 6.14: Spanwise distributions of efficiency (η), temperature ratio (τ) and pressure ratio (π) for the NASA rotor 67 redesign

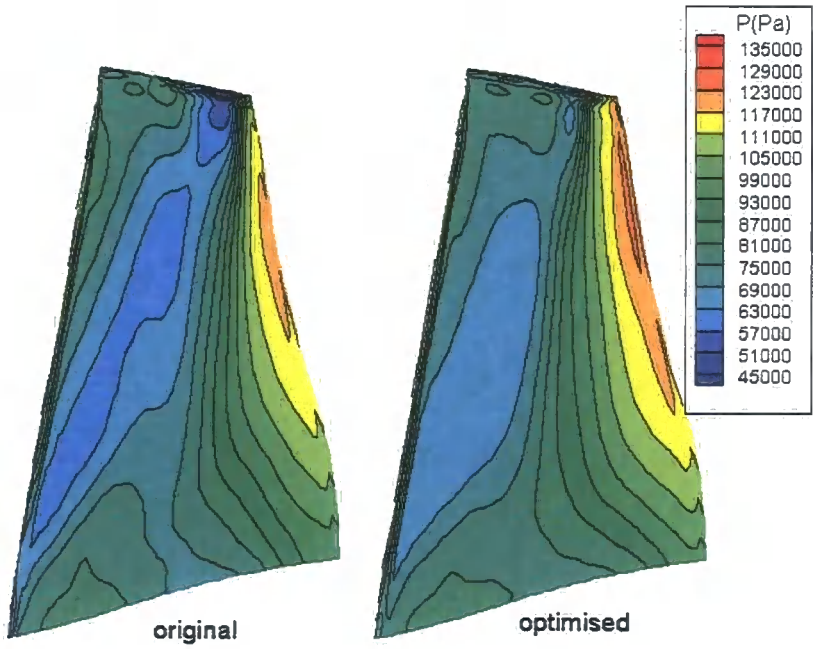


Figure 6.15: Pressure contours on blade suction surfaces of the original and optimised blades for the NASA rotor 67 redesign

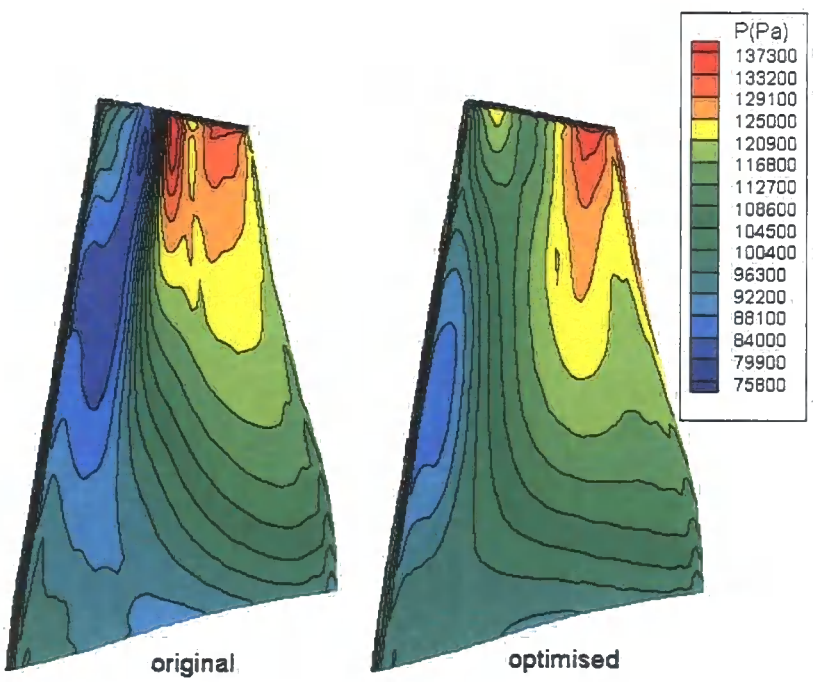


Figure 6.16: Pressure contours on blade pressure surfaces of the original and optimised blades for the NASA rotor 67 redesign

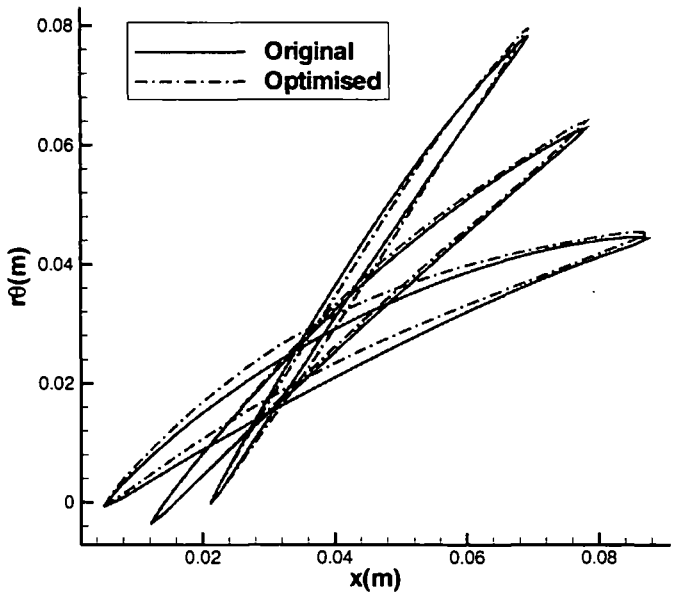


Figure 6.17: Blade geometry comparison between the original blade and the optimised blade for the NASA rotor 67 redesign

The blade geometry change against the original design is shown in Fig.6.17. In the hub region, the blade section has increased camber, which is in line with the increased blade loading around the hub region as revealed in Fig.6.14. There is also significant geometry change from the mid-span to the tip region where the tip sections have been changed from MCA into a reverse-cambered type of profile. Although the geometric change is small in the mid-span region, it is relatively larger in the hub and tip regions and it is much larger than the manufacturing tolerance. This geometric change is normally difficult to be achieved using a trial and error design approach.

6.2.2 Redesign of DLR Compressor Stage at a Single Operating Point

The DLR stage (Dunker et al., 1984) is a transonic axial flow compressor stage without inlet guide vanes. The stage was designed for a rotational speed of 20,260 rpm with a stagnation pressure ratio of 1.51 at an equivalent mass flow rate of 17.3 kg/s. The rotor inlet diameter is 39.8 cm with a hub/tip ratio of 0.51. The rotor consists of 28 MCA-profiled blades with a tip chord length of 6.0 cm. The blade solidities vary between 2.0 and 1.34 from hub to tip. The stator has 31 controlled diffusion blades with solidity varying from 1.68 at the hub to 1.05 at the tip.

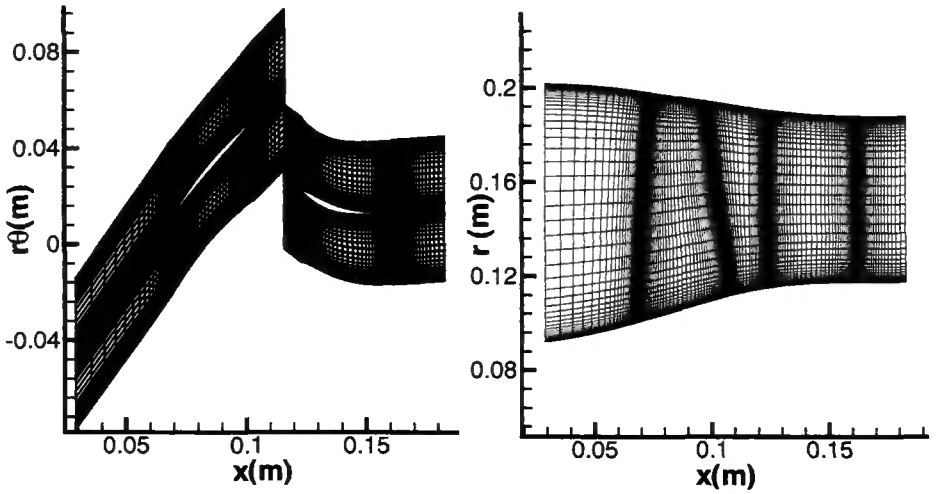


Figure 6.18: Blade to blade view and meridional view of the DLR stage mesh

The single passage H mesh consists of two rows with a mesh density of 110x37x39 for the rotor and 90x35x39 for the stator in the axial, circumferential and radial directions, respectively. The tip and hub clearances are treated by the simple pinched-up approach and 3 cells are used to accommodate the tip or hub gap. The meridional view and blade to blade view of the mesh are shown in Fig.6.18.

The calculated performance map is compared with the experimental data in Fig.6.19. As can be seen the calculated performance map has a narrower stable operating range. From the experimental pressure ratio curve, it is clear that the pressure ratio decreases with the reduction of mass flow rate, which indicates the experimental flow field is quite close to stall. This cannot usually be predicted by the current flow solver by solving the RANS equations in a single passage computational domain. Except for that, the calculated efficiency and pressure ratio versus mass flow rate curves agree reasonably well with the experimental data. The calculated choke mass flow rate is also quite close to the experimental value.

The redesign is carried out at the stage peak efficiency operating point of the original design with its performance listed in Table 6.2. There are 154 design variables with 14 design variables on each of 11 spanwise sections parameterizing perturbations to the rotor blade shape and 121 design variables with 11 design variables on each of 11 spanwise sections to perturb the stator blade shape.

It takes around 4 days to complete 36 design cycles. The evolution of the objective

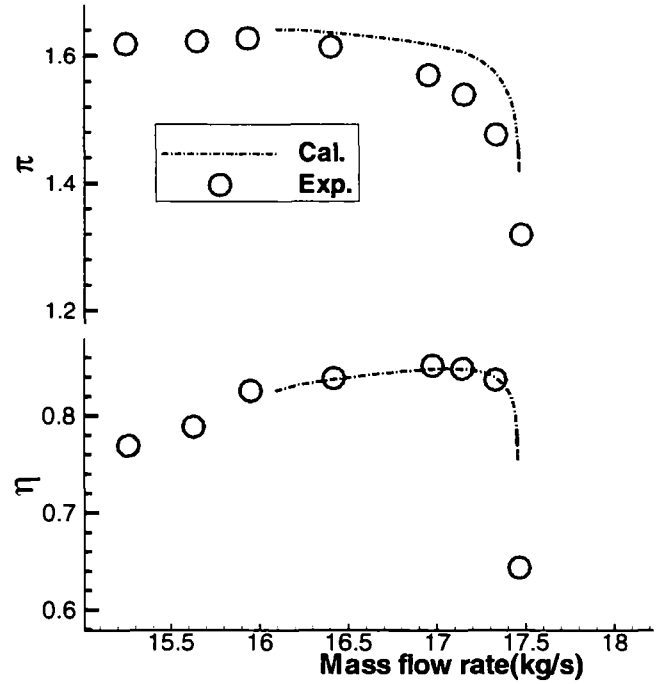


Figure 6.19: Comparison of calculated performance map with experimental data for the DLR stage

function as well as two constraints with design cycles is depicted in Fig.6.20. 80% of the total drop in the objective function is achieved within the first 10 design cycles, while the remaining 26 design cycles consumes most of the elapsed time with only 20% drop. This is probably due to the shortcoming of the steepest descent method, which converges slowly when the curvature of the design space is quite different in different directions. It can also be seen from Fig.6.20 that the constraints are violated significantly for the first few design cycles and are better satisfied when the design process converges.

Table 6.2: Performance comparison between the original DLR stage and the optimised blade

	mass flow rate(kg/s)	pressure ratio	isentropic efficiency(%)
original	17.07	1.5987	85.11
optimised	17.07	1.5987	85.83
change	0.0%	0.0%	+0.72

The performance of the original blade and the optimised design is listed in Table 6.2. The optimised design has an efficiency that is 0.72% point higher than its original counterpart, while it has the same mass flow rate and pressure ratio as the original values. The optimised blades have an increase in efficiency over the whole span at the

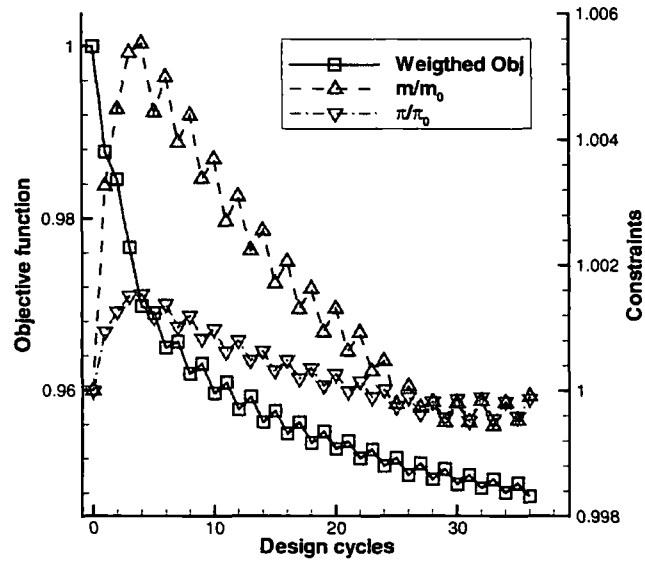


Figure 6.20: Evolution of the objective function and two constraints with design cycles for the DLR stage redesign

stage exit (Fig.6.21). It can also be seen that the pressure ratio and temperature ratio are decreased in both the hub and tip regions and increased in the mid-span region suggesting a spanwise loading shift towards the mid-span region from both endwall regions. This reduced loading around endwall regions is believed to contribute to the achieved efficiency gain by reducing flow losses there. Examination of the pressure coefficient [1] distributions on the three span locations of the rotor (Figs.6.22(a), 6.22(b) and 6.22(c)) suggests that there is also a chord-wise loading shift from the leading edge towards the trailing edge. As Figs.6.22(a) and 6.22(b) reveal, near the hub and mid-span regions, blade loading is shifted mainly from the leading edge region to the mid-chord region. While in the tip region, the loading shift is from the blade leading edge region to the trailing edge region.

The geometry of the original rotor and stator is compared with that of the optimised blades at three span locations: 25% span, 50% span and 85% span (Figs.6.23 and 6.24). The major geometry change is seen to occur in the rotor blade with most change taking place near the rotor blade trailing edge. During the design process, the round rotor leading edge shape is well preserved (Fig.6.23).

[1]The pressure coefficient definition is given by: $Cp = \frac{p-\bar{p}_i}{\bar{p}_i^* - \bar{p}_i}$, with $\bar{p}_i$ as the mass averaged static pressure at an inlet plane and $\bar{p}_i^*$ as the mass averaged total pressure at an inlet plane. The total pressure is calculated at a reference frame attached to the corresponding blade row.

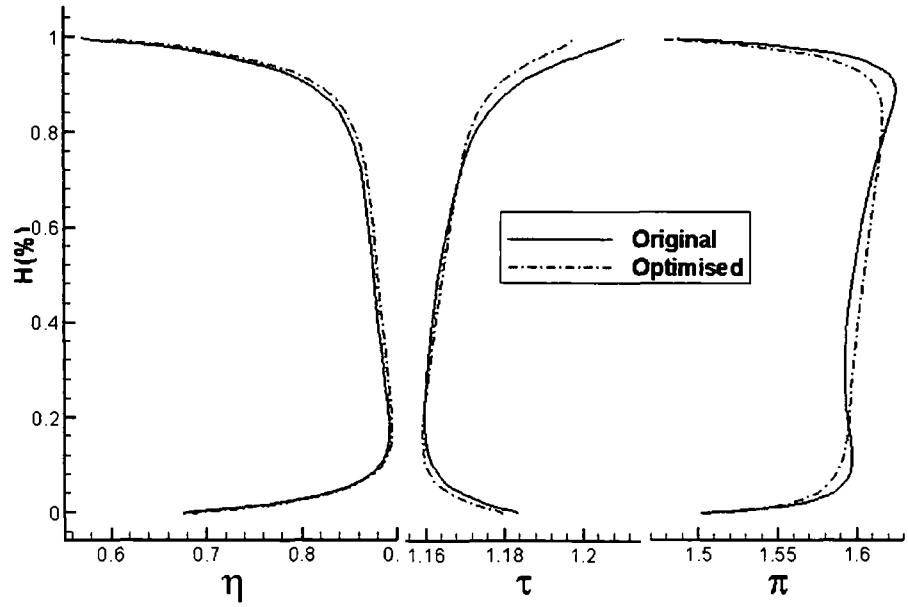
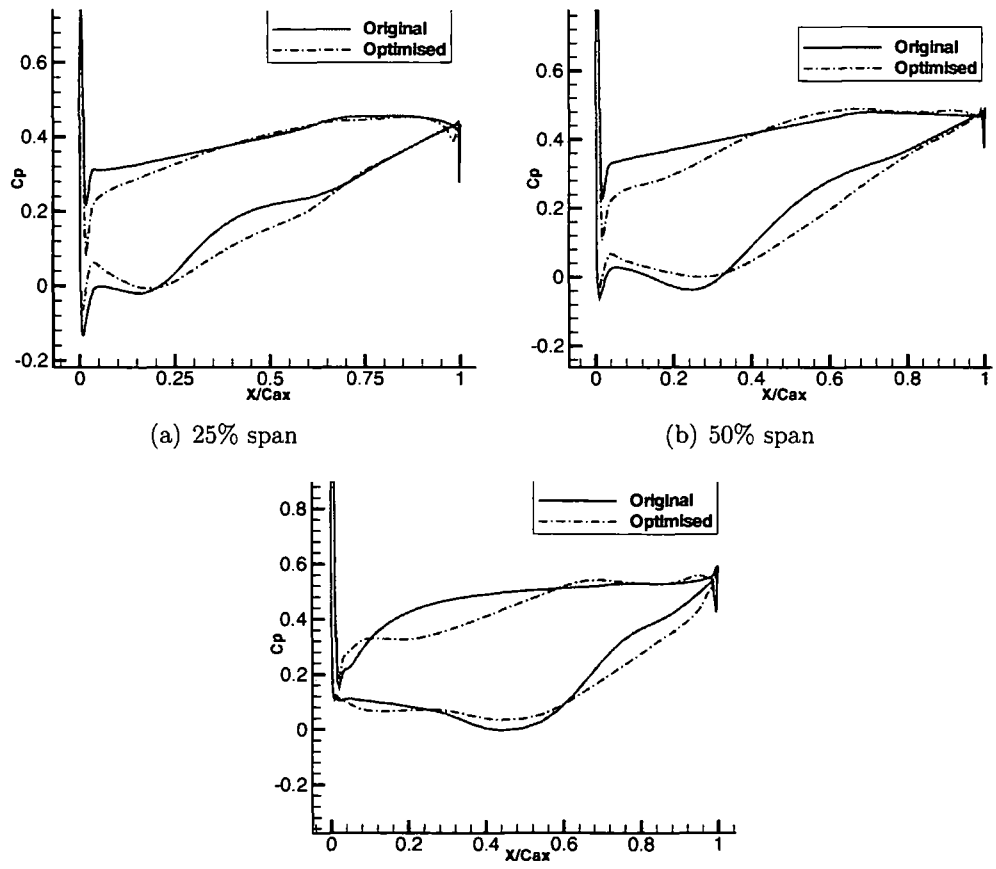


Figure 6.21: Spanwise distributions of efficiency (η), temperature ratio (τ) and pressure ratio (π) for the DLR stage redesign



(a) 25% span

(b) 50% span

(c) 85% span

Figure 6.22: Pressure coefficient distributions on different span sections of the original DLR rotor and the optimised rotor

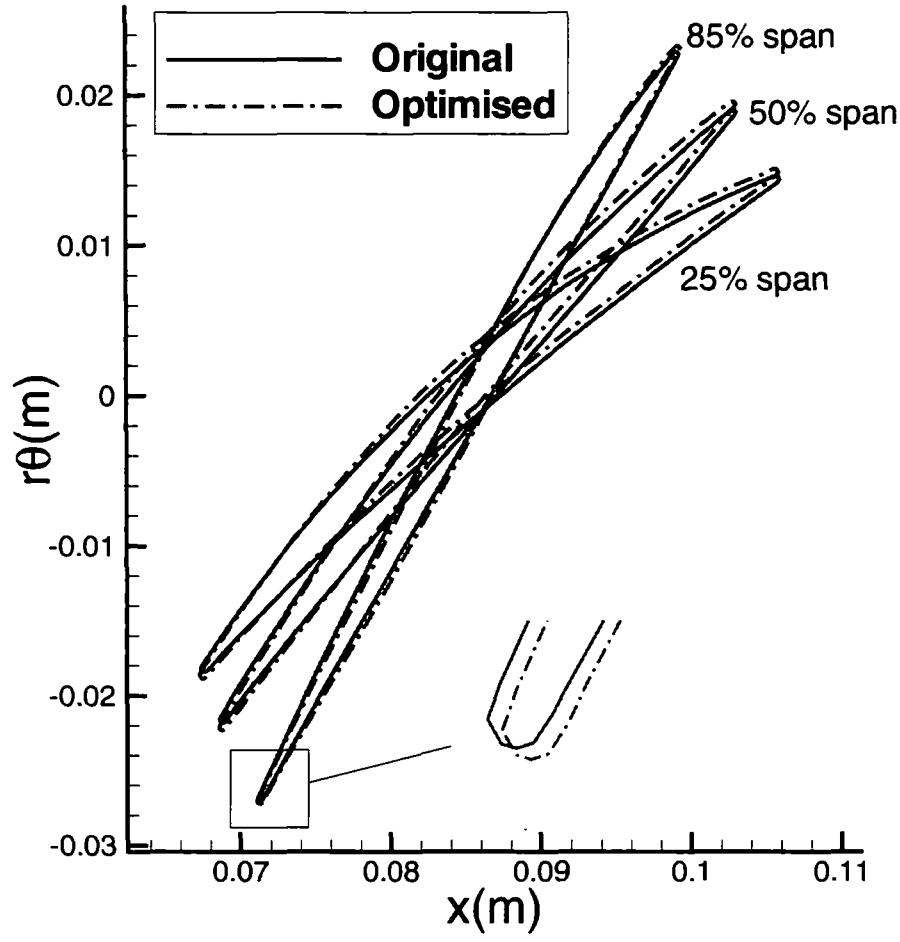


Figure 6.23: Rotor blade geometry comparison between the original DLR rotor and the optimised rotor

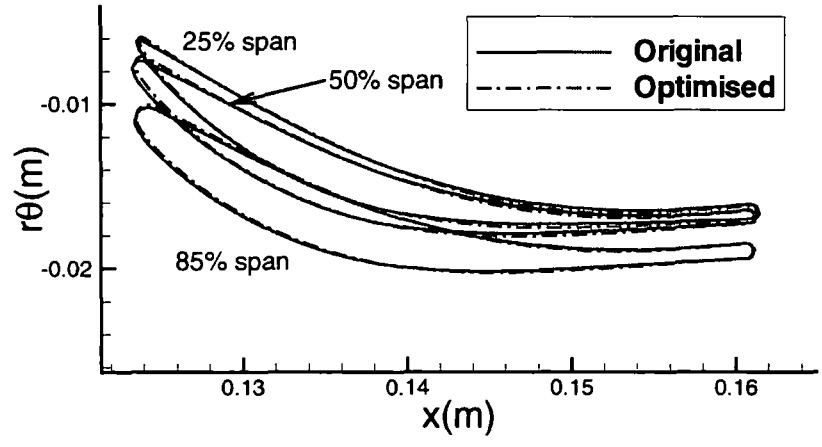
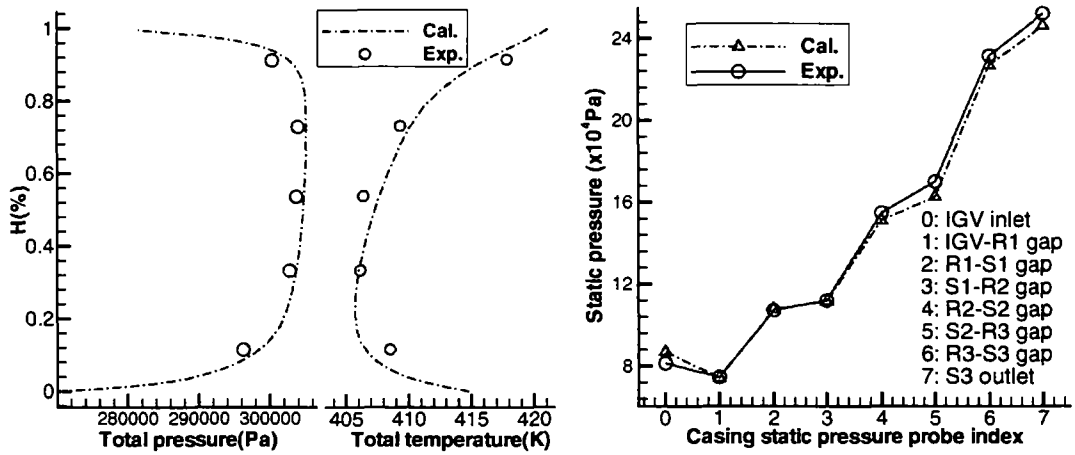


Figure 6.24: Stator blade geometry comparison between the original DLR stator and the optimised stator

6.2.3 Redesign of ATC Compressor

The ATC three-stage transonic test compressor is a high specific flow, high pressure ratio compressor designed using 3D aerodynamic design methods to achieve high performance (Li and Wells, 1999). The compressor has a design mass flow of 26.3 kg/s at a pressure ratio of 3:1. The compressor has been extensively tested both aerodynamically and aeromechanically in the rig to evaluate its aerodynamic performance and structural integrity. For the work reported here, the flow field inside the machine at its design point is simulated. The calculated spanwise distribution of circumferentially averaged stagnation pressure and temperature at the compressor exit is in close agreement with the experimental data (Fig.6.25(a)). The calculated casing static pressure distribution also has a close agreement with the experimental data (Fig.6.25(b)).



(a) Spanwise distributions of stagnation pressure and temperature at the ATC compressor exit

(b) Casing static pressure distributions

Figure 6.25: Comparison of the calculated flow field with experimental data for ATC

Redesign of IGV-Rotor-Stator Configuration

The first three rows, including the IGV, the first rotor and first stator, are taken out of the 7-row ATC compressor for a quick case study. The computational domain after the first stator is extended to allow for the application of a subsonic exit boundary condition (the static pressure is specified at hub or tip with the simple radial equilibrium to obtain the static pressure at other radial locations). Fig.6.26 shows the blade to blade view and meridional view of the three-bladerow mesh. The mesh has 49 mesh points in the

radial direction and 37 mesh points in the circumferential direction for all the three rows. The IGV, rotor and stator have 125, 124 and 117 mesh points in the axial direction respectively.

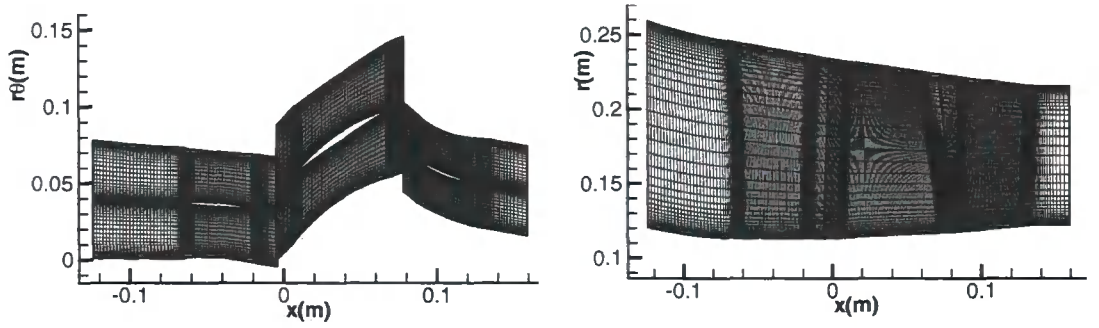
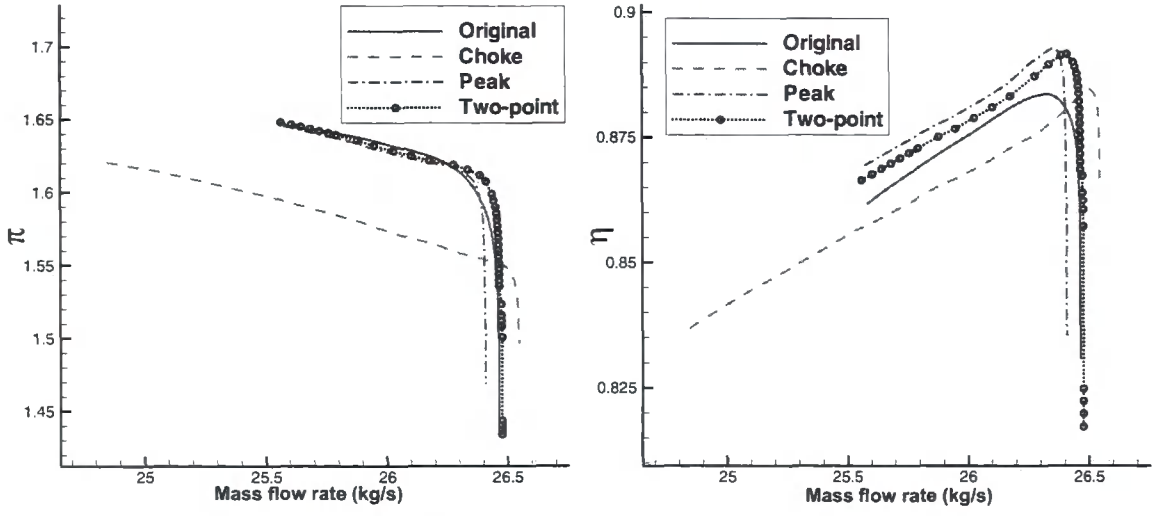


Figure 6.26: Blade to blade view and meridional view of the IGV-Rotor-Stator mesh

In the design optimization process, the IGV remains unchanged while the rotor and stator are allowed to be changed. 187 design variables are used to parameterize perturbations to the rotor blade shape, and 154 design variables are used for the stator blade shape. These design variables are distributed on 11 span sections dividing a blade span into 10 approximately equal intervals with 17 design variables on each radial section of the rotor and 14 design variables on each radial section of the stator.



(a) pressure ratio versus mass flow

(b) efficiency versus mass flow

Figure 6.27: Performance map comparison between different ATC 3-row designs

As a first attempt, this three-row configuration is redesigned at the stage peak efficiency point to further increase its isentropic efficiency while maintaining its mass flow rate and

pressure ratio. The evolution of the weighted objective function and two constraints with design cycles is depicted in Fig.6.28. The design process converges after about 20 design cycles with about 1.0% point efficiency gain and negligible change in mass flow rate and pressure ratio. The disappointing point of this redesign, denoted as 'peak' in Figs.6.27(a) and 6.27(b), is the noticeable reduction in the choke mass flow rate.

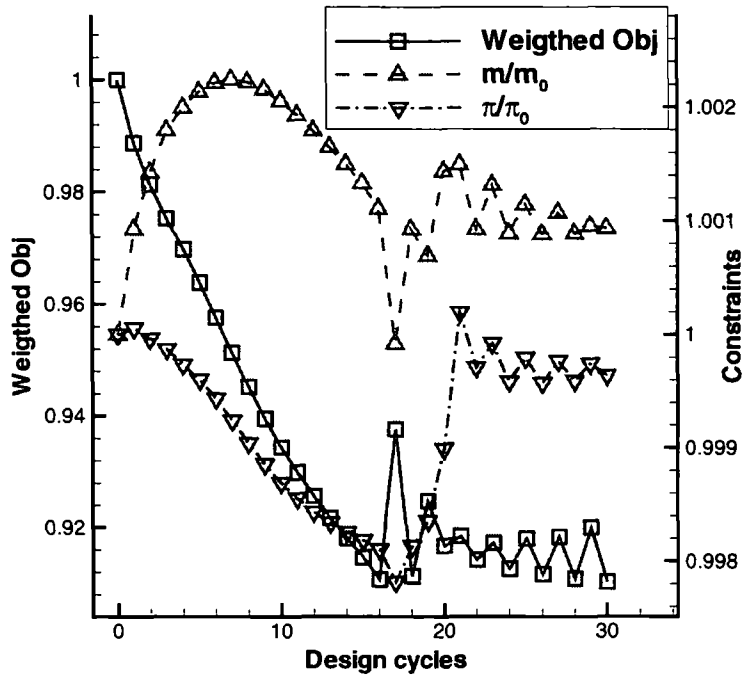


Figure 6.28: Evolution of the weighted objective function and two constraints with design cycles for the ATC 3-row redesign at the peak efficiency point

As the second attempt, a redesign is carried out at a near choke mass flow rate of the three-row configuration, attempting to get a design which can not only deliver higher efficiency at the original stage peak efficiency operating point but also maintain the choke mass flow rate. The evolution of the weighted objective function and two constraints with design cycles is drawn in Fig.6.29. The design process also converges after about 20 design cycles with 1.5% point efficiency gain and negligible change in mass flow rate and pressure at the near choke operating point. Unfortunately the redesign, denoted as 'choke' in Figs.6.27(a) and 6.27(b), deteriorates its performance unacceptably at operating points of lower mass flow rate from the near choke value, though it even increases the choke mass flow rate.

Based upon the outcome of the two single-point design optimizations, an attempt of

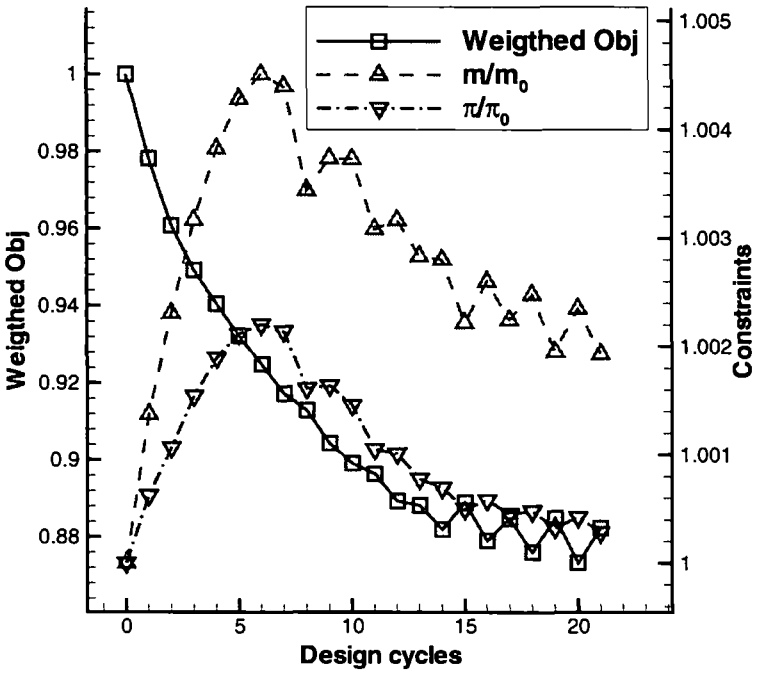


Figure 6.29: Evolution of the weighted objective function and two constraints with design cycles for the ATC 3-row redesign at a near choke point

two-operating-point design optimization, taking into account of performance at the two operating points, is made to achieve a redesign increasing the stage peak efficiency without lowering the choke mass flow rate. The objective function to be minimised is given by

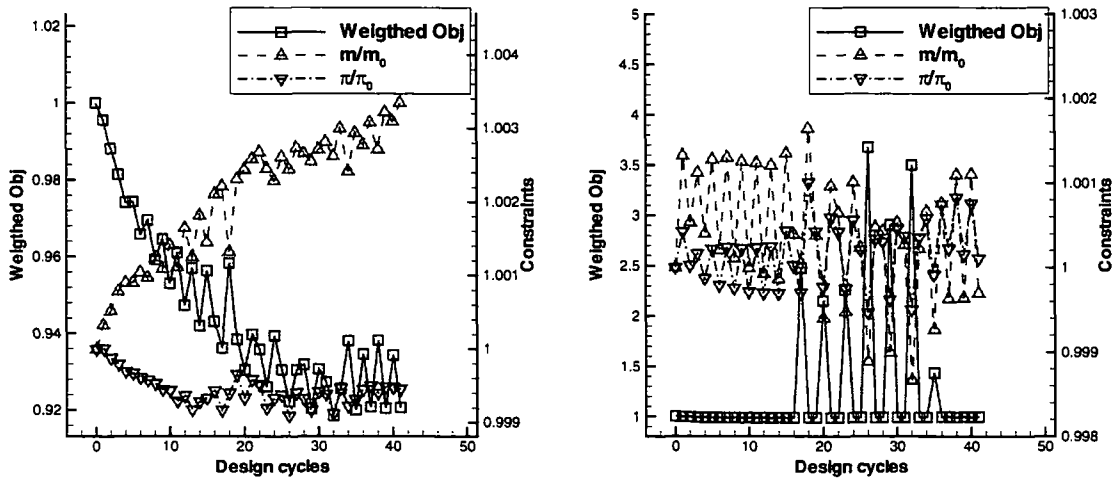
$$I = \sigma_{peak} \cdot I_{peak} + \sigma_{choke} \cdot I_{choke}$$

The design target can be achieved by increasing the efficiency at the stage peak efficiency point of the original design and preventing the choke mass flow rate from decreasing. This is to suggest that more priority should be put on the peak efficiency point, therefore σ_{peak} is assigned a value of 0.8 and $\sigma_{choke} = 0.2$. The gradient of the total objective function to a design variable is expressed as follows

$$\frac{dI}{d\alpha} = \sigma_{peak} \cdot \frac{dI_{peak}}{d\alpha} + \sigma_{choke} \cdot \frac{dI_{choke}}{d\alpha}$$

Though only two operating points are considered in this demonstration case, the parallelized code can handle as many operating points as one likes. The change of sub-objective functions I_{peak} and I_{choke} and their corresponding constraints with design cycles is shown in Fig.6.30, where one can see that there are many zigzags on the sub-objective function curves, while the objective functions in Fig.6.29 and Fig.6.28 are

much smoother. This is due to the balance of the performance between two different operating points. The sub-objective function at the peak efficiency point mainly acts as an objective function to be minimized (Fig.6.30(a)), while the sub-objective function at the near choke point mainly plays a constraint role (Fig.6.30(b)).



