Multidisciplinary Concurrent Optimization of Gas Turbine Blades



Omar R. Valero Ricart

Department of Engineering Science

University of Oxford

St Catherine's College



Submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy in Engineering Science

Trinity Term 2015

Acknowledgements

Firstly, I would like to express my sincere gratitude to my supervisor Prof. He for his patience, generosity, motivation and immense knowledge. His guidance and support have taught me and helped me in my academic and personal life during all these years of DPhil. I could not have imagined having a better supervisor and mentor.

Besides my supervisor, I would like to thank my thesis committee: Prof. Chew and Dr. McGilvray for willing to be my examiners.

My sincere thanks also goes to my colleagues in Siemens Industrial Turbomachinery Ltd: Dr. Wang, who provided me invaluable help during the project and also Dr. Li and Mr. Wells who provided funding for this work and their insightful feedback. A special thanks also goes to Dr. Rahmati who helped me during the first year of my DPhil and further discussions after.

I thank my fellow lab mates and friends for the work we shared and fun we had these last years.

I would like to thank my family: my parents and brother for their unconditional love throughout my life. Their advice always guide me, no matter how far we are.

And last, but not least important, I want to thank Noelia for being my support and happiness since we started this adventure 5 years ago. You always lift me up when I fall and put a smile on my face, even on the worst days.

Abstract

Multidisciplinary Concurrent Optimization of Gas Turbine Blades

Doctor of Philosophy in Engineering Science

Omar R. Valero Ricart

St Catherine's College

Trinity Term 2015

This thesis presents the study and development of optimization methods that can perform concurrent aerodynamic-aeroelastic blade optimization in a multi-bladerow environment, and for realistic turbomachinery blade geometries.

The Nonlinear Harmonic Phase Solution Method has been chosen as the flow solution method of this work because of its capability to calculate the aeroelasticity features of interest (blade flutter) and the main flow aerodynamic performance in steady flow timescales.

The first optimization method that is shown is a more generic non-gradient method with an improved version of a quadratic Response Surface Model. The new Re-Scaled Response Surface Model has shown marked convergence and performance improvements against traditional surrogate models. However, the computational cost of this method for cases with a large number of design variables limits its real applications.

A gradient-based adjoint method is presented next as a cost-independent alternative that can accomplish efficient multi-bladerow optimization for a large number of variables within the current levels of computational power. The continuous adjoint system has been developed based on the same methodology as the flow solution method and it shows a more consistent relation between the flow field and its corresponding adjoint field, in agreement with the "anti-physics" information path. An adjoint interface treat-

ment has been developed as an extension of the flow harmonic interface treatment. This unique treatment allows capture of the damping sensitivities of the vibrating blade to shape changes in adjacent rows. The application of this method to the design optimization of compressor and turbine stages has shown its capability to perform efficient multi-component and multi-disciplinary design optimization of turbomachinery blades.

Contents

Contents vii
List of Figures xiii
List of Tables xix
Nomenclature xxvi
1 Introduction 27
1.1 Introduction 27
1.2 Challenges in Advanced Turbomachinery Design 28
1.3 Methods for Design and Optimization 29
1.3.1 CFD Methods 30
1.3.1.1 Efficient Methods for Aero-mechanical Design Optimization 30
1.4 Optimization Methods 31
1.4.1 Flow Solver 31
1.4.2 Shape Parameterization 32
1.4.3 Optimization Strategy 32
1.4.3.1 Gradient-based Methods 33
1.4.3.2 Non-gradient-based Methods 34
1.5 Problem Statement for Present Work 34
1.6 Thesis Overview 35
2 Literature Review 39
2.1 CFD in Turbomachinery Design 39
2.2 Three-dimensional Steady Flow Methods 40
2.2.1 Single-bladerow steady flow calculations 40

2.2.2 Multi-stage steady flow calculations 41
2.3 Three-dimensional Unsteady Flow Methods 42
2.3.1 Direct Unsteady Solution 42
2.3.2 Single-passage (Phase-shift) Flow Methods 43
2.3.3 Harmonic (Frequency-domain) Methods 44
2.3.3.1 Linearised harmonic Modelling 45
2.3.3.2 Nonlinear harmonic Modelling 45
2.4 The Adjoint Method in Turbomachinery 47
2.5 Aerodynamic Design Optimization 49
2.5.1 Non-gradient Methods in Aerodynamic Design 50
2.5.2 Adjoint Method in Aerodynamic Design 50
2.6 Aeroelastic Design Optimization 51
2.7 Multidisciplinary Optimization Methods 52
2.7.1 Constrained (Mechanical) Optimization 52
2.7.2 Concurrent Optimization 54
2.8 Summary 55
3 Nonlinear Harmonic Phase Solution Method 57
3.1 Governing Equations 57
3.2 Flow Solution Method 60
3.3 Turbulence Models 62
3.4 Discretization of the Equations 64
3.4.1 Spatial Discretization 65
3.4.2 Pseudo Time Marching Solution 68
3.5 Boundary Conditions 70
3.5.1 Inlet Boundary Condition 70
3.5.2 Exit Boundary Condition 71
3.5.3 Solid Wall Boundary Condition 72
3.5.4 Periodic Boundary Condition 73

3.6 Multi-bladerow Treatment . . . 74
3.6.1 Mixing-Plane Treatment . . . 74
3.6.2 Harmonic Interface Treatment . . . 78
3.7 Validation of the Flow Solver . . . 81
3.8 Summary . . . 86
4 Non-gradient Based Optimization with Re-scaled Response Surface Model . . . 89
4.1 Database: Latin Hypercube Sampling . . . 89
4.2 Candidate Search: Differential Evolution . . . 91
4.3 Metamodel: Response Surface Model . . . 94
4.3.1 Re-Scaled Response Surface Model . . . 96
4.4 Method Validation . . . 101
4.4.1 Optimization System . . . 101
4.4.2 Validation Against a One-dimensional Function . . . 103
4.4.3 Validation Against a Two-dimensional Function . . . 106
4.5 Summary . . . 108
5 Optimization Case Studies using Re-scaled Response Surface Method . . . 111
5.1 Introduction . . . 111
5.2 Structural Modeling: FE-CFD interactions . . . 111
5.3 Blade Flutter Evaluation . . . 112
5.4 Shape Parameterization . . . 115
5.5 Objective Function . . . 116
5.6 Design Optimization Process . . . 121
5.7 Design Optimization Cases and Sensitivity Analysis . . . 122
5.7.1 NASA Rotor 67 . . . 124
5.7.2 DLR Transonic Compressor Stage . . . 125
5.7.3 Industrial Low-pressure Turbine Stage . . . 127
5.8 Results . . . 128
5.8.1 NASA Rotor 67 Aeroelastic-only Optimization . . . 128
5.8.2 DLR Transonic Stage Aeroelastic-only Optimization . . . 133

5.8.3 Industrial LP Stage Aeroelastic Optimization . . . 138
5.8.3.1 Aeroelastic-only Stage Design Optimization . . . 139
5.8.3.2 Concurrent Aeroelastic Stage Design Optimization . . . 140
5.8.3.3 Multi-disciplinary Optimization Comparison . . . 141
5.9 Summary . . . 146
6 The Adjoint Method . . . 147
6.1 Adjoint Method Principle . . . 147
6.2 Steady Adjoint Method: Euler Flow Equations . . . 150
6.3 Summary . . . 153
7 Continuous Unsteady Adjoint System . . . 155
7.1 Overview . . . 155
7.2 Continuous Unsteady Adjoint Method Principle . . . 157
7.3 Unsteady Adjoint Equations . . . 159
7.4 Discretization of the Equations . . . 163
7.4.1 Spatial Discretization . . . 163
7.4.2 Temporal Discretization . . . 164
7.5 Boundary Conditions . . . 165
7.6 Periodic Boundary Condition: Inter-blade Phase Angle . . . 170
7.7 Gradients of the Objective Function . . . 172
7.7.1 Entropy generation rate gradient . . . 174
7.7.2 Mass flow rate gradient . . . 175
7.7.3 Pressure ratio gradient . . . 176
7.7.4 Work sum gradient . . . 176
7.8 Summary . . . 177
8 Multi-bladerow Treatments . . . 179
8.1 Adjoint Mixing-plane . . . 179
8.2 Adjoint Harmonic Interface . . . 184
8.3 Validation of the Adjoint Harmonic Interface . . . 187
8.4 Summary . . . 189

9 Adjoint based Concurrent Design Optimization System . . . 191
9.1 Introduction . . . 191
9.2 Gradient Calculation . . . 191
9.3 Search Method: Steep Descent . . . 194
9.4 Design Optimization Process . . . 195
9.5 Summary . . . 197
10 Results and Optimization Case Studies using Adjoint . . . 199
10.1 Aeroelastic-only Optimization of Stator Blades . . . 200
10.1.1 DLR transonic compressor stage . . . 201
10.1.2 Industrial LP turbine stage . . . 204
10.2 NASA Rotor 67 Aeroelastic-only Optimization . . . 207
10.3 DLR Transonic Stage Aeroelastic-only Optimization . . . 211
10.3.1 Aeroelastic-only Rotor Blade Design Optimization . . . 211
10.3.2 Aeroelastic-only Stage Design Optimization . . . 211
10.3.3 Multi-component Optimization Comparison . . . 212
10.3.4 Overall Aerodynamic Performance . . . 216
10.4 Industrial LP Stage Aeroelastic Optimization . . . 217
10.4.1 Aeroelastic-only Rotor Blade Design Optimization . . . 218
10.4.2 Aeroelastic-only Stage Design Optimization . . . 218
10.4.3 Multi-component Optimization Comparison . . . 219
10.4.4 Concurrent Aeroelastic Stage Design Optimization . . . 222
10.4.5 Multi-disciplinary Optimization Comparison . . . 223
10.5 Industrial LP Stage Aerodynamic Optimization . . . 227
10.5.1 Aerodynamics-only Stage Design Optimization . . . 227
10.5.2 Concurrent Aerodynamic Stage Design Optimization . . . 228
10.5.3 Multi-disciplinary Optimization Comparison . . . 229
10.6 Summary . . . 233
11 Conclusions and Future Work . . . 235
11.1 Conclusions . . . 235

11.2 Future Work 236
References 239
A Unsteady Adjoint Navier-Stokes Equations 249
A.1 Domain Integral 255
A.2 Boundary Integral 256
B Unsteady Adjoint Navier-Stokes Solid Wall Boundary Condition 259
B.1 Inviscid terms 259
B.2 Viscous terms 261

List of Figures

2.1 Single-passage domain with flow periodic boundary conditions 44

3.1 Cylindrical coordinates frame of reference 59
3.2 Two-dimensional view of stress tensors applied to a fluid element 59
3.3 Three-dimensional mesh of a linear cascade passage 64
3.4 2D cartesian finite volume mesh. Element sizes are defined by $\triangle x$ and $\triangle y$ 66
3.5 Mixing-plane interface in a single-passage domain for rotor-stator and stator-rotor configurations 75
3.6 Illustration of flow characteristic jumps going across the interface 77
3.7 Relative position between domains at different time-instances t_1 and t_2 . . 80
3.8 Time-trace of the flow variable vector U over a period of unsteadiness . . . 80
3.9 Computational mesh of the linear cascade. Blade to blade view (left) and meridional view (right) 82
3.10 Time-averaged surface pressure coefficient 84
3.11 First harmonic pressure coefficient (amplitude and phase angles). 85
3.12 Aerodynamic damping coefficient at different inter-blade phase angles . . 85
3.13 First harmonic pressure coefficient (amplitude and phase angles). 86

4.1 Comparison of sampling distributions strategies 91
4.2 Example of mutation and crossover steps for two variables 94
4.3 Objective function convergence for both CFD and RSM evaluations 97
4.4 Design space at optimization cycle i with first set of candidates (red points) 99

4.5 Design space at optimization cycle $i+1$ with new set of candidates (blue points) . . . 100
4.6 Design space at optimization cycle $i+1$ with a triggered re-scale . . . 100
4.7 New design space after the re-scale at optimization cycle $i+1$ . . . 100
4.8 Relative error of objective function predictions. RSM and RRSM approaches 101
4.9 Flowchart of the Re-scaled RSM non-gradient method for validation purposes . . . 102
4.10 One-dimensional analytical function f(x) . . . 103
4.11 Design space of optimization cycle nº 1. Samples are shown in black while candidate point is red. . . . 104
4.12 1D f(x) and RRSM prediction at optimization cycle nº 1 after the first re-scale. . . . 105
4.13 1D f(x) and RRSM prediction at optimization cycle nº 1 after the second re-scale. . . . 105
4.14 Two-dimensional analytical function f(x) . . . 106
4.15 2D f(x) and RRSM prediction at optimization cycle nº 1. Samples are shown in black while candidate point is red. . . . 107
4.16 2D f(x) and RRSM prediction at optimization cycle nº 1 after first re-scale. 107
4.17 2D f(x) and RRSM prediction at optimization cycle nº 1 after second re-scale. 108

5.1 Imported finite element mesh in Abaqus . . . 113
5.2 Absolute displacements (mm) of first bending mode. . . . 113
5.3 Local work contours (J/cycle) of NASA rotor 67 blade (first bending mode) 114
5.4 Coordinate transformation done by Hicks-Henne functions. NASA rotor 67 117
5.5 Local contours of tangential coordinate change, $\delta_{r\theta}$. 1 perturbation node at $x_i = 0.5$ and $r_i = 0.5$ with factor $a = 1x10^{-4}$ . . . 117
5.6 Flowchart of the non-gradient optimization method with the enhanced RRSM technique . . . 122
5.7 Aerodynamic damping coefficient at different inter-blade phase angles . . 123

5.8 Computational domain of the NASA rotor 67. Blade to blade view (left) and meridional view (right) . . . 125
5.9 Computational domain of the DLR transonic compressor stage. Blade to blade view (left) and meridional view (right) . . . 126
5.10 Computational domain of an industrial low-pressure turbine stage. Blade to blade view (left) and meridional view (right) . . . 127
5.11 Hicks-Henne function perturbation nodes (reduced coordinates). NASA rotor 67 . . . 129
5.12 Convergence of objective function and constraints. NASA rotor 67 . . . 130
5.13 NASA rotor 67 pressure distribution contours. Pressure side. . . . 131
5.14 NASA rotor 67 pressure distribution contours. Suction side . . . 131
5.15 Local work contours on NASA rotor 67 (first bending mode). Pressure side. 132
5.16 Local work contours on NASA rotor 67 (first bending mode). Suction side . 132
5.17 Blade geometry comparison. NASA rotor 67 optimization . . . 133
5.18 Hicks-Henne function perturbation nodes (reduced coordinates). DLR stage . . . 134
5.19 Convergence of objective function and constraints. DLR stage . . . 135
5.20 Local Mach number contours at 50% span. DLR stage . . . 135
5.21 Local Mach number contours at 75% span. DLR stage . . . 136
5.22 Pressure coefficient distributions of DLR rotor blade . . . 136
5.23 Local work contours on DLR blade surface (first bending mode). Pressure side . . . 137
5.24 Local work contours on DLR blade surface (first bending mode). Suction side . . . 137
5.25 DLR stage profiles at 25%, 50% and 75% span locations . . . 138
5.26 Contours of tangential coordinate change, $\delta_{r\theta}$. DLR stage . . . 138
5.27 Hicks-Henne function perturbation nodes (reduced coordinates). LP turbine stage . . . 139
5.28 Convergence of objective function and constraints. LP turbine stage . . . 141

5.29 Local Mach number contours at 50% span. LP turbine stage 142
5.30 Pressure coefficient distributions of rotor blade. LP turbine aeroelastic redesign 143
5.31 Local work contours on LP turbine blade (first bending mode). Pressure side 144
5.32 Local work contours on LP turbine blade (first bending mode). Suction side 144
5.33 Stage profiles at 25%, 50% and 75% span. LP turbine aeroelastic redesign . 145
5.34 Contours of tangential coordinate change, $\delta_{r\theta}$. LP turbine aeroelastic redesign 145

7.1 Single-passage with adjoint periodic boundaries 172

8.1 Rotor-stator interface with adjoint characteristics $\widetilde{\lambda}_{ci}$. 182
8.2 Relative position between domains at two time instances t_1 and t_2 (physical time domain) 185
8.3 Time-traces of the adjoint variable $\lambda(t_a)$ and $\lambda(t_p)$ over a period of unsteadiness 186
8.4 Unsteady reflections from adjacent stators 188

9.1 Flowchart of the unsteady adjoint optimization system 197

10.1 Hicks-Henne function perturbation nodes (reduced coordinates). 81 variables 200
10.2 Convergence of objective function and constraints. DLR stator redesign . 202
10.3 Local work contours on DLR blade (first bending mode). Pressure side . . 202
10.4 Local work contours on DLR blade (first bending mode). Suction side . . . 203
10.5 Blade geometry comparison. DLR stator redesign 203
10.6 Convergence of objective function and constraints. LP turbine stator redesign 205
10.7 Local work contours on LP turbine blade (first bending mode). Pressure side 205

10.8 Local work contours on LP turbine blade (first bending mode). Suction side206
10.9 Blade geometry comparison. LP turbine stator redesign 206
10.10 Convergence of objective function and constraints. NASA rotor 67 208
10.11 NASA rotor 67 pressure distribution contours. Pressure side 208
10.12 NASA rotor 67 pressure distribution contours. Suction side 209
10.13 Local work contours on NASA rotor 67 (first bending mode). Pressure side 209
10.14 Local work contours on NASA rotor 67 (first bending mode). Suction side . 210
10.15 Blade geometry comparison. NASA rotor 67 210
10.16 Convergence of objective function and constraints. DLR stage redesign . . 212
10.17 Work sum convergence comparison 213
10.18 Local work contours on DLR blade (first bending mode). Pressure side . . 214
10.19 Local work contours on DLR blade (first bending mode). Suction side . . . 214
10.20 Stage profiles at 25%, 50% and 75% span. DLR stage redesign 215
10.21 Contours of tangential coordinate change, $\delta_{r\theta}$. DLR stage redesign 215
10.22 Local Mach number contours at 50% span. DLR stage redesign 216
10.23 Local Mach number contours at 75% span. DLR stage redesign 216
10.24 Pressure coefficient distributions of rotor blade. DLR stage redesign 217
10.25 Convergence of objective function and constraints. LP turbine stage aeroelastic-only redesign 219
10.26 Local work contours on blade (first bending mode). Pressure side. LP turbine stage aeroelastic-only redesign 220
10.27 Local work contours on blade (first bending mode). Suction side. LP turbine stage aeroelastic-only redesign 221
10.28 Stage profiles at 25%, 50% and 75% span. LP turbine stage aeroelastic-only redesign 221
10.29 Convergence of objective function and constraints. LP turbine stage concurrent aeroelastic redesign 222
10.30 Local work contours on blade (first bending mode). Pressure side. LP turbine stage aeroelastic redesign 224