(a) peak efficiency point

(b) near choke point

Figure 6.30: Evolution of the sub-objectives and constraints with design cycles at the two operating points for the ATC 3-row two-point redesign

From Figs.6.27(a) and 6.27(b), by comparing the two-operating-point design with the single-point design optimization at the stage peak efficiency point, one can see that the performance curves of the two-point design are shifted towards higher mass flow rate and the original design choke margin is maintained as a result of the design constraint on the near choke operating point. The peak efficiency point of the two-point design has also been shifted towards higher mass flow and it differs from the original stage peak efficiency point. The pressure ratio versus mass flow rate characteristic of the two-point design is steeper than the original design. The maximum efficiency gain seems to be achieved between the two chosen optimization points. The peak efficiency has been improved by 1.0% point. Overall, the two-point design optimization is a better design over the other two.

The difference in blade geometry between the original blades and all the three optimised blades is shown in Figs.6.31, 6.32 and 6.33. There is considerable geometry change in all the optimised blades against the original blades. Both the rotor and stator blades have more geometry changes in the hub region with increased cambers than in the tip

region (Figs.6.31 and 6.33). Between different optimised blades, the optimised blades at the near choke point have less camber increase than the other two designs. There is also quite a significant geometry difference between these optimised designs particularly around the rotor leading edge region (Fig.6.32). Comparing the two-point design with the peak efficiency design, the blades geometry is generally very close except for the rotor leading edge region with a maximum difference up to 1.0 mm.

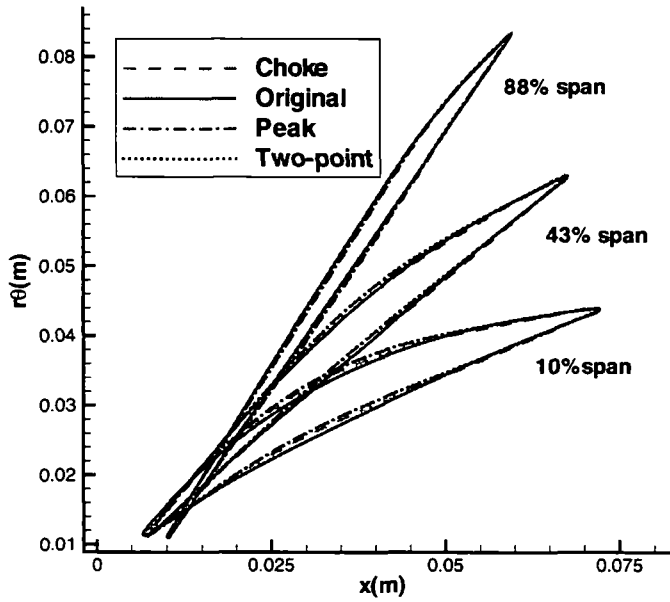


Figure 6.31: Rotor blade geometry comparison between different ATC 3-row designs

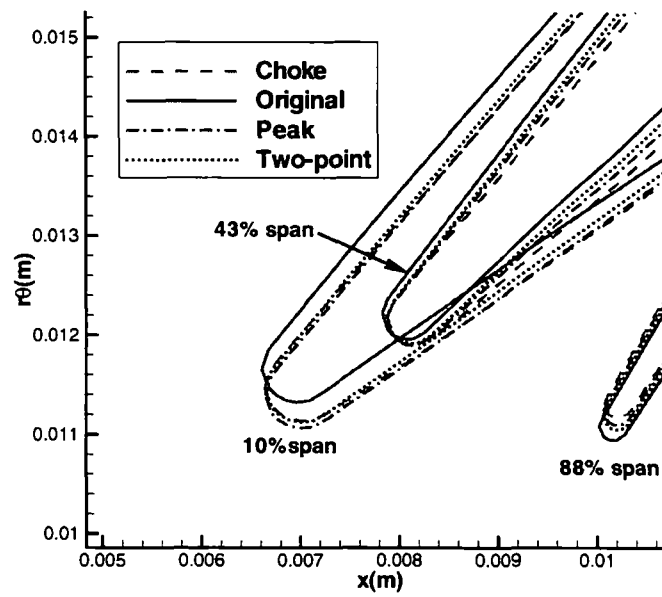


Figure 6.32: Close-up of rotor blade leading edges for different ATC 3-row designs

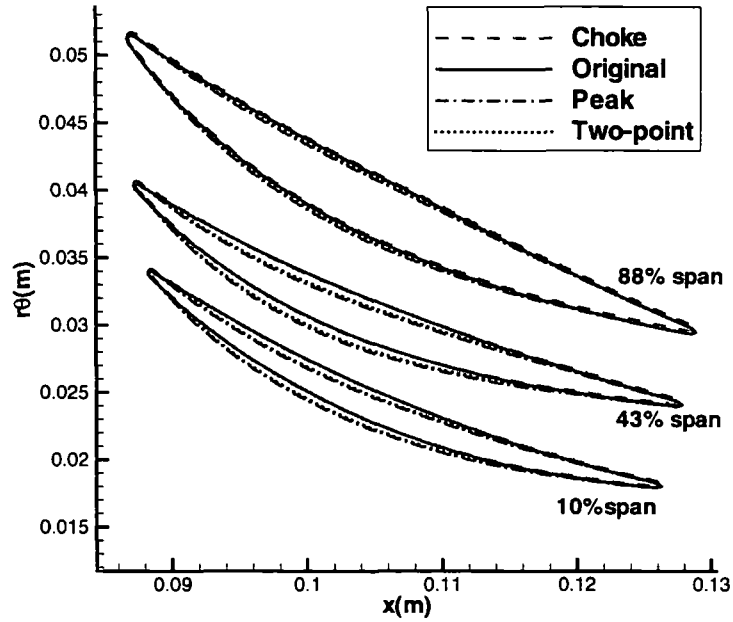


Figure 6.33: Stator blade geometry comparison between different ATC 3-row designs

Redesign of the 7-Bladerow Configuration

In this case study, the computational domain consists of 7 blade rows. Each row has 45 mesh points in the circumferential direction and 37 in the radial direction. The number of mesh points in the axial direction varies for different rows and are listed in Table 6.3. The blade to blade view of a mid-span section and meridional view of the mesh are shown in Fig.6.34, with every fourth mesh point shown in the axial direction and every third mesh point shown in the circumferential direction for a clear view.

Table 6.3: Number of axial mesh points and design variables for each row of the ATC compressor

	IGV	R1	S1	R2	S2	R3	S3
Number of axial mesh points	125	124	102	107	117	111	124
Number of design variables	0	187	154	165	187	165	165

In the design optimization, the IGV is unchanged as for the three-row design optimization, whilst the other 6 rows are allowed to change, with the number of design variables for each row listed in Table 6.3, resulting in 1023 design variables in total. For each row, the design variables are distributed on 11 spanwise sections with the same number of design variables for each section.

A single-point design optimization is carried out at the original design point of the

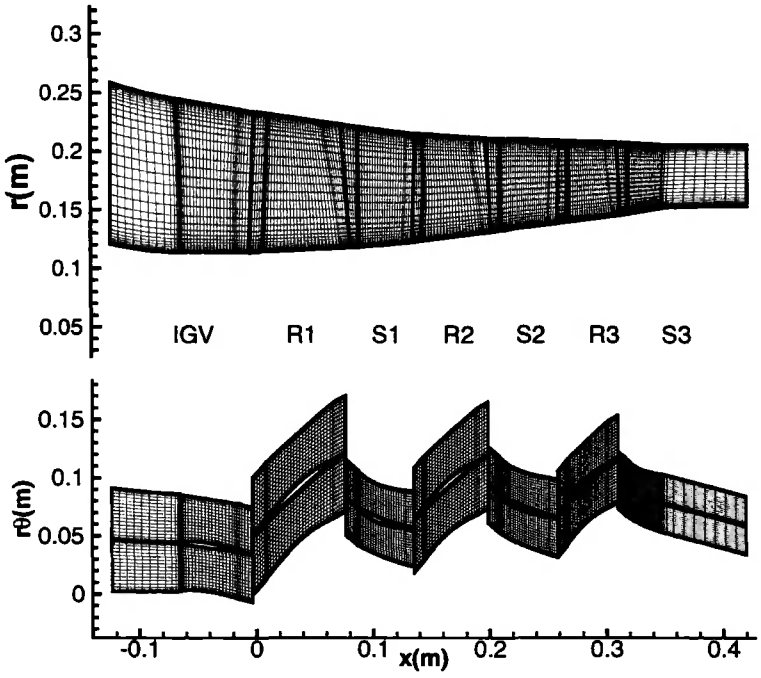


Figure 6.34: Meridional view and blade to blade view of the ATC mesh

compressor. 29 design cycles are completed over 11 days. The history of the objective function and the two constraints with design cycles is shown in Fig.6.35 where a 20% reduction of the objective function is obtained. The behavior of the two constraints with design cycles is qualitatively the same as that of the other single-point case studies: there is more violation at the beginning and the constraints are much better satisfied when the objective function progresses to its convergence.

The performance comparison between the original compressor and the optimised one is listed in Table 6.4. The optimised design has an efficiency that is 2.47% point higher than the original one with 0.34% increase in mass flow rate and 0.08% decrease in pressure ratio. Fig.6.36 compares the spanwise distribution of efficiency, stagnation temperature ratio and stagnation pressure ratio at the compressor exit. The optimised design has higher efficiency than the original design over the whole span. The stagnation temperature ratio in the optimised design is lower than the original design over the whole span, indicating less work done into the working fluid. The stagnation pressure ratio in the optimised design is increased in the tip region and decreased in the hub region.

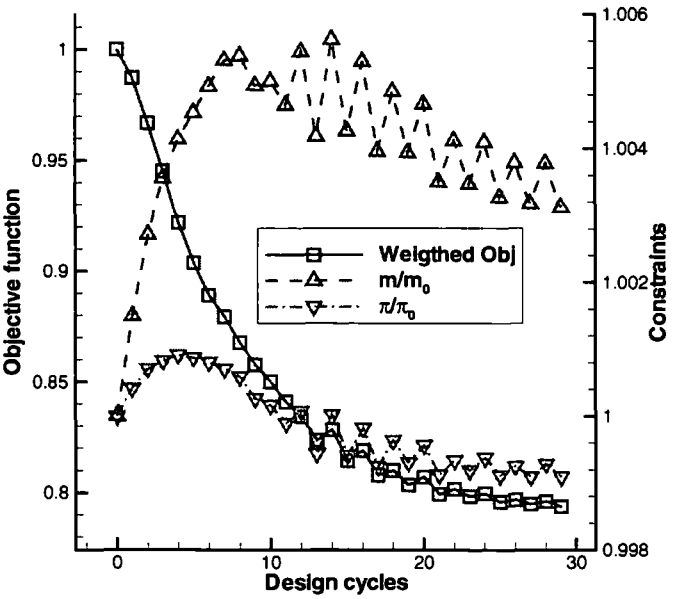


Figure 6.35: Evolution of the objective function and two constraints with design cycles for the ATC 7-row redesign

Table 6.4: Performance comparison between the original ATC compressor and the optimised compressor

	mass flow rate (kg/s)	pressure ratio	isentropic efficiency(%)
original	26.46	2.9885	86.81
optimised	26.55	2.9860	89.28
change	+0.34%	-0.08%	+2.47

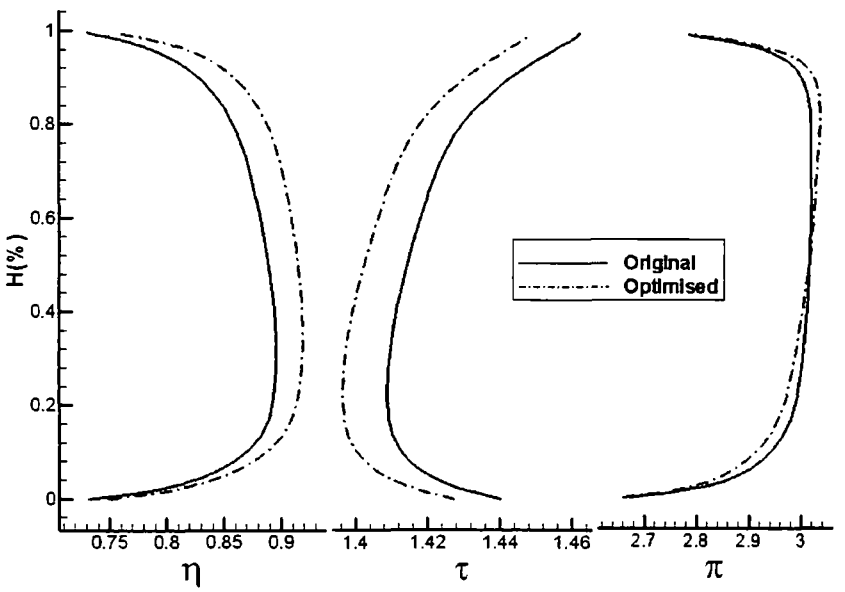
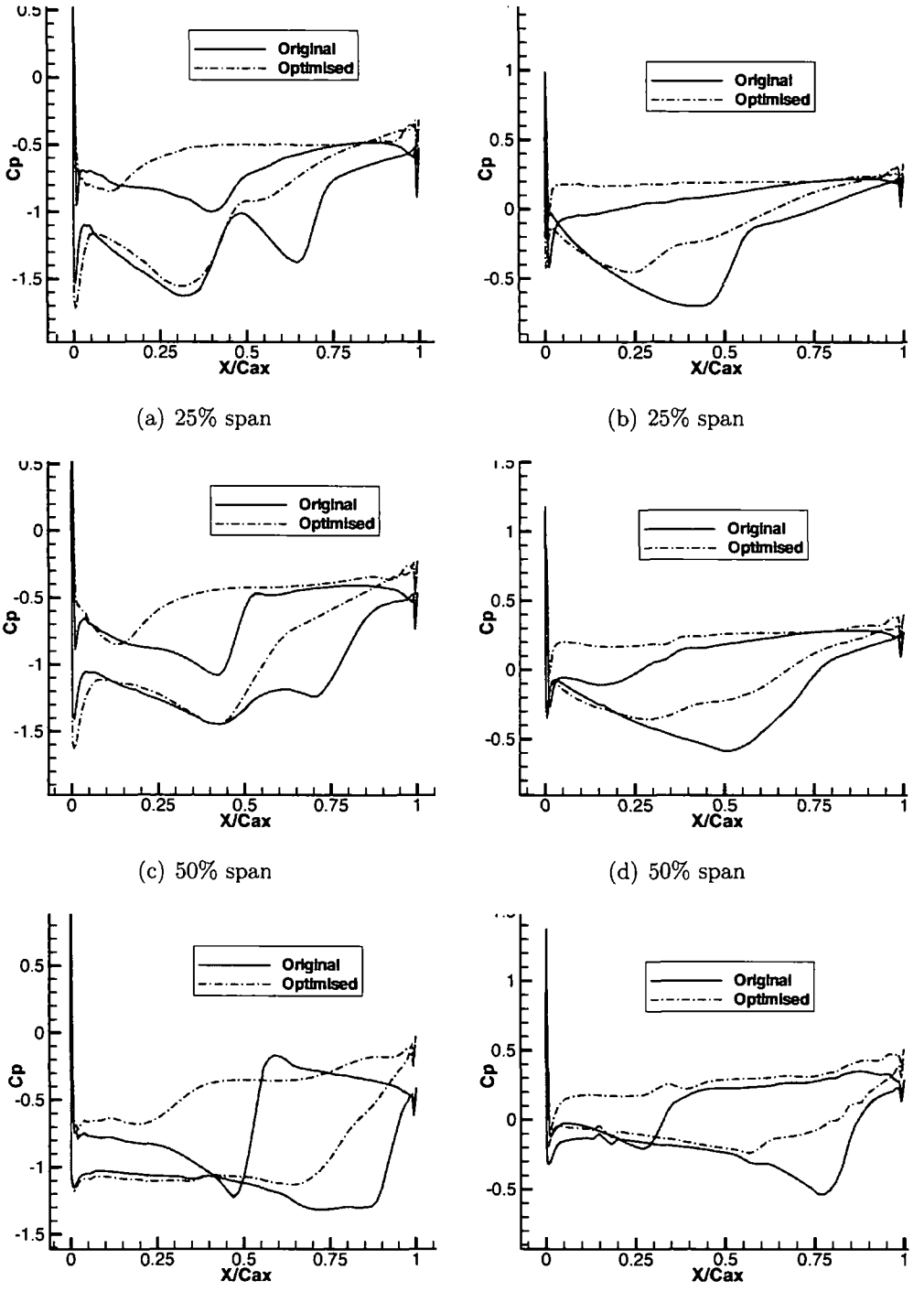


Figure 6.36: Comparison of efficiency (η), temperature ratio (τ) and stagnation pressure ratio (π) spanwise distributions for the ATC 7-row redesign



(a) 25% span

(b) 25% span

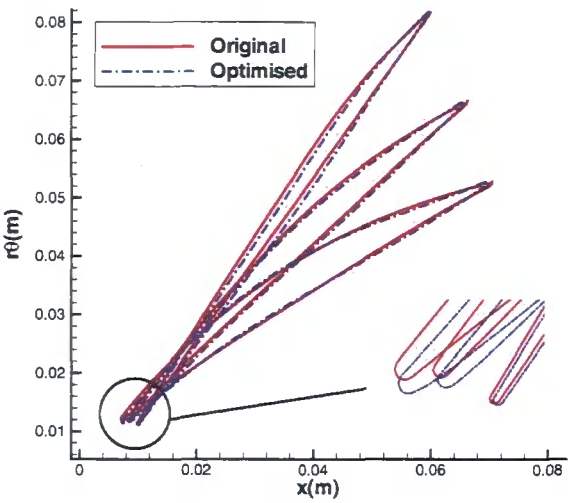
(c) 50% span

(d) 50% span

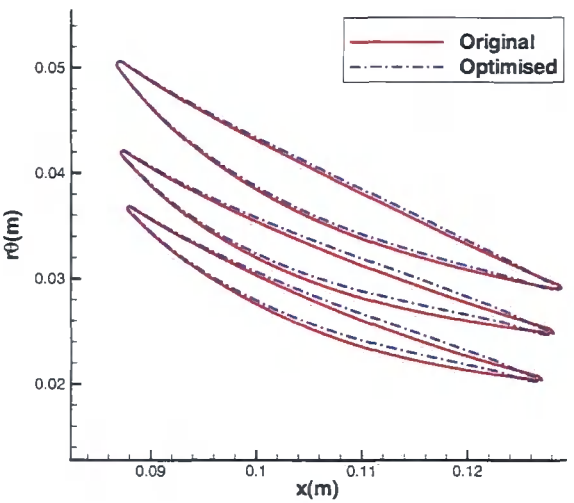
(e) 85% span

(f) 85% span

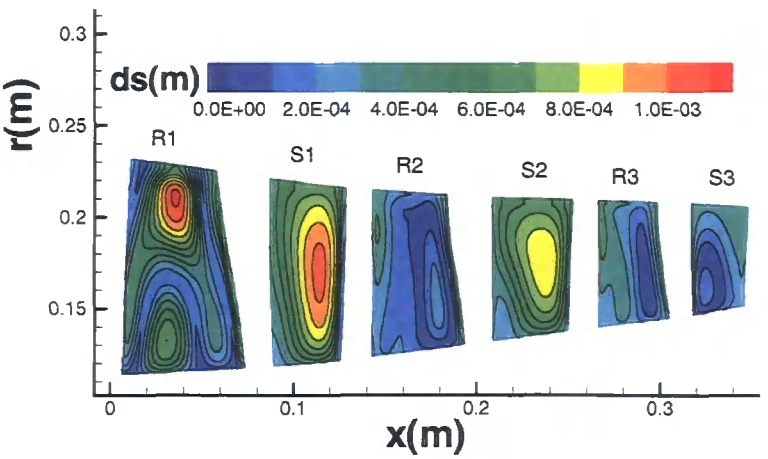
Figure 6.37: Pressure coefficient distributions on different span sections of the first rotor (left) and the second rotor (right) for the ATC 7-row redesign



(a) Rotor 1



(b) Stator 1



(c) Geometry change on blade surfaces

Figure 6.38: Blade geometry change for the ATC 7-row redesign

Pressure coefficient distributions on three different spanwise locations of the first and second rotors are presented in Figs.6.37. The optimised design shifts blade loading of the first two rotors upstream relative to the original designs. The strength of passage shocks inside the first and second rotors is significantly reduced resulting in the efficiency increase in the optimised design. With design constraints on the mass flow rate and overall stagnation pressure ratio, the design optimization has driven the redesign of the three transonic stages to different matching conditions as evidenced by the increased pressure coefficients at the trailing edge of rotors 1 and 2.

The change of blade geometry is revealed in Figs.6.38(a), 6.38(b) and 6.38(c). Figures 6.38(a) and 6.38(b) compare the original blade geometry to the optimised blade geometry of the first rotor and stator. The blade profile has been changed significantly in the leading edge region, particularly near the hub. There is also a considerable change in geometry in the tip region where the original MCA type of profile has been replaced by a reverse cambered type of profile. The first stator is decambered with increased turning in the front part of the airfoil and reduced turning towards the trailing edge. During the design process, the round leading edge shape is well preserved (Fig.6.38(a)). Fig.6.38(c) shows the distance change (ds in Fig.6.38(c)) of each mesh point on one side of the blades. Due to the nature of the shape perturbation parameterization in the present work, corresponding mesh points on both sides of the blades have the same distance change (except for the leading edge region) during the design optimization. Hence, the distance change on one side of the blades is shown only. The first rotor, the first stator and the second stator have major geometry change with the maximum change up to 1.1 mm.

6.3 Summary

The close agreement of the lift coefficient gradient by the adjoint method with that by the finite difference method demonstrates the validity of the adjoint method in a gradient calculation and its appropriate implementation. These are further confirmed by the inverse design well recovering the target pressure coefficient distribution and the target airfoil profile. The close agreement of the gradient results by the adjoint

method with those by the finite difference method for a 2D rotor-stator stage case demonstrates the appropriate formulation and implementation of the adjoint mixing-plane treatment. The characteristics based adjoint mixing-plane treatment is also shown to be nonreflective. The validity of the adjoint mixing-plane treatment is further confirmed by the optimization of the 2D compressor stage configuration, also showing the added benefit of carrying out design optimization in a multi-bladerow environment.

Four redesigns have been carried out, producing encouraging improvements. The redesign of NASA rotor 67 delivers 1.77% point increase in efficiency whilst maintaining mass flow rate and stagnation pressure ratio at values close to those specified for the original design. The optimised DLR stage has an increase in stage efficiency of 0.72% point at the same stagnation pressure ratio and mass flow rate. The two-operating-point redesign of the first three rows of the ATC compressor generates a design which has higher efficiency over the original design at the original stage peak efficiency point and ensures the choke mass flow rate does not decrease at the same time. The ATC 7-row compressor is optimised to achieve 2.47% point efficiency increase while maintaining the original design point stagnation pressure ratio and mass flow rate. The multi-bladerow design cases, particularly the redesign of ATC 7-row compressor have demonstrated the appropriate formulation and implementation of the proposed adjoint mixing-plane treatment and the capability of the design system in aerodynamic design optimization for multi-bladerow turbomachines.

Chapter 7

Nonlinear Harmonic Phase Solution Method for Unsteady Flows

This chapter presents the unsteady Reynolds Averaged Navier-Stokes equations and their solution methods for blade flutter predictions, forming an essential part of the basis for the aeromechanical-related design optimization to be introduced later. The frequency domain method - Nonlinear Harmonic Phase Solution method is adopted here to discretize in time the URANS equations at three phases of a period of unsteadiness, leading to three sets of coupled steady-like equations. The developed unsteady flow solver is validated against both a semi-analytic method and experimental data.

7.1 Governing Equations

When dealing with blade flutter, a computational domain vibrates with a vibrating blade. The deformation of a control volume is not normally restricted to a certain coordinate direction. Each mesh point generally has a velocity of which the three components in the axial, circumferential and radial directions are not zero. Flux calculations through the boundary of a control volume need to account for fluxes due to the motion of its bounding surface at the three coordinate directions. The URANS equations in a cylindrical coordinate system are given by

$$\frac{\partial U}{\partial t}+\frac{\partial(F-Uu_g-V_x)}{\partial x}+\frac{\partial(G-Uv_g-V_\theta)}{r\partial\theta}+\frac{\partial r\,(H-Uw_g-V_r)}{r\partial r}-S=0 \quad (7.1)$$

where u_g, v_g and w_g are grid/mesh velocity components in the axial, circumferential and radial directions respectively. The definition of inviscid and viscous fluxes and the

source term in the above equations is the same as those defined for the RANS equations (Eq.3.1). For the sake of simplicity and convenience in the following illustration, a shorthand is introduced:

$$R(U) = \frac{\partial(F - Uu_g - V_x)}{\partial x} + \frac{\partial(G - Uv_g - V_\theta)}{r\partial\theta}\frac{\partial r\,(H - Uw_g - V_r)}{r\partial r} - S$$

Eq.7.1 can be simplified accordingly as follows:

$$\frac{\partial U}{\partial t} + R(U) = 0 \tag{7.2}$$

In a Nonlinear Harmonic Phase Solution method, the flow solution is expressed as the sum of its time-averaged value plus one harmonic:

$$U = \bar{U} + U_A \sin(\omega t) + U_B \cos(\omega t) \tag{7.3}$$

where U is a function of both space and time, while $\bar{U}$, U_A and U_B are functions of space only. The corresponding governing equations with the above flow solution substituted are given by

$$[U_A \cos(\omega t) - U_B \sin(\omega t)]\,\omega + R\,[U(\omega t)] = 0 \tag{7.4}$$

Though the above equation is still time-dependent, there are only three sets of unknowns (U_A, U_B and $\bar{U}$). Choosing the three particular phases $\omega t = -\frac{\pi}{2}, 0, \frac{\pi}{2}$ and using -1, 0 and 1 to denote the three phases for simplicity and convenience, one has

$$\begin{aligned} U_{-1} &= U\left(\omega t = -\frac{\pi}{2}\right) \\ U_0 &= U\left(\omega t = -0\right) \\ U_1 &= U\left(\omega t = \frac{\pi}{2}\right) \end{aligned}$$

The flow solution at the three phases can be expressed in terms of $\bar{U}, U_A$ and U_B:

$$\begin{aligned} U_{-1} &= \bar{U} - U_A \\ U_0 &= \bar{U} + U_B \\ U_1 &= \bar{U} + U_A \end{aligned}$$

In return, U_A, U_B and $\bar{U}$ can be expressed in terms of the flow solution at the three phases:

$$\begin{aligned} U_A &= \frac{U_1 - U_{-1}}{2} \\ U_B &= U_0 - \frac{U_1 + U_{-1}}{2} \\ \bar{U} &= \frac{U_1 + U_{-1}}{2} \end{aligned} \tag{7.5}$$

Substituting the three phases into Eq.7.4 yields

$$\begin{aligned} U_B\omega + R(U_{-1}) &= 0 \\ U_A\omega + R(U_0) &= 0 \\ -U_B\omega + R(U_1) &= 0 \end{aligned} \tag{7.6}$$

Substituting U_A and U_B in Eq.7.5 into Eq.7.6 yields the following equations expressed in terms of flow solutions at the three phases only:

$$\begin{aligned} \left(U_0 - \frac{U_1 + U_{-1}}{2}\right)\omega + R(U_{-1}) &= 0 \\ \frac{U_1 - U_{-1}}{2}\omega + R(U_0) &= 0 \\ -\left(U_0 - \frac{U_1 + U_{-1}}{2}\right)\omega + R(U_1) &= 0 \end{aligned} \tag{7.7}$$

With boundary conditions specified, the above system is self-contained. Compared with the RANS equations, the above equations have an extra term corresponding to the time derivative term of the URANS equations. The URANS equations at the three phases can be solved in a similar way to that for the RANS equations with the extra term being treated as a source term. A pseudo time derivative term is added to the three sets of equations, so that the Runge-Kutta method can be used to time-march their solutions to a steady state. In this way, the flow solver for solving the three sets of equations can be easily extended from a steady RANS flow solver.

7.2 Boundary Conditions

It is normally desirable to use non-reflective boundary conditions at the inlet and exit of a computational domain in an unsteady flow field calculation. However if the inlet

and exit boundaries are quite far away from a blade leading and trailing edge, the reflective boundary condition implementation for solving the RANS equations can also be used without distorting the flow solution undesirably. Both non-reflective boundary conditions and reflective boundary conditions are implemented in the flow solver solving the URANS equations at the three phases. The non-reflective implementation is a local 1D treatment (Giles, 1991). The reflective implementation satisfies the necessary compatibility requirement and is the same as the one implemented in the steady flow solver. For the first validation case to be presented later, non-reflective boundary conditions are used at both an inlet and an exit. However in the design optimization case studies followed, the reflective boundary conditions are used in the present URANS flow solver. This is because the corresponding non-reflective adjoint boundary conditions are not available in the adjoint solver.

Along a viscous solid wall, a slip boundary condition with the log-law is applied in the same way as for the RANS equations.

The periodic boundary condition for the three-phase URANS solver is quite different from the periodic boundary condition for a steady flow solver. For blade flutter predictions, along a pair of periodic boundaries, the flow solution is not exactly the same, but has a phase difference. The phase difference is called inter-blade phase angle ϕ. For example, the flow solution at the lower periodic boundary is expressed as follows:

$$U^L = \bar{U}^L + U_A^L \sin(\omega t) + U_B^L \cos(\omega t) \tag{7.8}$$

with $\bar{U}^L, U_A^L$ and U_B^L calculated from the flow solution at the three phases (Eq.7.5) at the lower periodic boundary. Then the flow solution at the upper periodic boundary can be updated by

$$U^U = \bar{U}^L + U_A^L \sin(\omega t + \phi) + U_B^L \cos(\omega t + \phi) \tag{7.9}$$

Specifically at the three phases, the updating formula are given by

$$\begin{aligned} U_{-1}^U &= \bar{U}^L - U_A^L \cos\phi + U_B^L \sin\phi \\ U_0^U &= \bar{U}^L + U_A^L \sin\phi + U_B^L \cos\phi \\ U_1^U &= \bar{U}^L + U_A^L \cos\phi - U_B^L \sin\phi \end{aligned} \tag{7.10}$$

In the same way, the flow solution at the lower periodic boundary is updated using the flow solution at the upper periodic boundary:

$$U^L = \bar{U}^U + U_A^U \sin(\omega t - \phi) + U_B^U \cos(\omega t - \phi) \tag{7.11}$$

At the three phases, the updating formula are given by

$$\begin{aligned} U_{-1}^L &= \bar{U}^U - U_A^U \cos\phi - U_B^U \sin\phi \\ U_0^L &= \bar{U}^U - U_A^U \sin\phi + U_B^U \cos\phi \\ U_1^L &= \bar{U}^U + U_A^U \cos\phi + U_B^U \sin\phi \end{aligned} \tag{7.12}$$

7.3 Validation of the Unsteady Flow Solver

Two numerical investigations are carried out to assess the validity of the methodology in time-periodic unsteady flow calculations and its appropriate implementation. In the first case, the unsteady flow field around a flat plate is simulated with results compared with semi-analytic ones. The second case examines the unsteady flow field inside a 3D oscillating linear cascade of which experimental data are available for comparison with numerical results.

7.3.1 Validation against a semi-analytic method

As the first step validation, the flow solver is used to calculate the unsteady pressure jump coefficient across a flat plate cascade of zero thickness subject to incoming wakes. The flat plate has a stagger angle of 30 degrees with a pitch/chord ratio of 0.5. The computational domain is an H type mesh covering one blade passage with its inlet and exit planes extended one axial chord upstream of its leading edge and downstream of its trailing edge respectively. Since the Euler flow equation is solved, a uniform mesh with 361 mesh points in the axial direction and 61 mesh points in the pitchwise direction is used. At zero incidence, the time-averaged flow field is uniform at a Mach number of 0.7. The incoming wakes are specified in terms of velocity and density disturbances in

a harmonic form with the assumption of a constant pressure and total temperature:

$$\begin{aligned}
\Delta u &= u - u_0 = u_0 \delta \cos(\omega t + \phi y) \\
\Delta v &= v - v_0 = v_0 \delta \cos(\omega t + \phi y + \pi) \\
\Delta \rho &= \rho - \rho_0 = \frac{\gamma - 1}{\gamma} \frac{\rho_0^2 (u_0^2 - v_0^2)}{p_0} \delta \cos(\omega t + \phi t) \\
\Delta p &= p - p_0 = 0
\end{aligned}$$

where y is the normalized pitch, ϕ is the inter-blade phase angle assigned a value of -400 degrees in this calculation. The tangential velocity disturbance is in anti-phase with the axial velocity disturbance giving a wake orientation of -30 degrees with respect the axial direction. ω is calculated based on a reduce frequency k of 13.96 which is based on the axial chord C_{ax} and the mean axial velocity $\bar{u}$:

$$\omega = \frac{k \cdot \bar{u}}{C_{ax}} \tag{7.13}$$

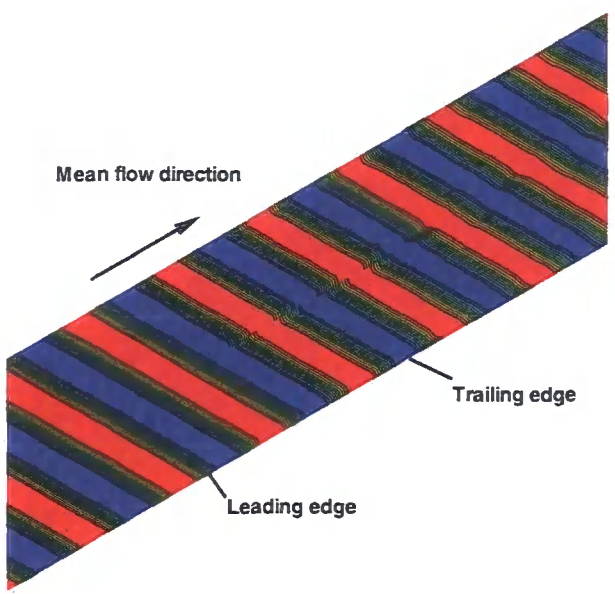


Figure 7.1: Entropy distribution at one instant (reconstructed from a single passage calculation)

Fig.7.1 presents the entropy contours of the unsteady flow field at one instant. The entropy contours show a qualitatively good agreement with those by He (2008) using the same methodology and Chen et al. (2001) using the time-linearized nonlinear harmonic method.