10.31 Local work contours on blade (first bending mode). Suction side. LP turbine stage aeroelastic redesign 224
10.32 Local Mach number contours at 50% span. LP turbine stage aeroelastic redesign 225
10.33 Pressure coefficient distributions of rotor blade. LP turbine aeroelastic design 225
10.34 Stage profiles at 25%, 50% and 75% span. LP turbine stage aeroelastic redesign 226
10.35 Contours of tangential coordinate change, $\delta_{r\theta}$. LP turbine stage aeroelastic redesign 226
10.36 Convergence of objective function and constraints. LP turbine stage aerodynamic-only redesign 227
10.37 Convergence of objective function and constraints. LP turbine stage concurrent aerodynamic redesign 228
10.38 Local Mach number contours at 50% span. LP turbine stage aerodynamic redesign 230
10.39 Pressure coefficient distributions of rotor blade. LP turbine stage aerodynamic redesign 230
10.40 Local work contours on blade (first bending mode). Pressure side. LP turbine stage aerodynamic redesign 231
10.41 Local work contours on blade (first bending mode). Suction side. LP turbine stage aerodynamic redesign 231
10.42 Stage profiles at 25%, 50% and 75% span. LP turbine stage aerodynamic redesign 232
10.43 Contours of tangential coordinate change, $\delta_{r\theta}$. LP turbine stage aerodynamic redesign 232

List of Tables

3.1 Geometric parameters and operating conditions of the validation case (Huang et al., 2009) 84

5.1 Sensitivity analysis NASA rotor 67. Values and relative change of obj. function parameters 125

5.2 Sensitivity analysis DLR stage. Values and relative change of obj. function parameters 126

5.3 Sensitivity analysis LP turbine stage. Values and relative change of obj. function parameters 127

5.4 Optimization settings and computational costs 128

5.5 Final values, absolute and relative differences. NASA rotor 67 redesign . . 129

5.6 Final values, absolute and relative differences. DLR stage redesign 134

5.7 Absolute and relative differences after convergence. LP turbine stage aeroelastic-only redesign 140

5.8 Absolute and relative differences after convergence. LP turbine stage concurrent aeroelastic redesign 140

8.1 Work sum gradients. DLR compressor stage 189

8.2 Work sum gradients. LP turbine stage 189

10.1 Optimization settings and computational costs 200

10.2 Final values, absolute and relative differences. DLR stator redesign 201

10.3 Absolute and relative differences after convergence. LP turbine stator redesign 204
10.4 Final values, absolute and relative differences. NASA rotor 67 207
10.5 Final values, absolute and relative differences. DLR rotor redesign 211
10.6 Final values, absolute and relative differences. DLR stage redesign 212
10.7 Absolute and relative differences after convergence. LP turbine rotor aeroelastic-only redesign 218
10.8 Absolute and relative differences after convergence. LP turbine stage aeroelastic-only redesign 219
10.9 Absolute and relative differences after convergence. LP turbine concurrent aeroelastic redesign 222
10.10 Absolute and relative differences after convergence. LP turbine stage aerodynamic-only redesign 227
10.11 Absolute and relative differences after convergence. LP turbine stage concurrent aerodynamic redesign 228

Nomenclature

Roman Symbols

A	Area, non-dimensional vibration amplitude
a_i	Hicks-Henne function size factor
b_x, b_r	Hicks-Henne function tangential factor
c	Blade axial chord
c_f	Friction coefficient
C_p	Pressure coefficient
c_p	Constant pressure specific heat capacity
c_v	Constant volume specific heat capacity
E_s	Strain Energy
F	Differential weight, axial derivative flow variable vector
f	Flow residual perturbation, nominal frequency
G	Circumferential derivative flow variable vector
H	Radial derivative flow variable vector
h	Enthalpy

I	Objective function
k	Effective thermal conductivity, step size damping factor, reduced frequency
L	Reference length scale
$\dot{m}$	Mass flow rate
M	Adjoint analytical objective function, number of Latin samples
N	Number of blades
n	Normal vector, number of nodal diameters
$\bar{P}$	Mass averaged total pressure
$\bar{p}$	Mass averaged local pressure
p	Static pressure
P^*	Total pressure
Pr	Prandtl number
q	Heat flux
R	Ideal gas constant, residual operator of flow equations
r	Radial coordinate
r_d	Reduced radial coordinate
r_i	Crossover random defined number
Re	Reynolds number
$\bar{s}$	Entropy generation rate
S	Source terms of flow equations vector, surface integral

s Step size

T Static temperature, time period

t Time

U Flow variable vector

u Axial velocity

u_g, v_g, w_g Mesh velocity components

V Viscous flow vector, volume

v Circumferential velocity

v_n Normal velocity

v_x, v_θ, v_r Mesh vibration components

W Work sum, velocity magnitude

w Radial velocity

x Axial coordinate

x_d Reduced axial coordinate

x_i Independent variable scalar

Greek Symbols

α Design variable vector, phase angle

α_i Design variable scalar

β_i Response Surface Model polynomial coefficients

δ Logarithmic-decrement (aero damping)

$\delta_{r\theta}$ Hicks-Henne function

γ Adiabatic index

λ Adjoint variable

μ_l Laminar viscosity

μ_t Turbulent viscosity

ω Angular frequency of vibration

ϕ Characteristic jump

π Pressure ratio

ρ Density

σ Inter-blade phase angle (IBPA), wall stress

σ_i Objective function weight factor

τ Shear stress, pseudo-time

θ Circumferential coordinate

ξ Aerodynamic damping

Subscripts

$-1,0,1$ Values at phase angles $-\frac{\pi}{2}$,0,$\frac{\pi}{2}$ respectively

a Adjoint time related variable

A,B Harmonic coefficients

d Dissipative flux related terms

i Inviscid flux related terms

L Lower boundary

p Physical time related variable

U Upper boundary

v Viscous flux related terms

Acronyms / Abbreviations

AD Automatic Differentiation

ANN Artificial Neural Network

CFD Computational Fluid Dynamics

CPU Central Processing Unit

CR Crossover Probability

DE Differential Evolution

FEA Finite Element Analysis

FE Finite Element

GA Genetic Algorithm

IBPA Inter-blade Phase Angle

LP Low Pressure

NASA National Aeronautics and Space Administration

RANS Reynolds Averaged Navier-Stokes

RBF Radial Basis Function

RRSM Re-scaled Response Surface Model

RSM Response Surface Model

URANS Unsteady Reynolds Averaged Navier-Stokes

Chapter 1

Introduction

1.1 Introduction

The introduction of turbomachinery 75 years ago as a way of propelling military and, later on civilian aircraft, led to an increase of research in fluid mechanics and heat transfer centered on the gas turbine engine. Durable, efficient and high-performance engines were needed for a large new generation of aircraft performing a wide range of missions. Now, turbomachines are key components of our daily life in a wide range of applications (e.g. power generation, propulsions systems and mechanical drivers). Compressors input mechanical energy into the working fluid while turbines extract energy out of the working fluid. Oil and gas industries, power plants and airlines want to operate their gas turbines with as little fuel and environmental impact and as much output as possible. Each new generation of turbomachines needs to aim to operate at the highest turbine inlet temperature practically achievable for maximum thrust or power output and at the highest thermodynamic efficiency. As the turbine inlet temperature has been raised, accurate means of predicting the flow and heat transfer in the hot section were generally not possible, relying in expensive, but yet not accurate enough, series of tests. This problem has motivated analytical and computational research in fluid mechanics and heat transfer to minimize expensive testing and establish new ways to learn more

aero-thermal physics from either or both validated solutions and the test results. New design methods and computational methods have now developed to the point where fully three-dimensional, unsteady numerical simulations are, still with some limitations, possible to be applied in a turbine design process.

1.2 Challenges in Advanced Turbomachinery Design

The gas turbine industry is often considered as mature and the blade shape design process is considered to be well established. In spite of this, the mechanical and thermal design of a gas turbine is a highly complex task, costing millions of dollars and employing many engineers for months/years. However, the designers are still facing more and more challenges in terms of size, thermal efficiency and power output. During a blade design, advances are made through increasing levels of sophistication and detailed analyses. In terms of blading design, aero-thermal efficiency has already reached more or less its limits, and this has been achieved almost using mainly steady flow methods. Consequently, an aero-thermal designer may wonder why unsteady flow effects should be taken into account in blade shape design, due to its complexity and difficulty compared to its steady counterpart.

There are two types of problems related to unsteady flow effects: the blade-row interactions and stability margins that can affect aero-thermal performance and the aero-mechanic performance due to flow-induced vibrations (forced response and flutter). In the past, at conventional operating (or design) conditions, the time-averaged or 'mean' flow around blades has been considered in design. The blade-row interaction effects (unsteady losses) are generally small, while also the vibration problems are not significant either. Therefore, it has been enough to focus in aerodynamic performance with sufficient aero-mechanic stability margins. This largely aerodynamic dominated design approach, with constraints in other disciplines, is still the most popular and has been applied with success in turbomachinery design.

However, the highly competitive market is driving the development of turbomachines

towards highly loaded and compact configurations, which inevitably leads to enhanced interactions between the surrounding working fluid and the blades, even at design conditions. The intensified unsteadiness and its effects on time-averaged performance must be re-evaluated. Moreover, blades are now more vulnerable to flow-induced aeroelastic stability problems and, as a consequence, the aeroelastic performance, blade flutter and forced response have to be also checked more closely in the design process, challenging the widely used design rules on aero-mechanical stability margins.

Furthermore, in conventional designs, when blade stability margins are enough, blade aeroelasticity is typically evaluated after a successful aerodynamic design. If a blade flutter or forced response problem is found, a design modification is required in order to meet the margins. Compared to the aerodynamic shape designs, this largely trial and error procedure is restrictive, particularly for the aeroelastic part. It tends to be far less intuitive for an engineer to modify a blade shape design to control and manipulate unsteady flow to achieve a required aeroelastic design target (e.g. to increase aerodynamic damping). This is in clear contrast to the aerodynamic shape design where the fluid-physics based reasoning established in the past 20-30 years can readily help shaping a blade (e.g. sweep, blade lean) for a better aerodynamic performance. Moreover, any blade shape design modification for the aeroelastic performance may not be able to keep the original satisfactory aerodynamic performance. This lengthy aero-mechanical iterative path can be very costly both in money and time. It is increasingly the case that these two disciplines should no longer be treated either separately, or even just sequentially.

1.3 Methods for Design and Optimization

Conventionally, blade design used to be analysed by engineers with appropriate expertise to decide how to change blade shape so that its performance could be improved. But with the progress of computer technology and numerical methods for fluid dynamics, a new way to investigate the fluid motion inside turbomachines is possible. This has allowed assessment of the aerodynamic performance with more details in a much shorter

period and at lower cost. The modern development of computer technologies keeps producing everyday more powerful computers, which allow us to calculate more complex problems. Computational Fluid Dynamics (CFD) results, which are carefully executed and analysed, provide valuable information to help our understanding of complex flow problems (such as unsteady flow) and improve the design methodologies.

1.3.1 CFD Methods

The first numerical methods that were developed made use of simplifying assumptions in order to evaluate the flow through a turbomachine in affordable CPU time. These methods gave way to fully three-dimensional methods for steady flow in the late 1980s and have been developed and used until now. In fact, steady flow methods are still the workhorse of the CFD design in turbomachinery aerodynamics blade design.

However, turbomachinery blades work in a highly unsteady environment. Blade rows are in relative motion with each other, with strong interactions between them. With the current competitive market, it is highly important to analyse aero-mechanical behaviour of the blades in the blade design process. A full time-domain unsteady three-dimensional method is still expensive and prohibitive for turbomachinery design optimization. This has motivated the development of efficient methods that can evaluate the aero-mechanical features under realistic computational times.

1.3.1.1 Efficient Methods for Aero-mechanical Design Optimization

The first approaches truncated the computational domain to single blade passages in the circumferential direction and the difference in unsteadiness between adjacent blade passages is modelled with phase-shifted periodic boundary conditions.

Other methods made use of Fourier series to model periodic unsteadiness with a harmonic treatment that avoided expensive time-integration and solved the flow equations in the frequency domain instead. These first methods neglected the nonlinearities in the flow by solving the linearised flow equations (Giles, 1992; Hall and Lorence, 1993; Verdon

and Caspar, 1982). More recently, nonlinearities started to be captured and these methods are still very popular (Chen et al., 2001; He and Ning, 1998) and (Hall et al., 2002).

1.4 Optimization Methods

With these new numerical methods, there has also been a large development of the so called optimization algorithms. These algorithms try to maximize the performance of a design by modifying its parameters with minimal intervention from the designer. There have been many types of optimization methods used historically for turbomachinery. These methods can be categorized in different ways. For example, they can be classified as inverse design methods and direct design methods, deterministic methods and non-deterministic methods, gradient-based methods and non-gradient-based methods, high-order and low-order methods, etc. Moreover, one optimization method can belong to more than one of the above categories.

In the present work, the attention is focused on direct design methods, where the geometry of the surfaces to be optimized is changed iteratively until optimum flow performance is obtained.

In any optimization method, the objective function is the value that the algorithm aims to improve or decrease. This function relates the target improvement to be achieved with the main target parameters such as efficiency, aeroelastic damping, etc.

The most important elements of a design optimization method are:

1.4.1 Flow Solver

The flow solver is the basis for which the whole optimization system is built. It gives information of the flow field in order to determine how a blade profile should be changed. But also it verifies that a change in the blade profile will deliver an expected performance gain. It is important that a flow solver can predict the performance of interest with an adequate accuracy. The flow solver is used every cycle of the design optimization and,

because of that, it must be both fast and accurate.

1.4.2 Shape Parameterization

Modern turbomachinery blades have rich three dimensional features such as twist, lean, sweep and so on. Many different variables can be involved in the description of a blade. Shape parameterization is of vital importance in turbomachinery blade design optimization. Differences in the way a shape is parameterized can lead to different final results and improvements. There is a computational cost associated with the addition of more variables. A good shape parameterization must be capable of producing realistic blades and covering most of the important geometry features of the blade within an efficient number of variables. One way to parameterize the shape is by parameterizing the shape perturbation instead. The advantage of this is that is possible to return to the original shape just by setting the perturbation back to zero. An example of this is to represent the shape perturbation as a linear superposition of "bump" functions (Hicks and Henne, 1978; Hicks et al., 1974). The way this shape perturbation is translated to the input files and the mesh is also important.

1.4.3 Optimization Strategy

Many different methods exist to solve the optimization problem. They can be subdivided depending on the order of the derivative of the objective function that is used to predict the amount of change a design variable can produce. Zero order methods, also known as non-gradient methods, use only the function values in their search for the minimum. Most of these methods make use of artificial intelligence to mimic the designer. First and second order methods, also called gradient methods, use respectively the first and second order derivatives of the objective function to drive the search towards the optimum. In the present work, both gradient and non-gradient methods are considered.

1.4.3.1 Gradient-based Methods

In a gradient-based design optimization, the sensitivities of the objective function to design variables are required to determine the change of a blade shape in a way that improves the blading performance. This blade change is usually evaluated by solving the flow governing equations or its linearised version, and it has to be done for each design variable. The time cost of gradient evaluation for many practical cases where there are hundreds or even more design variables can be prohibitive. The common drawback of these methods, intrinsically related to their gradient-based nature, is their tendency to be trapped in local extremes.

There are different types of gradient-based methods according to the way in which gradients of an objective function to design variables are calculated. Some examples are the finite difference method, the linearised method, the complex variable method and the adjoint method. The gradients obtained by the finite difference method normally suffer from step size dependence and computational cost. However this method has been used frequently due to its simplicity. The complex variable method is an implementation of the finite difference method with complex variables, which reduces the shortcoming of step size dependence. The linearised method evaluates the gradient by linearising the flow equations with respect to a design variable.

The main disadvantage of these three methods is that the time cost for evaluating the gradients is directly proportional to the number of design variables, which prevents the use of a large number of design variables. This and their proneness to be trapped in local minimums is the cause why non-gradient methods are often favoured over them.

The adjoint method, in contrast, offers a way to calculate gradients of an objective function to a large number of design variables at a cost that is almost independent of the number of design variables. This feature of the adjoint method makes it very attractive for design optimization with a large number of design variables. However, due to its complexity in implementation and difficulty in interpretation, the adjoint method has

not been widely applied.

1.4.3.2 Non-gradient-based Methods

Non-gradient methods only need values of an objective function, and not its gradient. These stochastic systems analyse statistically, with distribution patterns, possible blade shapes that could yield a performance improvement. Non-gradient methods can be very expensive, as CPU time can be wasted evaluating the flow fields of designs that result in no final improvement. One way of alleviating this cost is by using approximation models, also known as surrogate models or metamodels. These models mimic the behaviour of the flow solutions as closely as possible while being computationally cheaper to evaluate. Surrogate models are built based on intelligently chosen data points. However, the cost of setting up a surrogate model is still proportional to the number of design variables, and their applications are thus restricted to a moderate number of variables which limits their applications for realistic multi-bladerow 3D blade geometries. Nevertheless, non-gradient methods are more popular than gradient methods mainly due to their easy implementation and capability of finding a global optimum.

1.5 Problem Statement for Present Work

Current turbomachinery blade designs increasingly point to a need to look at aerodynamics and aeroelasticity in a more simultaneous (or concurrent) way in order to avoid long time-scales and expensive iterations in a design process. However, there has been little or no development for this type of concurrent optimizations. Most of the multidisciplinary (aerodynamic-aeroelastic) methods that have been developed are simply aerodynamic optimizers with mechanical constraints, and the few concurrent optimizations that have been performed only work for single-bladerow cases (as reviewed in Chapter 2). This can be restrictive as the multi-bladerow interaction effects (which may be considerable), are not included. In fact, Ekici and Hall (2007) highlighted the influence of neighbouring blade rows on the unsteady aerodynamic response of a blade row in a

multistage turbomachine. Moreover, Li and He (2005) showed that up to 100% change in the rotor aeroelastic damping can result from the stator reflection for a transonic compressor, which was supported recently by Rahmati et al. (2013).