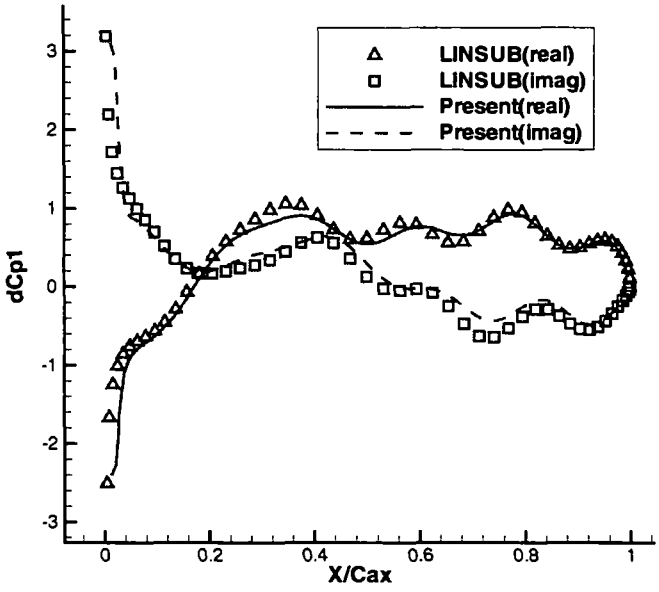


Figure 7.2: Pressure jump coefficients across the flat plate

A quantitative comparison of the calculated flow field by the present flow solver to that by the well-established semi-analytic method (Whitehead, 1987) is depicted in Fig.7.2 with a very good agreement. The pressure jump coefficient across a flat plate, in the linear harmonic method, is usually obtained in a complex variable manner with real and imaginary parts, defined as follows:

$$dCp1 = \frac{\Delta p}{\rho W w \sin\theta} \tag{7.14}$$

where W is the absolute mean flow velocity, w is the amplitude of the incoming wake velocity ($\delta \cdot W$ in this case), θ is the angle between the wake orientation and the flat plate (60 degrees in this case).

7.3.2 Validation against 3D linear cascade experimental data

In this section, the flow field inside a 3D oscillating linear cascade is investigated by the present three-phase flow solver. Detailed experimental data are available for this case to be compared with numerical results.

The experimental investigation was conducted by Yang (Yang and He, 2003; Yang, 2004) at Durham University to enhance the understanding of 3D blade aeroelastic mechanisms and to produce test data of realistic configurations for validation of ad-

vanced 3D aeromechanical methods. Geometric parameters of the test cascade are listed in Table 7.1 with operating conditions in Table 7.2.

Table 7.1: Experimental geometrical parameters of the linear cascade

Airfoil type	controlled diffusion
Chord length, C	0.15 m
Span length	0.19 m
Stagger angle,	14.2 deg
Pitch length	0.09 m
Bending mode direction	normal to the absolute chord
Bending amplitude at tip Am_{tip}	6% chord
Bending amplitude at hub Am_{tip}	0.5% chord

Table 7.2: Experimental operational conditions for the linear cascade

Measured inlet flow angle,	37.5 deg
Isentropic exit velocity,	19.5 m/s
Reynolds number, (based on blade chord and isentropic exit velocity)	1.95e5
Reduced frequency,	0.4
Nominal frequency,	8.28 (Hz)
Ambient conditions	1 atm & 20 deg

The computational domain has a dimension of $0.48 \times 0.09 \times 0.19$ m in the axial, circumferential and radial directions, respectively. The radial coordinates of mesh points on the same radial mesh section are the same. The radial coordinate of the hub mesh section is set to 100.27 m. This value is used to maintain the difference between the pitch distance at the hub and that at the tip is within acceptably small range[1].

A single passage H-type mesh is adopted. The mesh has a density of $133 \times 40 \times 40$ in the axial, circumferential and radial directions, respectively. Fig.7.3 shows the blade to blade view and meridional view of the computational mesh for the linear cascade. In the axial direction, 70 mesh points are distributed along the blade surface on both pressure and suction sides, 29 mesh points are allocated before the blade leading edge and 34 mesh points after the blade trailing edge. The mesh points in the circumferential direction are clustered towards blade surfaces, and mesh points in the radial direction are clustered towards both end walls as Fig.7.3 reveals.

[1]The flow solver solves the flow governing equations in a cylindrical coordinate system. Simulating the flow field in a linear cascade using this flow solver, the radial coordinate at the hub of the linear cascade should be big enough.

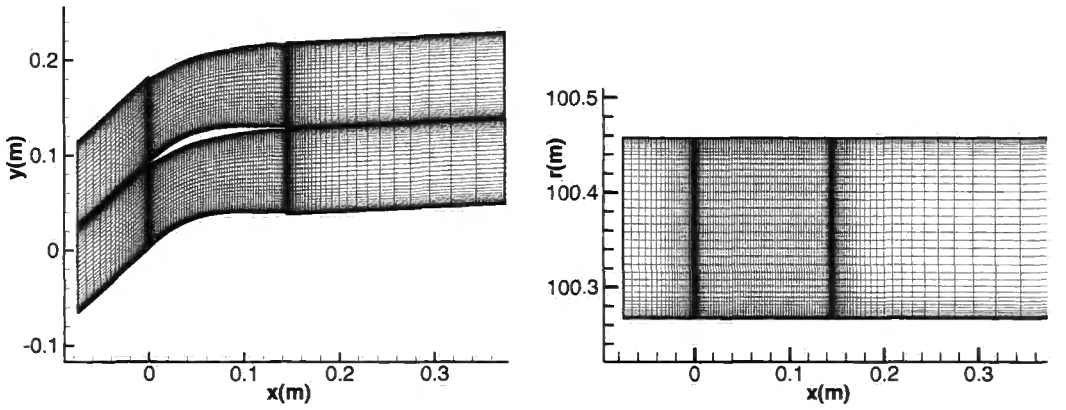


Figure 7.3: Blade to blade view (left) and meridional view (right) of the linear cascade mesh

At the inlet boundary, the total pressure and total temperature and flow angles are specified. The total pressure has a profile in the radial direction and this profile is symmetric about its centre line. The total pressure profile is figured out according to the loss profile measured by Yang (2004). A constant total temperature is used. The absolute swirl angle is set to be 37.5 degrees. The pitch angle is set to zero. At the exit boundary, the static pressure is specified as 1 atm.

The linear cascade operates at a low flow speed with a small Mach number (less than 0.3). The density-based flow solver normally converges extremely slowly when the Mach number in the flow field is approaching zero (the flow field tends to be incompressible). In order to ensure the flow solver has a good convergence when simulating the flow field around the linear cascade, the isentropic exit velocity is increased from the experimental value of 19.5 m/s to 66 m/s in the computation by increasing the total pressure at the inlet. The Reynolds number is kept the same as the experimental value to maintain the similarity of calculated results. As a consequence, the nominal vibration frequency is increased accordingly to give the same reduced frequency of 0.4.

The time-averaged/steady pressure coefficients with its definition given in 7.15 on the blade surface at six different span sections (in terms of percentage of the blade height measured from the blade hub) are presented in Fig.7.4.

$$C_p = \frac{p - p_{exit}}{p_0^* - p_{exit}} \tag{7.15}$$

where p_0^* is the total pressure at the mid span of the inlet plane. p_{exit} is the static pressure at the exit of the computational domain.

Experimental results are compared with computational results by the steady flow solver used for aerodynamic design optimization and computational results by the unsteady flow solver. The computational results by the unsteady flow solver are identical to those by the steady flow solver, which proves the consistency of the unsteady flow solver with the steady flow solver. The calculated results are generally in a very good agreement with measured data. It is shown that on the 50% span and 70% span locations, the experimental data show an abrupt change of pressure coefficient on the blade suction surface. This abrupt change in the pressure coefficient, as explained by Yang and He (2003), is due to a bubble type laminar flow separation which the computation is normally unable to capture in the absence of a turbulence transition model. The blade unloading phenomenon in the blade endwall regions is reasonably captured in the computation. However the computation slightly under predicts the unloading effect compared with the experimental data. The flow unloading can result from the secondary flow due to endwalls and blade tip clearance. The computation is conducted without the inclusion of tip clearance while the experimental configuration has a tip gap of 0.5% chord to accommodate the blade vibration.

The first harmonic pressure coefficients and phase angles are presented in Fig.7.5 and Fig.7.6 respectively for an inter-blade phase angle of 60 deg. The definitions of the first harmonic pressure coefficient and the phase angle are given by

$$\begin{aligned} |C_{p1}| &= \frac{\sqrt{p_A^2 + p_B^2}}{(p_0 - p_{exit})Am_{tip}} \\ \alpha &= \tan^{-1}\frac{p_A}{p_B} \end{aligned} \tag{7.16}$$

with the pressure on the blade surface expressed as

$$p = \bar{p} + p_A \sin(\omega t) + p_B \cos(\omega t)$$

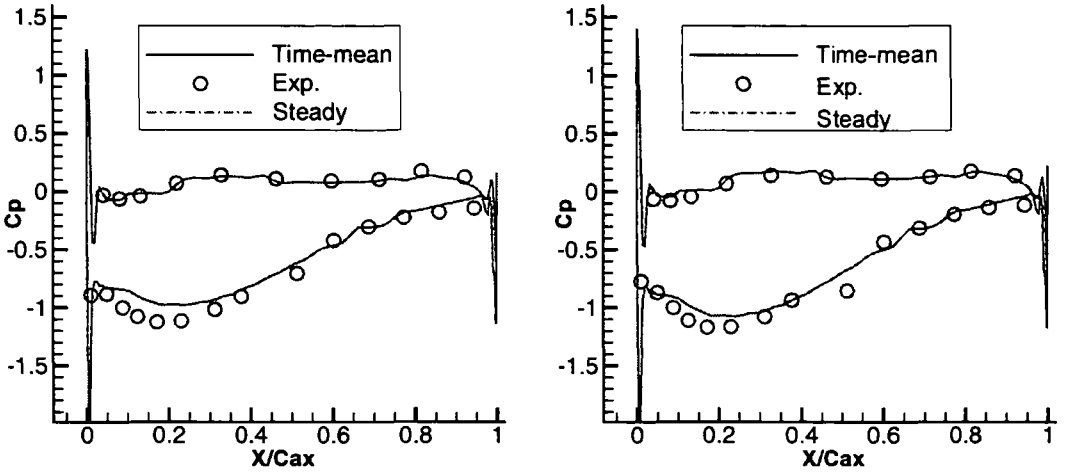
The unsteady flow field induced by the controlled blade oscillation in the experiment is assumed to be largely linear. This is to say, the amplitude of the unsteadiness in the unsteady flow field is proportional to the amplitude of the blade vibration amplitude. In this experiment/calculation, the blade tip has the largest vibration amplitude. The inclusion of the blade tip amplitude in the first harmonic pressure coefficient definition (expression 7.16) eliminates the dependence of the coefficient on the vibration amplitude.

Overall, both the first harmonic pressure coefficients and their phase angles by the computation are in a reasonably good agreement with their experimental counterparts. The flow separation as revealed in the steady flow field is also apparent in the unsteady flow field, especially clearly revealed by the abrupt change of the phase angle of the unsteady pressure field.

Though they are not presented here, a series of computations at this nominal vibration frequency with different inter-blade phase angles were conducted. All those results show favorable comparison with corresponding experimental data. This concludes that the Nonlinear Harmonic Phase Solution method with one harmonic is effective to capture the flow unsteadiness induced by blade vibration and is implemented appropriately.

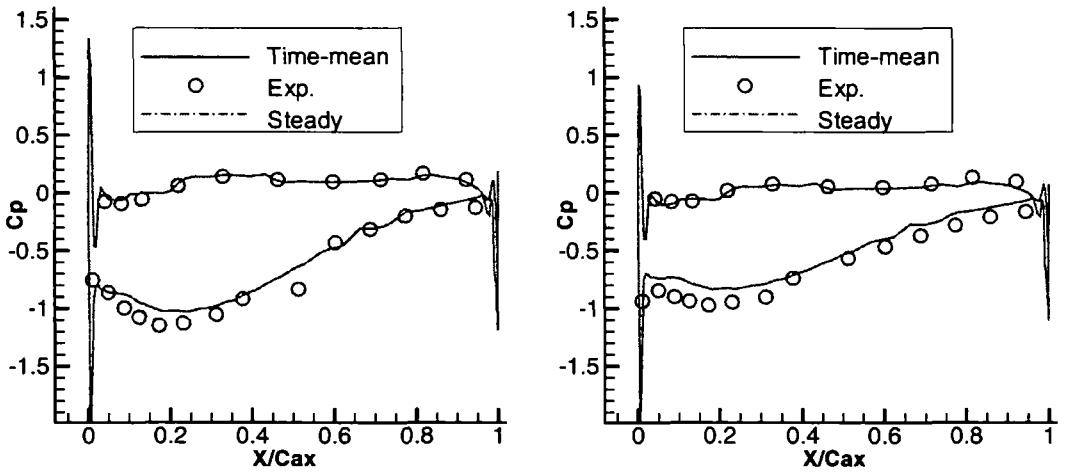
7.4 Summary

An unsteady flow solver is developed based upon the Nonlinear Harmonic Phase Solution method for turbomachinery blade flutter predictions. The flow solver solves the URANS equations at the three phases of a period of unsteadiness: $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$. With a phase-shift periodic boundary condition, one blade passage together is used to reduce the size of the problem (therefore reduction in real time cost). Two validation cases are examined. Firstly, the developed flow solver is validated against a semi-analytic method using the well-known case of a flat-plate subject to incoming unsteady wakes. The pressure jump coefficients by the present flow solver are in a close agreement with those by the semi-analytic method. Secondly, computations are conducted for a 3D low speed linear cascade composed of oscillating blades with results compared with experimental data. Reasonably good comparisons of time-averaged pressure coefficients, first harmonic pressure coefficients and phase angles at different span locations are achieved. It is concluded that the Nonlinear Harmonic Solution Phase Solution method has been properly implemented, and the flow solver is able to simulate time-periodic unsteady flows of single dominant frequency with desirable accuracy.



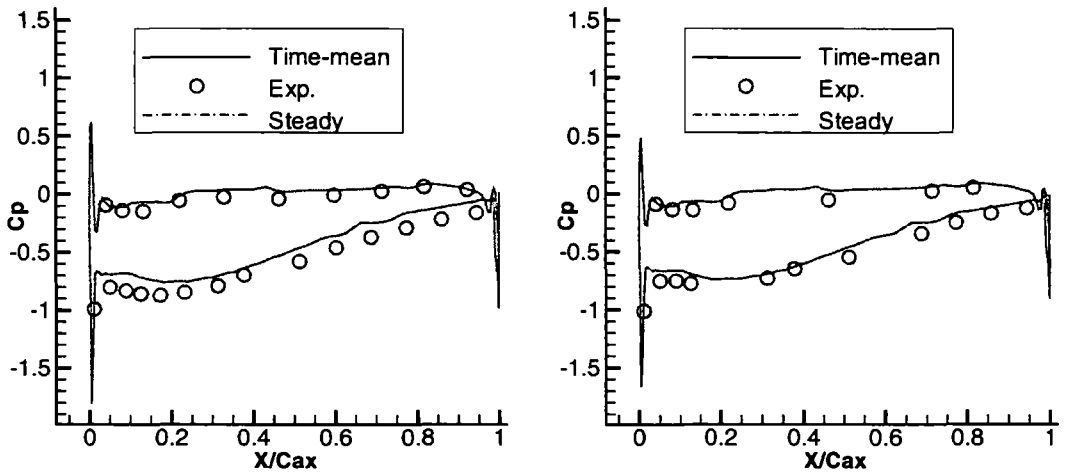
(a) 20% span

(b) 50% span



(c) 70% span

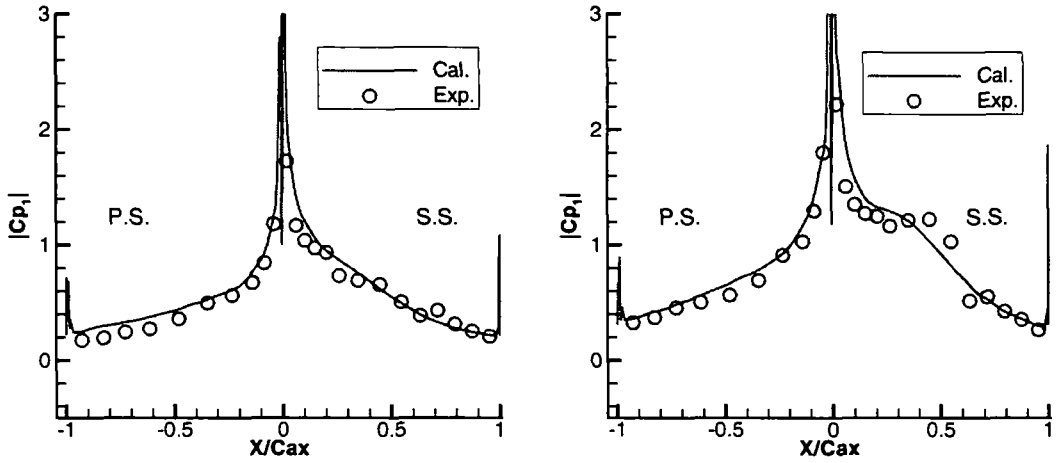
(d) 90% span



(e) 95% span

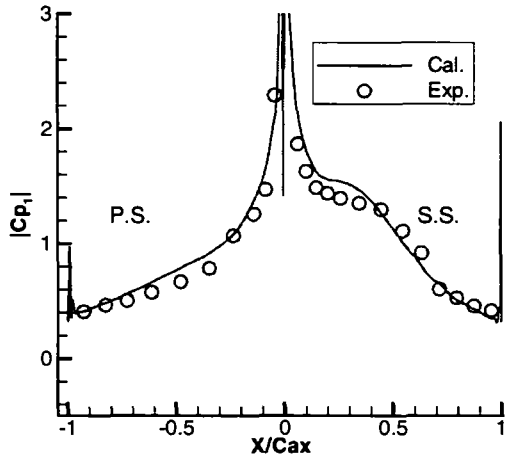
(f) 98% span

Figure 7.4: Comparison of pressure coefficients on the blade surface at six different span locations (Time-mean: Time-mean flow solution constructed from the three-phase flow solution, Exp.: Experimental data, Steady: Steady flow solution)

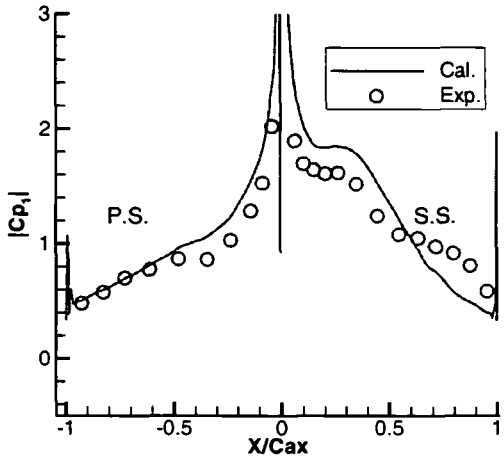


(a) 20% span

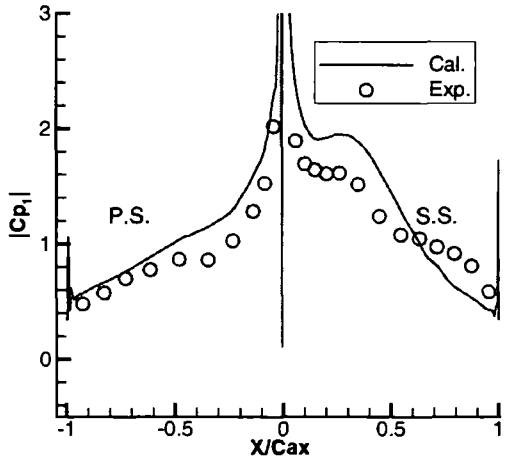
(b) 50% span



(c) 70% span

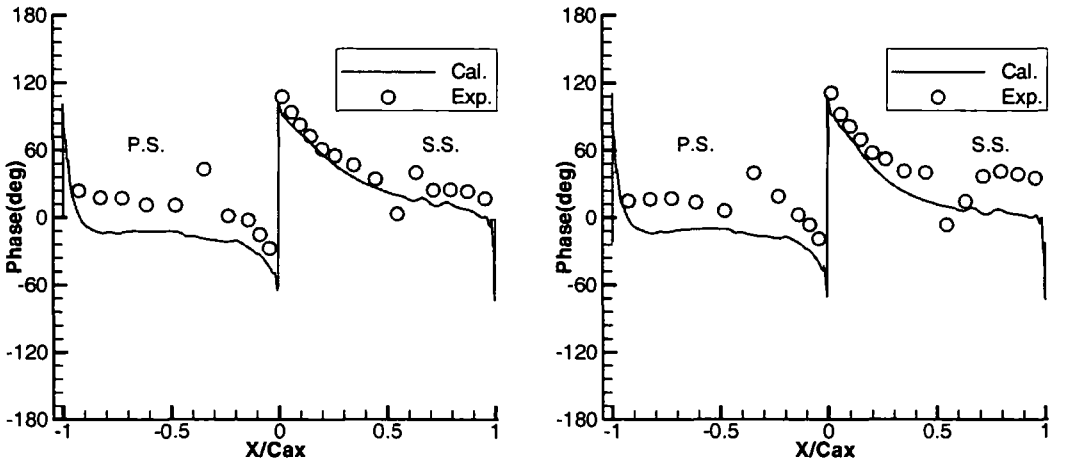


(d) 90% span



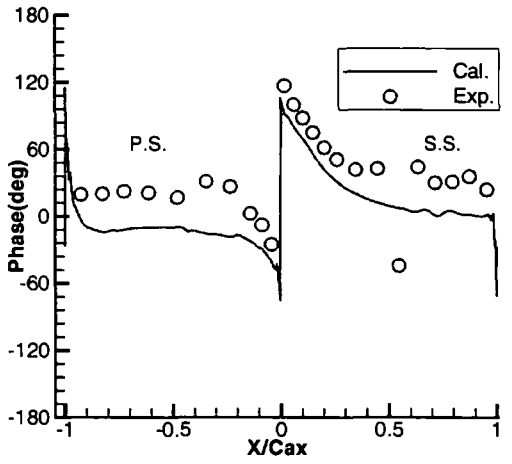
(e) 95% span

Figure 7.5: Comparison of the first harmonic pressure coefficients on the blade surface at five different span locations (P.S.: Pressure Surface, S.S.: Suction Surface)

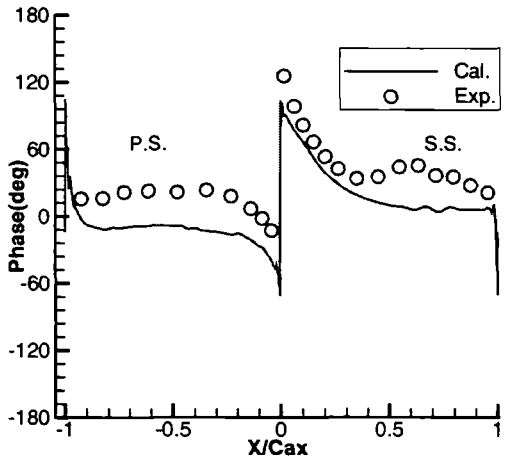


(a) 20% span

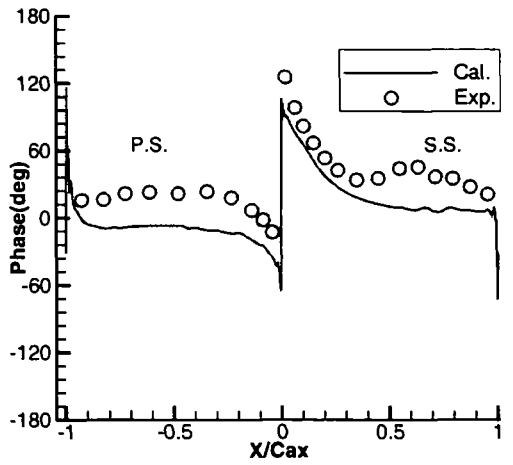
(b) 50% span



(c) 70% span



(d) 90% span



(e) 95% span

Figure 7.6: Comparison of phase angles of the first harmonic pressure on the blade surface at five different span locations (P.S.: Pressure Surface, S.S.: Suction Surface)

Chapter 8

Development of Unsteady Flow Adjoint System

There are two slightly different ways to formulate the adjoint system based upon the URANS equations for aero-damping sensitivity calculations. One can start by linearizing the URANS equations in a partial differential equation form, deriving the corresponding unsteady adjoint system, then formulating it at the three phases using the Nonlinear Harmonic Phase Solution method so that it can be solved efficiently. This is similar to what has been done by Nadarajah and Jameson (2006) and is illustrated in Appendix E. One may also start from the flow equations at the three phases by linearizing the three-phase flow equations first, then deriving the adjoint system. The former approach can be called a full continuous approach. The latter one can be called a semi-continuous approach, because the latter approach starts from the URANS equations which have been discretized in time at three time instants in a period. It is noted that the two approaches result in some differences in the adjoint equations, the periodic boundary condition and the final gradient expression as revealed in Appendix E. In the present work, the semi-continuous approach is adopted, because it is easier to formulate the adjoint system and less complicated to calculate the final gradients by this approach.

8.1 Unsteady Adjoint Systems

This section first attempts to derive an unsteady adjoint system based upon the unsteady flow equations at the three phases in a way as concise as section 4.1 is pre-

sented. Then the corresponding adjoint system in a partial differential equation form is obtained by applying the illustrated adjoint formulation principle.

8.1.1 Adjoint Formulation Principle

The objective function in this context is a function of the flow solution at the three phases and mesh coordinates:

$$I = I\left(U_{-1}, U_0, U_1, X\right) \tag{8.1}$$

Linearizing the objective function with respect to a design variable yields

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial U_{-1}}\tilde{U}_{-1} + \frac{\partial I}{\partial U_0}\tilde{U}_0 + \frac{\partial I}{\partial U_1}\tilde{U}_1 + \frac{\partial I}{\partial \alpha} \tag{8.2}$$

where

$$\begin{aligned} \tilde{U}_i &= \frac{\partial U_i}{\partial X}\frac{\partial X}{\partial \alpha} \quad \text{i = -1, 0, 1} \\ \frac{\partial I}{\partial \alpha} &= \frac{\partial I}{\partial X}\frac{\partial X}{\partial \alpha} \end{aligned}$$

Linearizing the flow equations at the three phases (Eq.7.7) gives the linearized flow equations:

$$\begin{aligned} \left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2}\right)\omega + \frac{\partial R\left(U_{-1}\right)}{\partial U_{-1}}\tilde{U}_{-1} + \frac{\partial R\left(U_{-1}\right)}{\partial \alpha} &= 0 \\ \frac{\tilde{U}_1 - \tilde{U}_{-1}}{2}\omega + \frac{\partial R\left(U_0\right)}{\partial U_0}\tilde{U}_0 + \frac{\partial R\left(U_0\right)}{\partial \alpha} &= 0 \\ -\left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2}\right)\omega + \frac{\partial R\left(U_1\right)}{\partial U_1}\tilde{U}_1 + \frac{\partial R\left(U_1\right)}{\partial \alpha} &= 0 \end{aligned} \tag{8.3}$$

where

$$\frac{\partial R\left(U_i\right)}{\partial \alpha} = \frac{\partial R\left(U_i\right)}{\partial X}\frac{\partial X}{\partial \alpha} \quad i = -1, 0, 1$$

Here it is assumed that ω does not depend on α or $\frac{d\omega}{d\alpha}$ is small and negligible. Compared with Eq.4.7, there are three sets of equations in the above with each set having an extra term. Multiplying the three sets of linearized equations in Eq.8.3 with the adjoint

variables at the three phases λ_{-1}, λ_0 and λ_1 respectively and summing up yields

$$\lambda_{-1}^T \left[\left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2} \right) \omega + \frac{\partial R(U_{-1})}{\partial U_{-1}} \tilde{U}_{-1} + \frac{\partial R(U_{-1})}{\partial \alpha} \right] + \lambda_0^T \left[\left(\frac{\tilde{U}_1 - \tilde{U}_{-1}}{2} \right) \omega + \frac{\partial R(U_0)}{\partial U_0} \tilde{U}_0 + \frac{\partial R(U_0)}{\partial \alpha} \right] + \lambda_1^T \left[- \left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2} \right) \omega + \frac{\partial R(U_1)}{\partial U_1} \tilde{U}_1 + \frac{\partial R(U_1)}{\partial \alpha} \right] = 0 \tag{8.4}$$

The transpose of the adjoint variables is used, because both the adjoint variables and the linearized equations are column vectors. Rearranging the above yields

$$\lambda_{-1}^T \left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2} \right) \omega + \lambda_0^T \left(\frac{\tilde{U}_1 - \tilde{U}_{-1}}{2} \right) \omega - \lambda_1^T \left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2} \right) \omega + \lambda_{-1}^T \left[\frac{\partial R(U_{-1})}{\partial U_{-1}} \tilde{U}_{-1} + \frac{\partial R(U_{-1})}{\partial \alpha} \right] + \lambda_0^T \left[\frac{\partial R(U_0)}{\partial U_0} \tilde{U}_0 + \frac{\partial R(U_0)}{\partial \alpha} \right] + \lambda_1^T \left[\frac{\partial R(U_1)}{\partial U_1} \tilde{U}_1 + \frac{\partial R(U_1)}{\partial \alpha} \right] = 0 \tag{8.5}$$

Subtracting the above from the gradient expression in Eq.8.2 gives

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial U_{-1}} \tilde{U}_{-1} + \frac{\partial I}{\partial U_0} \tilde{U}_0 + \frac{\partial I}{\partial U_1} \tilde{U}_1 + \frac{\partial I}{\partial \alpha} - \lambda_{-1}^T \left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2} \right) \omega - \lambda_0^T \left(\frac{\tilde{U}_1 - \tilde{U}_{-1}}{2} \right) \omega + \lambda_1^T \left(\tilde{U}_0 - \frac{\tilde{U}_1 + \tilde{U}_{-1}}{2} \right) \omega - \lambda_{-1}^T \left[\frac{\partial R(U_{-1})}{\partial U_{-1}} \tilde{U}_{-1} + \frac{\partial R(U_{-1})}{\partial \alpha} \right] - \lambda_0^T \left[\frac{\partial R(U_0)}{\partial U_0} \tilde{U}_0 + \frac{\partial R(U_0)}{\partial \alpha} \right] - \lambda_1^T \left[\frac{\partial R(U_1)}{\partial U_1} \tilde{U}_1 + \frac{\partial R(U_1)}{\partial \alpha} \right] \tag{8.6}$$

Collecting like terms with $\tilde{U_{-1}}$, $\tilde{U_0}$ and $\tilde{U_1}$, one has

$$\frac{dI}{d\alpha} = \left[\frac{-\lambda_1^T + \lambda_{-1}^T + \lambda_0^T}{2} \omega + \frac{\partial I}{\partial U_{-1}} - \lambda_{-1}^T \frac{\partial R(U_{-1})}{\partial U_{-1}} \right] \tilde{U}_{-1} + \left[\left(-\lambda_{-1}^T + \lambda_1^T \right) \omega + \frac{\partial I}{\partial U_0} - \lambda_0^T \frac{\partial R(U_0)}{\partial U_0} \right] \tilde{U}_0 + \left[\frac{-\lambda_1^T + \lambda_{-1}^T - \lambda_0^T}{2} \omega + \frac{\partial I}{\partial U_1} - \lambda_1^T \frac{\partial R(U_1)}{\partial U_1} \right] \tilde{U}_1 + \frac{\partial I}{\partial \alpha} - \lambda_{-1}^T \frac{\partial R(U_{-1})}{\partial \alpha} - \lambda_0^T \frac{\partial R(U_0)}{\partial \alpha} - \lambda_1^T \frac{\partial R(U_1)}{\partial \alpha} \tag{8.7}$$

U_{-1}, U_0 and U_1 are independent flow variables at three different phases of a period of unsteadiness. In order to eliminate the dependence of the gradient $\frac{dI}{d\alpha}$ on the flow variable sensitivities $(\tilde{U}_i)$, the terms multiplying $\tilde{U}_{-1}, \tilde{U}_0$ and $\tilde{U}_1$ need to vanish, resulting

in the following three sets of adjoint equations:

$$\begin{aligned} \frac{-\lambda_1^T + \lambda_{-1}^T + \lambda_0^T}{2}\omega + \frac{\partial I}{\partial U_{-1}} - \lambda_{-1}^T \frac{\partial R(U_{-1})}{\partial U_{-1}} &= 0 \\ \left(-\lambda_{-1}^T + \lambda_1^T\right)\omega + \frac{\partial I}{\partial U_0} - \lambda_0^T \frac{\partial R(U_0)}{\partial U_0} &= 0 \\ \frac{-\lambda_1^T + \lambda_{-1}^T - \lambda_0^T}{2}\omega + \frac{\partial I}{\partial U_1} - \lambda_1^T \frac{\partial R(U_1)}{\partial U_1} &= 0 \end{aligned} \tag{8.8}$$

It is noted that the terms multiplying with ω in the above equations have a very different pattern from those in the linearized equations (Eq.8.3) or in the nonlinear equations (Eq.7.7).

The final gradient expression is independent of flow variable sensitivities as intended:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} - \lambda_{-1}^T \frac{\partial R(U_{-1})}{\partial \alpha} - \lambda_0^T \frac{\partial R(U_0)}{\partial \alpha} - \lambda_1^T \frac{\partial R(U_1)}{\partial \alpha} \tag{8.9}$$

8.1.2 Unsteady Adjoint Equations

The adjoint formulation principle illustrated in the above is applied in this section to the derivation of the adjoint system based upon the three-phase Euler equations in a partial differential equation form. The resultant adjoint system is composed of three sets of steady-like equations.

An objective function for an aeromechanic related design optimization is a function of the three-phase flow solution and mesh:

$$I = \int_{\partial D} M(U_{-1}, U_0, U_1, X)\, ds \tag{8.10}$$

The objective function sensitivity to a design variable α is given by

$$\frac{dI}{d\alpha} = \int_{\partial D} \left(\frac{\partial M}{\partial U_i}\tilde{U}_i + \frac{\partial M}{\partial \alpha}\right) ds + \int_{\partial D} M\tilde{ds} \tag{8.11}$$

The Einstein summation convention is used here for a more compact illustration in the following.

The linearized three-phase Euler equations are written as

$$\begin{aligned} &\left(\tilde{U}_0 - \tfrac{\tilde{U}_1+\tilde{U}_{-1}}{2}\right)\omega + \left[\tfrac{\partial(A-u_g I)\tilde{U}}{\partial x} + \tfrac{\partial(B-v_g I)\tilde{U}}{r\partial\theta} + \tfrac{\partial r(C-w_g I)\tilde{U}}{r\partial r} - D\tilde{U}\right]_{-1} = f_{-1} \\ &\tfrac{\tilde{U}_1-\tilde{U}_{-1}}{2}\omega + \left[\tfrac{\partial(A-u_g I)\tilde{U}}{\partial x} + \tfrac{\partial(B-v_g I)\tilde{U}}{r\partial\theta} + \tfrac{\partial r(C-w_g I)\tilde{U}}{r\partial r} - D\tilde{U}\right]_0 = f_0 \\ &-\left(\tilde{U}_0 - \tfrac{\tilde{U}_1+\tilde{U}_{-1}}{2}\right)\omega + \left[\tfrac{\partial(A-u_g I)\tilde{U}}{\partial x} + \tfrac{\partial(B-v_g I)\tilde{U}}{r\partial\theta} + \tfrac{\partial r(C-w_g I)\tilde{U}}{r\partial r} - D\tilde{U}\right]_1 = f_1 \end{aligned} \tag{8.12}$$

Multiplying the above three equations with their corresponding adjoint variables λ_{-1}, λ_0 and λ_1 and summing up, one has

$$\lambda_{-1}^T\left(\tilde{U}_0-\frac{\tilde{U}_1+\tilde{U}_{-1}}{2}\right)\omega+\lambda_0^T\left(\frac{\tilde{U}_1-\tilde{U}_{-1}}{2}\right)\omega-\lambda_1^T\left(\tilde{U}_0-\frac{\tilde{U}_1+\tilde{U}_{-1}}{2}\right)\omega+ \quad (8.13)$$
$$\lambda_i^T\left[\frac{\partial\left(A-u_gI\right)\tilde{U}}{\partial x}+\frac{\partial\left(B-v_gI\right)\tilde{U}}{r\partial\theta}+\frac{\partial r\left(C-w_gI\right)\tilde{U}}{r\partial r}-D\tilde{U}-f\right]_i=0$$

Integrating the above equation over a computational domain, performing integration by parts for the spatial partial differential terms and rearranging gives

$$\int_D\left[\frac{-\lambda_{-1}^T-\lambda_0^T+\lambda_1^T}{2}\omega\tilde{U}_{-1}+\left(\lambda_{-1}^T-\lambda_1^T\right)\omega\tilde{U}_0+\frac{-\lambda_{-1}^T+\lambda_0^T+\lambda_1^T}{2}\omega\tilde{U}_1\right]dv \quad (8.14)$$
$$+\int_{\partial D}\lambda_i^T\left[\left(A-u_gI\right)\tilde{U}n_x+\left(B-v_gI\right)\tilde{U}n_\theta+\left(C-w_gI\right)\tilde{U}n_r\right]_i ds$$
$$-\int_D\left[\frac{\partial\lambda^T}{\partial x}\left(A-u_gI\right)+\frac{\partial\lambda^T}{r\partial\theta}\left(B-v_gI\right)+\frac{\partial\lambda^T}{\partial r}\left(C-w_gI\right)+\lambda^TD\right]_i\tilde{U}_idv-\int_D\lambda_i^Tf_idv=0$$

Subtracting the right hand side of the above equation from the objective function gradient (Eq.8.11) yields the augmented gradient expression:

$$\frac{dI}{d\alpha}=\int_{\partial D}\frac{\partial M}{\partial\alpha}ds+\int_{\partial D}M\tilde{ds}+\int_D\lambda_i^Tf_idv \quad (8.15)$$
$$-\int_D\left[\frac{-\lambda_{-1}^T-\lambda_0^T+\lambda_1^T}{2}\omega\tilde{U}_{-1}+\left(\lambda_{-1}^T-\lambda_1^T\right)\omega\tilde{U}_0+\frac{-\lambda_{-1}^T+\lambda_0^T+\lambda_1^T}{2}\omega\tilde{U}_1\right]dv$$
$$-\int_{\partial D}\left[\lambda^T\left(A-u_gI\right)n_x+\lambda^T\left(B-v_gI\right)n_\theta+\lambda^T\left(C-w_gI\right)n_r-\frac{\partial M}{\partial U}\right]_i\tilde{U}_ids$$
$$+\int_D\left[\frac{\partial\lambda^T}{\partial x}\left(A-u_gI\right)+\frac{\partial\lambda^T}{r\partial\theta}\left(B-v_gI\right)+\frac{\partial\lambda^T}{\partial r}\left(C-w_gI\right)+\lambda^TD\right]_i\tilde{U}_idv$$

The objective of the adjoint formulation requires the final gradient expression be free from flow variable sensitivities. This can be achieved by vanishing the domain integrals and boundary integrals related to flow variable sensitivities separately:

$$-\int_D\left[\frac{-\lambda_{-1}^T-\lambda_0^T+\lambda_1^T}{2}\omega\tilde{U}_{-1}+\left(\lambda_{-1}^T-\lambda_1^T\right)\omega\tilde{U}_0+\frac{-\lambda_{-1}^T+\lambda_0^T+\lambda_1^T}{2}\omega\tilde{U}_1\right]dv \quad (8.16)$$
$$+\int_D\left[\frac{\partial\lambda^T}{\partial x}\left(A-u_gI\right)+\frac{\partial\lambda^T}{r\partial\theta}\left(B-v_gI\right)+\frac{\partial\lambda^T}{\partial r}\left(C-w_gI\right)+\lambda^TD\right]_i\tilde{U}_idv=0$$

$$\int_{\partial D}\left[\lambda^T\left(A-u_gI\right)n_x+\lambda^T\left(B-v_gI\right)n_\theta+\lambda^T\left(C-w_gI\right)n_r-\frac{\partial M}{\partial U}\right]_i\tilde{U}_ids \quad (8.17)$$

The flow variables at the three phases (U_{-1}, U_0 and U_1) are independent of each other, and the flow variable sensitivities can not always be zero in the interior domain. The

sufficient and necessary condition to satisfy Eq.8.16 is to vanish the terms multiplying with $\tilde{U}_{-1}$, $\tilde{U}_0$ and $\tilde{U}_1$ respectively, leading to the three sets of steady-like adjoint equations:

$$\begin{aligned}
&\frac{\lambda_{-1}^T+\lambda_0^T-\lambda_1^T}{2}\omega+\left[\frac{\partial\lambda^T}{\partial x}(A-u_gI)+\frac{\partial\lambda^T}{r\partial\theta}(B-v_gI)+\frac{\partial\lambda^T}{\partial r}(C-w_gI)+\lambda^TD\right]_{-1}=0\\
&\left(-\lambda_{-1}^T+\lambda_1^T\right)\omega+\left[\frac{\partial\lambda^T}{\partial x}(A-u_gI)+\frac{\partial\lambda^T}{r\partial\theta}(B-v_gI)+\frac{\partial\lambda^T}{\partial r}(C-w_gI)+\lambda^TD\right]_{0}=0\\
&\frac{\lambda_{-1}^T-\lambda_0^T-\lambda_1^T}{2}\omega+\left[\frac{\partial\lambda^T}{\partial x}(A-u_gI)+\frac{\partial\lambda^T}{r\partial\theta}(B-v_gI)+\frac{\partial\lambda^T}{\partial r}(C-w_gI)+\lambda^TD\right]_{1}=0
\end{aligned} \tag{8.18}$$

The sufficient and necessary condition to satisfy Eq.8.17 gives the boundary condition for the adjoint equations to be presented later. The derivation of the boundary conditions needs the flow boundary conditions to be considered.

With the adjoint equation and boundary conditions satisfied, the original gradient expression (Eq.8.11) has changed into

$$\frac{dI}{d\alpha}=\int_{\partial D}\frac{\partial M}{\partial\alpha}ds+\int_{\partial D}M\tilde{ds}+\int_D\lambda_i^Tf_idv \tag{8.19}$$

8.1.3 Boundary Conditions

There are four types of boundary conditions for the solution of the derived adjoint system in the proceeding section: the subsonic inlet boundary condition, the subsonic exit boundary condition, the viscous solid wall boundary condition and the phase-shift periodic boundary condition. The adjoint inlet and exit boundary conditions are derived based upon the reflective flow inlet and exit boundary conditions. They are therefore reflective, but satisfy the compatibility requirements: the number of incoming adjoint characteristics to the computational domain is equal to the number of boundary conditions to be specified. At each temporal phase, the inlet, exit and solid wall boundary conditions assume a very similar form to those for a steady adjoint system. While the phase-shift periodic boundary condition is quite different from the one for the unsteady flow equations.

• Inlet Boundary Condition

Similar to the inlet boundary condition for the steady adjoint system, there is one equation corresponding to one incoming adjoint characteristic for each of the three sets of adjoint equations (Eqs.8.18). Furthermore at each phase, the boundary condition assumes the same form as that for the steady adjoint equation:

$$\lambda_i^T \frac{\partial F_i}{\partial p_i} \cdot n_x - \frac{\partial M}{\partial p_i} = 0 \quad i = -1, 0, 1 \tag{8.20}$$

It should be pointed out that $\frac{\partial M}{\partial p_i}$ does not normally have the same form at the three phases, because the flow solution at the three phases does not contribute to the objective function equally.

• Exit Boundary Condition

At an exit, the boundary condition at each phase is also similar to that for a steady adjoint system:

$$\lambda_i^T \frac{\partial F}{\partial q_i} \cdot n_x - \frac{\partial M}{\partial q_i} = 0 \quad q_i = \rho_i, u_i, v_i, w_i \text{ with } i = -1, 0, 1 \tag{8.21}$$

$\frac{\partial M}{\partial q_i}$ does not have the same form at the three phases due to a different contribution of the flow solution at the three phases to the objective function.

• Solid Wall Boundary Condition

Along a solid wall boundary, at each phase, the following boundary condition needs to be applied:

$$\lambda_2 n_x + r\lambda_3 n_\theta + \lambda_4 n_r + \lambda_5 (u_g n_x + v_g n_\theta + w_g n_r) = \frac{\partial M}{\partial p} \tag{8.22}$$

The following extra conditions are for a viscous solid wall with the wall shear stress calculated by the log-law:

$$\begin{aligned}
\sigma_x &= \frac{\tau_w}{W}(\lambda_2 + u_g \lambda_5) \\
\sigma_\theta &= \frac{\tau_w}{W}(\lambda_3 + \frac{v_g}{r} \lambda_5) \\
\sigma_r &= \frac{\tau_w}{W}(\lambda_4 + w_g \lambda_5) \\
\frac{\partial \lambda_5}{\partial n} &= 0 \quad \text{for an adiabatic wall} \\
\lambda_5 &= 0 \quad \text{for a non-adiabatic wall}
\end{aligned} \tag{8.23}$$

• Periodic Boundary Condition

As revealed in the derivation of the periodic boundary condition for a steady adjoint equation in section 4.3, along a pair of periodic boundaries there is a pair of the boundary integrals (left hand side of Eq.8.17 with $M = 0$) for each temporal phase. Here we can not vanish each pair at each temporal phase separately, but we need to vanish the three pairs for the three temporal phases all together.

Refer to Fig.3.1, Eq.8.17 is rearranged as follows:

$$\begin{aligned} &\int_{ab} \left[\lambda^T (A - u_g I) n_x + \lambda^T (B - v_g I) n_\theta + \lambda^T (C - w_g I) n_r\right]_i \tilde{U}_i ds \\ &+ \int_{cd} \left[\lambda^T (A - u_g I) n_x + \lambda^T (B - v_g I) n_\theta + \lambda^T (C - w_g I) n_r\right]_i \tilde{U}_i ds \;=\; 0 \end{aligned} \tag{8.24}$$

in which $[A \cdot n_x + (B - v_g I) \cdot n_\theta + C \cdot n_r]\tilde{U}$ is a flux perturbation, which can be expressed using one symbol F for a more compact derivation in the following. Transposing the above equation (Eq.8.24 is a scalar. Transposing it does not change the equation.) and replacing the flux perturbation with F, one has

$$\int_{ab} F_i^T \lambda_i ds + \int_{cd} F_i^T \lambda_i ds = 0 \tag{8.25}$$

ab and cd are a pair of periodic boundaries as sketched in Fig.3.1. Expanding the above equation at the three phases yields

$$\begin{aligned} &\int_{ab} \left(F^T \lambda\right)_{-1} ds + \int_{cd} \left(F^T \lambda\right)_{-1} ds + \\ &\int_{ab} \left(F^T \lambda\right)_0 ds + \int_{cd} \left(F^T \lambda\right)_0 ds + \\ &\int_{ab} \left(F^T \lambda\right)_1 ds + \int_{cd} \left(F^T \lambda\right)_1 ds = 0 \end{aligned} \tag{8.26}$$

At a pair of phase-shift periodic boundaries, the flow solution has a constant phase difference. By neglecting higher order harmonic components in the flux calculation, the flux along a pair of periodic boundaries should also have a constant phase difference. Hence the flux perturbation should also have a constant phase difference along a pair of periodic boundaries. At the lower periodic boundary, ab in Fig.3.1, the flux perturbation F is given by

$$F^{ab} = \bar{F} + F_A sin(\omega t) + F_B cos(\omega t)$$

Then the flux perturbation at the upper periodic boundary, dc in Fig.3.1, can be given by

$$F^{dc} = \bar{F} + F_A sin(\omega t + \phi) + F_B cos(\omega t + \phi)$$

Substituting the above flux perturbations into Eq.8.26 and rearranging yields

$$\begin{aligned} & \int_{ab} \left[\left(\bar{F} - F_A\right)^T \lambda_{-1}^L + \left(\bar{F} + F_B\right)^T \lambda_0^L + \left(\bar{F} + F_A\right)^T \lambda_1^L \right] ds \\ - & \int_{ab} \left[\left(\bar{F} - F_A \cos\phi + F_B \sin\phi\right)^T \lambda_{-1}^U + \left(\bar{F} + F_A \sin\phi + F_B \cos\phi\right)^T \lambda_0^U \right. \\ & \left. + \left(\bar{F} + F_A \cos\phi - F_B \sin\phi\right)^T \lambda_1^U \right] ds = 0 \end{aligned} \tag{8.27}$$

Collecting like terms multiplying with $\bar{F}$, F_A and F_B and rearranging gives

$$\begin{aligned} & \int_{ab} \left\{ \left[\left(\lambda_{-1}^L + \lambda_0^L + \lambda_1^L\right) - \left(\lambda_{-1}^U + \lambda_0^U + \lambda_1^U\right) \right] \bar{F} \right. \\ + & \left[\left(-\lambda_{-1}^L + \lambda_1^L\right) - \left(-\lambda_{-1}^U \cos\phi + \lambda_0^U \sin\phi + \lambda_1^U \cos\phi\right) \right] F_A \\ + & \left. \left[\lambda_0^L - \left(\lambda_{-1}^U \sin\phi + \lambda_0^U \cos\phi - \lambda_1^U \sin\phi\right) \right] F_B \right\} ds = 0 \end{aligned} \tag{8.28}$$

Because $\bar{F}$, F_A and F_B are independent of each other, the sufficient and necessary condition to satisfy the above equation is to vanish the terms multiplying with $\bar{F}$, F_A and F_B respectively, leading to the following three equations:

$$\begin{aligned} \lambda_{-1}^L + \lambda_0^L + \lambda_1^L &= \lambda_{-1}^U + \lambda_0^U + \lambda_1^U \\ -\lambda_{-1}^L + \lambda_1^L &= -\lambda_{-1}^U \cos\phi + \lambda_0^U \sin\phi + \lambda_1^U \cos\phi \\ \lambda_0^L &= \lambda_{-1}^U \sin\phi + \lambda_0^U \cos\phi - \lambda_1^U \sin\phi \end{aligned} \tag{8.29}$$

Taking λ_i^L as unknowns and solving the equations yields the update formula for λ_i^L at a lower periodic boundary:

$$\begin{aligned} \lambda_{-1}^L &= \frac{1}{2} \times \left[\lambda_{-1}^U (1 + cos\phi - sin\phi) + \lambda_0^U (1 - sin\phi - cos\phi) + \lambda_1^U (1 - cos\phi + sin\phi) \right] \\ \lambda_0^L &= \lambda_{-1}^U sin\phi + \lambda_0^U cos\phi - \lambda_1^U sin\phi \\ \lambda_1^L &= \frac{1}{2} \times \left[\lambda_{-1}^U (1 - cos\phi - sin\phi) + \lambda_0^U (1 + sin\phi - cos\phi) + \lambda_1^U (1 + cos\phi + sin\phi) \right] \end{aligned} \tag{8.30}$$

Taking λ_i^U as unknowns and solving the equations one can get the formula for λ_i^U at an upper periodic boundary:

$$\begin{aligned} \lambda_{-1}^U &= \frac{1}{2} \times \left[\lambda_{-1}^L (1 + cos\phi + sin\phi) + \lambda_0^L (1 + sin\phi - cos\phi) + \lambda_1^L (1 - cos\phi - sin\phi) \right] \\ \lambda_0^U &= -\lambda_{-1}^L sin\phi + \lambda_0^L cos\phi + \lambda_1^L sin\phi \\ \lambda_1^U &= \frac{1}{2} \times \left[\lambda_{-1}^L (1 - cos\phi + sin\phi) + \lambda_0^L (1 - sin\phi - cos\phi) + \lambda_1^L (1 + cos\phi - sin\phi) \right] \end{aligned} \tag{8.31}$$

8.2 Worksum and its Gradient

Worksum

In this work, blade flutter is quantified by worksum, which is the total energy exchange between a blade and its working flow in a blade vibration period. A positive worksum indicates energy is transferred from the working fluid into a blade, implying negative aero-damping and aeroelasticity instability. If grid/mesh vibration velocity components v_x, v_θ and v_r are given by

$$\begin{aligned} v_x &= v_x^A \sin(\omega t) + v_x^B \cos(\omega t) \\ v_\theta &= v_\theta^A \sin(\omega t) + v_\theta^B \cos(\omega t) \\ v_r &= v_r^A \sin(\omega t) + v_r^B \cos(\omega t) \end{aligned}$$

and the pressure field is given by

$$p = \bar{p} + p_A \sin(\omega t) + p_B \cos(\omega t)$$

Then the worksum is calculated by integrating the unsteady perturbation pressure weighted by mesh vibration velocities on a blade surface in one period:

$$W = \int_0^T \int_{bs} (p - \bar{p}) \left(v_x \cdot n_x + v_\theta \cdot n_\theta + v_r \cdot n_r\right) ds dt$$

where $_{bs}$ denotes a blade surface, $T = \frac{1}{f} = \frac{2\pi}{\omega}$ is the blade vibration period, n_x, n_θ and n_r are the outward normals of a blade surface. Substituting the expressions of v_x, v_θ, v_r and p into the worksum expression and integrating it in time yields

$$W = \int_{bs} \frac{1}{2f} \left[p_A \left(v_x^A \cdot n_x + v_\theta^A \cdot n_\theta + v_r^A \cdot n_r\right) + p_B \left(v_x^B \cdot n_x + v_\theta^B \cdot n_\theta + v_r^B \cdot n_r\right)\right] ds \tag{8.32}$$

Replacing p_A and p_B using $\frac{p_1 - p_{-1}}{2}$ and $p_0 - \frac{p_1 + p_{-1}}{2}$ respectively, one gets

$$\begin{aligned} W = \int_{bs} \frac{1}{2f} \Bigg[& \frac{p_1 - p_{-1}}{2} \left(v_x^A \cdot n_x + v_\theta^A \cdot n_\theta + v_r^A \cdot n_r\right) \\ & + \left(p_0 - \frac{p_1 + p_{-1}}{2}\right) \left(v_x^B \cdot n_x + v_\theta^B \cdot n_\theta + v_r^B \cdot n_r\right) \Bigg] ds \end{aligned} \tag{8.33}$$

Worksum Gradient

According to the worksum definition in Eq.8.32 or Eq.8.33, worksum is a function of the pressure, the blade vibration frequency and grid/mesh vibration velocities (vibration

mode shape). The gradient of worksum to a design variable is given by

$$\begin{aligned}\frac{dW}{d\alpha} &= \int_{bs} \frac{1}{2f}\left[\frac{\tilde{p}_1-\tilde{p}_{-1}}{2}\left(v_x^A\cdot n_x+v_\theta^A\cdot n_\theta+v_r^A\cdot n_r\right)\right. \\ &\quad \left.+\left(\tilde{p}_0-\frac{\tilde{p}_1+\tilde{p}_{-1}}{2}\right)\left(v_x^B\cdot n_x+v_\theta^B\cdot n_\theta+v_r^B\cdot n_r\right)\right] ds \\ &\quad +\frac{\partial W}{\partial \alpha}\end{aligned} \tag{8.34}$$

where $\frac{\partial W}{\partial \alpha}$ accounts for the sensitivity due to blade geometry changes (change of n_x, n_θ, and n_r), the mode shape change (change of v_x, v_θ and v_r) and the vibration frequency change (change of f) when the pressure terms are fixed. It should be noted that, in a practical design optimization, the change of the blade vibration mode shape and frequency due to a change in a design variable can not be obtained by the present flow solver and adjoint solver and is neglected. Rearranging the worksum gradient (Eq.8.34) gives

$$\frac{dW}{d\alpha} = \int_{bs} \frac{1}{2f}\left[-\frac{V_A+V_B}{2}\tilde{p}_{-1}+V_B\tilde{p}_0+\frac{V_A-V_B}{2}\tilde{p}_1\right] ds+\frac{\partial W}{\partial \alpha} \tag{8.35}$$

where $V_A = v_x^A\cdot n_x+v_\theta^A\cdot n_\theta+v_r^A\cdot n_r$ and $V_B = v_x^B\cdot n_x+v_\theta^B\cdot n_\theta+v_r^B\cdot n_r$. The worksum is calculated on a blade surface which corresponds to a solid wall boundary, where the term $\frac{\partial M}{\partial p}$ needs to be specified. With the worksum being part of an objective function to be optimised, $\frac{\partial M}{\partial p}$ is not zero. $\frac{\partial M}{\partial p}$ at the three phases is given as follows:

$$\frac{\partial M}{\partial p_{-1}} = -\frac{1}{2f}\frac{V_A+V_B}{2} \tag{8.36}$$

$$\frac{\partial M}{\partial p_0} = \frac{1}{2f}V_B \tag{8.37}$$

$$\frac{\partial M}{\partial p_1} = \frac{1}{2f}\frac{V_A-V_B}{2} \tag{8.38}$$

8.3 Verifying the Unsteady Adjoint Solver

A 2D blade and a 3D blade are used to investigate the gradient of worksum to its vibration amplitude. The 3D blade is the 3D linear cascade used in the validation of the unsteady flow solver in the preceding chapter. The 2D blade is one radial section of the 3D linear cascade. The worksum of a blade vibration has a quadratic relation to its vibration amplitude as detailed in the following section, providing an analytic gradient result against which the worksum gradient by the developed adjoint solver can be verified.

Worksum Gradient to Vibration Amplitude

From the definition of the worksum in section 8.2, we know that worksum is actually an integration of the product of the unsteady pressure on the blade surface and mesh points movement at the direction at which the pressure is exerted. The velocity of mesh points can be linearly scaled by the amplitude of a blade vibration. The amplitude of the unsteadiness in the flow field is also proportional to the amplitude of a blade vibration. Therefore the worksum has a quadratical relation to the amplitude of a blade vibration under other given conditions. This quadratical relation can be mathematically given by

$$W = H \cdot A^2 \tag{8.39}$$

where A is the vibration amplitude of a blade vibration, H is independent of the blade vibration amplitude and determined by other conditions (boundary conditions, the blade shape, etc). Normalizing the above equality by a reference value $W_{ref} = H \cdot A_{ref}^2$ yields

$$\bar{W} = \frac{W}{W_{ref}} = \frac{H \cdot A^2}{H \cdot A_{ref}^2} = \left(\frac{A}{A_{ref}}\right)^2 = \bar{A}^2$$

where $\bar{W}$ and $\bar{A}$ are called the normalized worksum and amplitude respectively. The gradient of the normalized worksum to the normalized vibration amplitude can therefore be obtained analytically:

$$\frac{d\bar{W}}{d\bar{A}} = 2\bar{A}$$

If the reference amplitude is chosen to be the same as the one at which the gradient is calculated, namely $A_{ref} = A$, then the gradient of the normalized worksum to the normalized amplitude is always 2.0.

A series of computations under different inter-blade phase angles and vibration frequencies for the two blades are performed. The calculated gradients of the normalized worksum to the normalized amplitude are listed in tables 8.1 and 8.2. The worksum gradients to vibration amplitude by the adjoint solver agree reasonably well with the analytic results with maximum relative difference no more than 5% for the 2D case and no more than 8% for the 3D case.

Table 8.1: Worksum gradient for a 2D controlled diffusion blade

vibration frequency (Hz)	IBPA (degrees)	Adjoint gradient	Analytic gradient
28	0	1.93	2.0
28	90	2.04	2.0
42	0	2.10	2.0
42	90	2.08	2.0

Table 8.2: Worksum gradient for a 3D controlled diffusion blade

vibration frequency (Hz)	IBPA (deg)	Adjoint gradient	Analytic gradient
28	0	1.89	2.0
28	90	2.14	2.0
42	0	2.15	2.0
42	90	2.09	2.0

8.4 Summary

An adjoint system is formulated based upon the unsteady flow equations at the three phases of a period of unsteadiness: $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$. Comparing with a steady adjoint system, the adjoint system is a coupled system of three sets of equations, corresponding to the unsteady flow equations at the three phases. The inlet, exit and solid wall boundary conditions for the three-phase adjoint system have a similar form to those for a steady adjoint system. However, the phase-shift periodic boundary condition for the unsteady adjoint system assumes a more complicated form than that for the unsteady flow equations. Nevertheless, the adjoint system can fairly easily be extended from a steady adjoint system and solved in the same manner. The gradient of a normalized worksum to its normalized amplitude is calculated by the adjoint solver and compared with the analytic value, with a good agreement achieved.

Chapter 9

Concurrent Aeromechanic and Aerodynamic Design Optimization

This chapter first supplements the elements expounded in the previous two chapters for a concurrent aerodynamic and aeromechanic design optimization. Then three case studies are presented to demonstrate the methodologies elaborated in the preceding two chapters and this chapter: a redesign of a 2D rotor blade at a single operating point, a redesign of the DLR rotor at a single operating point and at three operating points. Different from the cases presented in Chapter 6 regarding aerodynamic design optimization, the cases presented in this chapter will take account of both aerodynamic and aeromechanic performance of a blade in a design optimization simultaneously and explicitly. It is therefore called concurrent aeromechanic and aerodynamic design optimization.

9.1 Introduction

The blade flutter problem is modeled by a loosely coupled approach. This approach splits the coupled aerodynamics and structural dynamics into two sequential problems, with the structural dynamics problem modeled first and followed by the unsteady aerodynamics problem. The structural dynamics here is solved using the commercial software package-ANSYS to provide the mode shape and vibration frequency of a structural mode. The mode shape and vibration frequency are therefore fed into a CFD code to obtain the corresponding unsteady flow solutions. It is clear that the unsteady flow solution is directly determined by the blade geometry, the mode shape

and frequency of a structural mode.

In a concurrent aeromechanic and aerodynamic design optimization, both the blading aeromechanic performance and blading aerodynamic performance are extracted from the single unsteady flow solution at the three phases $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$. The aerodynamic performance is derived from the time-averaged flow solution constructed from the three-phase flow solutions. The aeromechanic performance (aero-damping) is derived from the first harmonic component of the flow solution also constructed from the three-phase flow solutions.

The flow field for a blade flutter prediction is largely linear (Moffatt and He, 2003). Therefore the time-averaged flow solution constructed from the three-phase flow solutions will be more or less the same as a steady flow solution for aerodynamic performance predictions. The influence of the mode shape and vibration frequency of a structural mode on the time-averaged flow solution, and therefore the blading aerodynamic performance, is small and negligible.

A change in a design variable results in changes in a blade shape, which in turn will change the vibration frequency and mode shape of a structure mode. All these changes eventually lead to changes in the unsteady flow solutions at the three phases, therefore the blading aerodynamic and aeromechanic performance. Fig.9.1 schematically unveils the dependency between different parameters. α is a design variable. X_{bs} denotes the

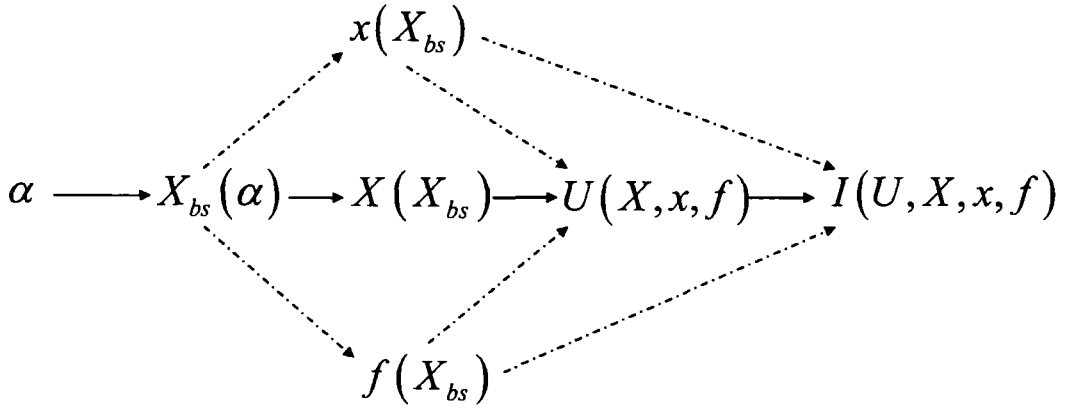


Figure 9.1: The dependency between different parameters for blading aeromechanics predictions

mesh coordinates on a blade surface, and is a function of α. X denotes the mesh

coordinates of a computational domain, which in turn is determined by X_{bs}. x and f are the mode shape and vibration frequency of a structural mode respectively. Both x and f are a function of X_{bs}. U denotes an unsteady flow solution at the three phases. It is determined directly by X, x and f through the unsteady flow equations. I is an objective function, which is an explicit function of U, X, x and f and represents the blading aerodynamic and aeromechanic performance.

The above dependency relation is embedded in the following gradient expression of the objective function to a design variable:

$$\begin{aligned}\frac{dI}{d\alpha} &= \frac{\partial I}{\partial U}\left(\frac{\partial U}{\partial X}\frac{dX}{dX_{bs}}\frac{dX_{bs}}{d\alpha}+\frac{\partial U}{\partial x}\underline{\frac{dx}{dX_{bs}}}\frac{dX_{bs}}{d\alpha}+\frac{\partial U}{\partial f}\underline{\frac{df}{dX_{bs}}}\frac{dX_{bs}}{d\alpha}\right) \\ &+ \frac{\partial I}{\partial X}\frac{dX}{dX_{bs}}\frac{dX_{bs}}{d\alpha}+\frac{\partial I}{\partial x}\underline{\frac{dx}{dX_{bs}}}\frac{dX_{bs}}{d\alpha}+\frac{\partial I}{\partial f}\underline{\frac{df}{dX_{bs}}}\frac{dX_{bs}}{d\alpha}\end{aligned} \tag{9.1}$$

In the loosely coupled approach used in the present work, the mode shape x and vibration frequency f are obtained through a FE analysis by the commercial code -ANSYS according to a FE blade surface mesh interpolated from a CFD blade surface mesh (X_{bs}). This tells that the sensitivities of a mode shape and vibration frequency to the blade geometry ($\frac{dx}{dX_{bs}}, \frac{df}{dX_{bs}}$) needs to be given by a FE analysis. These sensitivity calculations will cost considerable computing resources at a situation with hundreds of design variables if they are approximated by the finite difference method. With a commercial code, it is impossible to implement an adjoint code for an efficient sensitivity calculation. Therefore in the present work, those dependencies labeled in a dash-dot line in Fig.9.1 are neglected or those sensitivities as underlined in Eq.9.1 are set to zeros. The gradient expression in Eq.9.1 is approximated by

$$\frac{dI}{d\alpha} \approx \frac{\partial I}{\partial U}\frac{\partial U}{\partial X}\frac{dX}{dX_{bs}}\frac{dX_{bs}}{d\alpha}+\frac{\partial I}{\partial X}\frac{dX}{dX_{bs}}\frac{dX_{bs}}{d\alpha} \tag{9.2}$$

The neglect of the change of the mode shape and vibration frequency in a gradient calculation can be viewed in a similar way as to a frozen eddy viscosity. In each design iteration, the mode shape and vibration frequency are frozen like a frozen eddy viscosity in gradient calculations. While the mode shape and vibration frequency will be updated once a blade shape is updated in each design iteration, similar to the treatment of freezing the eddy viscosity.

9.2 Objective Function

The objective function is a weighted sum of some aeromechanic and aerodynamic performance parameters. The aeromechanic performance concerned here is blade flutter which is quantified by worksum (positive worksum corresponds to negative aero-damping) as defined in the preceding chapter. The aerodynamic performance concerned here includes mass flow rate, pressure ratio and entropy generation rate. If an aerodynamically good design is found to have poor aeromechanic performance (insufficient aero-damping) at some operating conditions, then a design optimization intent is to increase the aero-damping of the original design without deteriorating its aerodynamic performance. Therefore the objective function takes the following form to increase the aero-damping and set a limit to the change of mass flow rate and pressure ratio and the increase of entropy generation rate (entropy generation rate is allowed to decrease as much as it can, but not allowed to increase too much):

$$\begin{aligned} I &= \frac{W}{W_{ref}} + \sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right)^2 + \sigma_2 \left(\frac{\pi}{\pi_0} - 1 \right)^2 \\ &+ \sigma_3 \max \left(sign \left(1, \Delta\bar{s} - \Delta\bar{s}_0 \right), 0 \right) \times \left(\frac{\Delta\bar{s}}{\Delta\bar{s}_0} - 1 \right)^2 \end{aligned} \tag{9.3}$$

where W_{ref} is a positive reference value used to normalize the worksum. The absolute value of the initial worksum can be taken as the reference value, if it is not too small.

9.3 Structural Modeling and FE-CFD Interpolation

As pointed out in Chapter 2, blade flutter is modeled in a loosely coupled fashion in this work by splitting the coupled two-domain problem into two sequential problems in two domains. The first step of this approach is to get the vibration frequency and mode shape of a structural mode of interest. The calculation of vibration frequencies and mode shapes of a blade is conducted using the well-known finite element analysis software-ANSYS.

The modal analysis for a blade and the corresponding flow field analysis for the flow field in which the blade is immersed are carried out in different spacial domains with normally different types of meshes. The CFD mesh for a turbomachinery flow field

analysis discretizes the space occupied by the working fluid surrounding a blade, while the FE mesh discretizes the space occupied by the blade itself. A CFD mesh used by this flow solver is a structured mesh consisting of hexahedral elements only. The CFD mesh usually clusters towards the leading and trailing edges and endwalls of a blade to better resolve the flow field of higher spacial gradient in those regions. However a FE mesh is normally unstructured with a different distribution of mesh points in both spanwise (more uniformly distributed) and streamwise (fewer mesh points are required on a blade surface) directions. In this case study, the element type chosen is the SOLID95 in the ANSYS element library. The element is defined by 20 nodes having three degrees of freedom per node: translations in the nodal x, y, and z directions. It can tolerate irregular shapes without much loss of accuracy and is well suited to model curved boundaries like turbomachinery blade surfaces. The blade surface is the interface between the two meshes. These differences between the two meshes imply that along the blade surface the two meshes do not coincide with each other through their mesh points on the blade surface as revealed in Fig.9.2, resulting in a need to perform interpolation between the two meshes to transfer information. This loosely coupled method requires a one-way information transformation of the mode shape from a FE mesh to a CFD mesh.

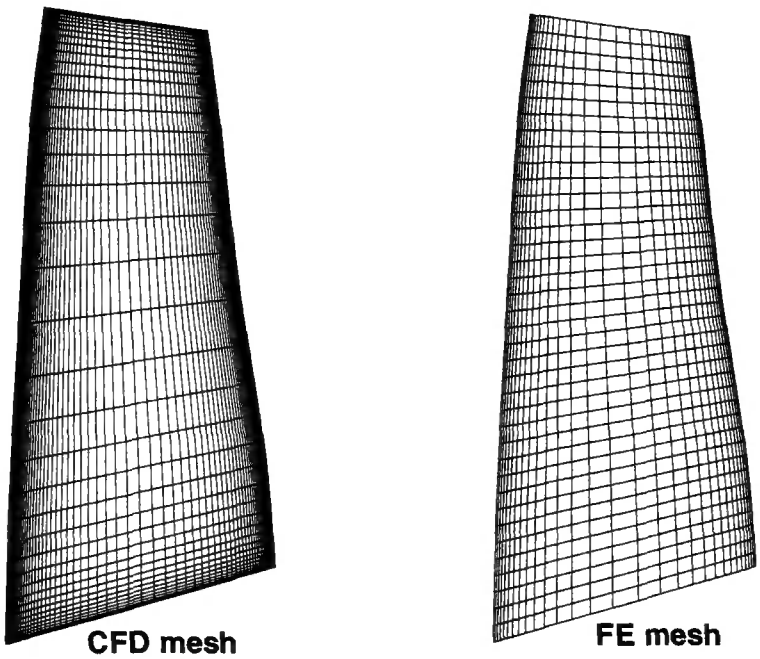


Figure 9.2: CFD and FE blade surface meshes

Before a FE analysis, a FE blade surface mesh is generated by interpolation according to a CFD blade surface mesh. This surface mesh is then imported to ANSYS to perform the modal analysis. The modal analysis will provide the natural vibration frequency and mode shape at a specified blade vibration mode. The mode shape needs to be interpolated back to the original CFD blade surface mesh for a corresponding unsteady flow analysis. The mode shape interpolated on the CFD blade surface mesh is further linearly distributed to the whole computational domain along corresponding mesh lines with zero mode displacement at the inlet and exit of a computational domain.