It is clear that in order to perform efficient aerodynamic and aeroelastic optimization, turbomachinery blades must be optimized in a multi-bladerow environment, where bladerow interactions can be timely taken into account during an optimization process. In order to do this in affordable time, the flow solver must be efficient enough to be used on each cycle of the optimization process. Aeroelasticity sensitivity calculations need to solve the unsteady flow equations, which may increase the computational time up to three orders of magnitude (He, 2008) for conventional time-domain methods compared to steady flow calculations. Moreover, an efficient flow solver will not be enough; the optimization method must be efficient too, as the very large number of variables needed for a realistic shape design optimization points to the requirement for the adjoint method as a candidate for efficient sensitivity calculations.

In summary, there is a case for a design optimization method simultaneously possessing the following desired attributes:

- Concurrent aerodynamics and aeroelasticity design optimization.
- Multi-bladerow or multi-component capability for unsteady flows.
- Adjoint gradient-based method.

As it will be further enforced in the literature review, such a method is not available. This in fact, is the primary motivation of the present new method development.

1.6 Thesis Overview

This research work is focused on two more specific objectives: the first one is to address the lack of concurrent aerodynamic-aeroelastic blade optimization in a multi-bladerow

environment; the second is to perform concurrent optimization for realistic turbomachinery blade geometries within industrial timescales.

The Nonlinear Harmonic Phase Solution Method (He, 2008) is chosen as the flow solution method of this work because of its capability to calculate the aeroelasticity features of interest (blade flutter) and the main flow aerodynamic performance in steady flow timescales. It also provides a good basis on which an adjoint system can be formulated in an intuitive and similar manner.

The design optimization work is started with a more generic non-gradient method. A surrogate model, with an improved version of a quadratic Response Surface Model, has been developed as an alternative optimization method to better illustrate the differences in complexity and performance compared to the adjoint method. The new Re-Scaled Response Surface Model has shown marked convergence and performance improvements against traditional surrogate models based on standard response surface methodologies. The need for efficient sensitivity calculation has been highlighted in the non-gradient design optimization case studies.

The adjoint method is exploited in this work as the main optimization strategy, as it is capable of performing multi-bladerow optimization cost-independently of the number of variables and so, the only method that could accomplish efficient multi-bladerow optimization within the current levels of computational power.

The continuous adjoint system, that has been newly re-derived from the Nonlinear Harmonic Phase Solution Method, has helped to better understand physically and concisely the relation between the adjoint equation and its unsteady periodic boundary conditions with those of the flow equation. This unique interaction with the Nonlinear Harmonic Phase Solution Method has enabled the development of an adjoint interface treatment to perform concurrent optimization in an unsteady multi-bladerow environment.

In terms of the thesis organization, after the overall background perspectives and problem statement introduced in Chapter 1, the literature highlighting the state of art leading

to the present work is reviewed in Chapter 2. Next, the flow solution method is presented and explained in detail in Chapter 3. The first optimization method that is introduced is the non-gradient-based methodology enhanced with the re-scaled response surface approach in Chapter 4. Its application to concurrent optimization of three case studies is shown in Chapter 5. The adjoint method is introduced next, first with an illustration of the method for steady flows in Chapter 6 and next with a full thorough derivation of the unsteady adjoint method in Chapter 7. The multi-bladerow treatment used in this work is explained in Chapter 8 along with a validation of the unique adjoint harmonic interface treatment. Lastly, the adjoint optimization system is described in Chapter 9 and its optimization results are shown in Chapter 10. To conclude, Chapter 11 summarises the research work and gives some suggestions for future work.

Chapter 2

Literature Review

This chapter focuses first on the development of numerical methods for turbomachinery, and more specifically, three-dimensional methods for steady and unsteady flows. Next, the development of different automated methods for aerodynamic shape optimization is reviewed, including the adjoint method. Because of the relevance of the adjoint method to this research, a specific section dedicated to the development of adjoint methods in turbomachinery is presented. Lastly, the most recent multidisciplinary and concurrent optimization methods for single bladerow configurations are reviewed, concluding the state of art for relevant developments so far.

2.1 CFD in Turbomachinery Design

Numerical methods are the workhorse of turbomachinery design. The application of numerical methods to turbomachinery has a lineage that can be traced back to the 1940s, at a time when methods were formulated before digital computers that could use them and the computational power that we have today would have been simply unimaginable.

How could those 1970s computers tackle these problems when even modern computers are struggling to compute unsteady flow? Simplifying assumptions can be made which allows reduction of the time required to evaluate the flow through a compressor or tur-

bine blade passage from thousands to seconds of CPU time. In fact, through-flow and quasi-three-dimensional blade-to-blade methods dominated the subject until fully 3D methods for steady single-bladerow became available in the late 1970s and steady multistage simulations were performed in the 1990s.

Three-dimensional CFD methods for steady flow solutions in turbomachinery have been maturing for about twenty years, and along through-flow and quasi-three-dimensional methods, they are an important tool for the designer who can choose between different balances of accuracy and run-time. In this chapter, the development of fully 3D methods will be reviewed, first for steady flow and next for unsteady flow.

2.2 Three-dimensional Steady Flow Methods

There are many features in turbomachinery blade shaping that are fully three-dimensional (blade lean effects, tip flows, secondary flows, leaks and bleeds, etc.) and cannot be predicted by through-flow or quasi-3D approaches

2.2.1 Single-bladerow steady flow calculations

Time-marching the Reynolds-averaged Navier-Stokes equations is the most popular approach to calculate fully 3D flow fields. Many different algorithms have been developed, but the most popular is time-marching with local time stepping and multi-grid to speed up convergence to a steady solution (Dawes, 1986; Denton, 1985).

The effects of 3D blade features like blade twist, lean and sweep, as well as tip leakage flows, were well predicted with single-bladerow calculations and promptly added as part of designers methods (Denton and Xu, 1999). However, there is a limitation, for single-bladerow methods, on the ability to predict secondary flows if the shape and thickness of the inlet boundary layer is not known beforehand.

2.2.2 Multi-stage steady flow calculations

In turbomachinery, adjacent blade rows are in relative rotation, and so there is an unsteady interaction between them. Because an unsteady 3D calculation would be too expensive, some early approaches for predicting the time-average of the unsteady flow from a steady calculation were developed. The first attempt (Denton, 1979) simply averaged circumferentially the flow leaving one row before entering into the next one. This averaging assumed that there was a mixing process occurring instantaneously at a 'mixing plane' instead of gradually through the downstream row. The issues with this approach were, firstly, that there was no guarantee that the mixing loss generated should match that generated by the gradual mixing that occurs in reality. Secondly, that the averaging process generated errors due to the imposition of a circumferentially uniform flow too close to the respective blade trailing and leading edges, which lead to spurious reflection errors. However, improved mixing plane treatments have been developed (Denton, 1992; Saxer and Giles, 1993) that addressed the reflectiveness and conservation issues of early approaches and allows a mixing plane to be located close to the leading and trailing edges of the respective bladerows. After these developments, 3D multistage steady flow methods have been used consistently to predict the overall machine performance.

A more rigorous approach than the mixing plane was studied by Adamczyk (1985), who attempted to deal with the averaging of the unsteady flow systematically by time-averaging the nonlinear flow equations results in a set of time-averaged equations that contain deterministic stress terms. This research illustrated the possibility of calculating unsteady flow features without losing the advantage of steady flow timescales and introduced the concept of deterministic stresses that has been widely used in the unsteady flow method developments that are reviewed next.

2.3 Three-dimensional Unsteady Flow Methods

The most important issue regarding the development of optimization methods for aeroelasticity is the computational cost of the unsteady flow equations; the flow equations have to be solved at each design cycle, and therefore, the CFD analysis has to be computationally efficient. However, solving the unsteady RANS equations directly is very time-consuming.

There have been active developments in many different levels of modelling fidelities. They range from the direct time-domain methods to the frequency-domain methods. In a design optimization context, it is important to choose a method which, on one hand is efficient for unsteady flow aeroelasticity, and on the other hand provides a good basis on which automated optimization methods can be developed without too much effort. These methods and their suitability to work with optimization methods are reviewed next.

2.3.1 Direct Unsteady Solution

The most straightforward way to calculate a solution for the URANS equations is by using a time-marching method in the time domain (He, 1989; He and Denton, 1994). The disadvantage of this method is its huge computational cost. An unsteady flow solution takes 10-20 times more computational time to calculate than a steady flow solution, even for the same mesh (Graf et al., 1999).

Moreover, the difference in blade counts between neighbouring rotor and stator blade rows defines the circumferential wavelength of the unsteady disturbances. As a result direct blade-to-blade periodicity can no longer be assumed. In order to model the correct length scale of unsteadiness, a whole annulus must be calculated, and this adds an extra factor between 20 and 100. Hence, a nonlinear unsteady flow solution calculated in a time domain is usually two to three orders of magnitude more expensive than a steady flow solution (He, 2008).

2.3.2 Single-passage (Phase-shift) Flow Methods

The large computational resource requirement can be reduced if the computational domain is truncated to a single blade row in the axial direction or a single blade passage in the circumferential direction.

The unsteady flow field among passages is not identical as in the steady flow field. There is a difference between two adjacent blade passages with a fixed phase (inter-blade phase angle) that can be known beforehand. Therefore, if a single blade passage computational domain is taken, the flow variable at the lower periodic boundary and the flow variable at the upper periodic boundary have to satisfy a phase-shifted (or time-lagged) periodicity. This is shown in Figure 2.1 where a single blade passage computational domain is represented.

Different methods have been developed to account for this phase-shifted periodic condition. The first method, proposed by Erdos et al. (1977), is known as the Direct Store method, where flow variables at the periodic boundaries are stored for one period of time and subsequently used with the current solution to correct the values at each boundary according to the phase-shifted periodicity. The drawback of this method is its large computer memory requirement, which is inversely proportional to the frequency.

Giles (1988) proposed a time-inclination method for the phase-shifted periodic condition. This method has the time plane inclined along the blade pitchwise direction according to the corresponding inter-blade phase angle. The phase-shifted periodicity can be directly applied by equating flow variables at the periodic boundary of the transformed computational domain. The benefits of this method are that there is no extra storage required and the convergence rate is much faster than the Direct Store method. However there is a strong restriction on the time inclination angle, thus the computational domain required (He, 2010).

Fourier series can also be used to implement the phase-shift periodicity at the boundaries of a single-passage domain, despite the flow being nonlinear. In fact, the nonlin-

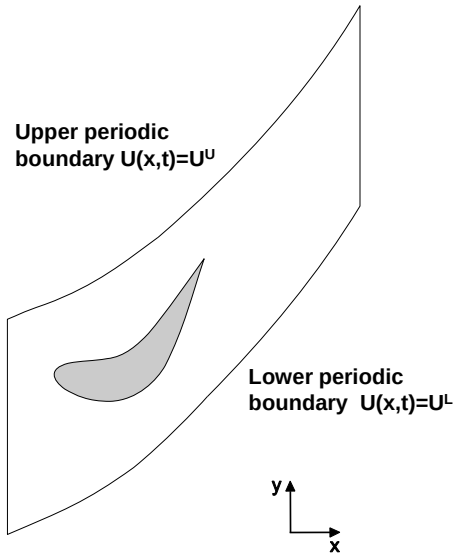


Figure 2.1: Single-passage domain with flow periodic boundary conditions

ear periodic flow variable at a known frequency can be expressed by Fourier series, with the zero order term corresponding to the time-averaged value. The Shape Correction method developed by He (1989), represents the lower and upper periodic boundaries of the unsteady domain in terms of a timewise Fourier series of Nth order, but the first few harmonics should be sufficient for most practical situations.

The original method has been extended to three-dimensional applications with rotor-stage configurations (Dewhurst and He, 2000) and multiple disturbances (Li and He, 2002). However, in terms of computational cost, the Shape Correction method is still quite time-consuming. For a test case with 2 blade passages (Huang, 2006), it still needs about 30-50 times more CPU time than a steady flow solution with the same mesh.

2.3.3 Harmonic (Frequency-domain) Methods

The unsteadiness of interest for aeroelasticity in turbomachinery can be attributed to time-periodic aerodynamic disturbances due to blade-row interactions such as upstream wakes and inlet/exit distortions, and those due to mechanic disturbances like blade vibration. As a result, the induced unsteadiness has a strong temporal periodicity. This

periodicity is a very important feature which can be used to greatly reduce time costs when solving the unsteady flow equations. Frequency-domain methods are a more efficient alternative to time-domain methods, and make use of the Fourier series to model periodic unsteadiness with a harmonic treatment and avoid the time-accurate integration that is responsible for the computational inefficiency of time-domain methods.

2.3.3.1 Linearised harmonic Modelling

The methodology of linear harmonic methods starts by assuming that the flow variable can be decomposed into a steady and an unsteady part. Assuming that the magnitude of the unsteadiness is much smaller than the steady part, the nonlinear effects can be neglected. As a consequence, the solution to these unsteady time-linearised equations is now expressed as a linear superposition of harmonics. With this harmonic assumption, the unsteady time-linearised equations can be converted to the frequency domain into steady-like equations with harmonic amplitude. The resulting system of equations can be solved efficiently, with a cost that is twice of the steady flow solution with just one harmonic. These types of methods are known as Linearised Harmonic Methods and they have been approached by different authors (Giles, 1992; Hall and Lorence, 1993; Verdon and Caspar, 1982). The drawback of this method is that nonlinear effects cannot be captured by this method, as it has been built with the assumption of small disturbances and linear unsteadiness.

2.3.3.2 Nonlinear harmonic Modelling

The nonlinear effects can be significant when there is shock or flow separation in the flow field, which is a common situation for modern turbomachines with high loads and high speed. He and Ning (1998), and later on Chen et al. (2001) developed the Nonlinear Harmonic Method in order to capture these nonlinearities in the flow field. The main difference compared to its linear harmonic counterpart is that the flow variable is decomposed in a time-averaged and an unsteady part. The unsteadiness is no longer neglected and the flow decomposition leads to two sets of equations, one for the time-

averaged part and the other for the unsteady part. The time-averaged equations contain extra deterministic stress terms, which are dependent on the unsteady solution. Using the harmonic formulation, the deterministic stress terms can be approximated by coupling the solution of the time-averaged and the unsteady equations. This method is still more efficient than nonlinear time-domain methods, but it neglects higher order coupling among harmonics. Effectively, a nonlinear harmonic method in its original form is a first order harmonic balance.

A full harmonic balanced analysis was further developed by Hall et al. (2002) and McMullen (2003) to extend the technique to the whole computational domain. The Harmonic Balance Method expresses an unsteady flow solution as a full Fourier series where the time derivative is worked out using the unsteady flow solution at equally separated phases of unsteadiness. At those discrete phases, Fourier transformation is used to get back the time derivatives of the flow variables. A selling point of this phase-solution method is the treatment of turbulence models. In the Nonlinear Harmonic Method, the turbulence model has to be frozen in order to avoid linearisation of the turbulence eddy viscosity. With a phase-solution approach, the turbulence transport equations can be solved in the same way as for other full nonlinear time-domain solutions. More demonstrations and applications have been made by Ekici and Hall (2007, 2008). Particularly for the study of blade flutter and forced response with a linear structural dynamic mode, a harmonic solution retaining only one harmonic would be sufficiently accurate for the unsteady flow of interest.

A simpler approach, when there is only one harmonic of interest, is the Nonlinear Harmonic Phase Solution Method proposed by He (2008). This new approach no longer uses equally spaced phases, but two particular phases $\left(0, \frac{\pi}{2}\right)$ when only linear unsteadiness is considered, or three particular phases: $\left(\frac{-\pi}{2}, 0, \frac{\pi}{2}\right)$ when accounting for nonlinear unsteadiness (i.e. the coupling between the harmonic and the time-averaged flow). The Nonlinear Harmonic Phase Solution Method effectively translates a problem of solving unsteady flow to equivalently solving two or three steady-like flows, and hence, it is highly

efficient. Furthermore, an interface treatment was developed by Rahmati et al. (2013) to extend the solution method to a multi-bladerow environment and calculate the unsteady reflections from adjacent rows. This method has been applied with success to aeroelastic design optimization (He and Wang, 2011), and it is the methodology of choice for the present work.

Nonlinear Harmonic Methods, due to inclusion of nonlinearity, effectively address the convergence difficulties of the conventional time-linearised methods as shown by He (2008). The unsteady nonlinear harmonic flow solver will provide the time-averaged aero-thermal and aeroelastic performance at the same time. This has a clear appeal in multidisciplinary blade design optimization. When the number of harmonics retained is low, both the Harmonic Balance Method and the Nonlinear Harmonic Phase Solution Method are equally as efficient as the time-linearised methods.

A distinctive advantage, however, comes when considering their suitability for developing the adjoint method. In order to formulate the corresponding adjoint system, the flow equations need to be linearised with respect to a design variable. As a consequence, using time-linearised methods means that the adjoint system will include two sets of equations: one corresponding to the time-averaged flow equation and another corresponding to the linear harmonic flow equation. This will add extra complexity to the problem and make it counter-intuitive. Using one of the frequency domain nonlinear harmonic methods instead, where the flow equations have not been previously linearised, makes the development of the adjoint system more intuitive and consistent with its steady counterpart.

2.4 The Adjoint Method in Turbomachinery

The adjoint method has been proven as a cost effective approach to calculate gradients of an objective function to a large number of variables. The first application to aerodynamic design optimization based on CFD was pioneered by Jameson (1988, 1989), and since then, the adjoint method has been widely used in many disciplines of fluid mechanics

(Mohammadi and Pironneau, 2004). However, it has not been until the last decade that it gained attention in the CFD community. This can be due to the complexity of the method in its formulation and implementation, as Giles and Pierce (2001) pointed out.