This FE-CFD interpolation interface was developed by Moffatt et al. (Moffatt and He, 2003; Ning et al., 2003; Moffatt et al., 2005) at Durham University for forced response predictions. The accuracy and robustness of the FE-CFD interpolation interface was demonstrated in (Moffatt and He, 2003). The FE modal analysis process is scripted using ANSYS macro commands in an automated way, so that it can be integrated into the design optimization system coded in FORTRAN to automate the whole process without any human interference once a problem is set up.

9.4 Design Optimization Process

The concurrent aeromechanic and aerodynamic design optimization process is broken down into the following procedures and shown schematically in Fig.9.3.

Step 1: Problem setup This is the first step of a design optimization process. It includes the specification of the blade geometry, boundary conditions, design variables, etc.

Step 2: Modal analysis The modal analysis is conducted by the FE analysis software package-ANSYS to get the blade vibration mode shape and frequency at a required vibration mode. The mode shape of the FE mesh points on the blade surface will be interpolated to the corresponding CFD mesh points on the blade surface by a FE-CFD mesh interpolation interface.

Step 3: Solve the unsteady flow equation The mode shape exported in the modal analysis is imported to the unsteady flow solver and interpolated linearly along mesh

lines to the whole computational domain with zero mode displacement at the far field (the inlet and exit planes) to set up the required meshes at the three phases $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$. Together with the boundary conditions, the unsteady flow equation is solved with the flow solution at the three phases obtained. The three-phase solutions in each cycle plays two important roles: 1) setting the right base steady (time-mean) and unsteady states for calculating the gradients for the next design cycle, 2) providing direct solutions to correct/minimize the errors in the 'projected' or perturbed fields based on the previous gradients.

Step 4: Solve the corresponding adjoint equation Once the unsteady flow solutions at the three phases are obtained, the corresponding adjoint systems can be formulated and solved accordingly. The order in which the flow equation and the adjoint equation are solved can not be changed. The flow equation must be solved before the adjoint equation, because the flow solution forms the coefficients of the linear adjoint system.

Step 5: Calculate gradients With the obtained flow solution and adjoint solution, the gradients of an objective function to all design variables can be obtained very efficiently with mesh perturbations.

Step 6: Calculate the step size The step size which determines the amount of change to the blade shape is determined by the empirical rule as detailed in Chapter 5.

Step 7: Update the blade geometry According to the steepest descent method, the step size is taken at the negative gradient direction. The perturbations to the blade shape are first determined and then superimposed to the blade shape to update the blade shape. Procedures 2 - 7 form a design cycle and are repeated till a stopping criterion is reached.

In such a design optimization process, each design cycle needs the solution of three systems of equations. The first one is the solution of the eigenvalue problem for the blade vibration frequency and mode shape. The second one is the solution of the nonlinear unsteady flow equations by the Nonlinear Harmonic Phase Solution method. The last one is the solution of the corresponding adjoint equations. The main time cost

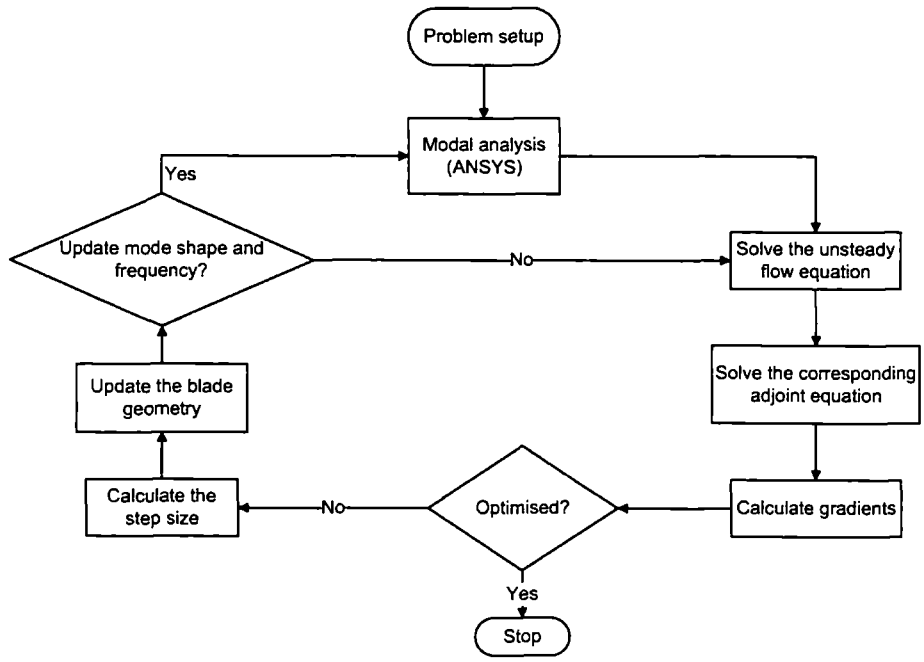


Figure 9.3: Flow chart of concurrent aeromechanic and aerodynamic design optimization

in one design cycle is consumed by the solution of the flow equations and the adjoint equations. The two systems usually consume more or less the same time cost. The solution of the eigenvalue problem (ANSYS solution) typically costs quite little time. The gradient calculation is conducted through mesh perturbation to get the sensitivity due to pure geometry perturbations. This means it can also consume considerable time compared with the time cost for the eigenvalue problem when there are hundreds of design variables. Nevertheless, the time cost for gradient evaluation is still acceptable and does not offset the great efficiency offered by the adjoint method, as explained in section 5.1 of Chapter 5.

9.5 Multi-Operating-Point Design Optimization

In reality, blade flutter rarely occurs at or near the design point. However, blade flutter at off-design conditions can be a big concern. For example, stall flutter, which occurs at high operating line with high pressure ratio and low mass flow rate, is the most common

type of flutter. If a design is found to have blade flutter at off-design conditions, performing an optimization over the blade at the problematic off-design conditions will not guarantee a good performance (especially aerodynamic performance) at its design point. In order to ensure the aerodynamic and aeromechanic performance of a blade at other operating points will not be worsened, when the aeromechanic performance of a blade is improved at one operating point, blading performance of more than one operating point needs to be considered simultaneously in a design optimization process. Similar to the multi-operating-point aerodynamic design optimization, it is natural and straightforward to extend the serial code for a single operating point concurrent design optimization to a parallel multi-operating-point case. The objective function to be minimized is formulated as a weighted sum of the sub-objective functions at the operating points considered:

$$I = \sum_{i}^{N} \sigma_i \cdot I_i \tag{9.4}$$

in which σ_i and I_i are the weighting factor and sub-objective function for the i^{th} operating point. N is the number of operating points to be considered in a design optimization. The weighting factor used in this work is independent of the flow solution. The gradient of the total objective function to a design variable α is accordingly a weighted sum of the sub-objective function gradients:

$$\frac{dI}{d\alpha} = \sum_{i}^{N} \sigma_i \cdot \frac{dI_i}{d\alpha} \tag{9.5}$$

Clearly, a sub-objective function I_i is a function of the flow solution at the i^{th} operating point only. Accordingly the corresponding gradient $\frac{dI_i}{d\alpha}$ calculation requires the flow solution and adjoint solution at the i^{th} operating point only. Therefore the calculation of the flow solution and adjoint solution for different operating points can be conducted in parallel without consulting each other. This apparently gives rise to parallelizing the solution process for design optimization considering performance at more than one operating point. The parallelization in fact is quite straightforward to implemented for this particular case and in return will significantly reduce real time cost, which is virtually the same as for the multi-operating-point aerodynamic design optimization as described in Chapter 5.

9.6 Design Optimization Case Studies

9.6.1 Redesign of 2D Rotor Blade Configuration

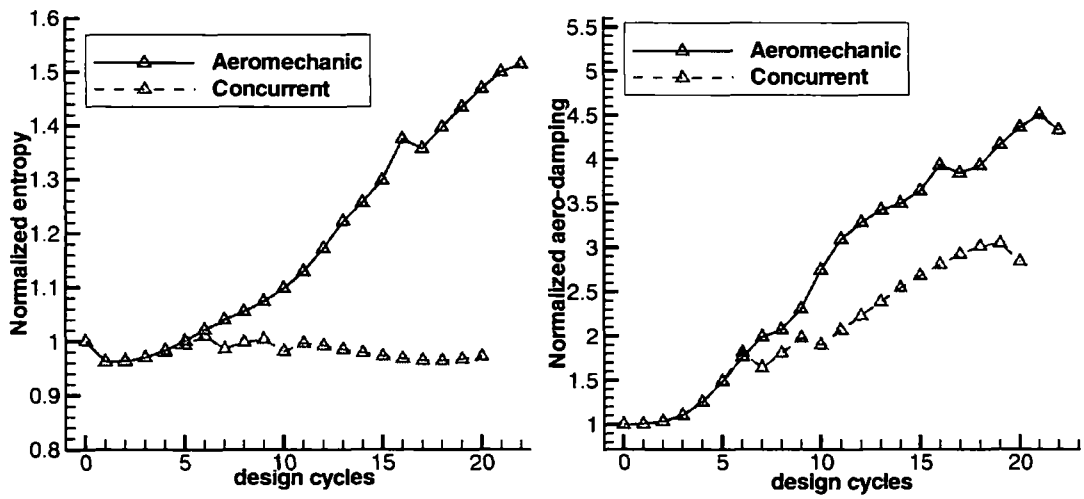
This design exercise is aimed to demonstrate the necessity of taking aerodynamics into account while increasing the aero-damping of turbomachinery blades and the feasibility of doing aeromechanic-related design optimizations using the adjoint method. A 2D section is taken from the 3D transonic fan-NASA rotor 67 as used in the aerodynamic design optimization in Chapter 6 and placed on a cylindrical surface. An H-type mesh has 37 mesh points in the circumferential direction and 110 mesh points in the axial direction with 27 mesh points before its leading edge and 20 after its trailing edge. Perturbations to the suction surface of the 2D blade are parameterized using 13 Hicks-Henne functions (Hicks and Henne, 1978) resulting in 13 design variables. The tangential thickness of the blade remains unchanged during the design process to avoid unrealistically thin blades.

The design target is to minimize the blade's worksum (maximizing the blade's aero-damping). This is done with or without aerodynamic constraints of mass flow rate, pressure ratio and entropy generation rate. In the design process, it is assumed that the blade vibrates at a mode shape which is a combination of flap and torsion with a frequency of 1000 Hz corresponding to a reduced frequency of around 2 with an IBPA of 98.18 degrees (a forward traveling wave mode with 7 nodal diameters). It is also assumed that the blade's vibration mode shape and frequency do not change with the update of the blade shape. For a pure aeromechanic design optimization, the objective function is the worksum. For a concurrent aeromechanic and aerodynamic design optimization, the objective function is a weighted sum of the worksum and the three aforementioned aerodynamic constraints (Eq.9.3).

Figures 9.4(a) and 9.4(b) depict the changes of entropy and aero-damping with design cycles for the two design scenarios: the pure aeromechanic design optimization, the current aeromechanic and aerodynamic design optimization. Fig.9.4(a) clearly reveals that without imposing aerodynamic constraints, the design process tends to increase the entropy generation rate dramatically. This implies worsening of the aerodynamic performance while the aero-damping is being increased, which should be avoided in a

design optimization. With the imposition of the aerodynamic constraints, the aero-damping can not be increased as much as without the aerodynamic constraints over the same design cycles. However the entropy generation rate is effectively maintained and the aero-damping is still increased considerably (Fig.9.4(b)).

The differences between the original blade profile, the optimised blade by the aeromechanic design and that by the concurrent design are shown in Fig.9.5. The aeromechanic design increases the camber of the part between the leading edge and the mid-chord. The increased camber accelerates the flow field more than the original blade does, leaving the flow field with a much stronger shock incurring more losses (Fig.9.6). The concurrent design reduces the camber of the part between the trailing edge and the mid chord and slightly increases the camber around the leading edge region. This geometry change allows the flow field to accelerate continuously with one shock wave. This shock wave is much weaker than the one produced by the aeromechanic design, as shown in Fig.9.6.



(a) entropy (b) aero-damping

Figure 9.4: Change of entropy and aero-damping with design cycles for the 2D rotor blade redesigns

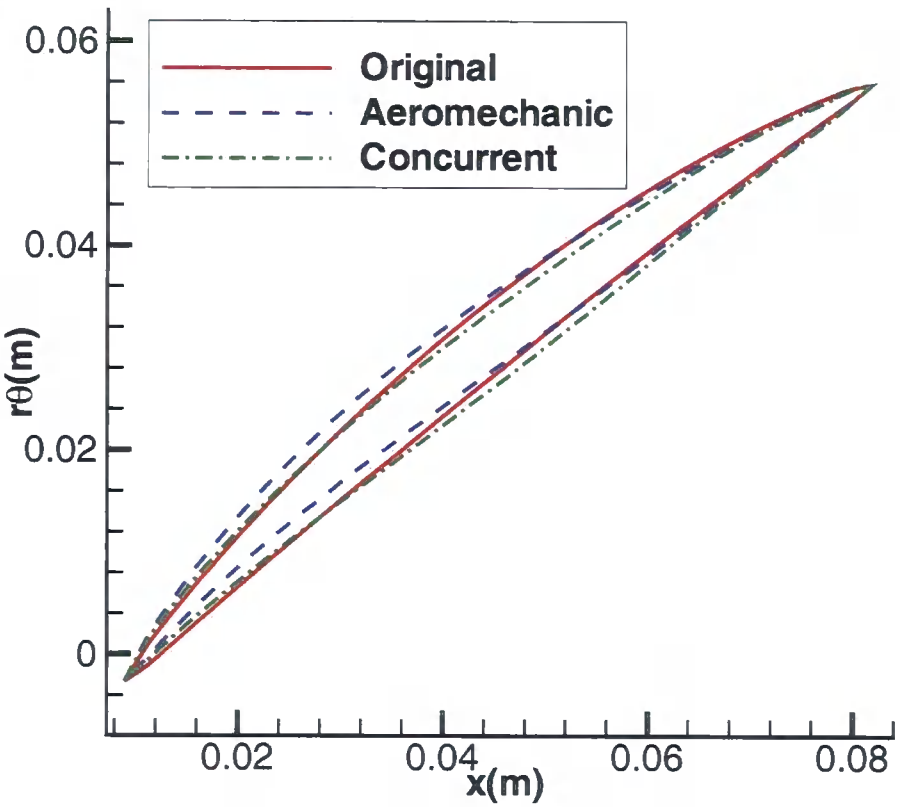


Figure 9.5: Blade geometry comparison between the original 2D rotor blade and the two optimised blades

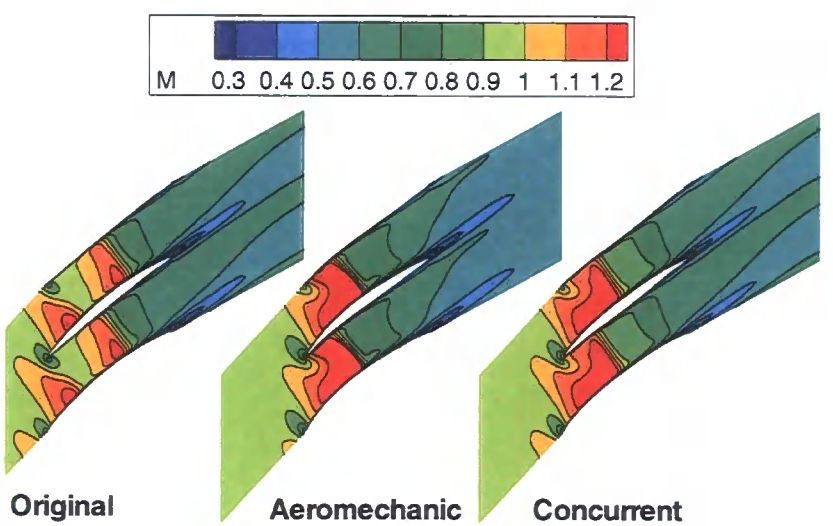


Figure 9.6: Comparison of Mach number contours between the three different designs for the 2D rotor blade redesigns

9.6.2 Redesign of DLR Rotor at a Single Operating Point

The rotor of the DLR stage used in one of the case studies in Chapter 6 is utilized here for a case study of the concurrent aerodynamic and aeromechanic design optimization using the adjoint method. At the rotor blade's first vibration mode, calculations show that the blade has sufficient aero-damping over its possible operating conditions along the design operating line. A hypothetic vibration mode is therefore created by lowering the natural vibration frequency (ANSYS output) by 20% here to make use of this rotor blade as a test case. The least aero-damping of the transonic DLR rotor, which is 0.1% in terms of the logarithmic decrement, occurs at the inter-blade phase angle of 25.71 degrees (corresponding to the second nodal diameter), when the rotor operates at a condition with the following aerodynamic performance parameters.

Aerodynamic performance parameters:	
Mass flow rate	16.25 kg/s
Pressure ratio	1.667
Efficiency	88.29%

and vibrates at the following hypothetic vibration mode:

Mode shape	the same as the blade's first natural vibration mode
Vibration frequency	589.4032 Hz (80% of the blade's first mode vibration frequency)

In the modal analysis, the chosen material for the rotor blade has a density of 4428.8 kg/m^3, Young modulus of $1.25 \times 10^{11} GPa$ and Poisson ratio of 0.27, which is close to the property of Titanium at ambient temperature.

The above operating condition of the blade is close to the blade's stall operating condition. This can be revealed from the performance map of the rotor blade (Figs.9.11(a) and 9.11(b)) and the pressure contours in Figs.9.7 and 9.8. Fig.9.7 shows pressure contours on the blade suction and pressure surfaces, while Fig.9.8 depicts pressure contours on a mid-span blade section. As it can be seen from these figures, the passage shock is pushed out of the blade passage well in front of the blade leading edge on the blade pressure surface. The operating point, at which the redesign is carried out, is chosen in such a way that it is as close as possible to the stall point but the flow solution process can still converge fully without any limit cycle, a necessary condition ensuring the adjoint solution based on this flow solution will converge without any problem.

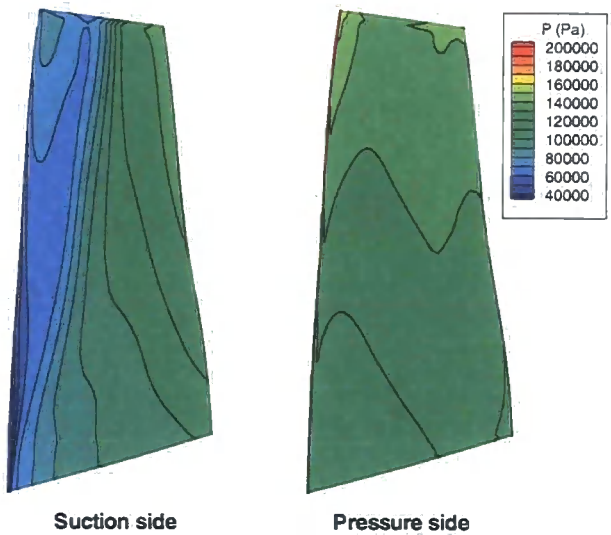


Figure 9.7: Pressure contours on the blade surfaces of the original DLR rotor

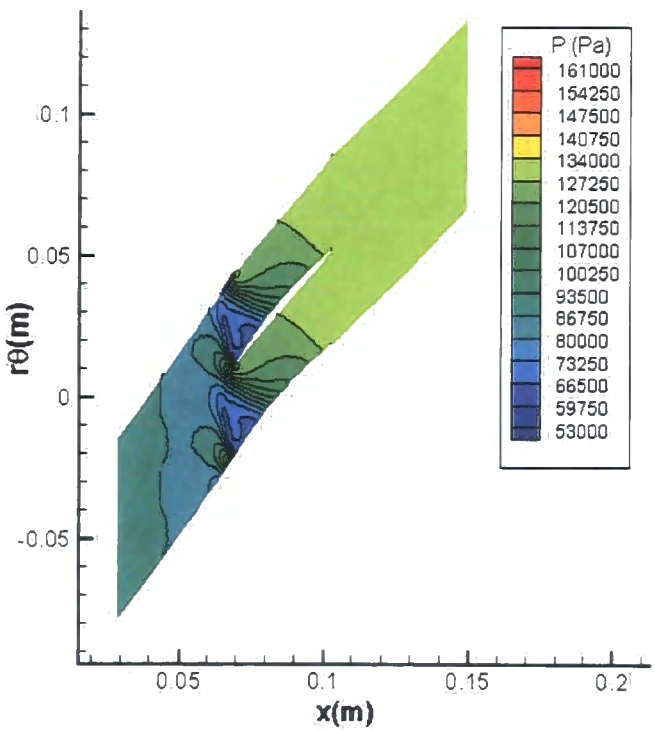


Figure 9.8: Pressure contours at the mid-span of the original DLR rotor

The design target is to increase the aero-damping of the blade while preventing unacceptable deterioration of the blade's isentropic efficiency and constraining the change of the mass flow rate and pressure ratio to a small range at this particular operating point.

The mesh has 120 mesh points in the axial direction, 37 mesh points in the circumferential direction and 39 mesh points in the radial direction. The blade to blade view and meridional view of the mesh are shown in Fig.9.9. 99 design variables are distributed on 11 radial mesh sections with 9 design variables on each section.

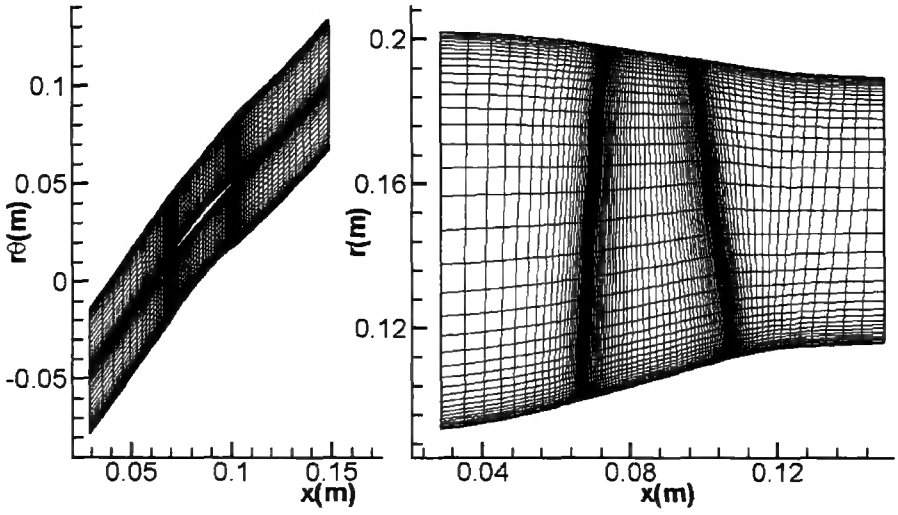
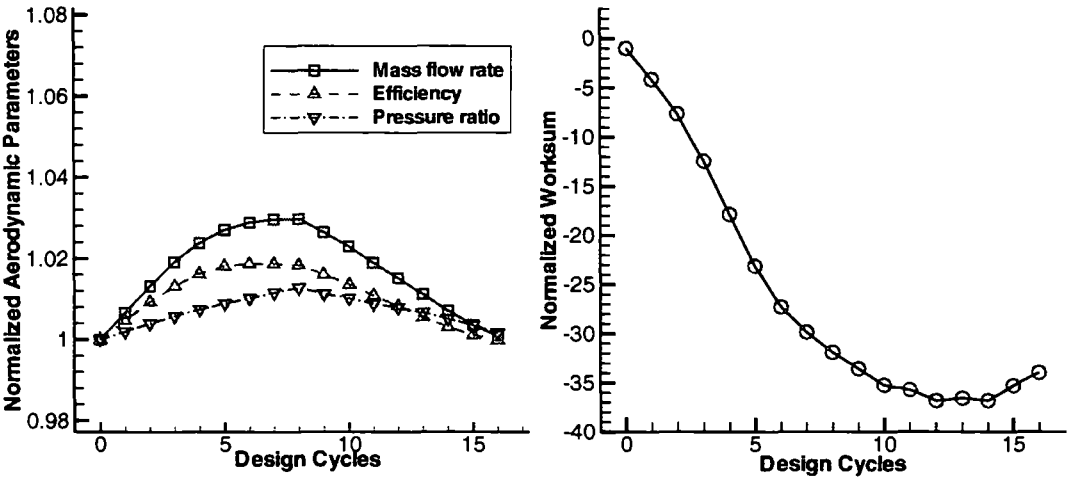


Figure 9.9: Blade to blade view and meridional view of the DLR rotor mesh

During the design process, the mode shape and vibration frequency of the blade are updated in every design cycle. In the update, the vibration frequency is always set to 80% of the blade's first natural vibration frequency. $\sigma_{1,2,3}$ are initially assigned the same value of 2×10^3 and changed to 2×10^5 after 8^{th} design cycle.

The changes of the constraints and worksum with design cycles are presented in Fig.9.10(a) and Fig.9.10(b) respectively. 16 design cycles are completed. The magnitude of worksum is increased dramatically to nearly 34 times its original value (3.4% in terms of the logarithmic decrement), while the mass flow rate, pressure ratio and efficiency are all kept more or less the same as their original values.

The performance map of the optimised blade is compared with that of the original blade in Figs.9.11(a) and 9.11(b). The performance of the optimised blade deteriorates



(a) Constraints versus design cycles

(b) Worksum versus design cycles

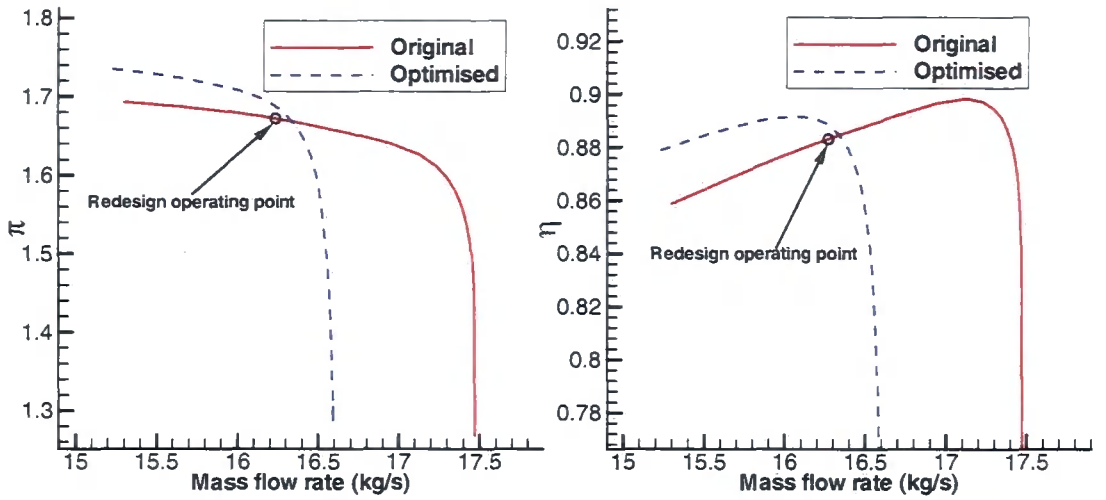
Figure 9.10: Change of normalized parameters with design cycles for the single point design optimization of the DLR rotor

dramatically at off-design conditions. The operating range of the new design is seriously reduced by its reduced choke mass flow rate.

It can be seen from Figs.9.11(a) and 9.11(b) that, the operating point, which has a lower mass flow rate than that of the peak efficiency point of the original design, now has higher mass flow rate than that of the peak efficiency point of the optimised design. That is to say, the operating point, close to the stall point of the original design, has become an operating point close to the choke point of the optimised design. The change of the position of the operating point in the performance map of the optimised design consequently improves its aero-damping performance at this operating point. Fig.9.12 shows the pressure contours on the blade suction and pressure surfaces of the optimised blade. Compared with those pressure contours in Fig.9.7, the passage shock in the optimised design is pushed downstream significantly, which is consistent with the position change of the operating point in the performance map of the optimised design.

The local work distributions on surfaces of the original (Fig.9.13) and optimised (Fig.9.14) blades show that, the blade suction surface is mainly a destabilizing region (positive work) with the zone immediately downstream of the shock as the major destabilizing region (positive work), while the blade pressure surface is mainly a stabilizing region with the strip immediately upstream of the passage shock as the main stabilizing re-

gion. When the passage shock is pushed downstream in the optimised design, the aero-damping increase is manifested through the increase of stabilizing region on the blade pressure surface (Fig.9.14).



(a) pressure ratio versus mass flow

(b) efficiency versus mass flow

Figure 9.11: Performance maps of the original DLR rotor and the single-point optimised rotor

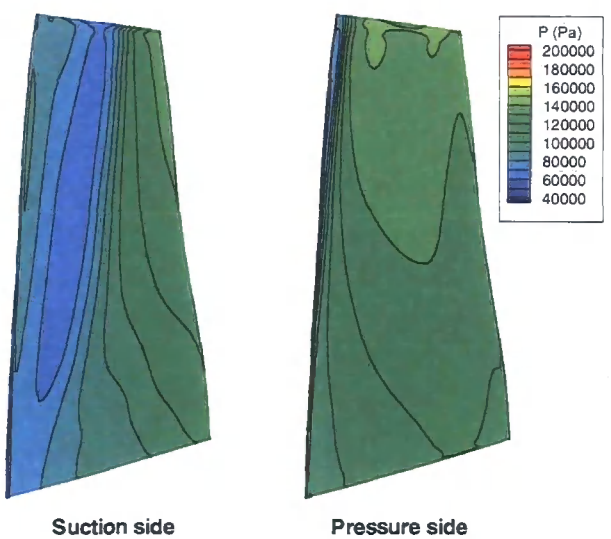


Figure 9.12: Pressure contours on the blade pressure and suction surfaces of the single-point optimised DLR rotor

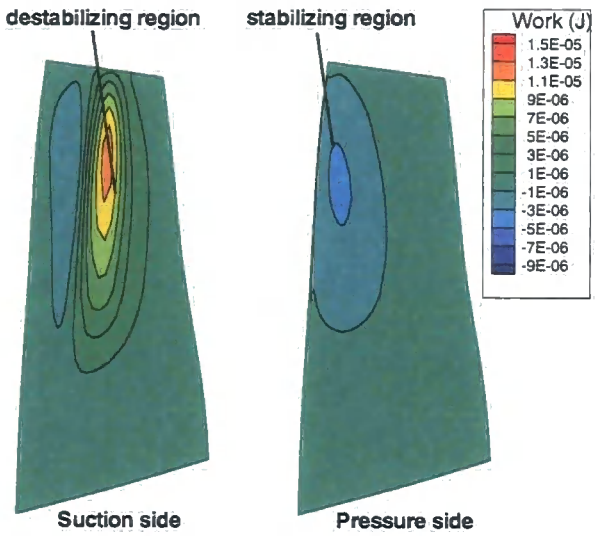


Figure 9.13: Local work distributions on the blade pressure and suction surfaces of the original DLR rotor

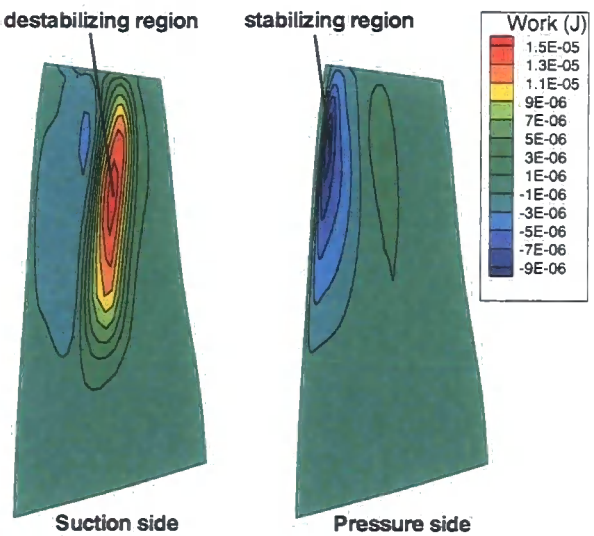


Figure 9.14: Local work distributions on the blade pressure and suction surfaces of the single-point optimised DLR rotor

9.6.3 Redesign of DLR Rotor at Multiple Operating Points

The redesign of the DLR rotor at a single operating point deteriorates its aerodynamic performance at its off-design conditions seriously as highlighted in Fig.9.11, though it increases the aero-damping a lot at the redesign point. This situation calls for a need to take account of performance at more than one operating point in a design optimization. Three operating points are considered in this multi-operating-point design optimization for the DLR rotor. One operating point is the one at which the previous single operating point design optimization is carried out. At this operating point, the aero-damping is quite low (0.1% in terms of the logarithmic decrement) and needs to be increased. Another operating point is the rotor peak efficiency point intended to maintain the peak efficiency in a design optimization. The last operating point has a mass flow rate close to the choke mass flow rate aimed at maintaining the choke mass flow rate of an optimised design. The objective function is a weighted sum of the three sub-objective functions (Eq.9.3) at the three operating conditions:

$$I = \sigma_{stall} I_{stall} + \sigma_{peak} I_{peak} + \sigma_{choke} I_{choke}$$

The weighting factors σ_{stall}, σ_{peak} and σ_{choke} are assigned values of 0.8, 0.1 and 0.1 respectively. The performance of the original DLR rotor at the three operating points are listed in Table 9.1.

Table 9.1: The original DLR rotor performance at three operating points

operating point	mass flow rate(kg/s)	pressure ratio	efficiency(%)	log-dec(%)
stall	16.25	1.667	88.29	0.1
peak	17.07	1.626	89.70	2.7
choke	17.46	1.421	83.47	3.4

Table 9.2: Performance of the multi-point optimised DLR rotor at three operating points

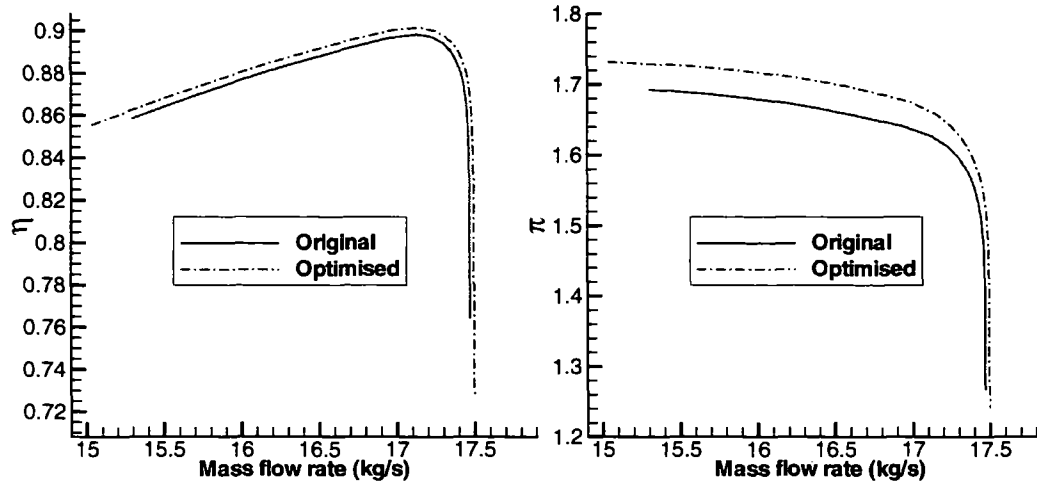
operating point	mass flow rate(kg/s)	pressure ratio	efficiency(%)	log-dec(%)
stall	16.78	1.716	89.57	2.0
peak	17.27	1.661	89.76	3.2
choke	17.52	1.449	83.60	3.9

The performance of the multi-point optimised rotor at the three operating points are given in Table 9.2. The optimised design has slightly higher mass flow rate, pressure ratio, isentropic efficiency at all three operating points. At the near stall operating

point, the aero-damping of the optimised design is 20 times of that of the original design. Even at other two operating points, the optimised design also has higher aero-damping over the original one.

The performance map of the optimised design is compared with that of the original design in Fig.9.15. In line with the aerodynamic performance gain indicated in Table 9.2, the optimised design has higher efficiency and pressure ratio over the whole operating line at the given operating speed. Fig.9.16 shows the overall aero-damping (in terms of the logarithmic decrement) of the original design and the optimised design at first few nodal diameters, when the blades operate at the near stall point. The optimised design has increased aero-damping at all these nodal diameters, with the lowest aero-damping occurring at the second nodal diameter at which the original design has the lowest aero-damping. That is to say the design optimization does not change the nodal diameter at which the lowest aero-damping occurs. Comparing the pressure contours in Fig.9.17 with those in Fig.9.7, one can see that shock position in the optimised design remains at almost the same position as the shock in the original design. ensuring the optimised design does not change the position of the original near stall operating point along the performance map of the optimised design. Fig.9.18 shows the local work distribution on a blade surface: the optimised design has increased the stabilizing work on the blade pressure surface and reduced the destabilizing work on the blade suction surface.

The blade geometry of the original blade, the optimised blade by the single point design optimization and the one by the three-point design optimization is compared at three different span locations in Figs.9.19, 9.20 and 9.21. The single point design has increased the blade camber from the leading edge to the rear part of the mid-chord with most camber increase occurring around the mid-span region (Fig.9.19) while it leaves the trailing edge region intact (Figs.9.20 and 9.21). On the contrary, the three-point design has increased the blade camber in the trailing edge region leaving the leading edge region intact (Figs.9.20 and 9.21).



(a) Efficiency versus mass flow rate (b) Pressure ratio versus mass flow rate

Figure 9.15: Performance maps of the original DLR design and the three-point optimised design

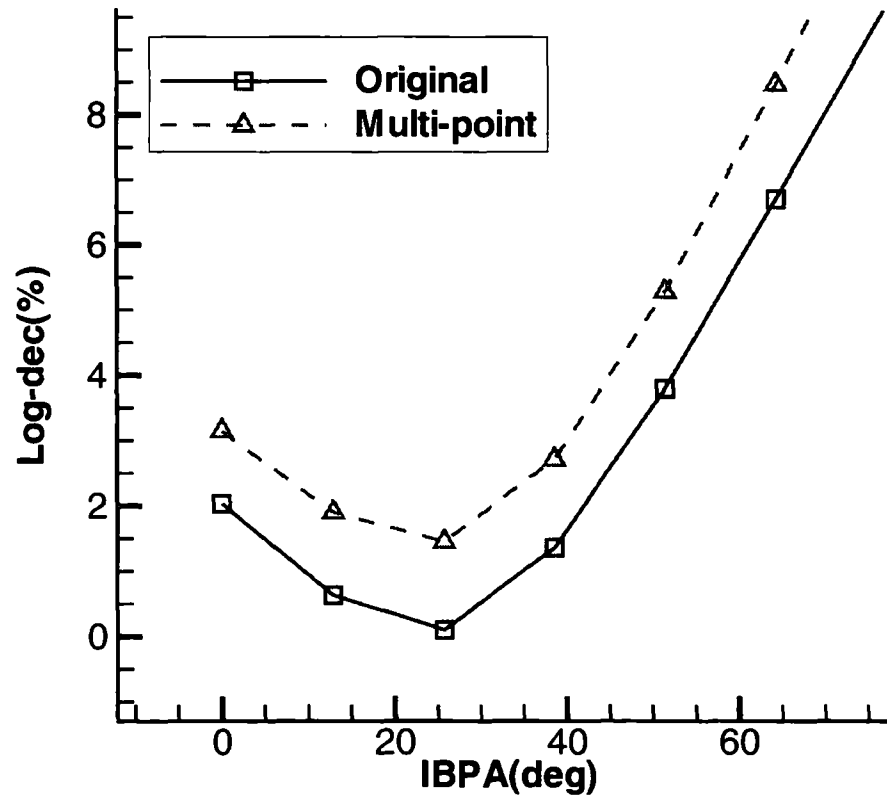


Figure 9.16: Aero-damping comparison at different inter-blade phase angles between the original DLR rotor and the three-point optimised rotor (near stall operating point)

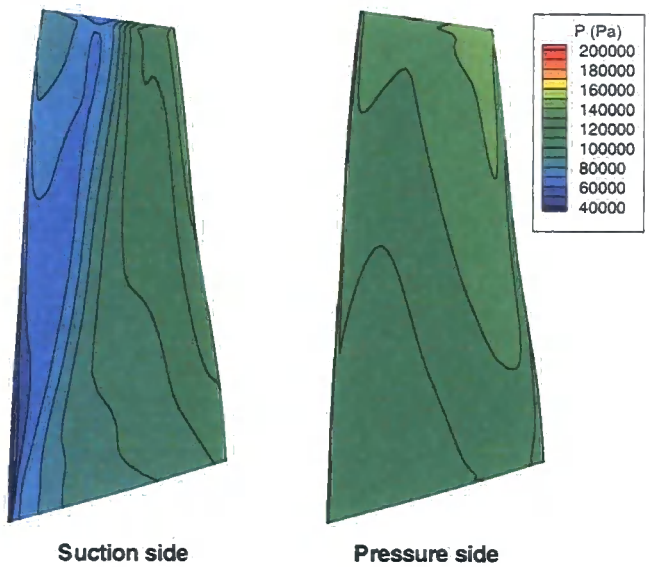


Figure 9.17: Pressure contours on the three-point optimised DLR rotor blade pressure and suction surfaces (near stall operating point)

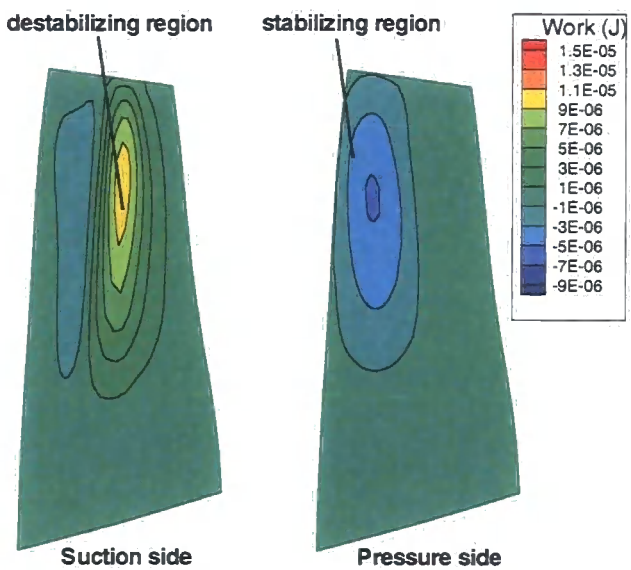


Figure 9.18: Local work distributions on the blade pressure and suction surfaces of the three-point optimised DLR rotor (near stall operating point)

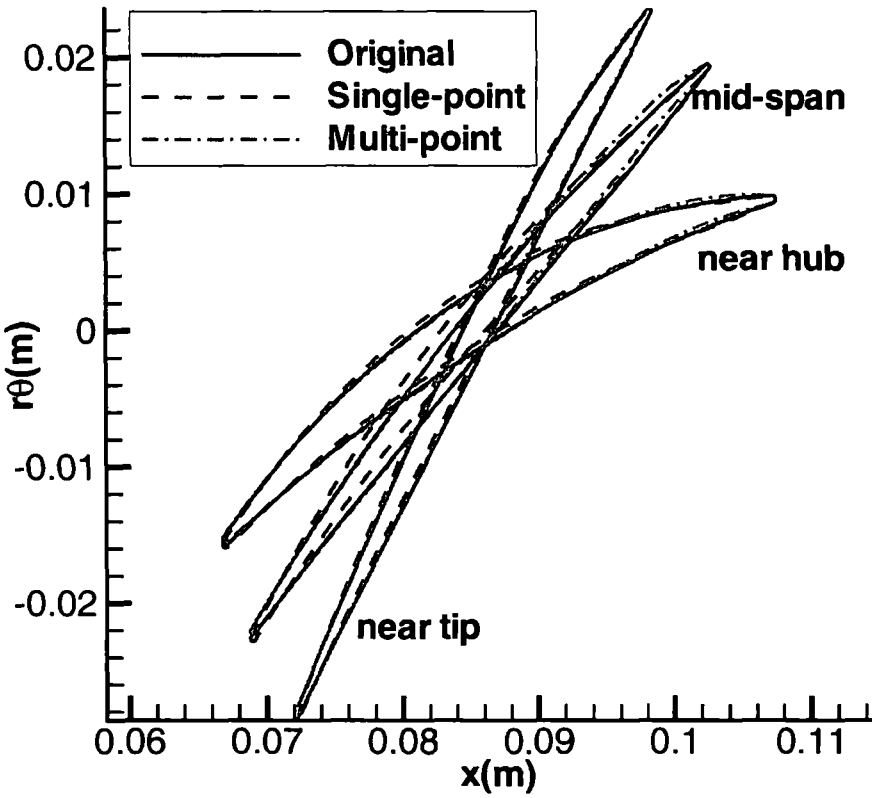


Figure 9.19: Blade geometry comparison between different DLR rotor designs

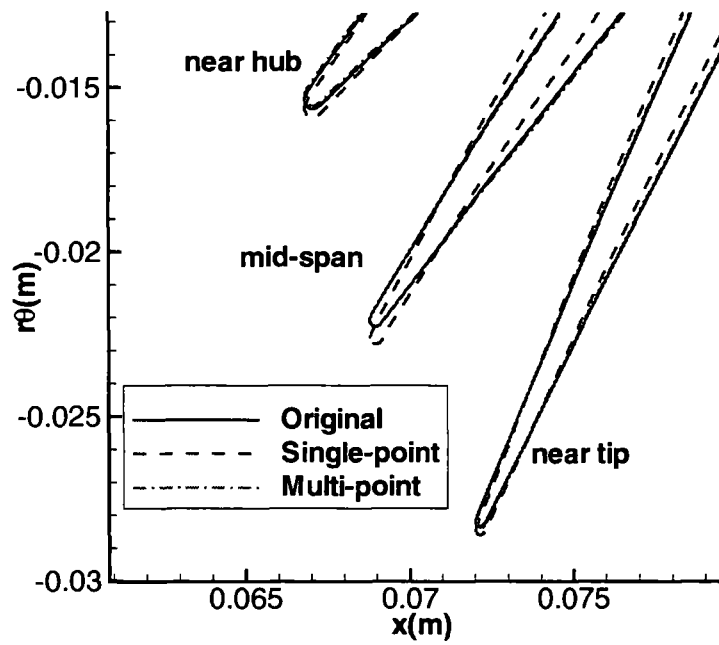


Figure 9.20: Close-up of blade geometry comparison around the leading edges of different DLR rotor designs

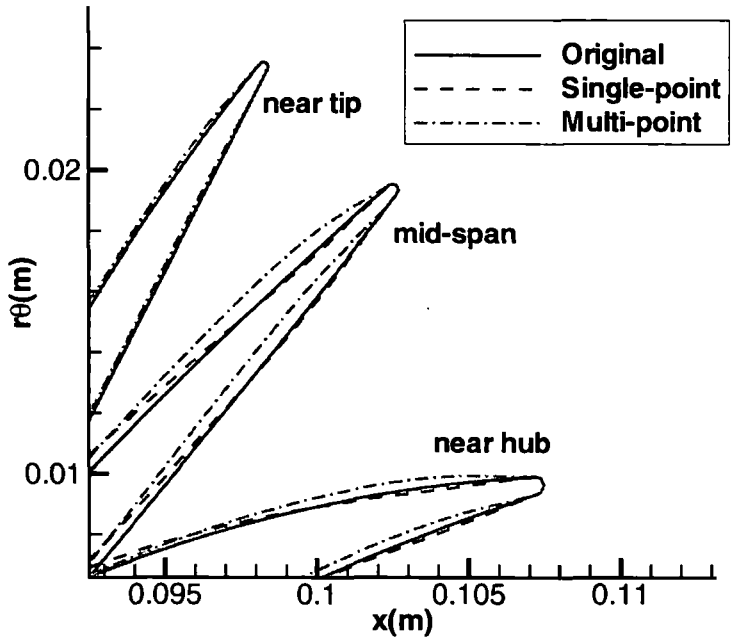


Figure 9.21: Close-up of blade geometry comparison around the trailing edges of different DLR rotor designs

9.7 Summary

Design optimization systems for turbomachinery concurrent aeromechanic and aerodynamic design optimization at either single operating point or multiple operating points are developed. For a design optimization at single operating point, the objective function is a weighted sum of the worksum, entropy generation rate, mass flow rate and pressure ratio. For a design optimization at multiple operating points, the objective function is a weighted sum of the sub-objective functionals at operating points of interest. The blade geometry is changed at a direction of negative gradients according to the steepest descent method. The blade shape perturbation is parameterized using Hick-Henne shape functions, as used for aerodynamic design optimizations.

Three design case studies are carried out. The first design optimization is for a 2D blade section. This case study has demonstrated the necessity of taking into account of blading aerothermal performance when performing an aeromechanic design optimization. It has also proved the validity of conducting a concurrent aeromechanic and aerodynamic design optimization using the adjoint method. The second case optimises

the aeromechanic performance of the DLR rotor at an aeroelastically problematic operating point while maintaining its aerodynamic performance at the same operating point. The case study has shown that the single operating point design optimization tends to deteriorate blading aerodynamic performance at off-design points significantly. The third case study is a three-operating-point concurrent design optimization of the same DLR rotor. This three-operating-point design optimization considers a near stall operating point, the rotor peak efficiency point and a near choke operating point. The three-point optimised design does not only have highly improved aero-damping over the original design at the aeromechanically problematic operating point, but also has a better aerodynamic performance over the whole operating range than the original design. This concludes that the developed design system is capable of doing concurrent aeromechanic and aerodynamic design optimizations, and proves the developed methodologies are valid in a blading concurrent design optimization and are appropriately implemented.

Chapter 10

Conclusions and Recommendations for Future Work

10.1 Conclusions

The two main research objectives have been successfully accomplished. One is the successful application of the adjoint method to turbomachinery blading aerodynamic design optimization in a multi-bladerow environment. The other is the integration of the adjoint method with the efficient unsteady flow solution method - the Nonlinear Harmonic Phase Solution method - to perform concurrent aeromechanic and aerodynamic design optimizations.

Aerodynamic Design Optimization

A basic adjoint solver is developed first, based upon the well-established continuous adjoint method, for a CFD-based aerodynamic design optimization. The validity of the adjoint method in gradient calculations and its appropriate implementation have been verified and demonstrated by two case studies: the lift coefficient gradient and the inverse design.

The capability of the basic adjoint solver is enhanced by the introduction of an adjoint mixing-plane treatment, which is the first of the kind in open literature, to the author's knowledge. This novel treatment is the counterpart of the flow mixing-plane treatment, which enables the communication of the adjoint solution in adjacent bladerow domains. Like the flow mixing-plane treatment, the adjoint mixing-plane treatment also features

the two important properties of conservation and non-reflectiveness. The appropriate formulation and implementation of the adjoint mixing-plane treatment are initially demonstrated by a gradient comparison investigation and a design optimization case study for a 2D compressor stage configuration.

Four aerodynamic design optimization case studies of realistic three-dimensional configurations are carried out and produce promising improvements over the original designs. The first case is the redesign of the NASA rotor 67 at a single operating point with 1.77% point increase of isentropic efficiency and well-satisfied constraints in mass flow rate and pressure ratio. The second redesign is a DLR compressor stage at the stage peak efficiency point. This is also a single operating point design optimization, but in a two-bladerow environment. The redesign offers 0.72% point increase of isentropic efficiency without violating constraints. The third case deals with the redesign of the first three rows of an industrial compressor of 7 blade rows (ATC). Two attempts are initially made to seek an optimised design which can offer a better performance map than the original design by carrying out a single operating point design optimization at two different operating points. The failed attempts call for the implementation of a parallel multi-operating-point design optimization. A two-point design optimization is carried out over the three-row configuration and delivers an optimised design of better performance map over the original one. The fourth case is attempted to demonstrate the ultimate capability of the design system for multi-bladerow turbomachines. The whole ATC compressor of 7 rows is put into a single operating point design optimization with the last 6 blade rows being changed with thousands of design variables in total. A remarkable efficiency gain of 2.47% point is achieved with acceptably small change of mass flow rate and pressure ratio.

Concurrent Aeromechanic and Aerodynamic Design Optimization

Among the various efficient solution methods for the URANS equations for turbomachinery aeromechanics, the Nonlinear Harmonic Phase Solution method is selected to integrate with the adjoint method for efficient gradient calculation for a concurrent aeromechanic and aerodynamic design optimization. The unsteady flow solver based upon the Nonlinear Harmonic Phase Solution method has been developed and validated

against both a semi-analytic method and experimental data, with favorable comparisons between the results by the flow solver and their counterparts by the semi-analytic method and experimental data. An adjoint system by the semi-continuous adjoint approach has been formulated according to the harmonic formulation of the unsteady flow equations at the three phases $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$. The validity of the adjoint system for gradient calculations is verified by the good comparison of the worksum gradients by the adjoint system against the analytic results.

The concurrent blading aeromechanic and aerodynamic design optimization has the great advantage of taking care of both blading aerodynamics and aeromechanics simultaneously in a design optimization process. This avoids the iterative process, to meet the often conflicting aerodynamic and aeromechanic requirements, which is otherwise required if a blade is either aerodynamically or aeromechanically optimised without coupling both of them simultaneously. To the author's knowledge, this represents the first effort of its kind in using the adjoint method for designing 3D turbomachine blades concerning both aerodynamics and aeromechanics simultaneously.

Three case studies have been conducted to investigate further the validity, necessity and benefit of carrying out concurrent aeromechanic and aerodynamic design optimization.

The two redesign scenarios of the 2D rotor configuration conclude:

1) an aeromechanic optimization without considering aerodynamic performances tends to deteriorate the aerodynamic performance unacceptably,

2) the implemented methodologies for a concurrent aeromechanic and aerodynamic design optimization is effective and beneficial.

The concurrent redesign of the DLR rotor at a hypothetic aeromechanically problematic operating point delivers an optimised design of much higher aero-damping without sacrificing its aerodynamic performance at this particular operating point. However the optimised has a much narrow operating range compared with the original design, pointing to the need of a multi-operating-point design optimization. Accordingly a three-operating-point concurrent design optimization is carried out for the DLR rotor. The optimised design not only has much higher aero-damping over the original design at the problematic operating point, but also has a better aerodynamic performance

over the whole operating range.

All the results from the test cases and application case studies have consistently verified the technical correctness of the methods development and their implementations, and demonstrated their validity and effectiveness.

10.2 Recommendations for Future Work

The potential of this work

The following two areas are identified as a potential of this work:

a) The developed method could be used for preliminary, multi-row optimisations when provided with an annulus line.

b) The design system could usefully be used for last minute design changes if, for example, a blade row was found to flutter.

Gradient Evaluation Scheme

In this research work, the adjoint method is the methodology for efficient gradient evaluation. Though the aerodynamic design optimization using the adjoint method can be carried out in a multi-bladerow environment, the design system is unable to perform a concurrent aeromechanic and aerodynamic design optimization in such an environment. As one knows, the unsteady flow equation can be solved using the Nonlinear Harmonic Phase Solution method in a multi-bladerow environment for turbomachinery blading aeromechanics. Hence it is worthwhile to investigate the possibility to extend the adjoint method to a concurrent aeromechanic and aerodynamic design optimization in a multi-bladerow environment. This extension will call for a scheme for transferring adjoint solution information between two adjacent bladerow domains. This scheme should be a counterpart of the one enabling the unsteady flow solution to be solved in a multi-bladerow environment and also should be mathematically consistent to obey the adjoint formulation principle.

Shape Parameterization Scheme

A shape perturbation parameterization scheme using Hicks-Henne like functions is employed in this developed design system. The biggest drawback of the scheme lies in the difficulty of preserving realistic geometry shape around a blade's leading edge and trailing edge. Though a remedy has been provided and produced reasonably satisfactory results, more generic methods should be sought for this purpose. In addition, the geometry constraint of constant tangential thickness is applied in a design optimization. This geometry constraint is applied due to the nature of the adopted mesh (circumferentially running mesh lines are parallel to the circumferential coordinate). It is quite difficult to constrain the normal thickness of a blade section profile in the current mesh, though it is more desirable from an industrial application point of view to do that.

Integrating with Other Design Optimization Methods

A gradient-based method has the well-known drawback of getting trapped at a local optimum solution, though it is claimed by Jameson (2003) that this does not pose a severe problem in industrial applications. This drawback can be relieved if a gradient-based method can be coupled with global search methods. For example, the gradients by the adjoint method can be used to construct a better response surface together with objective functionals in the response surface method. This will ideally lead to faster convergence compared with the original response surface method in a design optimization.

Bibliography

Adamczyk, J. (1985). Model equations for simulating flows in multistage-turbomachinery. ASME Paper, 85-GT-226. 2.2.1.2

Alonso, J. J., Kroo, I. M., and Jameson, A. (2002). Advanced algorithms for design and optimization of quiet supersonic platforms. AIAA Paper,02-0144. 2.4.1

Anderson, W. and Venkatakrishnan, V. (1997). Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. AIAA Paper,97-0643. 2.4.1, 5.3

Arens, K., Rentrop, P., Stoll, S., and Wever, U. (2005). An adjoint approach to optimal design of turbine blades. *Applied Numerical Mathematics*, 53:93–105. 2.4.3, 4, 5.3

Arnone, A. (1993). Viscous analysis of three-dimensional rotor flow using a multigrid method. Technical report, NASA TM106266. 6.2.1

Baldwin, B. and Lomax, H. (1978). Thin layer approximation and algebraic model for separated turbulent flows. AIAA Paper 78-257. 3.1, 3.2

Benini, E. (2004). Three-dimensional multi-objective design optimization of a transonic compressor rotor. *Journal of Propulsion and Power*, 20:559–565. 2.3, 5.3

Blom, F. (2000). Considerations on the spring analogy. *International Journal for Numerical Methods in Fluids*, 32:647 – 668. 1.3

Büche, D., Guidati, G., and Stoll, P. (2003). Automated design optimization of compressor blades for stationary, large scale turbomachinery. ASME Paper, GT2003-38421. 2.3, 5.3

Burguburu, S., Toussaint, C., Bonhomme, C., and Leroy, G. (2004). Numerical optimization of turbomachinery bladings. *Journal of Turbomachinery*, 126:91–100. 2.3, 5.3

Chen, T., Vasanthakumar, P., and He, L. (2001). Analysis of unsteady blade row interaction using nonlinear harmonic approach. *Journal of Propulsion and Power*, 17(3):651–658. 3.1, 7.3.1

Chima, R. (1998). Calculation of multistage turbomachinery using steady characteristic boundary conditions. AIAA Paper,98-0968. 2.1.1

Corral, R. and Gisbert, F. (2006). Profiled end-wall design using an adjoint Navier-Stokes solver. ASME Paper,GT2006-90650. 2.4.3

Cumpsty, N. and Greitzer, E. (2004). Ideas and methods of turbomachinery aerodynamics: A historical view. *Journal of Propulsion and Power*, 20(1):15–26. 2.1

de Vito, L., den Braembussche, R. V., and Deconinck, H. (2003). A novel two-dimensional viscous inverse design method for turbomachinery blading. *Journal of Turbomachinery*, 125:310–316. 2.3

Denton, J. (1992). The calculation of three-dimensional viscous flow through multistage turbomachines. *Journal of Turbomachinery*, 114:18–26. 2.1.1, 2.4.3

Denton, J. and Singh, U. (1979). Time-marching methods for turbomachinery flow calculations. 2.1.1, 3.4, 4.4

Denton, J. and Xu, L. (2002). The effects of lean and sweep on transonic fan performance. ASME Paper GT2002-30327. 2.3

Dunker, R., Rechter, H., Starken, H., and Weyer, H. (1984). Redesign and performance analysis of a transonic axial compressor stator and equivalent plane cascades with subsonic controlled diffusion blades. *Journal of Engineering for Gas Turbines and Power*, 106:279–287. 6.1.2, 6.2.2

Duong, H. A., Jung, S. K., Thi, N. A., Myong, R. S., and Cho, T. H. (2007). A three-dimensional moving structured mesh for aerodynamic design optimization. In *Proceedings of the 7th Asian CFD Conference*, Bangalore, India. 1.3

Duta, M. (2002). *The Use of the Adjoint Method for the Minimisation of Forced Vibration in Turbomachinery.* PhD thesis, University of Oxford. 2.2, 2.4.4, 2.5

Duta, M. and Giles, M. (2006). The use of automatic differentiation for adjoint CFD codes. In *European Conference on Computational Fluid Dynamics*, TU Delft,The Netherlands. 2.4.3

Duta, M., Giles, M., and Campobasso, M. (2002). The harmonic adjoint approach to unsteady turbomachinery design. *International Journal for Numerical Methods in Fluids*, 40:323–332. 2.4.4, 2.5

Duta, M., Shahpar, S., and Giles, M. (2007). Turbomachinery design optimization using automatic differentiated adjoint code. ASME Paper, GT2007-28329. 2.4, 2.4.3

Elliott, J. and Peraire, J. (1998). Constrained, multipoint shape optimization for complex 3D configurations. *Aeronautical Journal*, 100(1017):365–376. 5.2

Erdos, J., Alzner, E., and McNally, W. (1977). numerical solution of periodic transonic flow through a fan stage. *AIAA Journal*, 15(11). 2.2.1.1, 2.2.1.2

Florea, R. and Hall, K. (2001). Sensitivity analysis of unsteady inviscid flow through turbomachinery cascades. *AIAA Journal*, 39(6):1047–1056. 2.4.4

Giles, M. (1991). UNSFLO: A numerical method for the calculation of unsteady flow in turbomachinery. Technical report, Gas turbine laboratory, Massachusetts institute of technology. 2.1.1, 2.2.1.1, 3.4, 4.4, 7.2

Giles, M., Ghate, D., and Duta, M. (2006). Using automatic differentiation for adjoint CFD code development. In Cutup, B., Korunas, S., Sharon, R., and Priyadarshi, P., editors, *Recent Trends in Aerospace Design and Optimization*, pages 426–434. Tata McGraw-Hill, New Delhi. 2.4

Giles, M. and Pierce, N. (1997). Adjoint equations in CFD: duality, boundary conditions and solution behaviour. AIAA Paper,97-1850. 4.3, 4.5

Giles, M. and Pierce, N. (1998). On the properties of solutions of the adjoint Euler equations. In 6^{th} *ICFD Conference on Numerical Methods for Fluid Dynamics*, Oxford,UK. 4.3, 4.5

Giles, M. and Pierce, N. (2000). An introduction to the adjoint approach to design. *Flow,Turbulence and Combustion*, 65(3/4):393–415. 2.4, 4

Giles, M. and Pierce, N. (2001). Analytic adjoint solutions for the quasi-one-dimensional Euler equations. *Journal of Fluid Mechanics*, 426:327–345. 2.4

Hah, C. (1999). Aerodynamic lean and sweep for improvements in compressor performance. Lecture Series 1999-02 at the von Karman Institute for Fluid Dynamics, Brussels. 2.3

Hall, K. and Lorence, C. (1993). Calculation of three-dimensional unsteady flows in turbomachinery using the linearized harmonic Euler equations. *Journal of Turbomachinery*, 115(4):800–809. 2.2.1.2

Hall, K., Thomas, J., and Clark, W. (2002). Computation of unsteady nonlinear flows in cascades using a harmonic balance technique. *AIAA Journal*, 40(5):879–886. 2.2.1.2, 2.4.4

Harvey, N., Rose, M., Taylor, M., Shahpar, S., Hartland, J., and Gregory-Smith, D. (2000). Nonaxisymmetric turbine end wall design: Part I three-dimensional linear design system. *Journal of Turbomachinery*, 122:278–285. 2.3

He, L. (1989). An Euler solution for unsteady flows around oscillating blades. *Journal of Turbomachinery*, 112:714–722. 2.2.1.1, 2.2.1.2

He, L. (2008). Harmonic solution of unsteady flow around blade with separation. *to appear in AIAA Journal.* 2.1, 2.2.1.1, 2.2.1.2, 2.4.4, 7.3.1, D

He, L., Chen, T., Wells, R., Li, Y., and Ning, W. (2002). Analysis of rotor-rotor and stator-stator interferences in multi-stage turbomachines. *Journal of Turbomachinery*, 124:564–571. 3.1

He, L. and Denton, J. (1994). Three-dimensional time-marching inviscid and viscous solutions for unsteady flows around vibrating blades. *Journal of Turbomachinery*, 116:469–476. 2.2.1.1, 3.1, 3.5.2

He, L. and Ning, W. (1998). An efficient approach for analysis of unsteady viscous flows in turbomachines. *AIAA Journal*, 36(11):2005–2012. 2.2.1.2, 3.1

Health, M. (2002). *Scientific Computing: An Introductory Survey, Second Edition.* McGraw-Hill Companies, Inc. 1.3

Hicks, R. and Henne, P. (1978). Wing design by numerical optimization. *Journal of Aircraft*, 15(7):407–413. 5.3, 6.1.1, 9.6.1

Horlock, J. and Denton, J. (2005). A review of some early design practice using computational fluid dynamics and a current perspective. *Journal of Turbomachinery*, 127:5–13. 1.2

Hu, P., Choo, B., Zangeneh, M., and Rahmati, M. (2006). On design of transonic fan rotors by 3D inverse design method. ASME Paper, GT2006-91173. 2.3

Huang, X. (2006). *Three-Dimensional Unsteady Flow in Oscillating Turbine Blade Row.* PhD thesis, School of Engineering, University of Durham. 2.2, 2.2.1.1

Jameson, A. (1988). Aerodynamic design via control theory. *Journal of Scientific Computing*, 3(3):233–260. 2.4, 2.4.1

Jameson, A. (1989). Computational aerodynamics for aircraft design. *Journal of Science*, 245:361–371. 2.4, 2.4.1

Jameson, A. (2003). Aerodynamic shape optimization using the adjoint method. Lectures at the Von Karman Institute, Brussels. 2.4.1, 4, 5.1, 10.2

Jameson, A. and Alonso, J. J. (1996). Automatic aerodynamic optimization on distributed memory architectures. AIAA Paper, 96-0409. 2.4.1

Jameson, A. and Kim, S. (2003). Reduction of the adjoint gradient formula in the continuous limit. AIAA Paper,03-0040. 2.4.1

Jameson, A. and Martinelli, L. (2000). Aerodynamic shape optimization techniques based on control theory. In Capasso, V., Engl, H., and Periaux, J., editors, *Lecture Notes in Mathematics Vol. 1739.* Springer. 2.1

Jameson, A., Schmidt, W., and Turkel, E. (1981). Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes. AIAA Paper,81-1259. 3.5.1

Jang, C., Samad, A., and Kim, K. (2006). Optimal design of swept,leaned and skewed blades in a transonic axial compressor. ASME Paper, GT2006-90384. 2.3, 5.3

Keskin, A., Dutta, A., and Bestle, D. (2006). Modern compressor aerodynamic blading process using multi-objective optimization. ASME Paper, GT2006-90206. 2.3

Kim, C. (2007). Essential components for high-fidelity aerodynamic analysis and design. In *Proceedings of the 7th Asian CFD Conference*, Bangalore, India. 2.4, 2.4.1

Kim, H. and Nakahashi, K. (2005). Discrete adjoint method for unstructured Navier-Stokes solver. AIAA Paper,05-449. 1.3, 2.4, 2.4.1, 5.3

Kim, S. (2001). *Design Optimization of High-Lift Configurations Using A Viscous Adjoint-Based Method.* PhD thesis, Stanford University. 2.4, 2.4.1

Köller, U., Mönig, R., Kösters, B., and Schreiber, H. (2000). Development of advanced compressor airfoils for heavy-duty gas turbines Part I: Design and optimization. *Journal of Turbomachinery*, 122:397–405. 2.3

Küsters, B., Schreiber, H., Köller, U., and Mönig, R. (2000). Development of advanced compressor airfoils for heavy-duty gas turbines Part II: Experimental and theoretical analysis. *Journal of Turbomachinery*, 122:406–414. 2.3

Larosiliere, L., Wood, J., Hathaway, M., Medd, A., and Dang, T. (2002). Aerodynamic design study of advanced multistage axial compressor. Technical report, NASA/TP2002-211568. 2.1

Li, H. and He, L. (2003). Blade count and clocking effects on three-bladerow interaction in a transonic turbine. *Journal of Turbomachinery*, 125(4):632–640. 3.1

Li, H. and He, L. (2005). Blade aerodynamic damping variation with rotor-stator gap: a computational study using single-passage approach. *Journal of Turbomachinery*, 127:573–579. 3.1

Li, H., He, L., Li, Y., and Wells, R. (2006a). Blading aerodynamics design optimization with mechanical and aeromechanical constraints. ASME Paper,GT2006-90503. 6.2.1

Li, Y. and Wells, R. (1999). The three dimensional aerodynamic design and test of a three-stage transonic compressor. ASME Paper, GT1999-68. 2.3, 6.2.3

Li, Y., Yang, D., and Feng, Z. (2006b). Inverse problem in aerodynamic shape design of turbomachinery blades. ASME Paper,GT2006-91135. 2.4.3, 5.3

Lian, Y. and Liou, M. (2005). Multi-objective optimization of transonic compressor blade using evolutionary algorithm. *Journal of Propulsion and Power*, 21(6):979–987. 6.2.1

Lieber, L. (2003). Fluid dynamics of turbomachines. In Jr., E. L. and Roy, R., editors, *Handbook of Turbomachinery*, chapter 2. Marcel Dekker Ltd, 2 edition. 2.1

Lions, J. (1971). *Optimal control of systems governed by partial differential equations.* Springer-Verlag, Berlin. 2.4

Lotfi, O., Teixeira, J., Ivey, P., Kinghorn, I., and Sheard, A. (2006). Shape optimization of axial fan turbines using genetic algorithm and a 3D Navier-Stokes solver. ASME Paper, GT2006-90659. 2.3, 5.3

Mavriplis, D. J. (2008). Solution of the unsteady discrete adjoint for three-dimensional problems on dynamically deforming unstructured meshes. AIAA Paper 2008-0727. 2.2.1.1, 2.4.2

McMullen, M. (2003). *The Application of Nonlinear Frequency Domain Method to the Euler and Navier-Stokes Equations.* PhD thesis, Department of Aeronautics and Astronautics Stanford University. 2.2.1.2

Miller IV, P., Oliver, J., Miller, D., and Tweedt, D. (1996). BladeCAD: An interactive geometric design tool for turbomachinery blades. Technical report, NASA Technical Memorandum 107262. 5.3

Moffatt, S. and He, L. (2003). Blade forced response prediction for industrial gas turbines Part I: Methodologies. ASME Paper, GT2003-38640. 9.1, 9.3

Moffatt, S., Ning, W., Li, Y., Wells, R., and He, L. (2005). Blade forced response prediction for industrial gas turbines. *Journal of Propulsion and Power*, 21(4):707–714. 3.1, 9.3

Mohammadi, B. (1998). Shape optimization for 3D turbulent flows using automatic differentiation. *International Journal of Computational Fluid Dynamics*, 11:27–50. 2.4, 2.4.1

Mohammadi, B. and Pironneau, O. (2004). Shape optimization in fluid mechanics. *Annual Review of Fluid Mechanics*, 36:255–279. 2.4, 2.4.1

Molinari, M. and Dawes, W. (2006). Review of revolution of compressor design process and future perspectives. *Proceedings of the I MECH E Part C Journal of Mechanical Engineering Science*, 220(6):761–771. 2.1

Nadarajah, S. (2003). *The Discrete Adjoint Approach to Aerodynamic Shape Optimization.* PhD thesis, Stanford University. 2.4

Nadarajah, S. and Jameson, A. (2000). A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. AIAA Paper,00-0667. 2.4, 2.4.1, 6.1.1

Nadarajah, S. and Jameson, A. (2002). Optimal control of unsteady flows using a time accurate method. AIAA Paper,02-5436. 2.2.1.1, 2.4.2

Nadarajah, S. and Jameson, A. (2006). Optimum shape design for unsteady three dimensional viscous flows using a non-linear frequency domain method. AIAA Paper,06-3455. 2.4.2, 2.5, 8

Nielsen, E. and Anderson, W. (1999). Aerodynamic design optimization on unstructured meshes using the navier-stokes equations. *AIAA Journal*, 37(11):1411–1419. 1.3, 2.4, 2.4.1

Nielsen, E. and Kleb, B. (2005). Efficient construction of discrete adjoint operators on unstructured grids by using complex variables. AIAA Paper,05-0324. 2.4, 2.4.1

Ning, W. (1998). *Computation of unsteady flow in turbomachinery.* PhD thesis, School of Engineering, University of Durham. 2.2.1.2

Ning, W., Moffatt, S., Li, Y., and Wells, R. (2003). Blade forced response prediction for industrial gas turbines Part II: Verification and application. ASME Paper, GT2003-38642. 9.3