There are two ways of developing the final adjoint system: the continuous and the discrete adjoint method. In the continuous adjoint method, the nonlinear flow equations are linearised first with respect to a design variable. Next, the adjoint system is derived from the linearised flow equations and subsequently discretized. In the discrete adjoint method, the flow equations are discretized first from their partial differential form and linearisation is applied next to define the discrete adjoint system. The advantage of the discrete adjoint method is that it can calculate exact gradients of a discrete objective function with respect to design variables, which ensures a quick and full convergence. However it is more complex to develop. Efforts have been spent to reduce the complexity by using Complex Variables (Nielsen and Kleb, 2005) and Automatic Differentiation, AD (Mohammadi, 1998). The discrete codes also require more storage and CPU requirements than their continuous counterparts, as pointed out by Nadarajah and Jameson (2000). Particularly, an AD generated code will need three times more CPU time than its flow solver (Duta et al., 2007; Thomas et al., 2003), while the time cost of a continuous adjoint solver has been found to be similar to that of its flow solver. In the continuous adjoint method, there is freedom to discretize the adjoint equations in a similar way to the flow solver equations. The gradients by a continuous adjoint solver are subject to discretization errors for the flow equations and the adjoint equations, which can be reduced with mesh refinement, but still will cause the difference in value of the exact gradients of the objective function with respect to design variables.

Both methods can produce good designs if gradients given by the continuous approach are preconditioned (Giles et al., 2005; Giles and Pierce, 2000). More details about the advantages and disadvantages of the two approaches can be found in Giles and Pierce (2000) and Nadarajah and Jameson (2000). An adjoint system based on the RANS flow equations may have the eddy viscosity frozen to reduce complexity. There has been re-

search (Kim, 2007; Kim and Nakahashi, 2005; Marta and Shankaran, 2013; Nielsen and Anderson, 1999) on the benefits of doing this. The most recent conclusion was that there is a slight difference of gradients produced between freezing the eddy viscosity and not freezing it but the impact on the optimum solution is negligible and it does not change the final design (Marta and Shankaran, 2013). Moreover, the frozen eddy viscosity adjoint solver was faster and less expensive than the adjoint solver with full eddy viscosity update. Freezing the eddy viscosity seems to be a compromise between accuracy and complexity when developing an adjoint method.

2.5 Aerodynamic Design Optimization

In turbomachinery the flow field is intrinsically unsteady, due to the relative movement of adjacent blade rows. At early stages of design, blading aerodynamics methods are mainly based on steady flow (Cumpsty and Greitzer, 2004; Larosiliere et al., 2002). Within the blade passage, the flow field is assumed to be steady with the averaged effect from the adjacent rows. The steady flow assumption proves to work well and it is demonstrated by the vast range of blade designs and optimization methods that have achieved good aerodynamic performance without calculating the unsteady flow equations. The unsteadiness inside a turbomachine, particularly at its design operating condition, has a small effect on the time-averaged solution. To obtain a solution of the unsteady flow equations is much more computationally expensive than of the steady flow equations (He, 2008). Despite there having been great progress on computational power and development of efficient numerical methods for unsteady flows (as illustrated in section 2.3), the gap is still considerable and for some cases, a fully unsteady solution may not be easily justified. This is more relevant when these methods are used in optimization methods, where flow field calculations are called multiple times on each design cycle. Automated blade design optimization has been actively developed for modern gas turbines. Because of the simplicity described above when computing steady flows, first optimization methods were targeted to improved only aerodynamic performance from the time-averaged

flow. These methods, which are reviewed below, were the first early optimizers to be developed in the field of turbomachinery, initially for single-bladerow and later on for single-stage cases with verifiable performance gains relative to benchmark cases.

2.5.1 Non-gradient Methods in Aerodynamic Design

Due to its simplicity and possibility of finding the global optimum, non-gradient methods are predominant over gradient methods. Among the first methods, Chen et al. (2005) compared a non-gradient method against a gradient method for the optimization of a transonic compressor rotor and Yi et al. (2006) showed one of the first optimizations of 3D blades maximizing the adiabatic efficiency of transonic compressors. Non-gradient methods have been continuously improved due to their computational cost shortcoming. One of the first implementations of surrogate modelling can be found on the work done by Samad et al. (2006), where different models were tested and combined to enhance the performance of the NASA rotor 37. Since then, metamodels of all types have been actively developed to support designers to find high performance shapes for realistic geometries like axial compressors (Okui et al., 2013), radial turbines (Mueller et al., 2012), radial diffusers (Verstraete et al., 2012) and even marine propellers (Taheri and Mazaheri, 2012). The method used for radial diffusers by Verstraete et al. (2012) was recently improved for multi-objective (or multi-point) optimization within design and off-design conditions (Verstraete et al., 2014). There is little work, however, on multi-bladerow shape optimization with non-gradient methods. This is mainly because the number of variables can be too big if more than one row is shaped. The work from Lee et al. (2014) using a Kriging technique is one of the few non-gradient multi-bladerow shape optimization studies.

2.5.2 Adjoint Method in Aerodynamic Design

The history of adjoint method in turbomachinery design optimization is quite short. The adjoint solution of 2D Euler flow equations for turbomachinery aerodynamics was still

a topic of research ten years ago (Arens et al., 2005; Li et al., 2006). The first efforts in the development of adjoint methods for turbomachinery aerodynamics were made by Yang et al. (2003) and Wu et al. (2003) who applied the continuous adjoint method to 2D blade aerodynamic design optimization. Adjoint solvers for 3D turbomachinery were pioneered by Wu et al. (2005) and Papadimitriou and Giannakoglou (2006). Most of the research that has been done after has been limited to single-bladerow domains and benchmark cases (Luo et al., 2010; Mousavi and Nadarajah, 2011). In order to exploit the advantage of the adjoint method over non-gradient methods, Wang and He (2010a,b) developed the first adjoint approach for multi-bladerow and multi-stage optimization. They developed an adjoint mixing-plane interface as a

part of their flow solver in order to pass the time-averaged adjoint information among rows. The method was applied to single-bladerow benchmark cases like the NASA rotor 67 and a three-stage industrial compressor. A more recent adjoint design optimization with multi-bladerow shape optimization was conducted by Walther and Nadarajah (2012), who developed a non-reflecting mixing-plane formulation for a discrete adjoint system. Moreover, aerodynamic shape optimization is still a common topic of research with more interesting features to be developed, as it is shown in the work presented recently by Walther and Nadarajah (2015), who improved their previous work with a more refined discrete adjoint approach with the use of radial basis function for complex multi-block grids.

Despite a need to look at other disciplines, it seems clear that aerodynamics only (single-discipline) optimizers are still popular, and yet to be fully developed for multi-stage and multi-objective optimization with both non-gradient and gradient methodologies.

2.6 Aeroelastic Design Optimization

Aeroelasticity is the branch of physics that deals with the interactions between elastic and aerodynamic forces that take place when an elastic body is exposed to a fluid flow. In the context of turbomachinery blades, their mechanical integrity is influenced by blade

flutter and forced response.

In this work, only blade flutter is considered. Blade flutter is a dynamic self-excited instability of an elastic structure in a fluid flow, caused by an energy exchange between the fluid flow and the blade. Once a blade starts to vibrate, it induces unsteady aerodynamic forces which in return act on the vibrating blade. If the energy is absorbed by the blade, the aeroelastic instability will amplify the blade vibration, eventually causing blade failure. Blades must be designed carefully to avoid flutter by dissipating the net energy that is being exchanged. This is in fact an interaction process between blade structure dynamics and flow aerodynamics. The blade flutter prediction process can be split into two uncoupled problems in two domains. Due to the large mass fluid ratio, the blade vibration is independent of the unsteady aerodynamics and its vibration mode shape and frequency can be obtained through a finite element (FE) analysis. The unsteady aerodynamic forces induced by the blade vibration are obtained by solving the unsteady flow equations and the energy exchange between the fluid and the blade, which is determined by the energy method based on the unsteady solution. The uncoupled method has been validated by many researchers and widely used in turbomachinery aero-mechanics (Duta, 2002; Huang, 2006; Reddy and Srivastava, 1999; Srivastava et al., 2002).

2.7 Multidisciplinary Optimization Methods

The main topic of interest of this review is on combined aerodynamic and aeroelastic optimization. Parallel to the advanced development of aerodynamic shape optimization methods, there have been efforts addressed towards multidisciplinary methods capable of looking at both aerodynamics and aeroelasticity.

2.7.1 Constrained (Mechanical) Optimization

A starting point for researchers was to add mechanical constrains to the objective function in a way that the optimization method could behave similarly to the aerodynamic only optimization process, but having mechanical constraints and stability margins checked

at the end of the optimization. If the margins were not met, the design was discarded. The first approaches to constrained optimization were tested for benchmark cases. For the NASA rotor 67, Li et al. (2006) developed an aero-mechanical optimizer with a direct search method that scanned the constrained design space. The results showed an increase of 0.4% in efficiency while keeping maximum stress at allowed levels. Pierret (2005) optimized the rotor 67 blade with a Genetic Algorithm (GA) method and a Radial Basis Function (RBF) as metamodel for 3 operating conditions: near peak efficiency, chocked mass flow and near stall. His results were compared against aerodynamic-only results and they showed an improvement in the control of vibrations and maximum stress. The use of a RBF allowed the algorithm to converge 20 times faster. Lian and Liou (2006) also investigated the rotor 67 blade. Their new design decreased the maximum of von Mises stress while achieving an increase in pressure ratio of 1.88%.

The development of multidisciplinary optimization methods for realistic geometries has been started only in recent years. In relation to the optimization procedure, most existing optimization methods are still zero order with surrogate models. In fact, one of the earliest developments for realistic geometries was performed by Verstraete et al. (2010), who obtained significant reduction of the centrifugal stresses in a radial compressor blade without penalizing the performance by using a non-gradient GA supported by surrogate Artificial Neural Network (ANN). In addition, Chahine et al. (2012) compared the performance of different surrogate models in multidisciplinary optimization for the same case, and illustrated the impact of surrogate models in the overall computational efficiency of non-gradient methods. Using different versions of this ANN, a turbocharger radial turbine was optimized by Mueller et al. (2012), showing an improvement in both efficiency and moment of inertia. Other cases are the redesign of a turbocharger compressor by Van den Braembussche et al. (2012), a highly-loaded fan by Joly et al. (2012) with a 65% attenuation of the stress at the cost of 1% aerodynamic efficiency and the recent work by Astrua et al. (2012), who optimized the efficiency and off-design performance of an axial transonic compressor while maintaining its structural integrity. Deng et al. (2013) showed more recently the performance of the Kriging metamodel to im-

prove the overall aerodynamic performance with mechanical constraints for a transonic fan blade.

2.7.2 Concurrent Optimization

There has always been an interest in approaching turbomachinery design in a multidisciplinary way. However, focusing simply in one discipline and meeting acceptable margins in the others has been the way to pursue this for several years. There are two main drawbacks when applying this iterative approach in design optimization that can affect cost and quality of the solution significantly. Firstly, the fact that the constraints (or margins) are checked after finding an initial good candidate can be a waste of computational time. Secondly, during the optimization, because only one discipline is being truly optimized, the design space is limited and the influences of the design variables in the other disciplines are missed, which can hinder finding a true optimum. A more concurrent and simultaneous approach to multiple disciplines can provide significant advantages compared to constrained optimization.

Despite non-gradient methods being the most dominant in multidisciplinary constrained optimization, their computational cost is an inherent limitation for multi-bladerow shape optimization. The problem exists even for steady flow optimization methods and that is worsened when evaluating unsteady flow and considering their suitability for concurrent optimization. The adjoint method, however, is a very efficient alternative for simultaneous aerodynamic and aeroelastic evaluations, which can make the evaluation of sensitivities, a major part of the optimization design process, independent of the number of variables. Moreover, it seems to be the only affordable option for multi-bladerow shape optimization too. The only real drawbacks are the risk of being trapped in local minimum and its complexity, the reason why there has been little research in the last years, as it is very complex to model an adjoint system for unsteady flow equations.

However, there has been very promising work on multidisciplinary (concurrent) design optimization using the adjoint method. He and Wang (2011) developed a continuous ad-

joint method for concurrent aerodynamic-aeroelastic optimization focused on the enhancement of blade flutter stability and the minimization of blade forced response for two test cases. The first case showed an overall better aerodynamic performance with an increase in the damping behaviour of a DLR transonic rotor, while the other case reduced the forced response of the NASA rotor 67 by 25% without aerodynamic penalties. Another approach to calculate the aeroelastic properties was performed by Huang and Ekici (2014), who optimized the blade shapes of a vibrating cascade to maximize the aerodynamic damping using the harmonic balance technique and a discrete adjoint approach. This last work, however, is not concurrent but it is one of the few which has aimed to capture aeroelastic sensitivities instead of aerodynamics ones as in most previous blade design optimization studies. It should be emphasized that these two latest adjoint-based aeroelastic design optimization developments have been confined to single blade row configurations.

2.8 Summary

From the existing literature as reviewed, multidisciplinary aerodynamic and aero-mechanic shape optimization has been a topic of research with two important issues. In relation to non-gradient methods, their expensive computational cost is limiting their development and restricting it to single-bladerow shaping only. This will remain until more efficient surrogate modelling techniques are developed and/or the computational power grows enough to overcome this issue. Therefore, the adjoint method seems the only option to perform multi-bladerow optimization at the present. However its complexity is the reason why only limited research has been done. The concurrent adjoint method developed by He and Wang (2011) under the Nonlinear Harmonic Phase Solution Method (He, 2008) showed the potential of the adjoint method to perform concurrent optimization for single-bladerow cases. Now, after the development of a multi-bladerow interface treatment by Rahmati et al. (2013) for this flow solution method, there is now an opportunity to develop an adjoint method with a multi-bladerow treatment that can perform

multi-bladerow optimization in a multidisciplinary environment. This is, to the author's knowledge, the first work in concurrent aerodynamic-aeroelastic shape optimization of turbomachinery blades in a multi-bladerow (and multi-stage) environment.

Chapter 3

Nonlinear Harmonic Phase Solution Method

The Unsteady Reynolds Averaged Navier-Stokes equations and the flow solution method used in this work are presented in this chapter. The Nonlinear Harmonic Phase Solution Method developed by He (2008) discretizes the URANS equations in the frequency domain at three distinctive phases angles of a period of unsteadiness and calculates the flow field by solving a system of three coupled steady-like equations. This solution method is able to predict blade flutter efficiently and it sets the framework for the aeroelastic adjoint design optimization method that is presented in Chapter 7. The unsteady-flow solver has been validated extensively in previous literature (He, 2008; Rahmati et al., 2013) and used for design optimization (He and Wang, 2011). The main modeling approach is briefly presented here. In addition, an extra set of validations is also included at the end of the chapter.

3.1 Governing Equations

Turbomachines are mainly formed by rotatory components. Therefore it is convenient to use the flow governing equations in a cylindrical coordinate frame of reference as

it will make easier the implementation of the periodic boundary conditions. A simple schematic of the frame of reference is shown in Figure 3.1 with a velocity vector in its three cylindrical components: u, v and w. In order to model blade flutter, the computational domain has to deform to accommodate the vibrating blade. Each point in the mesh has an oscillatory movement that is represented with three mesh velocities in the axial, circumferential and radial directions. The URANS equations are written in cylindrical coordinates as follows:

$$\frac{\partial U}{\partial t}+\frac{\partial\left(F-Uu_g-V_x\right)}{\partial x}+\frac{\partial\left(G-Uv_g-V_\theta\right)}{r\partial\theta}+\frac{\partial r\left(H-Uw_g-V_r\right)}{r\partial r}-S=0 \tag{3.1}$$

where:

$$U=\begin{Bmatrix}\rho\\ \rho u\\ \rho v r\\ \rho w\\ \rho \mathrm{e}\end{Bmatrix};F=\begin{Bmatrix}\rho u\\ \rho u^2+p\\ \rho v r u\\ \rho w u\\ \rho h u\end{Bmatrix};G=\begin{Bmatrix}\rho v\\ \rho u v\\ \rho v^2 r+p r\\ \rho w v\\ \rho h v\end{Bmatrix};H=\begin{Bmatrix}\rho w\\ \rho u w\\ \rho v r w\\ \rho w^2+p\\ \rho h w\end{Bmatrix};S=\begin{Bmatrix}0\\ 0\\ 0\\ \left(\rho v^2+p\right)/r\\ 0\end{Bmatrix}$$

$$V_x=\begin{Bmatrix}0\\ \tau_{xx}\\ r\tau_{\theta x}\\ \tau_{rx}\\ u\tau_{xx}+v\tau_{\theta x}+w\tau_{rx}-q_x\end{Bmatrix};V_\theta=\begin{Bmatrix}0\\ \tau_{x\theta}\\ r\tau_{\theta\theta}\\ \tau_{r\theta}\\ u\tau_{x\theta}+v\tau_{\theta\theta}+w\tau_{r\theta}-q_\theta\end{Bmatrix};$$

$$V_r=\begin{Bmatrix}0\\ \tau_{xr}\\ r\tau_{\theta r}\\ \tau_{rr}\\ u\tau_{xr}+v\tau_{\theta r}+w\tau_{rr}-q_r\end{Bmatrix}$$

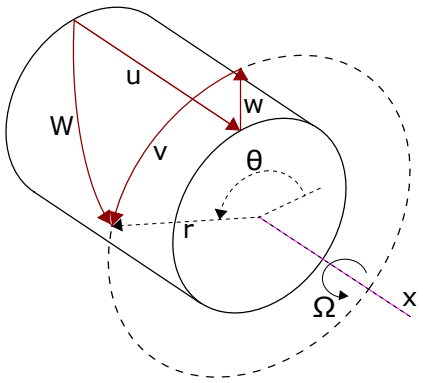


Figure 3.1: Cylindrical coordinates frame of reference

$\tau_{i,j}$ are the stress tensors according to the cylindrical frame of reference. Their notation is shown in Figure 3.2 for a two-dimensional x-r plane and they are defined as:

$$\tau_{xx} = \frac{2}{3}\mu\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{r\partial\theta} - \frac{\partial r w}{r\partial r}\right)$$

$$\tau_{\theta\theta} = \frac{2}{3}\mu\left(-\frac{\partial u}{\partial x} + 2\frac{\partial v}{r\partial\theta} - \frac{\partial r w}{r\partial r}\right)$$

$$\tau_{rr} = \frac{2}{3}\mu\left(-\frac{\partial u}{\partial x} - \frac{\partial v}{r\partial\theta} + 2\frac{\partial r w}{r\partial r}\right)$$

$$\tau_{x\theta} = \tau_{\theta x} = \mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{r\partial\theta}\right)$$

$$\tau_{xr} = \tau_{rx} = \mu\left(\frac{\partial w}{\partial x} + \frac{\partial r u}{r\partial r}\right)$$

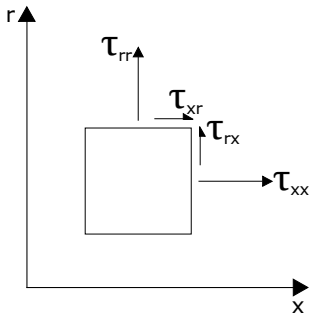


Figure 3.2: Two-dimensional view of stress tensors applied to a fluid element

q_i is the heat flux by thermal conduction:

$$q_x = -k\frac{\partial T}{\partial x}$$

$$q_\theta = -k\frac{\partial T}{r\partial\theta}$$

$$q_r = -k\frac{\partial rT}{r\partial r}$$

u_g, v_g and w_g are the corresponding grid velocities due to mesh blade vibration, μ is the fluid dynamic viscosity and k is the effective thermal conductivity defined as k=$\frac{C_p\mu}{Pr}$. In order to close the system, the gas state equation and the definitions of total specific energy, heat ratio and enthalpy are included as follows:

$$p = \rho RT$$

$$e = c_v T + \frac{1}{2}(u^2 + v^2 + w^2)$$

$$h = c_p T + \frac{1}{2}(u^2 + v^2 + w^2) = e + \frac{p}{\rho}$$

$$\gamma = \frac{c_p}{c_v}$$

3.2 Flow Solution Method

In order to illustrate better the flow solution method, Equation 3.1 can be simplified by collecting the spatial derivatives in a flow residual term $R(U)$ as follows:

$$\frac{\partial U}{\partial t} + R(U) = 0 \tag{3.2}$$

where:

$$R(U) = \frac{\partial\left(F - Uu_g - V_x\right)}{\partial x} + \frac{\partial\left(G - Uv_g - V_\theta\right)}{r\partial\theta} + \frac{\partial r\left(H - Uw_g - V_r\right)}{r\partial r} - S \tag{3.3}$$

As introduced in 2.3.3, the Nonlinear Harmonic Phase Solution Method assumes that the flow variable can be decomposed in a time-averaged part and a harmonic part:

$$U(x,t) = \bar{U}(x) + U_A(x)\sin\omega t + U_B(x)\cos\omega t \tag{3.4}$$

The flow variable U is a function of space and time but $\bar{U}, U_A$ and U_B are functions of space only. With this assumption, Equation 3.4 is added to Equation 3.2, thus eliminating the time derivative term and yielding the following:

$$\omega(U_A\cos\omega t - U_B\sin\omega t) + R\left(\bar{U} + U_A\sin\omega t + U_B\cos\omega t\right) = 0 \tag{3.5}$$

Equation 3.5 has substituted the time derivative term with an extra term that can be treated as a source term. However, the expression is still time-dependent and has three unknowns: $\bar{U}, U_A$, and U_B. At three particular phases $\omega t = -\frac{\pi}{2}, 0, \frac{\pi}{2}$, the flow variable $U(x,t)$ can be written as follows:

$$U\left(\omega t = -\frac{\pi}{2}\right) = U_{-1} = \bar{U} - U_A \tag{3.6a}$$

$$U(\omega t = 0) = U_0 = \bar{U} + U_B \tag{3.6b}$$

$$U\left(\omega t = \frac{\pi}{2}\right) = U_1 = \bar{U} + U_A \tag{3.6c}$$

Consequently, the three unknowns: $\bar{U}$,U_A, and U_B can also be expressed in terms of the flow solutions at these three phases:

$$\bar{U} = \frac{U_1 + U_{-1}}{2} \tag{3.7a}$$

$$U_A = \frac{U_1 - U_{-1}}{2} \tag{3.7b}$$

$$U_B = U_0 - \frac{U_1 + U_{-1}}{2} \tag{3.7c}$$

Substituting this back into Equation 3.5 yields the final system of three steady-like equations:

$$\omega\left(U_0 - \frac{U_1 + U_{-1}}{2}\right) + R\left(U_{-1}\right) = 0 \tag{3.8a}$$

$$\omega\left(\frac{U_1 - U_{-1}}{2}\right) + R\left(U_0\right) = 0 \tag{3.8b}$$

$$-\omega\left(U_0 - \frac{U_1 + U_{-1}}{2}\right) + R\left(U_1\right) = 0 \tag{3.8c}$$

The Nonlinear Phase Harmonic Solution method allows for the URANS equations to be reduced to a set of three steady-like equations which are simultaneously solved in a similar way to that of the steady RANS equations with the extra term being treated as a source term. This effectively reduces the computational cost of the unsteady problem to the same order of magnitude as an steady problem for blade flutter applications. In order to march their solutions to a steady state, a pseudo time derivative term is added to the three equations, so that the Runge-Kutta stepping technique (described in section 3.4) can be applied.

3.3 Turbulence Models

The effective viscosity that is used in a numerical simulation is the sum of the laminar viscosity μ_l, and the turbulent eddy viscosity μ_t. The laminar viscosity μ_l, is constant in

the whole domain and can be calculated using the next formula:

$$\mu_l = \frac{\rho u L}{Re} \tag{3.9}$$

where Re is the Reynolds number and ρ, u, and L are the corresponding reference density and relative velocity at the mid span region of the inlet domain and the axial chord length of a blade at its mid span as reference length scale. The Spalart-Allmaras turbulence model (Spalart and Allmaras, 1992) is implemented in this work in order to approximate the value of the turbulent eddy viscosity μ_t. This one-equation model solves the modeled transport equation for the kinematic eddy turbulent viscosity. The Spalart-Allmaras model was designed for aerospace applications but it has gained popularity in turbomachinery applications. The transport equation in cylindrical coordinates is written as follows:

$$\begin{aligned}\frac{\partial \rho\widetilde{\nu}u}{\partial x} + \frac{\partial \rho\widetilde{\nu}v}{r\partial\theta} + \frac{\partial r\rho\widetilde{\nu}w}{r\partial r} &= C_{b1}\left(1-f_{t2}\right)\tilde{S}\widetilde{\nu} \\ &+ 1.5\left\{\nabla\cdot\left[\left(1+\rho\widetilde{\nu}\right)\mu_l\right]\nabla\widetilde{\nu} + C_{b2}\rho\mu_l\nabla\widetilde{\nu}\cdot\triangle\widetilde{\nu}\right\} \\ &- \frac{\rho}{\mu_l}\left(C_{w1}f_w - \frac{C_{b1}}{\kappa^2}f_{t2}\right)\left(\frac{\widetilde{\nu}\mu_l}{d}\right)^2 + f_{t1}\rho\Delta U^2\end{aligned} \tag{3.10}$$

where

$$\kappa = 0.4187,\ C_{b1} = 0.1355,\ C_{b2} = 0.622,\ C_{w1} = \frac{C_{b1}}{k^2} + 1.5\left(1+C_{b2}\right)$$

$$c_{t1} = 1.0,\ c_{t2} = 2.0,\ c_{t3} = 1.2,\ c_{t4} = 0.5,\ c_{w2} = 0.3,\ c_{w3} = 2.0$$

$$\chi = \rho\widetilde{\nu},\ f_{t2} = c_{t3}\exp\left(-c_{t4}\chi^2\right),\ f_{v2} = 1 - \frac{\chi}{1+\chi f_{v1}},\ f_w = g\left(\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right)^{\frac{1}{6}}$$

$$\widetilde{S} = \omega + \frac{\widetilde{\nu}\mu_l}{(k\cdot d)^2}f_{v2},\ f_{t1} = c_{t1}g_t\exp\left[-c_{t2}\frac{\omega_t^2}{\triangle U^2}\left(d^2+g_t^2d_t^2\right)\right],\ r = \frac{\widetilde{\nu}\mu_l}{\widetilde{S}(k\cdot d)^2}$$

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

The solution variable $\widetilde{\nu}$ for the transport equation is a viscosity-like variable, often referred to as the Spalart-Allmaras variable. The turbulent eddy viscosity of interest, μ_t is

calculated afterwards from $\widetilde{\nu}$ in the following way:

$$\mu_t = \rho\widetilde{\nu}\mu_l f_{v1},\ f_{v1} = \frac{\chi^3}{\chi^3 + C_{v1}^3},\ C_{v1} = 7.1$$

3.4 Discretization of the Equations

The Nonlinear Harmonic Phase Solution Method allows the system of equations to be solved in the frequency domain and consequently to reduce the computational domain from a whole annulus to a sub-domain of one single passage between adjacent blades. In order to discretize the flow equations in a finite volume approach, Equation 3.5 is written in a conservative form over a control volume V_i as follows:

$$\int_{\partial V_i} \left[\left(F - Uu_g - V_x\right) n_x + \left(G - Uv_g - V_\theta\right) n_\theta + \left(H - Uw_g - V_r\right) n_r\right] dS = -\int_{V_i} S' dv \quad (3.11)$$

where $S' = S + \omega\left(U_A \cos\omega t - U_B \sin\omega t\right)$ contains all the source terms, including the resulting terms from the Nonlinear Harmonic Phase Solution Method . The control volume has a surface ∂V_i and a volume $\triangle V_i$. The computational domain of the single passage is formed by a set of 3D hexahedral elements as shown in Figure 3.3.

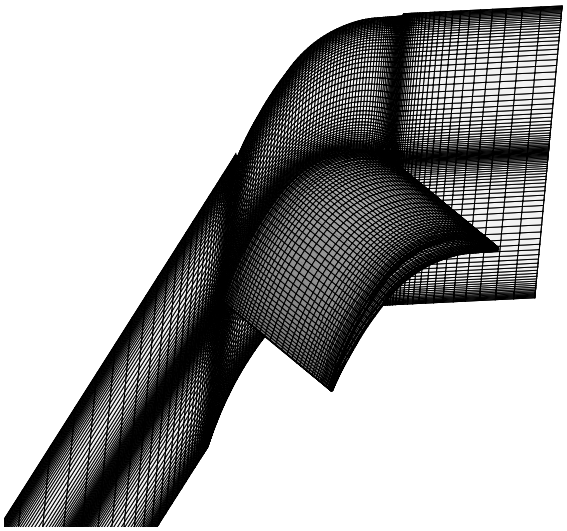


Figure 3.3: Three-dimensional mesh of a linear cascade passage

3.4.1 Spatial Discretization

The spatial discretization is done using a central difference method with second and fourth order artificial dissipation. Artificial dissipation is introduced in the discretization in order to avoid the instabilities of a standard central difference approach when there are shock regions in the flow field. The second order artificial dissipation removes non-physical spurious oscillations while the fourth order artificial dissipation improves the stability limit and makes the computational scheme more robust. The discretized convective fluxes from Equation 3.11 through a bounding surface are written as follows:

$$R(U) = \sum_{j=1}^{n} \left(Flux^{i}_{j+\frac{1}{2}} - Flux^{i}_{j-\frac{1}{2}} \right) + \left(Flux^{v}_{j+\frac{1}{2}} - Flux^{v}_{j-\frac{1}{2}} \right) + \left(Flux^{d}_{j+\frac{1}{2}} - Flux^{d}_{j-\frac{1}{2}} \right) - \triangle V_i S_i = 0 \quad (3.12)$$

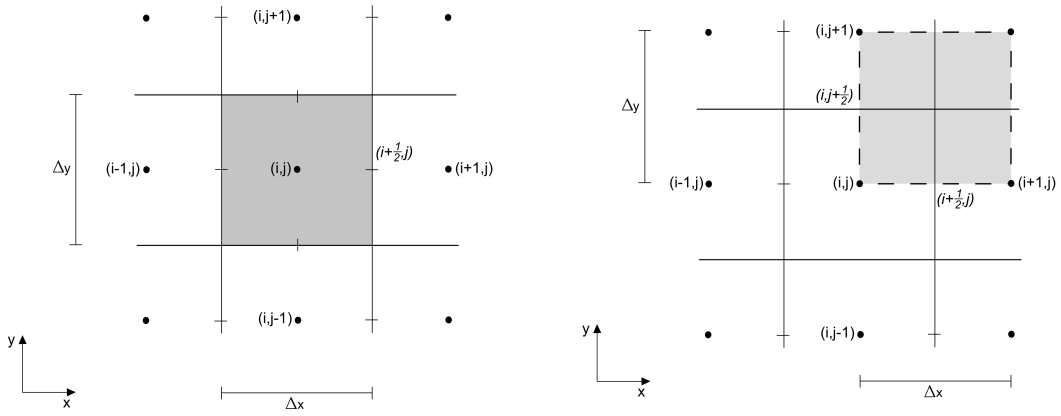
where n is the number of dimensions of the control volume and $Flux_{j+\frac{1}{2}}$ is the flux through the bounding surface with index $j+\frac{1}{2}$. This flux has three components: inviscid, viscid and dissipative, and it is denoted in the above equation with the superscripts i, v and d. The discretization of each of these convective fluxes is illustrated next with a two-dimensional control volume as an example. The corresponding two-dimensional structured mesh with quadrilateral elements is shown in Figure 3.4a. The principles of the finite volume method are also applicable to a three-dimensional mesh.

Inviscid flux

The two-dimensional inviscid flux along the bounding surface of the mesh shown in Figure 3.4a indexed as $\left(i+\frac{1}{2}, j\right)$ of the control volume index (i, j) is:

$$Flux^{i}_{i+\frac{1}{2},j} = \left(F - Uu_g\right)_{i+\frac{1}{2},j} S_x + \left(G - Uv_g\right)_{i+\frac{1}{2},j} S_y \quad (3.13)$$

where S_x and S_y are projections of the area of the bounding surface onto the x and y directions of the 2D control volume, respectively. The rest of the terms are expanded as



(a) Primary mesh (b) Staggered mesh

Figure 3.4: 2D cartesian finite volume mesh. Element sizes are defined by $\triangle x$ and $\triangle y$

follows:

$$\left(F - Uu_g\right)_{i+\frac{1}{2},j} = \frac{1}{2}\left[F\left(U_{i,j}\right) + F\left(U_{i+1,j}\right) - u_g\left(U_{i,j} + U_{i+1,j}\right)\right]$$

$$\left(G - Uv_g\right)_{i+\frac{1}{2},j} = \frac{1}{2}\left[G\left(U_{i,j}\right) + G\left(U_{i+1,j}\right) - v_g\left(U_{i,j} + U_{i+1,j}\right)\right]$$

The central difference scheme calculates the value of the fluxes at the bounding surfaces from the value of the flow variables at the two adjacent cell points.

Viscous flux

The viscous flux along the bounding surface indexed as $\left(i+\frac{1}{2},\, j\right)$ is:

$$Flux^{v}_{i+\frac{1}{2},j} = (V_x)_{i+\frac{1}{2},j}\, S_x + \left(V_y\right)_{i+\frac{1}{2},j}\, S_y \tag{3.14}$$

The central difference interpolation method of the inviscid flux can not be directly applied to the viscous flow. This is because the viscous flow terms contain spatial derivatives. Consequently, the value of these derivatives at the bounding surface has to be approximated differently. In order to do so, an extra mesh (staggered mesh) is defined as shown in Figure 3.4b. This auxiliary mesh defines the boundary surfaces of its elements in the lines that cross the cell points of the primary mesh. The spatial derivatives of the

viscous terms are calculated using the divergence theorem as follows:

$$\frac{1}{\triangle x \triangle y}\int_{\partial S} u \cdot n_x dS \approx \frac{u_{i+1,j} - u_{i,j}}{\triangle x} = \left.\frac{\partial u}{\partial x}\right|_{i+\frac{1}{2},j} \tag{3.15}$$

where $\triangle x$ and $\triangle y$ are the length of the sides of the corresponding element of the staggered mesh. Any gradient of the viscous terms can be discretized by replacing u with the corresponding flow variable and x with the coordinate of interest. Once all the gradients have been discretized, the three viscous terms of the flux expression above V_x, V_θ and V_r can be calculated.

Dissipative flux

Some compressible flow cases in turbomachinery may have shock waves generated in the blade passages. These discontinuities are characterized by very sharp gradients of the flow variables and strong dissipation. Many of the traditional methods like the second-order central difference scheme cannot be used directly since they lead to strong numerical dispersion and spurious oscillations around the shock region. There are various techniques that allow use of these schemes such as the artificial dissipation. The formulation that is followed in this method is the scalar dissipation model, which was introduced by Jameson et al. (1981). Adding an artificial term to the convective fluxes reduces these effects and makes the computational scheme more robust. The artificial dissipative flux is added to the convective flux as follows:

$$Flux^d_{i+\frac{1}{2},j} = \varepsilon^{(2)}_{i+\frac{1}{2},j}\triangle_{i+\frac{1}{2},j} - \varepsilon^{(4)}_{i+\frac{1}{2},j}\left(\triangle_{i+\frac{3}{2},j} - 2\triangle_{i+\frac{1}{2},j} + \triangle_{i-\frac{1}{2},j}\right) \tag{3.16}$$

where $\triangle_{i,j}$ is a forward difference operator, defined as:

$$\triangle_{i+\frac{1}{2},j} = VT_{i+\frac{1}{2},j}\left(U_{i+1,j} - U_{i,j}\right)$$

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

$\triangle V$ and $\triangle t$ are the volume and time-step of the corresponding element. The coefficients $\varepsilon^{(2)}$ and $\varepsilon^{(4)}$ make use of the pressure as sensors for strong gradients where dissipation may be needed and they are defined as:

$$\varepsilon^{(2)}_{i+\frac{1}{2},j} = \max\left(\min\left(0.05, c_2 \nu_{i+\frac{1}{2},j}\right), 0.0075\right)$$

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

where

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

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

The factors, denoted as c_i are smoothing constants: c blends the second and fourth order dissipation effects, c_2 is the corresponding second order dissipation factor with a typical range between $\frac{1}{4}$ and $\frac{1}{2}$. c_4 is the fourth order dissipation factor with a typical range between $\frac{1}{64}$ and $\frac{1}{32}$.