Oyama, A. and Liou, M. (2004). Transonic axial-flow blade optimization: Evolutionary algorithms/three-dimensional Navier-Stokes solver. *Journal of Propulsion and Power*, 20(4):612–619. 2.3, 5.2, 5.3, 6.2.1

Papadimitriou, D. and Giannakoglou, K. (2006). compressor blade optimization using a continuous adjoint formulation. ASME Paper,GT2006-90466. 2.4.3, 5.1

Pierret, S. (2005). Multi-objective and multi-disciplinary optimization of three dimensional turbomachinery blades. In *6th World Congress of Structural and Multidisciplinary Optimization*, Rio de Janeiro, Brazil. 5.2, 6.2.1

Pironneau, O. (1974). On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64(01):97–110. 2.4

Reddy, T. and Srivastava, R. (1999). Flutter and forced response analysis of cascades using a two-dimensionl linearized euler solver. Technical report, NASA/TM1999-209633. 2.2

Reuther, J., Jameson, A., Alonso, J., Remlinger, M., and Saunders, D. (1999a). Constrained multipoint aerodynamic shape optimisation using and adjoint formulation and parallel computers,part 1. *Journal of Aircraft*, 36(1):51C60. 2.4.1

Reuther, J., Jameson, A., Alonso, J., Remlinger, M., and Saunders, D. (1999b). Constrained multipoint aerodynamic shape optimisation using and adjoint formulation and parallel computers,part 2. *Journal of Aircraft*, 36(1):61–74. 2.4.1

Reuther, J., Jameson, A., Farmer, J., Martinelli, L., and Saunders, D. (1996). Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. AIAA Paper,96-0094. 2.4.1

Sadeghi, M. and Liu, F. (2005). Computation of cascade flutter by uncoupled and coupled methods. *International Journal of Computational Fluid Dynamics*, 19(8):559–569. 2.2

Samareh, J. (1999). A novel shape parameterization approach. Technical report, NASA TM-1999-209116. 5.3

Sieverding, F., Ribi, B., Casey, M., and Meyer, M. (2004). Design of industrial axial compressor blade sections for optimal range and performance. *Journal of Turbomachinery*, 126:323–331. 2.3, 5.3

Soto, O. and Löhner, R. (2000). CFD optimization using an incomplete-gradient adjoint approach. AIAA Paper,00-0666. 2.4.1

Soto, O. and Löhner, R. (2001). General methodologies for incompressible flow design problems. AIAA Paper,01-1061. 2.4.1

Soto, O. and Löhner, R. (2004). On the computation of flow sensitivities from boundary integrals. AIAA Paper,04-0112. 5.1

Spalart, P. and Allmaras, S. (1992). A one-equation turbulence model for aerodynamic flows. AIAA Paper 92-0439. 3.1, 3.2

Srivastava, R., Bakhle, M., Keith, T., and Stefko, G. (2002). Flutter analysis of a transonic fan. Technical report, NASA/TM2002-211818. 2.2

Strazisar, A., Wood, J., Hathaway, M., and Suder, K. (1989). Laser anemometer measurements in a transonic axial-flow fan rotor. Technical report, NASA Technical Paper 2879. 6.2.1

Thomas, J., Hall, K., and Dowell, E. (2003). A discrete adjoint approach for modelling unsteady aerodynamic design sensitivities. AIAA Paper,03-0041. 2.4, 2.4.4

Tsai, H., Wong, A., Cai, J., Zhu, Y., and Liu, F. (2001). Unsteady flow calculation with a parallel moving mesh algorithm. *AIAA Journal*, 39(6):1021–1029. 1.3

van Rooij, M., Dang, T., and Larosiliere, L. (2007). Improving aerodynamic matching of axial compressor blading using a three-dimensional multistage inverse design method. *Journal of Turbomachinery*, 129:108–118. 2.3

Vatsa, V. (1999). Computation of sensitivity derivatives of Navier-Stokes equations using complex variables. NASA Langley Research Center, Hampton, VA 23681. 2.3, 4.1, 4.3, 5.1, C

Wang, Z. (1999). Three-dimensional theory and design method of bowed-twisted blade and its application to turbomachines. Lecture Series 1999-02 at the von Karman Institute for Fluid Dynamics, Brussels. 2.3

Whitehead, D. (1987). Classical two-dimensional methods. Technical report, AGARD Manual on Aeroelasticity in Axial Flow Turbomachines, Unsteady Turbomachinery Aerodynamics, Vol. 1, AGARD-AG-298. 7.3.1

Wu, H., Liu, F., and Tsai, H. (2005a). Aerodynamic design of turbine blades using an adjoint equation method. AIAA Paper,05-1006. 2.4.3, 5.2

Wu, H., Yang, S., and Liu, F. (2003). Comparison of three geometric representation of airfoils for aerodynamic optimization. AIAA Paper,2003-4095. 1.3, 2.4.3, 5.3

Wu, X., Vahdati, M., Sayma, A., and Imregun, M. (2005b). Whole-annulus aeroelasticity analysis of a 17-bladerow wrf compressor using an unstructured navier-stokes solver. *International Journal of Computational Fluid Dynamics*, 19(3):211–223. 2.2

Yang, H. (2004). *3D Unsteady Flow in Oscillating Compressor Cascade.* PhD thesis, School of Engineering, University of Durham. 7.3.2, 7.3.2

Yang, H. and He, L. (2003). Experiment on linear compressor cascade with 3-D blade oscillation. ASME Paper, GT2003-38484. 7.3.2, 7.3.2

Yang, S., Wu, H., Liu, F., and Tsai, H. (2003). Aerodynamic design of cascades by using an adjoint equation method. AIAA Paper,03-1068. 2.4.3, 5.3, 6.1.1

Yi, W., Huang, H., and Han, W. (2006). Design optimization of transonic compressor rotor using CFD and genetic algorithm. ASME Paper, GT2006-90155. 2.3, 5.3

Appendix A

Adjoint Navier-Stokes Equations

The derivation of the adjoint Navier-Stokes equations based upon the RANS equations is given in this appendix.

Linearizing the RANS equations 3.1 with respect to a design variable, one has

$$\begin{aligned} &\frac{\partial\left(A\tilde{U}-A_v\tilde{U}-D_{xx}\tilde{U}_x-D_{x\theta}\tilde{U}_\theta-D_{xr}\tilde{U}_r\right)}{\partial x}+ \\ &\frac{\partial\left(B\tilde{U}-B_v\tilde{U}-D_{\theta x}\tilde{U}_x-D_{\theta\theta}\tilde{U}_\theta-D_{\theta r}\tilde{U}_r\right)}{r\partial\theta}+ \\ &\frac{\partial r\left(C\tilde{U}-C_v\tilde{U}-D_{rx}\tilde{U}_x-D_{r\theta}\tilde{U}_\theta-D_{rr}\tilde{U}_r\right)}{r\partial r}-D\tilde{U}=f \end{aligned} \tag{A.1}$$

where matrices A, B, C, D are defined as in Eq.4.14, and

$$A_v=\frac{\partial V_x}{\partial U}, B_v=\frac{\partial V_\theta}{\partial U}, C_v=\frac{\partial V_r}{\partial U}, \tilde{U}_x=\frac{\partial\tilde{U}}{\partial x}$$

$$\tilde{U}_\theta=\frac{\partial\tilde{U}}{r\partial\theta}, \tilde{U}_r=\frac{\partial r\tilde{U}}{r\partial r}, D_{ij}=\frac{\partial V_i}{\partial U_j}\ i,j=x,\theta,r$$

Multiplying the above equation with the adjoint variable and rearranging yields

$$\begin{aligned} \lambda^T\Bigg[&\frac{\partial\left(A\tilde{U}-A_v\tilde{U}-D_{xx}\tilde{U}_x-D_{x\theta}\tilde{U}_\theta-D_{xr}\tilde{U}_r\right)}{\partial x}+ \\ &\frac{\partial\left(B\tilde{U}-B_v\tilde{U}-D_{\theta x}\tilde{U}_x-D_{\theta\theta}\tilde{U}_\theta-D_{\theta r}\tilde{U}_r\right)}{r\partial\theta}+ \\ &\frac{\partial r\left(C\tilde{U}-C_v\tilde{U}-D_{rx}\tilde{U}_x-D_{r\theta}\tilde{U}_\theta-D_{rr}\tilde{U}_r\right)}{r\partial r}-D\tilde{U}-f\Bigg]=0 \end{aligned} \tag{A.2}$$

The product holds no matter what value the adjoint variable will take. Integrating the left hand side of the above equation over the whole computational domain, one has

$$\int_D \lambda^T \left[\frac{\partial\left(A\tilde{U} - A_v\tilde{U} - D_{xx}\tilde{U}_x - D_{x\theta}\tilde{U}_\theta - D_{xr}\tilde{U}_r\right)}{\partial x} + \right. \tag{A.3}$$
$$\frac{\partial\left(B\tilde{U} - B_v\tilde{U} - D_{\theta x}\tilde{U}_x - D_{\theta\theta}\tilde{U}_\theta - D_{\theta r}\tilde{U}_r\right)}{r\partial\theta} +$$
$$\left. \frac{\partial r\left(C\tilde{U} - C_v\tilde{U} - D_{rx}\tilde{U}_x - D_{r\theta}\tilde{U}_\theta - D_{rr}\tilde{U}_r\right)}{r\partial r} - D\tilde{U} - f \right] dv$$

Performing integration by parts once gives

$$\int_{\partial D} \left[\lambda^T\left(\tilde{F} - \tilde{V}_x\right)n_x + \lambda^T\left(\tilde{G} - \tilde{U}v_g - \tilde{V}_\theta\right)n_\theta + \lambda^T\left(\tilde{H} - \tilde{V}_r\right)n_r\right]ds \tag{A.4}$$
$$- \int_D \left[\frac{\partial\lambda^T}{\partial x}(A - A_v) + \frac{\partial\lambda^T}{r\partial\theta}(B - v_g I - B_v) + \frac{\partial\lambda^T}{\partial r}(C - C_v) + \lambda^T D\right]\tilde{U}dv$$
$$+ \int_D \left[\frac{\partial\lambda^T}{\partial x}\left(D_{xx}\tilde{U}_x + D_{x\theta}\tilde{U}_\theta + D_{xr}\tilde{U}_r\right) + \frac{\partial\lambda^T}{r\partial\theta}\left(D_{\theta x}\tilde{U}_x + D_{\theta\theta}\tilde{U}_\theta + D_{\theta r}\tilde{U}_r\right)\right.$$
$$\left. + \frac{\partial\lambda^T}{\partial r}\left(D_{rx}\tilde{U}_x + D_{r\theta}\tilde{U}_\theta + D_{rr}\tilde{U}_r\right)\right]dv$$
$$- \int_D \lambda^T f dv$$

Compared with expression 4.17, it can be seen that the above expression has extra terms attributable to viscous terms in the RANS equations. The last second domain integral in the above contains spatial derivatives of flow variable sensitivity, further integration by parts needs to be used to release the flow variable sensitivity out of their spatial derivative operators. Rearranging the last second domain integral in the above by collecting like terms with $\tilde{U}_x, \tilde{U}_\theta$ and $\tilde{U}_r$ yields

$$\int_D \left[\left(\frac{\partial\lambda^T}{\partial x}D_{xx} + \frac{\partial\lambda^T}{r\partial\theta}D_{\theta x} + \frac{\partial\lambda^T}{\partial r}D_{rx}\right)\tilde{U}_x + \right. \tag{A.5}$$
$$\left(\frac{\partial\lambda^T}{\partial x}D_{x\theta} + \frac{\partial\lambda^T}{r\partial\theta}D_{\theta\theta} + \frac{\partial\lambda^T}{\partial r}D_{r\theta}\right)\tilde{U}_\theta +$$
$$\left.\left(\frac{\partial\lambda^T}{\partial x}D_{xr} + \frac{\partial\lambda^T}{r\partial\theta}D_{\theta r} + \frac{\partial\lambda^T}{\partial r}D_{rr}\right)\tilde{U}_r\right]dv$$

Performing integration by parts to the above once gives

$$\begin{aligned}\int_{\partial D}\Bigg[&\left(\frac{\partial\lambda^T}{\partial x}D_{xx}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta x}+\frac{\partial\lambda^T}{\partial r}D_{rx}\right)\tilde{U}\cdot n_x+\\&\left(\frac{\partial\lambda^T}{\partial x}D_{x\theta}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta\theta}+\frac{\partial\lambda^T}{\partial r}D_{r\theta}\right)\tilde{U}\cdot n_\theta+\\&\left(\frac{\partial\lambda^T}{\partial x}D_{xr}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta r}+\frac{\partial\lambda^T}{\partial r}D_{rr}\right)\tilde{U}\cdot n_r\Bigg]\,ds\\-\int_D\Bigg[&\frac{\partial}{\partial x}\left(\frac{\partial\lambda^T}{\partial x}D_{xx}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta x}+\frac{\partial\lambda^T}{\partial r}D_{rx}\right)+\\&\frac{\partial}{r\partial\theta}\left(\frac{\partial\lambda^T}{\partial x}D_{x\theta}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta\theta}+\frac{\partial\lambda^T}{\partial r}D_{r\theta}\right)+\\&\frac{\partial}{\partial r}\left(\frac{\partial\lambda^T}{\partial x}D_{xr}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta r}+\frac{\partial\lambda^T}{\partial r}D_{rr}\right)\Bigg]\tilde{U}dv\end{aligned}\tag{A.6}$$

Substituting A.6 into A.4 and collecting like terms yields

$$\begin{aligned}\int_{\partial D}\Bigg[&\lambda^T\left(\tilde{F}-\tilde{V}_x\right)n_x+\lambda^T\left(\tilde{G}-\tilde{U}v_g-\tilde{V}_\theta\right)n_\theta+\lambda^T\left(\tilde{H}-\tilde{V}_r\right)n_r\\&+\left(\frac{\partial\lambda^T}{\partial x}D_{xx}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta x}+\frac{\partial\lambda^T}{\partial r}D_{rx}\right)\tilde{U}\cdot n_x\\&+\left(\frac{\partial\lambda^T}{\partial x}D_{x\theta}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta\theta}+\frac{\partial\lambda^T}{\partial r}D_{r\theta}\right)\tilde{U}\cdot n_\theta\\&+\left(\frac{\partial\lambda^T}{\partial x}D_{xr}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta r}+\frac{\partial\lambda^T}{\partial r}D_{rr}\right)\tilde{U}\cdot n_r\Bigg]\,ds\\-\int_D\Bigg[&\frac{\partial\lambda^T}{\partial x}(A-A_v)+\frac{\partial\lambda^T}{r\partial\theta}(B-v_gI-B_v)+\frac{\partial\lambda^T}{\partial r}(C-C_v)+\lambda^TD\\&+\frac{\partial}{\partial x}\left(\frac{\partial\lambda^T}{\partial x}D_{xx}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta x}+\frac{\partial\lambda^T}{\partial r}D_{rx}\right)\\&+\frac{\partial}{r\partial\theta}\left(\frac{\partial\lambda^T}{\partial x}D_{x\theta}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta\theta}+\frac{\partial\lambda^T}{\partial r}D_{r\theta}\right)\\&+\frac{\partial}{\partial r}\left(\frac{\partial\lambda^T}{\partial x}D_{xr}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta r}+\frac{\partial\lambda^T}{\partial r}D_{rr}\right)\Bigg]\tilde{U}dv\\&-\int_D\lambda^Tf\,dv\end{aligned}\tag{A.7}$$

Subtracting A.7 from the gradient expression in 4.13 and grouping domain and bound-

ary integral terms with $\tilde{U}$ separately, one has

$$\int_D \left[\frac{\partial \lambda^T}{\partial x}(A - A_v) + \frac{\partial \lambda^T}{r\partial\theta}(B - v_g I - B_v) + \frac{\partial \lambda^T}{\partial r}(C - C_v) + \lambda^T D \right. \tag{A.8}$$
$$+\frac{\partial}{\partial x}\left(\frac{\partial \lambda^T}{\partial x}D_{xx} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta x} + \frac{\partial \lambda^T}{\partial r}D_{rx}\right)$$
$$+\frac{\partial}{r\partial\theta}\left(\frac{\partial \lambda^T}{\partial x}D_{x\theta} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r}D_{r\theta}\right)$$
$$\left. +\frac{\partial}{\partial r}\left(\frac{\partial \lambda^T}{\partial x}D_{xr} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta r} + \frac{\partial \lambda^T}{\partial r}D_{rr}\right)\right] \tilde{U} dv$$

$$-\int_{\partial D} \left[\lambda^T \left(\tilde{F} - \tilde{V}_x\right) n_x + \lambda^T \left(\tilde{G} - \tilde{U} v_g - \tilde{V}_\theta\right) n_\theta + \lambda^T \left(\tilde{H} - \tilde{V}_r\right) n_r \right. \tag{A.9}$$
$$+\left(\frac{\partial \lambda^T}{\partial x}D_{xx} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta x} + \frac{\partial \lambda^T}{\partial r}D_{rx}\right)\tilde{U}\cdot n_x$$
$$+\left(\frac{\partial \lambda^T}{\partial x}D_{x\theta} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r}D_{r\theta}\right)\tilde{U}\cdot n_\theta$$
$$\left. +\left(\frac{\partial \lambda^T}{\partial x}D_{xr} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta r} + \frac{\partial \lambda^T}{\partial r}D_{rr}\right)\tilde{U}\cdot n_r - \frac{\partial M}{\partial U}\tilde{U}\right] ds$$

The remaining terms form the final gradient expression, which does not depend on flow variable sensitivity any more as intended and assumes the form in 4.23. The adjoint formulation requires both the domain integral A.8 and the boundary integral A.9 vanish. Anywhere inside a domain, the flow variable sensitivity there can not always be zero. The sufficient and necessary condition to vanish the above domain integral is to choose the adjoint variable in such a way that the term multiplying $\tilde{U}$ in the domain integral vanishes, leading to the adjoint Navier-Stokes equations:

$$\frac{\partial \lambda^T}{\partial x}(A - A_v) + \frac{\partial \lambda^T}{r\partial\theta}(B - v_g I - B_v) + \frac{\partial \lambda^T}{\partial r}(C - C_v) + \lambda^T D \tag{A.10}$$
$$+\frac{\partial}{\partial x}\left(\frac{\partial \lambda^T}{\partial x}D_{xx} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta x} + \frac{\partial \lambda^T}{\partial r}D_{rx}\right)$$
$$+\frac{\partial}{r\partial\theta}\left(\frac{\partial \lambda^T}{\partial x}D_{x\theta} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r}D_{r\theta}\right)$$
$$+\frac{\partial}{\partial r}\left(\frac{\partial \lambda^T}{\partial x}D_{xr} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta r} + \frac{\partial \lambda^T}{\partial r}D_{rr}\right) = 0$$

The above equation can be rearranged as follows:

$$(A - A_v)^T \frac{\partial \lambda}{\partial x} + (B - v_g I - B_v)^T \frac{\partial \lambda}{r\partial\theta} + (C - C_v)^T \frac{\partial \lambda}{\partial r} \tag{A.11}$$
$$+M^T\left(\frac{\partial D_x}{\partial x} + \frac{\partial D_\theta}{r\partial\theta} + \frac{\partial D_r}{\partial r}\right) + D^T\lambda = 0$$

$$
\begin{aligned}
M &= \frac{\partial U}{\partial U_v}, U_v = (p, u, v, w, T)^T \\
D_x &= (0, \sigma_{xx}, \sigma_{\theta x}/r, \sigma_{rx}, \phi_x)^T \\
D_\theta &= (0, \sigma_{x\theta}, \sigma_{\theta\theta}/r, \sigma_{r\theta}, \phi_\theta)^T \\
D_r &= (0, \sigma_{xr}, \sigma_{\theta r}/r, \sigma_{rr}, \phi_r)^T \\
\sigma_{xx} &= \frac{2}{3}\mu\left[2\left(\frac{\partial\lambda_2}{\partial x} + u\frac{\partial\lambda_5}{\partial x}\right) - \left(\frac{\partial\lambda_3}{\partial\theta} + v\frac{\partial\lambda_5}{r\partial\theta}\right) - \left(\frac{\partial\lambda_4}{\partial r} + w\frac{\partial\lambda_5}{\partial r}\right)\right] \\
\sigma_{\theta\theta} &= \frac{2}{3}\mu\left[2\left(\frac{\partial\lambda_3}{\partial\theta} + v\frac{\partial\lambda_5}{r\partial\theta}\right) - \left(\frac{\partial\lambda_2}{\partial x} + u\frac{\partial\lambda_5}{\partial x}\right) - \left(\frac{\partial\lambda_4}{\partial r} + w\frac{\partial\lambda_5}{\partial r}\right)\right] \\
\sigma_{rr} &= \frac{2}{3}\mu\left[2\left(\frac{\partial\lambda_4}{\partial r} + w\frac{\partial\lambda_5}{\partial r}\right) - \left(\frac{\partial\lambda_3}{\partial\theta} + v\frac{\partial\lambda_5}{\partial\theta}\right) - \left(\frac{\partial\lambda_2}{\partial x} + u\frac{\partial\lambda_5}{\partial x}\right)\right] \\
\sigma_{x\theta} &= \sigma_{\theta x} = \mu\left[\left(\frac{\partial\lambda_2}{r\partial\theta} + u\frac{\partial\lambda_5}{r\partial\theta}\right) + \left(r\frac{\partial\lambda_3}{\partial x} + v\frac{\partial\lambda_5}{\partial x}\right)\right] \\
\sigma_{xr} &= \sigma_{rx} = \mu\left[\left(\frac{\partial\lambda_2}{\partial r} + u\frac{\partial\lambda_5}{\partial r}\right) + \left(\frac{\partial\lambda_4}{\partial x} + w\frac{\partial\lambda_5}{\partial x}\right)\right] \\
\sigma_{r\theta} &= \sigma_{\theta r} = \mu\left[\left(\frac{\partial\lambda_4}{r\partial\theta} + w\frac{\partial\lambda_5}{r\partial\theta}\right) + \left(r\frac{\partial\lambda_3}{\partial r} + v\frac{\partial\lambda_5}{\partial r}\right)\right] \\
\phi_x &= k\frac{\partial\lambda_5}{\partial x}, \phi_\theta = k\frac{\partial\lambda_5}{r\partial\theta}, \phi_r = k\frac{\partial\lambda_5}{\partial r}
\end{aligned}
$$

Vanishing the boundary integral A.9 requires the corresponding flow boundary conditions at a boundary be taken into account of to derive the sufficient and necessary boundary conditions for the adjoint equations. At an inlet or exit boundary, it is assumed that viscous terms in A.9 are negligible, implying the boundary conditions at an inlet or exit boundary for the adjoint Navier-Stokes equations are the same as those for the adjoint Euler equations. However at a viscous solid wall boundary where the viscous effect is quite significant, the viscous terms in the boundary integral A.9 can not be neglected. Removing the dependence of the boundary integral on the flow variable sensitivity at a viscous solid wall boundary is detailed in Appendix B.

Appendix B

Adjoint Viscous Solid Wall Boundary Condition

This appendix illustrates the derivation of the boundary condition at a viscous solid wall for the adjoint Navier-Stokes equations. The boundary condition is the sufficient and necessary condition eliminating the dependence of the boundary integral A.9 on the flow variable sensitivity at a viscous solid wall boundary. The boundary integral is repeated in the following for an easier reference:

$$
\begin{aligned}
-\int_{\partial D}\Big[&\lambda^T\left(\tilde{F}-\tilde{V}_x\right)n_x+\lambda^T\left(\tilde{G}-\tilde{U}v_g-\tilde{V}_\theta\right)n_\theta+\lambda^T\left(\tilde{H}-\tilde{V}_r\right)n_r \\
&+\left(\frac{\partial\lambda^T}{\partial x}D_{xx}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta x}+\frac{\partial\lambda^T}{\partial r}D_{rx}\right)\tilde{U}\cdot n_x \\
&+\left(\frac{\partial\lambda^T}{\partial x}D_{x\theta}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta\theta}+\frac{\partial\lambda^T}{\partial r}D_{r\theta}\right)\tilde{U}\cdot n_\theta \\
&+\left(\frac{\partial\lambda^T}{\partial x}D_{xr}+\frac{\partial\lambda^T}{r\partial\theta}D_{\theta r}+\frac{\partial\lambda^T}{\partial r}D_{rr}\right)\tilde{U}\cdot n_r-\frac{\partial M}{\partial U}\tilde{U}\Big]\,ds
\end{aligned}
\tag{B.1}
$$

Split the boundary integral into the following two parts: the inviscid part and the viscous part.

$$
-\int_{\partial D}\left[\lambda^T\tilde{F}n_x+\lambda^T\left(\tilde{G}-\tilde{U}v_g\right)n_\theta+\lambda^T\tilde{H}n_r-\frac{\partial M}{\partial U}\tilde{U}\right]ds \tag{B.2}
$$

$$
\begin{aligned}
&- \int_{\partial D} \Big[\lambda^T \left(\tilde{V}_x n_x + \tilde{V}_\theta n_\theta + \tilde{V}_r n_r \right) \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) n_x \cdot \tilde{U} \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) n_\theta \cdot \tilde{U} \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) n_r \cdot \tilde{U} \Big] \, ds
\end{aligned}
\tag{B.3}
$$

The boundary integral B.1 can be eliminated by vanishing the above two parts separately. The first part B.2 can be eliminated as explained in the boundary condition section in Chapter 4. The following text is aimed at illustrating the elimination of the second part B.3.

$V_x n_x + V_\theta n_\theta + V_r n_r$ is the viscous flux vector of the flow equations. At a viscous solid wall with the log-law for a shear stress calculation, this flux vector is approximated by

$$
V_x n_x + V_\theta n_\theta + V_r n_r = \begin{pmatrix} 0 \\ \tau_x \\ \tau_\theta \cdot r \\ \tau_r \\ \tau_\theta \Omega r + k \frac{\partial T}{\partial n} \end{pmatrix}
$$

where τ_x, τ_θ and τ_r are the components of the wall shear stress τ_w in the x, θ and r directions. These components are calculated as follows with the wall shear stress computed using the log-law as explained in the boundary section in Chapter 3:

$$
\begin{aligned}
\tau_x &= \tau_w \cdot \frac{u}{W} \\
\tau_\theta &= \tau_w \cdot \frac{v_{rel}}{W} \\
\tau_r &= \tau_w \cdot \frac{w}{W}
\end{aligned}
\tag{B.4}
$$

where v_{rel} is the relative velocity in the circumferential direction defined as $v - \Omega r$. The wall shear stress τ_w is a function of u, v, w and ρ at a viscous solid wall boundary, consequently τ_x, τ_θ and τ_r are also functions of u, v, w and ρ. The sensitivity of the

stress terms[1] in terms of the flow variable sensitivity can be obtained by the chain rule:

$$\begin{aligned}
\widetilde{\tau_x} &= \frac{\partial \tau_x}{\partial u}\tilde{u} + \frac{\partial \tau_x}{\partial v}\tilde{v} + \frac{\partial \tau_x}{\partial w}\tilde{w} + \frac{\partial \tau_x}{\partial \rho}\tilde{\rho} \\
\widetilde{\tau_\theta} &= \frac{\partial \tau_\theta}{\partial u}\tilde{u} + \frac{\partial \tau_\theta}{\partial v}\tilde{v} + \frac{\partial \tau_\theta}{\partial w}\tilde{w} + \frac{\partial \tau_\theta}{\partial \rho}\tilde{\rho} \\
\widetilde{\tau_r} &= \frac{\partial \tau_r}{\partial u}\tilde{u} + \frac{\partial \tau_r}{\partial v}\tilde{v} + \frac{\partial \tau_r}{\partial w}\tilde{w} + \frac{\partial \tau_r}{\partial \rho}\tilde{\rho}
\end{aligned} \tag{B.5}$$

With the above stress terms substituted, the term $\lambda^T\left(\tilde{V}_x n_x + \tilde{V}_\theta n_\theta + \tilde{V}_r n_r\right)$ can be expanded as

$$\begin{aligned}
\lambda^T\left(\tilde{V}_x n_x + \tilde{V}_\theta n_\theta + \tilde{V}_r n_r\right) &= \lambda_2 \cdot \widetilde{\tau_x} + \lambda_3 \cdot \widetilde{\tau_\theta} r + \lambda_4 \cdot \widetilde{\tau_r} + \lambda_5 \cdot (\widetilde{\tau_\theta}\Omega r + k\frac{\partial \tilde{T}}{\partial n}) \\
&= \left[\lambda_2 \frac{\partial \tau_x}{\partial u} + (\lambda_3 + \lambda_5\Omega) r \frac{\partial \tau_\theta}{\partial u} + \lambda_4 \frac{\partial \tau_r}{\partial u}\right]\tilde{u} \\
&+ \left[\lambda_2 \frac{\partial \tau_x}{\partial v} + (\lambda_3 + \lambda_5\Omega) r \frac{\partial \tau_\theta}{\partial v} + \lambda_4 \frac{\partial \tau_r}{\partial v}\right]\tilde{v} \\
&+ \left[\lambda_2 \frac{\partial \tau_x}{\partial w} + (\lambda_3 + \lambda_5\Omega) r \frac{\partial \tau_\theta}{\partial w} + \lambda_4 \frac{\partial \tau_r}{\partial w}\right]\tilde{w} \\
&+ \left[\lambda_2 \frac{\partial \tau_x}{\partial \rho} + (\lambda_3 + \lambda_5\Omega) r \frac{\partial \tau_\theta}{\partial \rho} + \lambda_4 \frac{\partial \tau_r}{\partial \rho}\right]\tilde{\rho} \\
&+ \lambda_5 k \frac{\partial \tilde{T}}{\partial n}
\end{aligned} \tag{B.6}$$

The spatial derivative terms in B.3 can also be written in terms of the viscous flow variable sensitivity:

$$\begin{aligned}
&\left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx}\right) n_x \cdot \tilde{U} \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta}\right) n_\theta \cdot \tilde{U} \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr}\right) n_r \cdot \tilde{U} \\
&= \left(\frac{\partial \lambda^T}{\partial x} D^v_{xx} + \frac{\partial \lambda^T}{r\partial \theta} D^v_{\theta x} + \frac{\partial \lambda^T}{\partial r} D^v_{rx}\right) n_x \cdot \tilde{U}_v \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D^v_{x\theta} + \frac{\partial \lambda^T}{r\partial \theta} D^v_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D^v_{r\theta}\right) n_\theta \cdot \tilde{U}_v \\
&+ \left(\frac{\partial \lambda^T}{\partial x} D^v_{xr} + \frac{\partial \lambda^T}{r\partial \theta} D^v_{\theta r} + \frac{\partial \lambda^T}{\partial r} D^v_{rr}\right) n_r \cdot \tilde{U}_v
\end{aligned} \tag{B.7}$$

[1] to be precise, this sensitivity is a partial sensitivity considering the sensitivity due to the flow variable sensitivity. The adjoint formulation needs to eliminate the flow variable sensitivity appearing in the boundary integral. The geometric sensitivity does not need eliminating. For the sake of simplicity, the geometric sensitivity is not considered here.

where U_v is the viscous flow variable vector defined as $(\rho, u, v, w, T)^T$. The right hand side of the above equation can be written in the following shorthand:

$$X_1 \cdot \tilde{\rho} + X_2 \cdot \tilde{u} + X_3 \cdot r \cdot \tilde{v} + X_4 \cdot \tilde{w} + X_5 \cdot \tilde{T} \tag{B.8}$$

As it is well-known, viscous stress terms are spatial derivatives of velocity components and temperature. Accordingly the term multiplying $\tilde{\rho}$ must be zero:

$$X_1 = 0$$

Substituting B.6 and B.8 into B.3, collecting like terms with flow variable sensitivities and rearranging, one can have the following equations by vanishing the term multiplying $\tilde{\rho}$, $\tilde{u}$, $\tilde{v}$ and $\tilde{w}$ respectively:

$$\begin{aligned}
\lambda_2\frac{\partial \tau_x}{\partial \rho} + (\lambda_3 + \lambda_5\Omega)r\frac{\partial \tau_\theta}{\partial \rho} + \lambda_4\frac{\partial \tau_r}{\partial \rho} &= 0 \\
\lambda_2\frac{\partial \tau_x}{\partial u} + (\lambda_3 + \lambda_5\Omega)r\frac{\partial \tau_\theta}{\partial u} + \lambda_4\frac{\partial \tau_r}{\partial u} &= X_2 \\
\lambda_2\frac{\partial \tau_x}{\partial v} + (\lambda_3 + \lambda_5\Omega)r\frac{\partial \tau_\theta}{\partial v} + \lambda_4\frac{\partial \tau_r}{\partial v} &= X_3 r \\
\lambda_2\frac{\partial \tau_x}{\partial w} + (\lambda_3 + \lambda_5\Omega)r\frac{\partial \tau_\theta}{\partial w} + \lambda_4\frac{\partial \tau_r}{\partial w} &= X_4
\end{aligned} \tag{B.9}$$

Denote $\frac{\tau_w}{W}$ by f, the shear stresses can be rewritten as follows:

$$\tau_x = fu,\ \tau_\theta = fv_{rel},\ \tau_r = fw \tag{B.10}$$

The sensitivities of the stress terms can be written as follows:

$$\begin{array}{llll}
\frac{\partial \tau_x}{\partial u} = \frac{\partial f}{\partial u}u + f, & \frac{\partial \tau_x}{\partial u} = \frac{\partial f}{\partial u}u + f, & \frac{\partial \tau_\theta}{\partial u} = \frac{\partial f}{\partial u}v_{rel}, & \frac{\partial \tau_r}{\partial u} = \frac{\partial f}{\partial u}w \\
\frac{\partial \tau_x}{\partial v} = \frac{\partial f}{\partial v}u, & \frac{\partial \tau_x}{\partial v} = \frac{\partial f}{\partial v}u, & \frac{\partial \tau_\theta}{\partial v} = \frac{\partial f}{\partial v}v_{rel} + f, & \frac{\partial \tau_r}{\partial v} = \frac{\partial f}{\partial v}w \\
\frac{\partial \tau_x}{\partial w} = \frac{\partial f}{\partial w}u, & \frac{\partial \tau_x}{\partial w} = \frac{\partial f}{\partial w}u, & \frac{\partial \tau_\theta}{\partial w} = \frac{\partial f}{\partial w}v_{rel}, & \frac{\partial \tau_r}{\partial w} = \frac{\partial f}{\partial w}w + f \\
\frac{\partial \tau_x}{\partial \rho} = \frac{\partial f}{\partial \rho}u, & \frac{\partial \tau_x}{\partial \rho} = \frac{\partial f}{\partial \rho}u, & \frac{\partial \tau_\theta}{\partial \rho} = \frac{\partial f}{\partial \rho}v_{rel}, & \frac{\partial \tau_r}{\partial \rho} = \frac{\partial f}{\partial \rho}w
\end{array}$$

With the above sensitivity expressions substituted into Eq.B.9, one has

$$\begin{aligned}
\lambda_2\frac{\partial f}{\partial \rho}u + (\lambda_3 + \lambda_5\Omega)r\frac{\partial f}{\partial \rho}v_{rel} + \lambda_4\frac{\partial f}{\partial \rho}w &= 0 \\
\lambda_2(\frac{\partial f}{\partial u}u + f) + (\lambda_3 + \lambda_5\Omega)r\frac{\partial f}{\partial u}v_{rel} + \lambda_4\frac{\partial f}{\partial u}w &= X_2 \\
\lambda_2\frac{\partial f}{\partial v}u + (\lambda_3 + \lambda_5\Omega)r(\frac{\partial f}{\partial v}v_{rel} + f) + \lambda_4\frac{\partial f}{\partial v}w &= X_3 r \\
\lambda_2\frac{\partial f}{\partial w}u + (\lambda_3 + \lambda_5\Omega)r\frac{\partial f}{\partial w}v_{rel} + \lambda_4(\frac{\partial f}{\partial w} + f) &= X_4
\end{aligned} \tag{B.11}$$

Further arrangement leads to

$$\begin{aligned} X_2 &= \lambda_2 f \\ X_3 &= (\lambda_3 + \lambda_5 \Omega) f \\ X_4 &= \lambda_4 f \end{aligned} \tag{B.12}$$

where X_2, X_3 and X_5 are σ_x, σ_θ and σ_r respectively as defined in the boundary section in Chapter 4. The detailed expression of X_5 is given by

$$X_5 = k \frac{\partial \lambda_5}{\partial n}$$

If a viscous solid wall is adiabatic, then $\frac{\partial T}{\partial n} = 0$ and $\tilde{T} \neq 0$, the following condition about λ_5 needs applying at a viscous solid wall boundary:

$$\frac{\partial \lambda_5}{\partial n} = 0$$

If temperature is specified and fixed at any point along a viscous solid wall, then $\tilde{T} = 0$ and $\frac{\partial T}{\partial n} \neq 0$, the corresponding boundary condition about λ_5 is applied by

$$\lambda_5 = 0$$

Appendix C

Objective Function Gradients

As it reveals in section 4.3, the gradients of the integrand of the objective function to some flow variables are required to specify adjoint boundary conditions. This appendix presents these analytic expressions of these gradients.