3.4.2 Pseudo Time Marching Solution

Compared to the URANS equations, the system of three steady-like solutions that has been derived using the Nonlinear Harmonic Phase Solution Method has an extra term corresponding to the original time derivative term of the URANS equations. Because this extra term is being treated as a source term, there is no implicit time variable to march the equations forward. In order to march Equation 3.8 forward with a time-marching method, a pseudo time derivative term is added to these equations, which yields:

$$\frac{\partial U}{\partial \tau} + \omega\left(U_0 - \frac{U_1 + U_{-1}}{2}\right) + R\left(U_{-1}\right) = 0 \tag{3.17a}$$

$$\frac{\partial U}{\partial \tau} + \omega\left(\frac{U_1 - U_{-1}}{2}\right) + R\left(U_0\right) = 0 \tag{3.17b}$$

$$\frac{\partial U}{\partial \tau} + -\omega \left(U_0 - \frac{U_1 + U_{-1}}{2} \right) + R\left(U_1\right) = 0 \tag{3.17c}$$

The spatial derivatives of the URANS equations and the source terms can be collected in a new residual operator $R'_i(U_i)$ as follows:

$$R'_i(U_i) = \omega \left(U_A \cos(\omega t) - U_B \sin(\omega t)\right) + R(U_i) \quad i = -1, 0, 1 \tag{3.18}$$

where the subscript i denotes the phase of unsteadiness. A four step Runge-Kutta scheme is used for this flow solution method. Hence, for each phase of unsteadiness, the 4-step method below is applied as follows:

$$U_i^{n+\frac{1}{4}} = U_i^n - \frac{1}{4}\frac{\triangle \tau}{\triangle V}\left[R'_i\left(U_i^n\right)\right] \tag{3.19a}$$

$$U_i^{n+\frac{1}{3}} = U_i^n - \frac{1}{3}\frac{\triangle \tau}{\triangle V}\left[R'_i\left(U_i^{n+\frac{1}{4}}\right)\right] \tag{3.19b}$$

$$U_i^{n+\frac{1}{2}} = U_i^n - \frac{1}{2}\frac{\triangle \tau}{\triangle V}\left[R'_i\left(U_i^{n+\frac{1}{3}}\right)\right] \tag{3.19c}$$

$$U_i^{n+1} = U_i^n - \frac{\triangle \tau}{\triangle V}\left[R'_i\left(U_i^{n+\frac{1}{2}}\right)\right] \tag{3.19d}$$

The main advantages of the Runge-Kutta method is its flexibility and good stability properties. Moreover, the time step required for its stability is typically larger than other explicit methods. However, the flow residuals and the source terms $R'(U_i)$, must be solved on every time step. In order to accelerate the convergence of the solution process, local time-stepping and multi-grid method (He and Denton, 1994) techniques are used.

3.5 Boundary Conditions

The system of three steady-like equations is solved by adding a pseudo time derivative term which makes the system of equations hyperbolic. The compatibility requirement for a hyperbolic system states that the number of boundary conditions to be specified must be equal to the number of characteristics running into the computational domain. Four types of boundary conditions are implemented based on this compatibility. These are:

- Subsonic inlet boundary condition
- Subsonic exit boundary condition
- Viscous solid wall boundary condition
- Periodic boundary condition

3.5.1 Inlet Boundary Condition

The flow field at the inlet boundary is usually subsonic and there are four incoming flow characteristics and one outgoing flow characteristic running across the boundary. The four incoming flow characteristics are: one entropy disturbance convected downstream at local flow velocity, two vorticity disturbances convected downstream at local flow velocity and one acoustic disturbance convected downstream at a total speed of the local flow velocity plus the local sound speed. The one outgoing flow characteristic is an acoustic disturbance running upstream at a total speed of the local flow velocity minus the local sound speed. According to the characteristic requirements, four boundary conditions need to be specified, which fixes the value of the four flow variables at the inlet boundary. The four boundary conditions that are specified are:

1. Total pressure at inlet
2. Total gas temperature at inlet

3. Swirl velocity flow angles

4. Pitch velocity flow angle

The rest of the parameters and isentropic relations are calculated from the boundary conditions.

3.5.2 Exit Boundary Condition

From the above section, there are four outgoing flow characteristics and one incoming flow characteristic at the exit boundary. The one incoming flow characteristic is an acoustic disturbance traveling upstream at a speed of the local flow velocity minus the local sound speed. Consequently, one boundary condition needs to be specified to satisfy the characteristic requirements, that is the pressure field at the exit. There are three options to specify the static pressure at the exit:

- Static pressure at the hub or tip with the radial distribution being worked out with the equilibrium equation:
$$\frac{\partial p}{\partial r} = \rho v^2 \tag{3.20}$$

- Static pressure profile in the radial direction.
$$p = p(r) \tag{3.21}$$

  – As an specific case of the above, only the static pressure at both hub and tip can be specified and the static pressure distribution is worked out by linear interpolation.
$$p = p_{hub} + \left(p_{tip} - p_{hub}\right) \frac{p - p_{hub}}{p_{tip} - p_{hub}} \tag{3.22}$$

Once the static pressure field at the exit is defined, the rest of the flow variables (density and velocity components) can be extrapolated from the interior domain.

3.5.3 Solid Wall Boundary Condition

The first condition is that the pressure gradient normal to the wall is assumed to be zero:

$$\frac{\partial p}{\partial n} = 0 \tag{3.23}$$

The second condition is that the relative velocity normal to a solid wall is zero, whether it is viscous or invisicid.

$$v_n = \left(u - u_g\right) n_x + \left(v - v_g\right) n_\theta + \left(w - w_g\right) n_r = 0 \tag{3.24}$$

This is implemented in the code by setting the convective fluxes to zero for all the five equations:

$$U \cdot v_n = U\left[\left(u - u_g\right) n_x + \left(v - v_g\right) n_\theta + \left(w - w_g\right) n_r\right] = 0 \tag{3.25}$$

For a viscous solid wall boundary condition, the slip boundary condition also has to be implemented. This can be done with a refined boundary layer mesh or with a log-law function. The log-law is used in this work to calculate the wall shear stress τ_w without the need of a refined mesh. The calculation of the wall shear stress is done as follows:

$$\tau_w = \frac{1}{2} c_f \rho W^2 \tag{3.26}$$

$$W = \sqrt{\left(u^2 + v^2 + w^2\right)}$$

where W is the velocity magnitude and ρ the density of the fluid at the first mesh point away from the wall. c_f is the skin friction coefficient, which depends on the local Reynolds number as follows:

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

The Reynolds number is calculated locally based on the first element of the mesh at the wall and its normal height, denoted as $\triangle y$, which yields:

$$Re = \frac{\rho W \triangle y}{\mu_l}$$

As a last condition, if the wall is adiabatic, the zero temperature gradient normal to the wall is imposed:

$$\frac{\partial T}{\partial n} = 0 \tag{3.27}$$

3.5.4 Periodic Boundary Condition

The periodic boundary condition (Figure 2.1) for the three steady-like equations of the Nonlinear Harmonic Phase Solution Method is different from the periodic boundary condition of the steady-flow solver. The unsteady flow-variable is assumed to have a time-averaged and a harmonic part. Each blade is subject to unsteadiness of a circumferential traveling wave pattern. Since this wave-length is not the same as the blade pitch, there is no direct periodicity among blade passages as in the steady-flow case. Instead, there is a phase difference among periodic boundaries. This phase difference is called inter-blade phase angle σ and it is defined as:

$$\sigma = \frac{2\pi n}{N} \tag{3.28}$$

where N is the number of blades of the bladerow of interest and n the number of nodal diameters of the unsteady disturbance. The inter-blade phase angle also defines the circumferential wave-length. Based on this, the flow solution at the upper periodic boundary is phase-shifted between the upper and lower periodic boundaries by σ as follows:

$$U^U = \bar{U^L} + U_A^L \sin(\omega t + \sigma) + U_B^L \cos(\omega t + \sigma) \tag{3.29}$$

where $\bar{U^L}, U_A^L$ and U_B^L are calculated based on the flow solution at the three phases (Equation 3.7) at the lower periodic boundary. The flow solution at the lower periodic boundary

is updated from the flow solution at the upper periodic boundary as follows:

$$U^L = \bar{U^U} + U_A^U \sin(\omega t - \sigma) + U_B^U \cos(\omega t - \sigma) \tag{3.30}$$

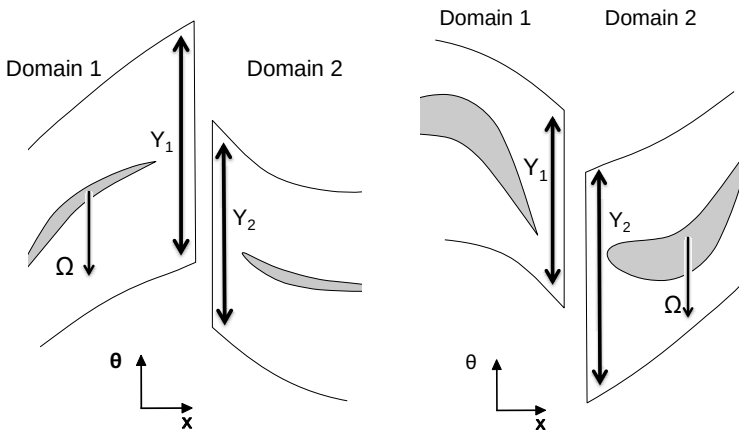
3.6 Multi-bladerow Treatment

The Nonlinear Harmonic Phase Solution Method assumes that the unsteady flow variable can be decomposed in a time-averaged and a harmonic part (Equation 3.4). In order to exchange the flow information among blade-rows, different methods need to be used for the time-averaged and the harmonic components. Firstly, for the time-mean flow, the mixing-plane treatment is used to pass the time-averaged values across a rotor-stator interface. Secondly, for the first harmonic unsteady disturbances, a harmonic interface is implemented .

3.6.1 Mixing-Plane Treatment

The flow mixing-plane treatment approach is used to pass the time-averaged flow variable across the interface. This treatment was initially proposed by Denton and Singh (1979) and the implementation is based on the work done by He and Sato (2000). The mixing-plane treatment has been previously validated (Rahmati et al., 2013) and used in design shape optimization by Wang and He (2010a,b). This method is numerically robust and has two important properties: non-reflectiveness, which is critical due to the small intra-row gap between adjacent blade rows, and conservation. The non-reflective property ensures the right propagation of the flow solution information across an interface while also forcing the conservation of mass, momentum and energy fluxes without artificial reflections to the interior domains that can corrupt the flow solution.

Figure 3.5 shows a schematic of the interface in a rotor-stator domain. The axial and circumferential direction correspond to the x and y coordinates. The conservation property



(a) Rotor-stator interface (b) Stator-rotor interface

Figure 3.5: Mixing-plane interface in a single-passage domain for rotor-stator and stator-rotor configurations

is illustrated as follows:

$$\frac{1}{Y_1}\int_{y1}\rho u dy = \frac{1}{Y_2}\int_{y2}\rho u dy \tag{3.31a}$$

$$\frac{1}{Y_1}\int_{y1}\left(\rho u^2 + p\right) dy = \frac{1}{Y_2}\int_{y2}\left(\rho u^2 + p\right) dy \tag{3.31b}$$

$$\frac{1}{Y_1}\int_{y1}\rho v u dy = \frac{1}{Y_2}\int_{y2}\rho v u dy \tag{3.31c}$$

$$\frac{1}{Y_1}\int_{y1}\rho w u dy = \frac{1}{Y_2}\int_{y2}\rho w u dy \tag{3.31d}$$

$$\frac{1}{Y_1}\int_{y1}\rho h u dy = \frac{1}{Y_2}\int_{y2}\rho h u dy \tag{3.31e}$$

where Y_1 and Y_2 are pitch lengths of domain 1 and domain 2 respectively. On each side of the interface, once the five circumferentially averaged fluxes (Equation 3.31) are obtained, a set of mean flow variables is defined as follows:

$$\bar{v} = \frac{F_3}{F_1} \tag{3.32a}$$

$$\bar{w} = \frac{F_4}{F_1} \tag{3.32b}$$

$$\bar{h} = \frac{F_5}{F_1} \tag{3.32c}$$

$$\bar{p} = \frac{F_2 + \sqrt{F_2^2 + \left(\gamma^2 - 1\right)\left(F_2^2 + F_3^2 + F_4^2 - 2F_1F_5\right)}}{\gamma + 1} \tag{3.32d}$$

$$\bar{u} = \frac{F_2 - \bar{p}}{F_1} \tag{3.32e}$$

$$\bar{\rho} = \frac{F_1}{\bar{u}} \tag{3.32f}$$

where F_1, F_2, F_3, F_4 and F_5 are the corresponding five circumferentially averaged fluxes (Equation 3.31). If these five fluxes are not equal across the interface, then the mean flow variables (Equation 3.32) will not be equal across the interface either. At the start of a solution process, the fluxes are not equal across the interface. The difference in these intermediate flow variables is used to define the corresponding difference or characteristic jumps traveling into a domain. The flow characteristic jumps are defined as follow:

$$\phi_1 = -c^2\left(\bar{\rho}_2 - \bar{\rho}_1\right) + \left(\bar{p}_2 - \bar{p}_1\right) \tag{3.33a}$$

$$\phi_2 = \rho c\left(\bar{v}_2 - \bar{v}_1\right) \tag{3.33b}$$

$$\phi_3 = \rho c\left(\bar{w}_2 - \bar{w}_1\right) \tag{3.33c}$$

$$\phi_4 = \rho c\left(\bar{u}_2 - \bar{u}_1\right) + \left(\bar{p}_2 - \bar{p}_1\right) \tag{3.33d}$$

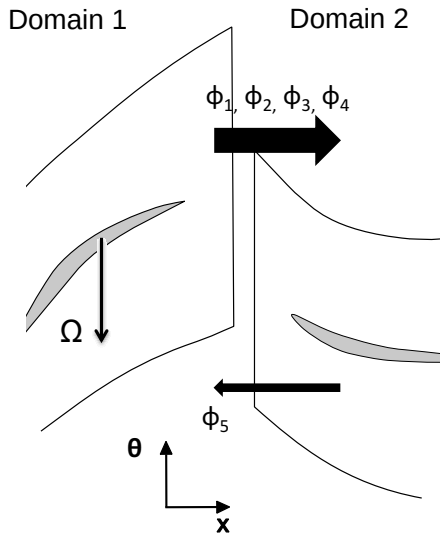


Figure 3.6: Illustration of flow characteristic jumps going across the interface

$$\phi_5 = -\rho c\,(\bar{u_2} - \bar{u_1}) + \left(\bar{p_2} - \bar{p_1}\right) \tag{3.33e}$$

where c is the speed of sound and $\bar{\rho}, \bar{p}, \bar{u}, \bar{v}$ and $\bar{w}$ are mean variables calculated from Equation 3.32 for each respective interface. ϕ_i denotes the five characteristics jumps, which are:

- ϕ_1 represents an entropy disturbance traveling downstream at a speed of u
- ϕ_2 and ϕ_3 represent both vorticity disturbances traveling downstream at a speed of u
- ϕ_4 represents an acoustic disturbance traveling downstream at a speed of $u + c$
- ϕ_5 represents an acoustic disturbance traveling upstream at a speed of $u - c$

It can be seen in Figure 3.6 that one flow characteristic (ϕ_5) travels into domain 1, while the other four flow characteristics (ϕ_1, ϕ_2, ϕ_3 and ϕ_4) travel into domain 2. For domain 1, the incoming flow characteristic (ϕ_5) is calculated using the definition in Equation 3.33, while the other four outgoing flow characteristics are extrapolated from the interior domain. For domain 2, the four incoming flow characteristics are calculated using

the definition in Equation 3.33 while the one outgoing flow characteristic is extrapolated from the interior domain.

Once all the corresponding flow characteristic jumps are obtained for one domain, they can be converted back to the primitive flow variable perturbations which can be used to update the local variables at the interface:

$$\left(\bar{\rho_2} - \bar{\rho_1}\right) = \frac{1}{c^2}\left(-\phi_1 + \frac{1}{2}\phi_4 + \frac{1}{2}\phi_5\right) \tag{3.34a}$$

$$(\bar{u_2} - \bar{u_1}) = \frac{1}{2\rho c}\left(\phi_4 - \phi_5\right) \tag{3.34b}$$

$$(\bar{v_2} - \bar{v_1}) = \frac{1}{\rho c}\phi_2 \tag{3.34c}$$

$$(\bar{w_2} - \bar{w_1}) = \frac{1}{\rho c}\phi_3 \tag{3.34d}$$

$$\left(\bar{p_2} - \bar{p_1}\right) = \frac{1}{2}\left(\phi_4 + \phi_5\right) \tag{3.34e}$$

3.6.2 Harmonic Interface Treatment

The approach used for passing the harmonic disturbance through the interface has been developed by Rahmati et al. (2013). With this interface treatment, propagation and reflection of pressure waves between adjacent blade rows are modeled with a domain consisting of only a single blade passage in each bladerow. In order to pass the harmonic disturbance through the interface, first each side of the interface obtains the boundary information from the corresponding adjacent row in a time series format. Then, a temporal Fourier transform is used to get the disturbance amplitudes corresponding to the first harmonic mode only.

Considering a stage configuration with rotor blades vibrating at a frequency ω_R, the fre-

quency of the corresponding unsteady-flow stator domain ω_S is determined by the basic frequency shift relation:

$$\omega_S = \omega_R + N_D\Omega \tag{3.35}$$

where N_D is the number of nodal diameters and Ω is the rotational speed. Once the interior domain mesh points are updated, two tasks are carried out by the harmonic interface treatment:

1. To identify the unsteady signals at both sides of the interface (Time-trace reconstruction)

2. To separate the sources of the unsteady components and retain only the corresponding component of the first harmonic (Fourier filtering)

Time-trace reconstruction

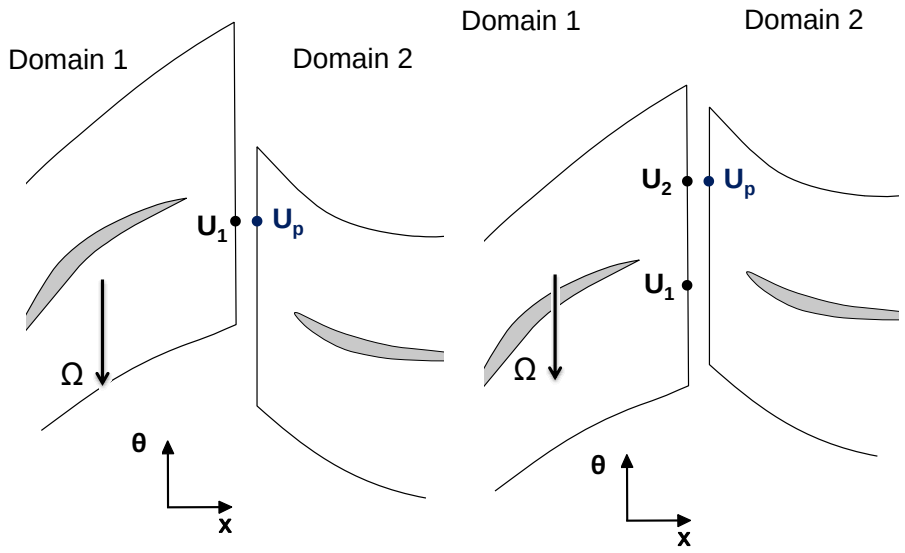
A time trace in the time domain needs to be built. For any points at the interface of domain 1, the instantaneous position relative to the domain on the other side of the interface (domain 2) is determined by its initial position, the rotational speed, and the relative time to a reference point. This is effectively done with a local second order interpolation and it is illustrated in Figure 3.7. The value of the flow variable at the point p, for different time instances at a vibration period can be defined as follows:

$$U_p\,(t = t_i) = U_i^{'}\,(i = 0, 1, 2, ..., n) \tag{3.36}$$

where $U_i^{'}$ is the flow variable at time t_i on domain 1 based on the harmonic assumption (Equation 3.4). A full time-trace at a period of unsteadiness looks like in Figure 3.8.

Fourier filtering

Once the time trace is generated for each interface, only the unsteadiness of interest must be retained while the rest has to be filtered out. This can be done performing a temporal



(a) Time instance $t = t_1$ (b) Time instance $t = t_2$

Figure 3.7: Relative position between domains at different time-instances t_1 and t_2

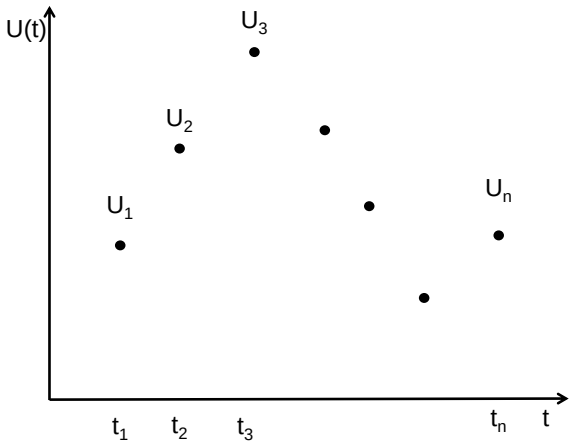


Figure 3.8: Time-trace of the flow variable vector U over a period of unsteadiness

Fourier transformation of the time trace as follows:

$$U_A = \frac{\omega}{\pi} \sum_{i=1}^{n} (U_i) \sin(\omega t_i) \triangle t_i \tag{3.37a}$$

$$U_B = \frac{\omega}{\pi} \sum_{i=1}^{n} (U_i) \cos(\omega t_i) \triangle t_i \tag{3.37b}$$

$$\bar{U} = \frac{\omega}{2\pi} \sum_{i=1}^{n} (U_i) \triangle t_i \tag{3.37c}$$

where ω is the angular velocity based on the frequency of vibration, n is the number of time steps used for the construction of the time trace $U(t_i)$ and $\triangle t_i$ is the size of the time steps between time intervals. After U_A, U_B and $\bar{U}$ are calculated, they are used to update the solution at the interface U_i for the three phases $\omega t = -\frac{\pi}{2}, 0, \frac{\pi}{2}$.

3.7 Validation of the Flow Solver

The unsteady flow field inside an oscillating low-speed turbine cascade is investigated in order to show the aeroelastic capabilities of an unsteady-flow solver based on the Non-linear Harmonic Phase Solution Method. The experimental data was collected by (Huang et al., 2009), whose work helped to produce test data for proper validation of numerical aero-mechanical methods.

The geometric parameters and operating conditions are shown in Table 3.1. The computational domain consists of a single passage H-type mesh with a grid dimension of $51x128x51$ in the pitchwise, streamwise and spanwise directions respectively. The blade-to-blade and meridional views of the computational mesh for the turbine cascade are shown in Figure 3.9. The computational results are compared with the corresponding experimental results at a Reynolds number of 2.2×10^5. First, the time-averaged flow field obtained by the solver is analyzed. The calculated pressure coefficients at four spanwise locations (10%, 30%, 50% and 80%) are compared with the corresponding experimental

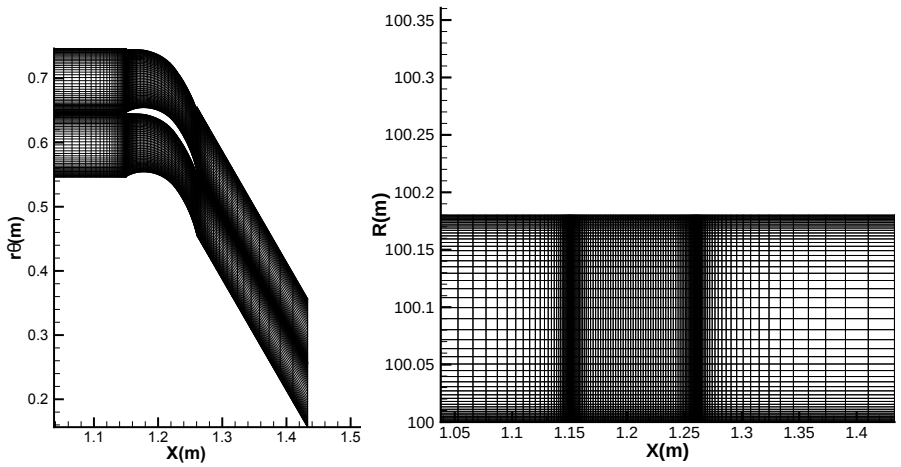


Figure 3.9: Computational mesh of the linear cascade. Blade to blade view (left) and meridional view (right)

results. The pressure coefficient definition is given next:

$$C_p = \frac{p - p_{exit}}{p_0^* - p_{exit}} \tag{3.38}$$

where p_{exit} is the static pressure at the exit and p_0^* is the total pressure at the inlet, both at midspan. The results are shown in Figure 3.10. It can be seen that the flow field is mainly two-dimensional, with little variation along the span direction. The computed surface distribution is in good agreement with the experimental data. Secondly, the unsteady flow field due to controlled vibration at a limited first bending reduced frequency k of 0.2. The reduced frequency is a measure of both spatial and temporal length scales and it is defined as:

$$k = \frac{\omega c}{V}$$

where $\omega = 2\pi f$ is the angular frequency and f is the nominal frequency of unsteadiness of 5.2 Hz. c is the reference length chosen as the chord of the blade and V is the reference velocity chosen as the inlet flow velocity. For blade flutter problems, a typical value of reduced frequency for the first bending or flap mode is around 0.3. Reduced frequencies

of blades need to be above this value for aeroelastic design purposes. The first harmonic of the unsteady pressure in the blade surface has the following form:

$$p = \bar{p} + p_A \sin(\omega t) + p_B \cos(\omega t) \tag{3.39}$$

The chordwise distribution of the amplitude and phase angle of the first harmonic unsteady pressure is shown in Figure 3.11 for different spanwise locations and two inter-blade phase angles: 180 and 150 degrees. The definition of the first harmonic pressure coefficient and phase angle is as follows:

$$\mid C_{p1} \mid = \frac{\sqrt{p_A^2 + p_B^2}}{\left(p_0 - p_{exit}\right) A_{tip}} \tag{3.40}$$

$$\alpha = \tan^{-1}\left(\frac{p_A}{p_B}\right) \tag{3.41}$$

where A_{tip} represents the non-dimensional vibration amplitude at the blade tip. It can be seen that the solver predicts all the main features such as the flow acceleration over the suction surface and the amplitude peak of the pressure term. Overall, both the first harmonic pressure coefficients and their phase angles are in good agreement with the experimental data.

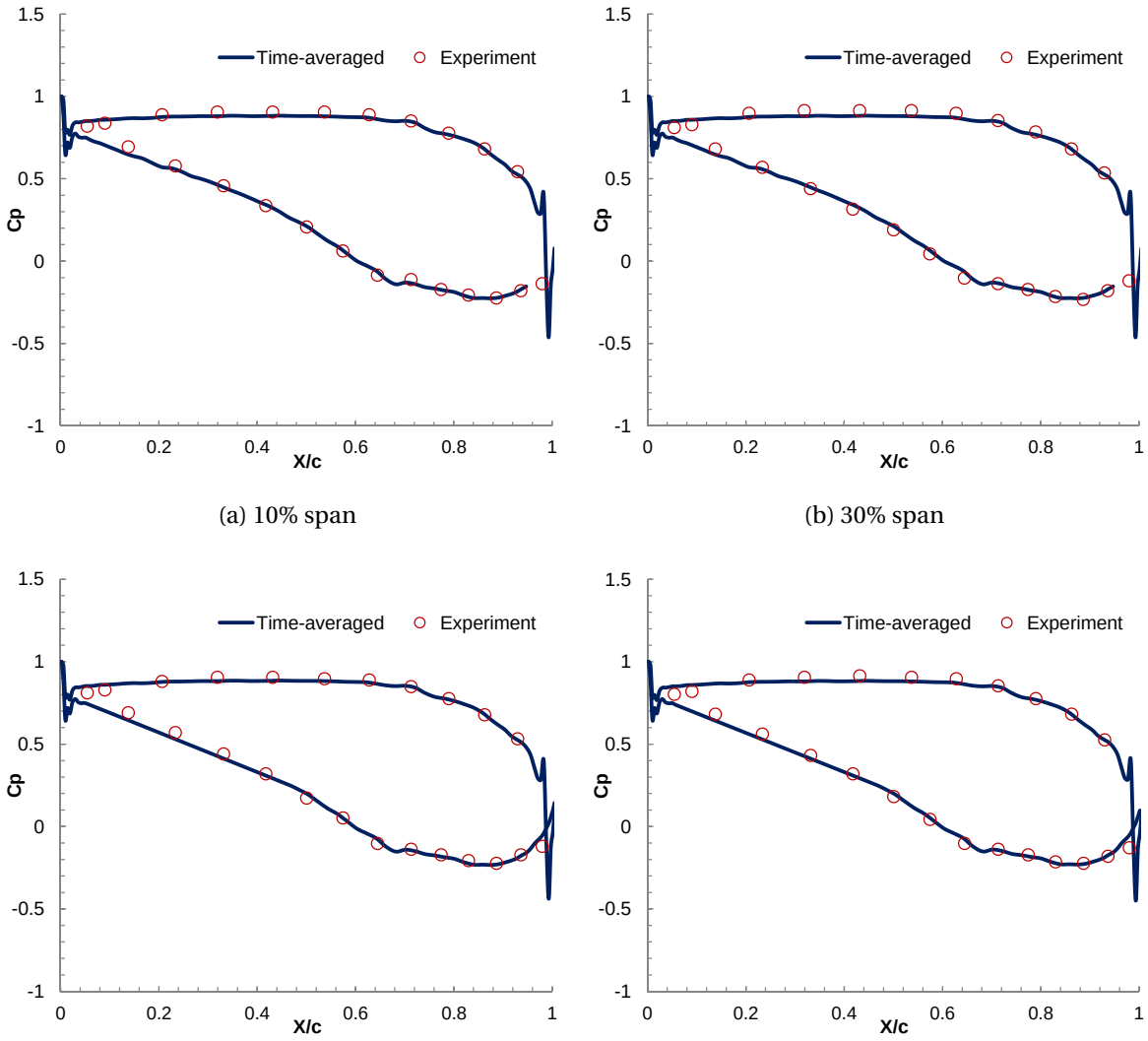
Lastly, the experimental measures of aerodynamic damping at different inter-blade phase angles are compared with the computed results. The formula for the aerodynamic damping is written below:

$$\xi = -\frac{W}{\left[\left(p_0^* - p_{exit}\right) \cdot A_{tip}^2 \cdot c^2 \cdot h\right]} \tag{3.42}$$

where c and h are the chord and height of the blade and W is the total work sum, explained in section 5.3. The aerodynamic damping is computed for 9 different inter-blade phase angles (Figure 3.12). Its distribution changes in a sinusoidal way as a function of the Inter-Blade Phase Angle (IBPA). The calculated results are again in very good agreement with the measured data.

Blade chord, C	0.143 m
Blade span, h	0.18 m
Pitch length, S	0.1 m
Stagger angle	40.0 deg
Inlet flow angle	0.0 deg
Vibration mode	First bending
Bending amplitude	3% c at tip, 0.3% c at hub
Reynolds number, Re	2.2×10^5
Reduced frequency, k	0.2
Nominal frequency, f	5.2 Hz

Table 3.1: Geometric parameters and operating conditions of the validation case (Huang et al., 2009)



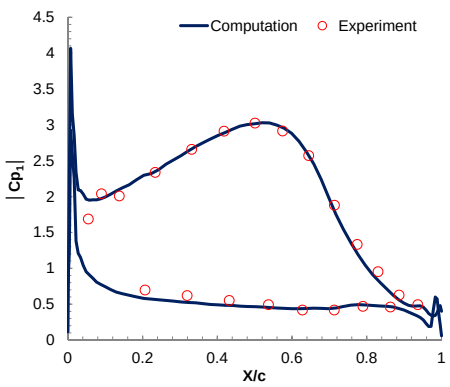
(a) 10% span

(b) 30% span

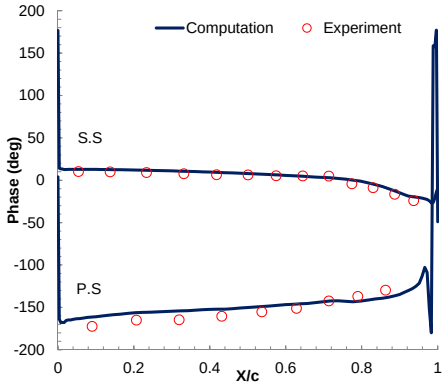
(c) 50% span

(d) 80% span

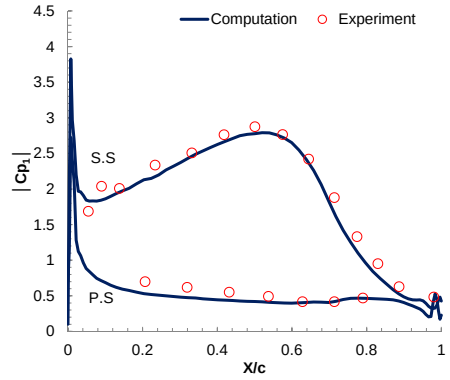
Figure 3.10: Time-averaged surface pressure coefficient



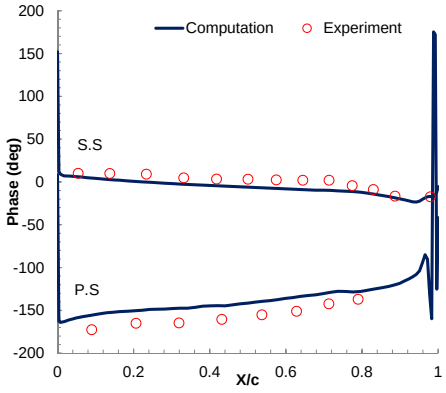
(a) Amplitude. IBPA=180 deg. 50% span



(b) Phase. IBPA=180 deg. 50% span



(c) Amplitude.IBPA=150 deg.50% span



(d) Phase. IBPA=150 deg. 50% span

Figure 3.11: First harmonic pressure coefficient (amplitude and phase angles).

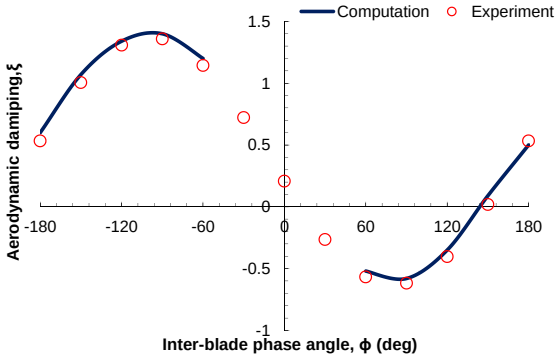
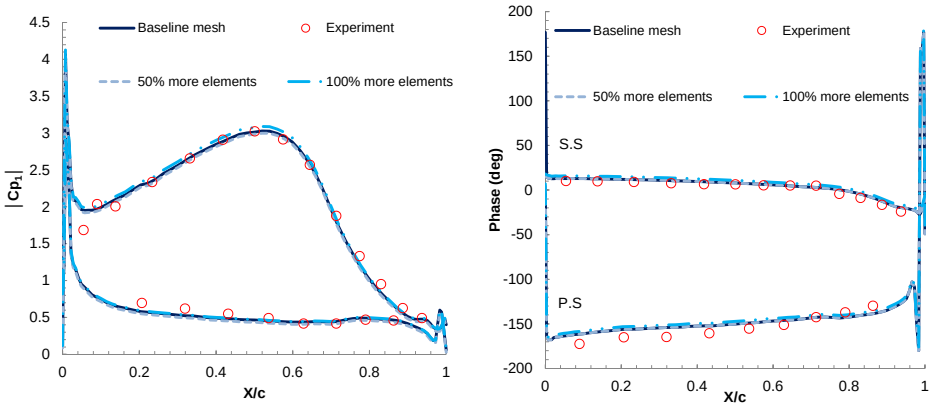


Figure 3.12: Aerodynamic damping coefficient at different inter-blade phase angles

Mesh convergence study

In addition to the steady and unsteady pressure validation, a mesh convergence study was carried out. The same linear cascade case was computed for three different meshes. One of these meshes was the same as the cases above, the next one had 50% more elements and the last one had 100% more elements. The first harmonic pressure coefficient and the phase angle distributions at midspan are compared for the three different cases with an Inter-Blade Phase Angle of 180 degrees. The results are shown in Figure 3.13. It can be seen how the first harmonic pressure coefficient and angle distributions are practically the same, which proves that the results from the validation above are no longer dependent on the mesh size.



(a) Amplitude. IBPA=180 deg. 50% span (b) Phase. IBPA=180 deg. 50% span

Figure 3.13: First harmonic pressure coefficient (amplitude and phase angles).