The substantial gradient of the objective function (expression 5.3) with respect to a design variable in general can be written as

$$\frac{dI}{d\alpha} = \frac{1}{\Delta\bar{s}_0}\frac{d\Delta\bar{s}}{d\alpha} + 2\sigma_1(\frac{\dot{m}}{\dot{m}_0} - 1)\frac{1}{\dot{m}_0}\frac{d\dot{m}}{d\alpha} + 2\sigma_2(\frac{\pi}{\pi_0} - 1)\frac{1}{\pi_0}\frac{d\pi}{d\alpha} \tag{C.1}$$

Entropy generation rate, mass flow rate and pressure ratio as defined in section 5.2 are integrals at the inlet and exit planes of a computational domain. For a clear presentation here, the gradient term $\frac{\partial M}{\partial U}$ is decomposed into the sum of three ones:

$$\frac{\partial M}{\partial U} = \frac{\partial M_1}{\partial U} + \frac{\partial M_2}{\partial U} + \frac{\partial M_3}{\partial U} \tag{C.2}$$

with $\frac{\partial M_1}{\partial U}$, $\frac{\partial M_2}{\partial U}$ and $\frac{\partial M_3}{\partial U}$ corresponding to the entropy generation rate, mass flow rate and pressure ratio respectively.

Referring to section 4.3, at an inlet or solid wall boundary, $\frac{\partial M}{\partial p}$ is required; at an exit boundary, $\frac{\partial M}{\partial \rho}$, $\frac{\partial M}{\partial u}$, $\frac{\partial M}{\partial v}$ and $\frac{\partial M}{\partial w}$ are required. It is noted that the objective function in 5.3 does not contain integrals at a solid wall boundary. Therefore we have $\frac{\partial M}{\partial p} = 0$ at a solid wall boundary. Those derivatives at an exit or inlet are given in detail in the following.

Entropy Generation Rate Gradient

The entropy generation rate gradient can be expanded as follows

$$\frac{d\Delta\bar{s}}{d\alpha} = \frac{d\bar{s}_e}{d\alpha} - \frac{d\bar{s}_i}{d\alpha}$$

At an inlet, the derivative with respect to static pressure is given by

$$\frac{\partial M_1}{\partial p} = -\frac{1}{\Delta s_0}\frac{1}{m}\left[\left(\frac{1}{\rho^r} - \frac{s\gamma}{\rho}\frac{d\rho}{dp}\right)\rho v_n + (s-\bar{s})\frac{d\rho v_n}{dp}\right] \tag{C.3}$$

where $m = \int_i \rho v_n ds$, $\frac{d\rho v_n}{dp}$ and $\frac{d\rho}{dp}$ are calculated by the complex variable method (Vatsa, 1999).

At an exit, the derivatives are given by

$$\begin{aligned}
\frac{\partial M_1}{\partial \rho} &= \frac{1}{\Delta s_0}\frac{1}{m}\left[\left(-\frac{s\gamma}{\rho}\right)\rho v_n + (s-\bar{s})\, v_n\right] \qquad \text{(C.4)}\\
\frac{\partial M_1}{\partial u} &= \frac{1}{\Delta s_0}\frac{1}{m}\left[(s-\bar{s})\,\rho n_x\right]\\
\frac{\partial M_1}{\partial v} &= \frac{1}{\Delta s_0}\frac{1}{m}\left[(s-\bar{s})\,\rho n_\theta\right]\\
\frac{\partial M_1}{\partial w} &= \frac{1}{\Delta s_0}\frac{1}{m}\left[(s-\bar{s})\,\rho n_r\right]
\end{aligned}$$

where $m = \int_e \rho v_n ds$.

Mass Flow Rate Gradient

The mass flow rate gradient can be broken down by

$$\frac{d\dot{m}}{d\alpha} = \frac{1}{2}\left(\int_i \frac{d\rho v_n}{d\alpha} ds \cdot N_{b1} + \int_e \frac{d\rho v_n}{d\alpha} ds \cdot N_{bn}\right)$$

At an inlet, the derivative is given by

$$\frac{\partial M_2}{\partial p} = 2\sigma\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0} \cdot 0.5\frac{\partial \rho v_n}{\partial p} N_{b1} \tag{C.5}$$

At an exit, the derivatives are given by

$$\begin{aligned}
\frac{\partial M_2}{\partial \rho} &= 2\sigma\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0} \cdot 0.5 v_n N_{bn} \qquad \text{(C.6)}\\
\frac{\partial M_2}{\partial u} &= 2\sigma\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0} \cdot 0.5\rho n_x N_{bn}\\
\frac{\partial M_2}{\partial v} &= 2\sigma\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0} \cdot 0.5\rho n_\theta N_{bn}\\
\frac{\partial M_2}{\partial w} &= 2\sigma\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0} \cdot 0.5\rho n_r N_{bn}
\end{aligned}$$

Pressure Ratio Gradient

When calculating the pressure ratio, the inlet mass averaged total pressure $\bar{P}_1^*$ is more or less fixed due to the specified total pressure profile at an inlet. It is therefore assumed that the inlet mass averaged total pressure is constant.

At an inlet, the only derivative is given by

$$\frac{\partial M_3}{\partial p} = 2\sigma \left(\frac{\pi}{\pi_0} - 1 \right) \frac{1}{\pi_0} \frac{1}{\bar{P}_1^*} (-\pi) \frac{1}{\dot{m}} \left(p_1^* - \bar{P}_1^* \right) \frac{\partial \rho v_n}{\partial p} \tag{C.7}$$

At an exit, the derivatives are given by

$$\begin{aligned}
\frac{\partial M_3}{\partial \rho} &= 2\sigma \left(\frac{\pi}{\pi_0} - 1 \right) \frac{1}{\pi_0} \frac{1}{\bar{P}_1^*} \frac{1}{\dot{m}} \left[\frac{1}{2} A \left(u^2 + v^2 + w^2 \right) \rho v_n + \left(p_2^* - \bar{P}_2^* \right) v_n \right] \\
\frac{\partial M_3}{\partial u} &= 2\sigma \left(\frac{\pi}{\pi_0} - 1 \right) \frac{1}{\pi_0} \frac{1}{\bar{P}_1^*} \frac{1}{\dot{m}} \left[A \rho u \rho v_n + \left(p_2^* - \bar{P}_2^* \right) \rho n_x \right] \\
\frac{\partial M_3}{\partial v} &= 2\sigma \left(\frac{\pi}{\pi_0} - 1 \right) \frac{1}{\pi_0} \frac{1}{\bar{P}_1^*} \frac{1}{\dot{m}} \left[A \rho v \rho v_n + \left(p_2^* - \bar{P}_2^* \right) \rho n_\theta \right] \\
\frac{\partial M_3}{\partial w} &= 2\sigma \left(\frac{\pi}{\pi_0} - 1 \right) \frac{1}{\pi_0} \frac{1}{\bar{P}_1^*} \frac{1}{\dot{m}} \left[A \rho w \rho v_n + \left(p_2^* - \bar{P}_2^* \right) \rho n_r \right]
\end{aligned} \tag{C.8}$$

Appendix D

Nonlinear Harmonic Phase Solution Method versus Time-linearized Linear Harmonic Method

The comparison of the Nonlinear Harmonic Phase Solution method and the time-linearized linear harmonic method will be illustrated here with the use of the Burger's equation for simplicity:

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial u^2}{\partial x} = 0 \tag{D.1}$$

The time linearized method decomposes the solution variable into its time averaged one $\bar{u}$ plus a perturbation u', namely,

$$u = \bar{u} + u'$$

where $\bar{u}$ does not depend on time any more, but u' is a function of both time and space. Substituting the expression of u into the Burger's equation yields

$$\frac{\partial(\bar{u} + u')}{\partial t} + \frac{1}{2}\frac{\partial(\bar{u}^2 + 2\bar{u}u' + u'^2)}{\partial x} = 0$$

$\frac{\partial \bar{u}}{\partial t} = 0$, because $\bar{u}$ is not a function of time. This equation can be further simplified as

$$\frac{\partial u'}{\partial t} + \frac{1}{2}\frac{\partial(\bar{u}^2 + 2\bar{u}u' + u'^2)}{\partial x} = 0 \tag{D.2}$$

Time-averaging the above equation, one has the time-averaged equation:

$$\frac{1}{2}\frac{\partial(\bar{u}^2 + \overline{u'^2})}{\partial x} = 0 \tag{D.3}$$

Subtracting Eq.D.3 from Eq.D.2 gives the perturbation equation:

$$\frac{\partial u'}{\partial t} + \frac{1}{2}\frac{\partial(2\bar{u}u' + u'^2 - \overline{u'^2})}{\partial x} = 0 \tag{D.4}$$

The time-linearized method assumes that u' is much smaller than $\bar{u}$ in terms of magnitude, therefore u'^2 and $\overline{u'^2}$ are of one order smaller than u' and are negligible. Eqs.D.3 and D.4 are accordingly reduced as follows

$$\frac{1}{2}\frac{\partial \bar{u}^2}{\partial x} = 0 \tag{D.5}$$

$$\frac{\partial u'}{\partial t} + \frac{\partial \bar{u}u'}{\partial x} = 0 \tag{D.6}$$

Eq.D.6 is still in time domain, direct solution of this equation in time domain is still costly. The unsteady flow field concerning blade flutter or forced response is usually time-periodically dominant with its period/frequency known *a priori*. The perturbation part of the steady flow solution can be expressed in terms of Fourier series:

$$u' = \sum_{i=1}^{\infty} \left[a_i \sin(i\omega t) + b_i \cos(i\omega t)\right] \tag{D.7}$$

It is normally enough to keep one harmonic for blade flutter or forced response prediction. The perturbation part is simplified accordingly:

$$u' = a_1 \sin(\omega t) + b_1 \cos(\omega t) \tag{D.8}$$

Substituting the above perturbation part expression into Eq.D.6 yields

$$\left[a_1 \cos(\omega t) - b_1 \sin(\omega t)\right]\omega + \frac{\partial \bar{u}\left[a_1 \sin(\omega t) + b_1 \cos(\omega t)\right]}{\partial x} = 0 \tag{D.9}$$

The above equation can be rearranged by collecting like terms of $\sin(\omega t)$ and $\cos(\omega t)$:

$$\left[a_1\omega + \frac{\partial \bar{u}b_1}{\partial x}\right]\cos(\omega t) + \left[-b_1\omega + \frac{\partial \bar{u}a_1}{\partial x}\right]\sin(\omega t) = 0$$

Both $\sin(\omega t)$ and $\cos(\omega t)$ can not always be zeros, therefore the terms multiplying them have to vanish to satisfy the above equation, leading to the two sets of time-linearized equations:

$$\begin{aligned} a_1\omega + \frac{\partial \bar{u}b_1}{\partial x} &= 0 \\ -b_1\omega + \frac{\partial \bar{u}a_1}{\partial x} &= 0 \end{aligned} \tag{D.10}$$

Now it can be seen that Eq.D.5 is independent of Eq.D.10, while Eq.D.10 does depend on the solution to Eq.D.5. The two equations are solved sequentially with Eq.D.5 solved first and followed by Eq.D.10.

Unfortunately, the splitting process, leading to the time-averaged equation and the linear harmonic equation, makes it difficult to formulate and then solve the corresponding adjoint system. The two equations have to be linearized with respect to a design variable to derive an adjoint system. Eq.D.10 is already a time-linearized equation, making it quite complicated to be linearized against a design variable.

In line with the time-linearized linear harmonic method, the Nonlinear Harmonic Phase Solution method bases itself upon the assumption that the unsteadiness in the unsteady flow field of interest is temporal-periodic with the flow solution as a whole expressed in terms of the summation of the time-averaged one plus one harmonic:

$$u = \bar{u} + a_1 \sin(\omega t) + b_1 \cos(\omega t)$$

With the u substituted into the time derivative term in Eq.D.1, the time derivative term can be replaced as follows

$$[a_1 \cos(\omega t) - b_1 \sin(\omega t)]\,\omega + \frac{1}{2}\frac{\partial u^2}{\partial x} = 0 \tag{D.11}$$

Notice that u^2 in Eq.D.1 is not replaced with its Fourier expression. Eq.D.11 has three sets of unknowns: a_1, b_1 and $\bar{u}$. In order to solve the equation, as He (2008) proposed, three particular phases are chosen: $\omega t = -\frac{\pi}{2}, 0, \frac{\pi}{2}$. Eq.D.11 can be written at the aforementioned three phases:

$$\begin{aligned} b_1\omega + \frac{1}{2}\frac{\partial u_{-1}^2}{\partial x} &= 0 \\ a_1\omega + \frac{1}{2}\frac{\partial u_0^2}{\partial x} &= 0 \\ -b_1\omega + \frac{1}{2}\frac{\partial u_1^2}{\partial x} &= 0 \end{aligned} \tag{D.12}$$

where subscripts $-1, 0$ and 1 denote the solution u at the three phases: $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$ respectively. a_1 and b_1 can be expressed in terms of the solution at the three phases, namely,

$$\begin{aligned} a_1 &= \frac{u_1 - u_{-1}}{2} \\ b_1 &= u_0 - \frac{u_1 + u_{-1}}{2} \end{aligned}$$

Subsequently Eq.D.13 can be written in terms of the solution at three phases only:

$$\begin{aligned}\left[u_0 - \frac{u_1 + u_{-1}}{2}\right]\omega + \frac{1}{2}\frac{\partial u_{-1}^2}{\partial x} &= 0 \\ \frac{u_1 - u_{-1}}{2}\omega + \frac{1}{2}\frac{\partial u_0^2}{\partial x} &= 0 \\ -\left[u_0 - \frac{u_1 + u_{-1}}{2}\right]\omega + \frac{1}{2}\frac{\partial u_1^2}{\partial x} &= 0\end{aligned} \tag{D.13}$$

Compared with the steady Eq.D.5, each of the equations at the three phases has an extra term corresponding to the time derivative in the unsteady equation. This extra term can be treated as a source term when these equations are solved. It can be seen that it is quite straightforward to extend a solver solving the steady equation to one solving the three equations. In addition, when linearizing the equations with respect to a design variable, there is little difference from linearizing a steady equation to a design variable, making it straightforward to derive the corresponding adjoint system and solve it.

Appendix E

Time-Domain Unsteady Adjoint Equations

An unsteady adjoint Euler equation is derived from an unsteady Euler flow equation using the continuous adjoint approach. Then the derived unsteady adjoint equation is discretized in time using the Nonlinear Harmonic Phase Solution method, leading to three sets of steady-like adjoint systems at the three phases: $(-\frac{\pi}{2}, 0, \frac{\pi}{2})$. The following three main points will be illustrated in the derivation.

1) The unsteady adjoint Euler equation has the same characteristic paths as the linearized unsteady Euler flow equation. The characteristic information of the unsteady adjoint system propagates at the same speed but in the opposite direction compared with their corresponding flow characteristic information.

2) The unsteady adjoint equation needs to be time-marched backwards in time: from the final time to the initial time. The adjoint field at the final time is null, while the initial adjoint field is a part of an unsteady adjoint solution (hence unknown).

3) In the context of unsteady turbomachinery flow field calculations, the inter-blade phase angle for phase-shift periodic boundaries in an unsteady adjoint solution is the same as that in the baseline unsteady flow solution, if a single blade passage with an inter-blade phase angle is used in a computation.

Different from an objective function in a design optimization based upon a steady flow field calculation, the objective function in the context of an unsteady flow based design optimization is normally an integral in both space and time:

$$I = \int_0^T \int_{\partial D} M(U, X)\, ds dt \tag{E.1}$$

The gradient of the objective function to a design variable is given by

$$\frac{dI}{d\alpha} = \int_0^T \int_{\partial D} \left(\frac{\partial M}{\partial U} \tilde{U} + \frac{\partial M}{\partial \alpha} \right) ds dt + \int_0^T \int_{\partial D} M \tilde{d} s dt \tag{E.2}$$

A linearized unsteady Euler flow equation with respect to a design variable is written as follows:

$$\frac{\partial \tilde{U}}{\partial t} + \frac{\partial (A - u_g I)\tilde{U}}{\partial x} + \frac{\partial (B - v_g I)\tilde{U}}{r \partial \theta} + \frac{\partial r (C - w_g I)\tilde{U}}{r \partial r} - D\tilde{U} - f = 0 \tag{E.3}$$

Multiplying the above flow equation with an adjoint solution variable yields

$$\lambda^T \left[\frac{\partial \tilde{U}}{\partial t} + \frac{\partial (A - u_g I)\tilde{U}}{\partial x} + \frac{\partial (B - v_g I)\tilde{U}}{r \partial \theta} + \frac{\partial r (C - w_g I)\tilde{U}}{r \partial r} - D\tilde{U} - f \right] = 0 \tag{E.4}$$

Integrating the above equation in both time and the computational domain gives

$$\int_0^T \int_D \lambda^T \left[\frac{\partial \tilde{U}}{\partial t} + \frac{\partial (A - u_g I)\tilde{U}}{\partial x} + \frac{\partial (B - v_g I)\tilde{U}}{r \partial \theta} + \frac{\partial r (C - w_g I)\tilde{U}}{r \partial r} - D\tilde{U} - f \right] dv dt = 0 \tag{E.5}$$

Performing integration by parts over the spatial and time derivative terms once and rearranging, one has

$$\begin{aligned}
&\int_0^T \int_D \frac{\partial \lambda^T \tilde{U}}{\partial t} dv dt - \int_0^T \int_D \frac{\partial \lambda^T}{\partial t} \tilde{U} dv dt \\
&+ \int_0^T \int_{\partial D} \lambda^T \left[(A - u_g I)\tilde{U} n_x + (B - v_g I)\tilde{U} n_\theta + (C - w_g I)\tilde{U} n_r \right] ds dt \\
&- \int_0^T \int_D \left[\frac{\partial \lambda^T}{\partial x} (A - u_g I) + \frac{\partial \lambda^T}{r \partial \theta} (B - v_g I) + \frac{\partial \lambda^T}{\partial r} (C - w_g I) + \lambda^T D \right] \tilde{U} dv dt \\
&- \int_0^T \int_D \lambda^T f dv dt = 0
\end{aligned} \tag{E.6}$$

Subtracting the left hand side of the above equation from the gradient expression in E.2 does not change the value of the original gradient:

$$\begin{aligned}
\frac{dI}{d\alpha} &= - \int_D \left(\lambda^T \tilde{U} \Big|_{t=T} - \lambda^T \tilde{U} \Big|_{t=0} \right) dv \\
&- \int_0^T \int_{\partial D} \lambda^T \left[(A - u_g I)\tilde{U} n_x + (B - v_g I)\tilde{U} n_\theta + (C - w_g I)\tilde{U} n_r - \frac{\partial M}{\partial U} \tilde{U} \right] ds dt \\
&+ \int_0^T \int_D \left[\frac{\partial \lambda^T}{\partial t} + \frac{\partial \lambda^T}{\partial x} (A - u_g I) + \frac{\partial \lambda^T}{r \partial \theta} (B - v_g I) + \frac{\partial \lambda^T}{\partial r} (C - w_g I) + \lambda^T D \right] \tilde{U} dv dt \\
&+ \int_0^T \int_{\partial D} \frac{\partial M}{\partial \alpha} ds dt + \int_0^T \int_{\partial D} M \tilde{d} s dt + \int_0^T \int_D \lambda^T f dv dt
\end{aligned} \tag{E.7}$$

General Unsteady Flow

At first, let's consider that the unsteady flow field is not temporally periodic. The adjoint formulation requires the final gradient be free from flow variable sensitivities. This is equivalent to vanish the following three integral terms:

$$\int_D \lambda^T \tilde{U}\Big|_{t=T} dv = 0 \tag{E.8}$$

$$\int_0^T \int_{\partial D} \lambda^T \left[(A - u_g I)\,\tilde{U} n_x + (B - v_g I)\,\tilde{U} n_\theta + (C - w_g I)\,\tilde{U} n_r - \frac{\partial M}{\partial U}\tilde{U} \right] ds dt = 0 \tag{E.9}$$

$$\int_0^T \int_D \left[\frac{\partial \lambda^T}{\partial t} + \frac{\partial \lambda^T}{\partial x}(A - u_g I) + \frac{\partial \lambda^T}{r\partial \theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r}(C - w_g I) + \lambda^T D \right] \tilde{U} dv dt \tag{E.10}$$

It should be noted that the flow variable sensitivity at the initial time $\tilde{U}\Big|_{t=0}$ is known in a linearized method (the initial condition for solving Eq.E.3), therefore $\int_D \lambda^T \tilde{U}\Big|_{t=0} dv$ does not depend on any unknown flow variable sensitivity and should not be forced to vanish. This term is a function of the adjoint solution at the initial time, and constitutes a part of the final gradient expression.

The necessary and sufficient condition to satisfy Eq.E.8 is

$$\lambda|_{t=T} = 0 \tag{E.11}$$

This gives an unsteady adjoint solution at the final time, from which the adjoint solution will be time-marched backwards in time till the initial time. This equation tells that the unsteady adjoint solution at the final time is zero in the interior domain.

The sufficient and necessary condition to satisfy Eq.E.9 will give the boundary condition for the unsteady adjoint equation. The inlet, exit and solid wall boundary conditions for a time-domain unsteady adjoint system take a similar form as those for the adjoint system derived in Chapter 8.

The sufficient and necessary condition to satisfy Eq.E.10 leads to the unsteady adjoint Euler equation:

$$\frac{\partial \lambda^T}{\partial t} + \frac{\partial \lambda^T}{\partial x}(A - u_g I) + \frac{\partial \lambda^T}{r\partial \theta}(B - v_g I) + \frac{\partial \lambda^T}{\partial r}(C - w_g I) + \lambda^T D = 0 \tag{E.12}$$

The unsteady adjoint Euler equation is a linear system. The Jacobian matrices in the linearized Euler flow equation (E.3) are identical to those in the above adjoint equation. This ensures that the adjoint equation has the same characteristic paths as the linearized flow equation.

The unsteady adjoint Euler equation is different from the steady adjoint equation with a pseudo time derivative term added (Eq.4.22): the signs before the spatial derivative terms in Eq.4.22 are minuses, while the signs before the spatial derivative terms in Eq.E.12 are pluses. The unsteady adjoint system is a final-boundary value problem, where an unsteady adjoint solution process starts at the final time and time-marches backwards in time. While the time integration methods, such as the Runge-Kutta method, are designed for initial-boundary value problems. In order to make use of the well-developed time integration methods to solve a steady adjoint system, the signs before those spatial derivatives are made opposite to the sign of the pseudo time derivative term.

The final gradient expression is therefore reduced to

$$\frac{dI}{d\alpha} = \int_{\partial D} \lambda^T \tilde{U}\Big|_{t=0} dv + \int_0^T \int_{\partial D} \frac{\partial M}{\partial \alpha} ds dt + \int_0^T \int_{\partial D} M \tilde{ds} dt + \int_0^T \int_D \lambda^T f dv dt \quad \text{(E.13)}$$

Time-periodic Flow

In the following, we are going to consider the situation where the unsteady flow solution is time-periodic. If an unsteady flow solution is time-periodic, and T is the period of the unsteadiness, then the unsteady flow solution (hence its sensitivity) at $t = 0$ and $t = T$ should be equal:

$$\tilde{U}\Big|_{t=T} = \tilde{U}\Big|_{t=0} \quad \text{(E.14)}$$

The sufficient and necessary condition to vanish the boundary integral:

$$\int_D \left(\lambda^T \tilde{U}\Big|_{t=T} - \lambda^T \tilde{U}\Big|_{t=0} \right) dv$$

is given by

$$\lambda|_{t=T} = \lambda|_{t=0} \quad \text{(E.15)}$$

The above equation also implies that the unsteady adjoint solution is time-periodic with a period of T.

The final gradient in E.13 is changed as follows:

$$\frac{dI}{d\alpha} = \int_0^T \int_{\partial D} \frac{\partial M}{\partial \alpha} dsdt + \int_0^T \int_{\partial D} M\tilde{d}sdt + \int_0^T \int_D \lambda^T f dvdt \tag{E.16}$$

In the context of a time-periodic unsteady flow solution for turbomachinery blades, a phase-shift period boundary condition with an inter-blade phase angle (ϕ) is usually used in conjunction with a single blade passage to reduce the size of a problem. The following derivation will illustrate the phase-shift periodic boundary condition for the corresponding unsteady adjoint Euler equation.

At phase-shift periodic boundaries, an objective function normally does not explicitly depend on flow solutions there, namely, $M = 0$. In Eq.E.9

$$(A - u_g I)\tilde{U}n_x + (B - v_g I)\tilde{U}n_\theta + (C - w_g I)\tilde{U}n_r$$

is a flux perturbation, which can be expressed using one symbol F for a more compact illustration. Then Eq.E.9 along a pair of periodic boundaries can be simplified as

$$\int_0^T \int_{ab} \lambda F^T dsdt + \int_0^T \int_{cd} \lambda F^T dsdt = 0 \tag{E.17}$$

where ab and cd refer to the pair of periodic boundaries as sketched in Fig.3.1. At periodic boundaries, if a flow solution is time-periodic and truncated to one harmonic, then it is accordingly valid to assume the flux (hence the flux perturbation) is also time-periodic and can be truncated to one harmonic. If the flux perturbation at a lower periodic boundary is expressed as

$$F^L = \bar{F} + F_A \sin(\omega t) + F_B \cos(\omega t) \tag{E.18}$$

where ^{L} denotes a lower periodic boundary. Then the flux perturbation at an upper periodic boundary can be expressed as

$$F^U = \bar{F} + F_A \sin(\omega t + \phi) + F_B \cos(\omega t + \phi) \tag{E.19}$$

where ^{U} denotes an upper periodic boundary. Substituting Eqs.E.18 and E.19 into Eq.E.17 gives

$$\begin{aligned} &\int_0^T \int_{ab} \lambda \left[\bar{F} + F_A \sin(\omega t) + F_B \cos(\omega t)\right]^T dsdt \\ &+ \int_0^T \int_{cd} \lambda \left[\bar{F} + F_A \sin(\omega t + \phi) + F_B \cos(\omega t + \phi)\right]^T dsdt = 0 \end{aligned} \tag{E.20}$$

With further rearrangement the above equation becomes

$$\begin{aligned}&\int_0^T \int_{ab} \lambda^L \left[\bar{F} + F_A \sin(\omega t) + F_B \cos(\omega t)\right]^T ds dt \\ &- \int_0^T \int_{ab} \lambda^U \left[\bar{F} + F_A \sin(\omega t + \phi) + F_B \cos(\omega t + \phi)\right]^T ds dt = 0\end{aligned} \tag{E.21}$$

$\bar{F}$, F_A and F_B can be also expressed as follows:

$$\begin{aligned}\bar{F} &= \frac{1}{T}\int_0^T F d\tau \\ F_A &= \frac{2}{T}\int_0^T F \sin(\omega\tau)\, d\tau \\ F_B &= \frac{2}{T}\int_0^T F \cos(\omega\tau)\, d\tau\end{aligned} \tag{E.22}$$

The first term on the left hand side of Eq.E.21 can be arranged as follows with the above substituted into it:

$$\begin{aligned}&\int_0^T \int_{ab} \lambda^L \left[\bar{F} + F_A \sin(\omega t) + F_B \cos(\omega t)\right]^T ds dt \\ &= \int_0^T \int_{ab} \lambda^L \left[\tfrac{1}{T}\int_0^T F^T d\tau + \tfrac{2}{T}\int_0^T F^T \sin(\omega\tau)\, d\tau \sin(\omega t) + \tfrac{2}{T}\int_0^T F^T \cos(\omega\tau)\, d\tau \cos(\omega t)\right] ds dt \\ &= \int_0^T \int_{ab} \int_0^T \lambda^L F^T \left[\tfrac{1}{T} + \tfrac{2}{T}\sin(\omega\tau)\sin(\omega t) + \tfrac{2}{T}\cos(\omega\tau)\cos(\omega t)\right] d\tau ds dt \\ &= \int_0^T \int_{ab} F^T \left[\tfrac{1}{T}\int_0^T \lambda^L dt + \tfrac{2}{T}\int_0^T \lambda^L \sin(\omega t)\, dt \sin(\omega\tau) + \tfrac{2}{T}\int_0^T \lambda^L \cos(\omega t)\, dt \cos(\omega\tau)\right] ds d\tau \\ &= \int_0^T \int_{ab} F^T \left[\bar{\lambda}^L + \lambda_A^L \sin(\omega\tau) + \lambda_B^L \cos(\omega\tau)\right] ds d\tau\end{aligned} \tag{E.23}$$

where

$$\begin{aligned}\bar{\lambda}^L &= \frac{1}{T}\int_0^T \lambda^L dt \\ \lambda_A^L &= \frac{2}{T}\int_0^T \lambda^L \sin(\omega t)\, dt \\ \lambda_B^L &= \frac{2}{T}\int_0^T \lambda^L \cos(\omega t)\, dt\end{aligned} \tag{E.24}$$

In a similar way, the second term on the left hand side of Eq.E.21 can be arranged as follows:

$$\begin{aligned}&\int_0^T \int_{ab} \lambda^U \left[\bar{F} + F_A \sin(\omega t + \phi) + F_B \cos(\omega t + \phi)\right]^T ds dt \\ &= \int_0^T \int_{ab} F^T \left[\tfrac{1}{T}\int_0^T \lambda^U dt + \tfrac{2}{T}\int_0^T \lambda^U \sin(\omega t + \phi)\, dt \sin(\omega\tau) + \tfrac{2}{T}\int_0^T \lambda^U \cos(\omega t + \phi)\, dt \cos(\omega\tau)\right] ds \\ &= \int_0^T \int_{ab} F^T \left[\tfrac{1}{T}\int_0^T \lambda^U dt + \tfrac{2}{T}\int_0^T \lambda^U \sin(\omega t)\, dt \sin(\omega\tau - \phi) + \tfrac{2}{T}\int_0^T \lambda^U \cos(\omega t)\, dt \cos(\omega\tau - \phi)\right] ds \\ &= \int_0^T \int_{ab} F^T \left[\bar{\lambda}^U + \lambda_A^U \sin(\omega\tau - \phi) + \lambda_B^U \cos(\omega\tau - \phi)\right] ds d\tau\end{aligned} \tag{E.25}$$

where

$$\begin{aligned} \bar{\lambda}^U &= \frac{1}{T}\int_0^T \lambda^U dt \\ \lambda_A^U &= \frac{2}{T}\int_0^T \lambda^U \sin(\omega t)\, dt \\ \lambda_B^U &= \frac{2}{T}\int_0^T \lambda^U \cos(\omega t)\, dt \end{aligned} \tag{E.26}$$

Substituting expressions E.23 and E.25 back into Eq.E.21, it is not difficult to find that the sufficient and necessary condition to satisfy Eq.E.21 is given by

$$\bar{\lambda}^L + \lambda_A^L \sin(\omega\tau) + \lambda_B^L \cos(\omega\tau) = \bar{\lambda}^U + \lambda_A^U \sin(\omega\tau - \phi) + \lambda_B^U \cos(\omega\tau - \phi) \tag{E.27}$$

The left hand side of the above equation is actually the adjoint solution at a lower periodic boundary:

$$\lambda^L = \bar{\lambda}^U + \lambda_A^U \sin(\omega\tau - \phi) + \lambda_B^U \cos(\omega\tau - \phi) \tag{E.28}$$

This is the formulation updating the unsteady adjoint solution at a lower periodic boundary using the adjoint solution at an upper periodic boundary.

Replacing $\omega\tau$ with $\omega\tau + \phi$ in Eq.E.27 and rearranging gives the following formulation for updating the unsteady adjoint solution at an upper periodic boundary using the unsteady adjoint solution at the corresponding lower periodic boundary:

$$\lambda^U = \bar{\lambda}^L + \lambda_A^L \sin(\omega\tau + \phi) + \lambda_B^L \cos(\omega\tau + \phi) \tag{E.29}$$

Comparing Eqs.E.28 and E.29 with Eqs.7.11 and 7.9, one can see that the phase-shift for both the adjoint equation and the flow equation is identical.

In order to solve the unsteady adjoint Euler equation (Eq.E.12) efficiently, the Nonlinear Harmonic Phase Solution method for an efficient unsteady flow solution can also be used for an efficient unsteady adjoint solution.

Transposing Eq.E.12 and collectively writing its source term and the spatial derivative term as $R^{adj}(\lambda)$, one has

$$\frac{\partial \lambda}{\partial t} + R^{adj}(\lambda) = 0 \tag{E.30}$$

The unsteady adjoint solution can be expressed as a whole in the following Fourier series with one harmonic included:

$$\lambda = \bar{\lambda} + \lambda_A \sin(\omega t) + \lambda_B \cos(\omega t) \tag{E.31}$$

In a similar way for the three-phase flow equations in Chapter 7, the three-phase adjoint equations are not derived in detail, but are given as follows:

$$\begin{aligned} \left(\lambda_0 - \frac{\lambda_1 + \lambda_{-1}}{2}\right)\omega + R^{adj}(\lambda_{-1}) &= 0 \\ \frac{\lambda_1 - \lambda_{-1}}{2}\omega + R^{adj}(\lambda_0) &= 0 \\ -\left(\lambda_0 - \frac{\lambda_1 + \lambda_{-1}}{2}\right)\omega + R^{adj}(\lambda_1) &= 0 \end{aligned} \tag{E.32}$$

Comparing the above three sets of equations with those in Eq.8.18, it is not difficult to find out the differences in the terms multiplying with ω.

The gradient (Eq.E.16) is composed of three parts. The last term in Eq.E.16 is related to the adjoint solution and its calculation is examined particularly in the following. The term is an integral in both space and time, it requires the reconstruct of an time-dependent adjoint solution and an time-dependent flow residual perturbation due to pure geometry changes. The time-dependent quantities can be reconstructed from the solutions at the three phases for both the adjoint solution and the flow residual perturbation.

Assuming the flow residual due to pure geometry changes with fixed flow variables can be decomposed in the same manner as the unsteady flow solution in Eq.7.3:

$$\lambda = \bar{f} + f_A \sin(\omega t) + f_B \cos(\omega t)$$

where $\bar{f}$, f_A and f_B can be obtained using the value of f at the three phases. Substituting the above and E.31 into the last term in the right hand of Eq.E.16 yields

$$\begin{aligned} &\int_0^T \int_D \lambda^T f \, dv dt \\ &= \int_0^T \int_D \left(\bar{\lambda} + \lambda_A \sin\omega t + \lambda_B \cos\omega t\right)^T \left(\bar{f} + f_A \sin\omega t + f_B \cos\omega t\right) dv dt \\ &= \int_D T\left(\bar{\lambda}\bar{f} + \frac{\lambda_A f_A}{2} + \frac{\lambda_B f_B}{2}\right) dv \end{aligned} \tag{E.33}$$

Substituting the following expressions into the above ends up with a much more complicated result than the corresponding term in the gradient expression by the semi-

continuous approach (Eq.8.19):

$$\begin{aligned}
\lambda_A &= \frac{\lambda_1 - \lambda_{-1}}{2} \\
\lambda_B &= \lambda_0 - \frac{\lambda_1 + \lambda_{-1}}{2} \\
\bar{\lambda} &= \frac{\lambda_1 + \lambda_{-1}}{2}
\end{aligned}$$

$$\begin{aligned}
f_A &= \frac{f_1 - f_{-1}}{2} \\
f_B &= f_0 - \frac{f_1 + f_{-1}}{2} \\
\bar{f} &= \frac{f_1 + f_{-1}}{2}
\end{aligned}$$