3.8 Summary

The Nonlinear Harmonic Phase Solution Method is used in this work. This flow solution method solves the URANS equations as a system of three steady-like solutions in the frequency domain, which allows for the computational domain to be truncated into one single passage, where phase-shift periodic boundary conditions are applied. The flow solver is able to calculate the flow field in a multi-bladerow environment due to the

mixing plane and harmonic interface treatment that exchanges the unsteady information across adjacent rows. The solver is validated against a 3D linear cascade case, with the results being compared against experimental data. The time-averaged and the first harmonic pressure (amplitudes and phase angles) coefficients are in good agreement with the experimental data. The aerodynamic damping is also satisfactorily compared for different inter-blade phase angles. The Nonlinear Harmonic Phase Solution Method has been proven as an efficient approach to simulate time-periodic unsteady flows for blade flutter applications. Moreover, the fundamentals of this flow solution method are essential for the development of an efficient multi-bladerow unsteady adjoint system.

Chapter 4

Non-gradient Based Optimization with Re-scaled Response Surface Model

This chapter presents the main elements of a concurrent non-gradient-based optimization method. The most distinctive feature of this method is that the standard Response Surface Model (RSM) is enhanced with an improved version called Re-scaled Response Surface model (RRSM). This new surrogate modelling technique is able to rescale the design space and redefine the response surface during a blade shape optimization process, leading to a much enhanced convergence compared to a standard RSM approach. As a result, the optimization method is able to perform automated concurrent blade aerodynamic and aeroelastic shape optimization of realistic 3D turbine stages in reasonable time for a moderate number of variables.

4.1 Database: Latin Hypercube Sampling

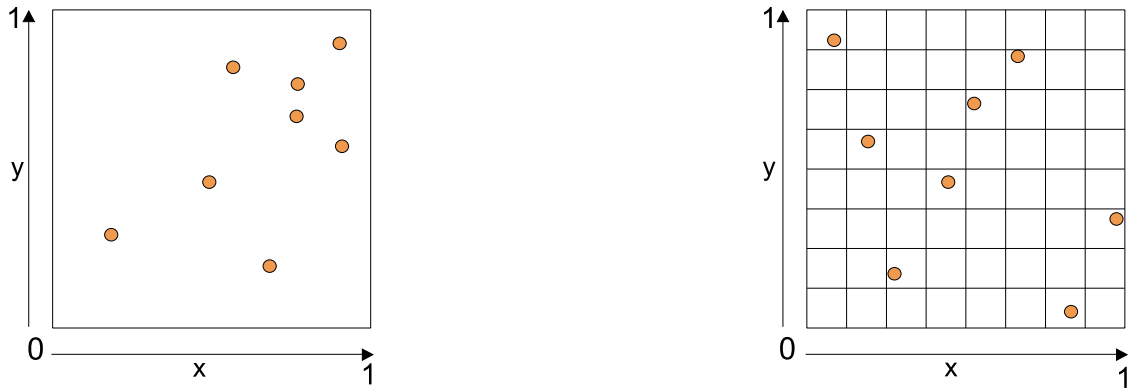
The database is the knowledge source to teach or build surrogate models. The database contains the set of samples that are used to define the surrogate model (metamodel) that will establish a relationship between the design variables or inputs, and the objective function or outputs. The accuracy of the predictions given by the metamodel will

strongly depend on the information that is stored in the database; a poor distribution of sample points can have a negative impact. Although samples can be created randomly and be used to populate the database, it is important to choose wisely the number of initial samples and their distribution in the design space. There are different and elaborated techniques that can be used to create databases. The method of choice for this work is called Latin Hypercube Sampling.

Latin hypercube sampling (LHS), initially proposed by McKay et al. (1979), is a statistical method that generates a set of samples from a multidimensional distribution. To better illustrate this, consider a design space of two dimensions as shown in Figure 4.1. A square grid with samples at different positions is a Latin square if there is only one sample for each row and each column. The Latin hypercube concept can be extended to an arbitrary number of dimensions. In a multi-dimensional space (a function of N variables), the range of each variable is divided into M equally spaced intervals. This number, M, is also equal to the number of sample points. In order to satisfy the latin hypercube Sampling requirements, only one sample is located in each aligned interval. Because there is no requirement for each specific sample to be chosen in a particular location, the maximum number of combinations for a Latin Hypercube of M divisions and N variables is given by:

$$\left(\prod_{n=0}^{M-1}(M-n)\right)^{N-1} = (M!)^{N-1} \tag{4.1}$$

As an example of an optimization case with $N = 2$ variables, a Latin Hypercube of $M = 4$ samples (a two-dimensional squared design space) will have 24 possible combinations. If the number of variables is increased to $N = 3$, a Latin Hypercube of $M = 4$ samples (a three-dimensional cubic design space) will have 576 possible combinations. An important feature of this sampling scheme is that it does not require more samples as the number of variables is increased. Despite this is true that more samples allow building more accurate metamodels, It is left to the designer to choose the balance between accuracy and cost.



(a) Random distribution

(b) Latin Hypercube Sampling

Figure 4.1: Comparison of sampling distributions strategies

4.2 Candidate Search: Differential Evolution

Non-gradient and a gradient-based methods mainly differ in their strategy to find a candidate design. Non-gradient methods do not need gradients of an objective function. The simplest approach they can follow is to randomly select a large number of candidates and evaluate the objective function of each of them. This process, very expensive and inefficient, is avoided by many designers. Instead, more elaborated approaches are followed such as the simulated annealing and evolutionary algorithms. It is the latter type that is implemented in this work.

Evolutionary algorithms were developed in the 70s by Holland (1975). These methods are based on Darwin's evolution theories, where populations of individuals evolve and adapt to the environment by different mechanisms, such as mutation, crossover and selection. The individuals with a higher fitness will have more chance to survive and reproduce. These type of methods have to start with a specified number of designs as initial population (e.g the initial set of samples from a database) and they require the evaluation of the entire population at each iteration. The objective function of each sample or individual is evaluated along with a fitness factor that is intended to be maximized. Evolutionary algorithms are still non-deterministic; there are no assumptions about the problem or ob-

jective function to be optimized and the evolutionary nature guides the search towards regions of the design space with expected improvement. However, there is no guarantee that an optimum solution will be found.

There are different evolutionary algorithms. The most popular ones are the Genetic Algorithm (GA) and the Differential Evolution (DE), the latter method being the one used in this work. The DE is a relatively new evolutionary algorithm developed by Price and Storn (1997). It does not require a continuous objective function, as is required by other optimization methods such as gradient descent and adjoint methods, and it is noise tolerant. A standard DE starts with a population of candidates (or individuals). These individuals are moved around in the design space (i.e their variables are changed) by using a simple mathematical method to combine the positions of existing individuals from the population. If the new position of an individual gives an improvement in the objective function, it is added as part of the population; otherwise it is discarded. The process is repeated and iterated for a certain number of n generations until, hopefully but not guaranteed, a satisfactory solution is found.

Each candidate or individual contains n variables or parameters:

$$\mathbf{x_c} = (x_1, x_2, ..., x_n) \tag{4.2}$$

An initial population of x individuals is randomly generated or created from a statistically sampled database. Until a fixed number of n generations have passed, the next process is repeated as following for each candidate $\mathbf{x_c}$ in the population:

- In order to evolve the individual, three parameter vectors $\mathbf{a_g}, \mathbf{b_g}$ and $\mathbf{c_g}$ are randomly chosen so that:

$$\mathbf{a_g} \neq \mathbf{b_g} \neq \mathbf{c_g} \neq \mathbf{x_g} \tag{4.3}$$

- This step known as 'mutation' is shown in Figure 4.2a. A trial (mutated) vector $\mathbf{y_g}$ containing n variables is defined based on a combination of the three vectors above

as:

$$y_i = a_i + F_i \cdot (b_i - c_i) \quad i = 1, ..., n \tag{4.4}$$

where F_i is a randomly defined constant $F_i \in]0,1[$ called differential weight.

- The next step is known as 'crossover' (Figure 4.2b). A new candidate vector $\mathbf{z_g}$ is defined by recombination (binary crossover) of the variables involving vectors $\mathbf{x_g}$ and $\mathbf{y_g}$ as follows:

$$z_i = \begin{cases} y_i & \text{if} \quad r_i \leq CR \\ x_i & \text{if} \quad r_i > CR \end{cases} \quad i = 1, ..., n \tag{4.5}$$

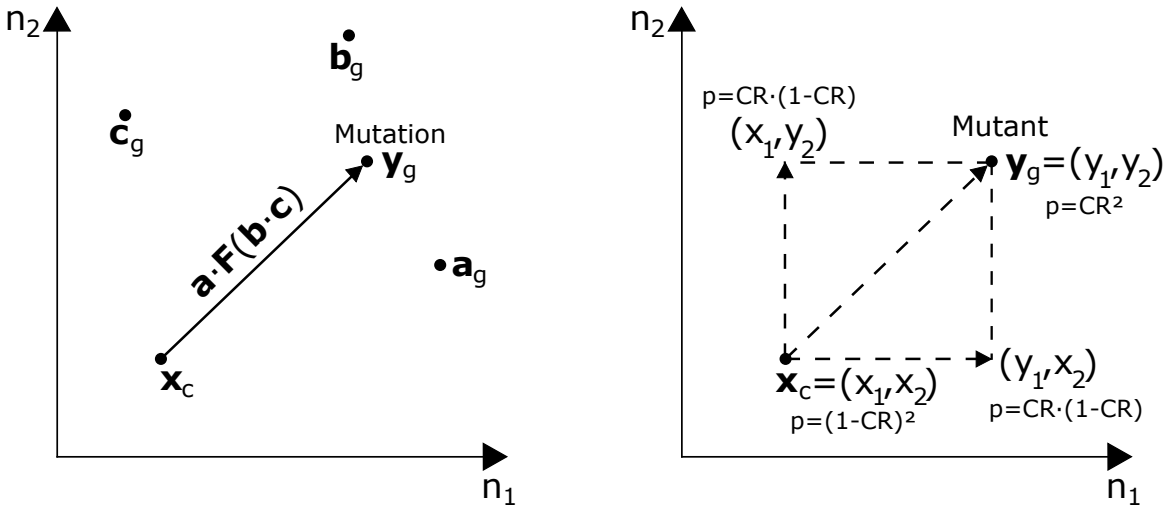
  where r_i is a randomly generated number ($0 \leq r_i < 1$) and CR is a user defined constant ($CR \in]0,1[$) called crossover probability. The new vector $\mathbf{z_g}$ will keep some variables from its 'parent' vector while the rest have been mutated. The probability for each combination of variables p is defined by theCR constant.

- The final stage involves the selection process and comparison of fitness quality, f, between the original candidate $\mathbf{x_g}$ and the new candidate $\mathbf{z_g}$.

$$\mathbf{x_{g+1}} = \begin{cases} \mathbf{z_g} & \text{if} \quad f(\mathbf{z_g}) \leq f(\mathbf{x_g}) \\ \mathbf{x_g} & \text{if} \quad f(\mathbf{z_g}) > f(\mathbf{x_g}) \end{cases} \quad i = 1, ..., n \tag{4.6}$$

  If the new candidate gives an improvement in the objective function over the original candidate, then $\mathbf{z_g}$ will substitute $\mathbf{x_g}$ for the new generation $g+1$.

After the maximum number of generations have been reached, the individual with the highest fitness and best objective function is chosen as the best candidate.



(a) Mutation step (y_g generation)

(b) Crossover step (z_g generation)

Figure 4.2: Example of mutation and crossover steps for two variables

4.3 Metamodel: Response Surface Model

The non-gradient method relies on predictions from a metamodel to save the need of performing expensive computations using a direct solver (e.g CFD). Once the metamodel is built, it is able to predict the performance of new candidates, and the evaluation of candidates using accurate solvers is limited to only those whose improvement of the objective function has been predicted first.

The Response Surface Model is a set of mathematical and statistical techniques in which the objective is to model the response of a system $\mathbf{y}$ influenced by n independent variables $\mathbf{x} = (x_1, x_2, ..., x_n)$, and is subject to observational errors, e, such as:

$$\mathbf{y} = f(x_1, x_2, ..., x_n) + e \tag{4.7}$$

The method was introduced by Box and Wilson (1951). The model itself is only an approximation, but it is easy to estimate and apply, considering that little is known about the real relations among variables. In practice, the real relationship between the final

output and the parameter values **x** is unknown, and subject to some uncertainty. However, the target of these type of methods is to estimate an optimum point that can lead to an objective function improvement, despite the estimation errors e, and inadequacies of the model. No physical laws are implemented. The relationship is modelled locally by linear, quadratic or cubic polynomials. In this work, the RSM is built by second-degree polynomials of the form:

$$f(\mathbf{x}) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{i} \beta_{i,j} x_i x_j \tag{4.8}$$

where β_o, β_i and $\beta_{i,j}$ are unknown coefficients. If more than one output is to be modelled (multi-dimensional output), Equation 4.8 is repeated for each dimension with different β values. The second-degree model is flexible, and it is the most common order for response surface approaches. It can take a variety of functional forms (maximize, minimize, or specific target), and it approximates the response surface locally. When building the RSM, it is mandatory that the number of samples from the database N_s, is greater than or equal to the number of unknown coefficients N_β, to avoid an under-determined system of equations. The β coefficients are commonly calculated (coefficient fitting) by least squares regression. The problem with most of the least squares coefficient fitting methods is that they need a larger number of samples than the minimum required by the β coefficients when the number of variables is large. Recently, some methods based on stochastic process have been used to fit these coefficients using nonlinear methods (Mishra, 2007). The main advantage over traditional least-squares methods is that there is no specific number of samples needed (apart from the β coefficients requirement) and, consequently, an accurate fitting of coefficients can normally be achieved with a smaller number of samples. Because of this, the DE algorithm by Mishra (2007) is chosen in this work for fitting the response surface coefficients.

The main drawbacks of the standard RSM approach is that some assumptions of the mathematical shape of the approximation are inherently made. These type of functions or surfaces will rarely correspond to the real response, which is in general mul-

timodal and nonlinear for aerodynamic and aero-mechanic optimization problems. In addition, the number of β coefficients or unknowns increases rapidly with the dimension of the problem. For a second-order RSM, the number of unknowns is proportional to $(n+2)(n+1)/2$ and for a third-order RSM, the proportion of unknowns increase to $(n+3)(n+2)(n+1)/6$. For a 10 variables problem, a second-order RSM needs at least 66 samples, and for a 20 variables problem, 231 samples. Hence, the method is not suited for an optimization case with a high number of variables. This can be partly alleviated by adding more samples and recomputing the β coefficients after each optimization cycle.

4.3.1 Re-Scaled Response Surface Model

One of the difficulties encountered during the present study is related to the convergence of the response surface model. The convergence is judged by the difference between the objective function prediction at the design variable values from the constructed response surface and the final objective function value obtained by a direct solution at the same design variable values. The fact that the surface coefficients are defined by least squares fitting makes it difficult to increase the resolution in a local area of the design space by simply adding more samples and refitting the surface coefficients again. With a standard RSM approach, there is a point where the difference between the performance given by the response surface and the direct solution at the same design variables cannot be reduced and hence, there is always a margin of uncertainty. Moreover, adding further sampling points may saturate the fitting technique and practically end up generating an oscillatory response surface, which does not help the convergence. This was quantified in a previous study that was carried out (Valero et al., 2014) and the magnitude of this uncertainty is shown in Figure 4.3 where the values of the objective function, I, predicted by the RSM and the ones calculated using direct solution (CFD) are compared. These oscillations are of a similar magnitude to the value of performance gain of interest during each design cycle (e.g. ~ 0.1% in stage efficiency), which, increases the risk of design cycle being stopped prematurely, preventing further meaningful design cycles to be carried out. It is important to recognise that even for a global optimization, its success will also have

to depend on how well the procedure identifies and captures the local performance variation in the design space. This consideration is still more important when searching in a local region for the best point, as more accuracy from the metamodel will be required in that area.

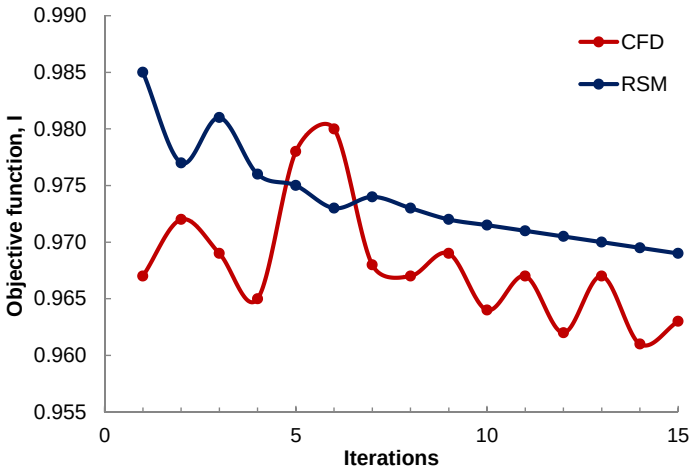


Figure 4.3: Objective function convergence for both CFD and RSM evaluations

The Re-scaled Response Surface Model (RRSM) is an approach that aims to address the convergence and accuracy issues of the standard RSM. Once the response surface of a design space has been constructed, the RRSM predicts the performance of the candidate points generated by the DE and the best candidates are sent to be evaluated by direct solution. The performance of these points is compared against the best point found so far, and:

- **If improvement is found**, the candidate is identified as a better design and a new optimization cycle is started; the response surface is refitted and a new set of candidate points is created by the DE.
- **If no improvement is found among all the candidates**, then the best solution found so far is identified as the centre of a new design space. The new space is created with a smaller size inside the original design space and a new set of sample points is defined by Latin Hypercube Sampling.

Figure 4.4 shows an example of the initial design space of a three variables optimization case. The black points represent the sample points and the grey point is the baseline design, located at the centre of the design space. A set of candidate solutions marked with the subscript i, which denotes the optimization cycle nº i, is evaluated by the direct solver. If there is improvement compared with the current best point (e.g. the baseline design for the first optimization cycle), the point that gives the best improvement is identified as the best point found so far, which is shown in the figure below coloured as green, while the rest of the points are coloured red. Because improvement has been found, a new optimization cycle starts ($i+1$); all the candidates that have been evaluated by the direct solution are stored in the database and the response surface coefficients are fitted again.

Figure 4.5 shows the design space of the next optimization cycle, $i+1$. The response surface coefficients have been refitted and a new set of candidate points is evaluated by direct solution. No improvement is found this time; all the points are coloured red and the green point from the previous optimization cycle remains as the best candidate. Because no improvement has been achieved, a re-scaling of the design space is now triggered.

Figure 4.6 represents the definition of the new design space. The re-scaling is performed around the best point found so far (i.e. the green point). A new set of samples (black points) is defined and all previous candidates that still are inside the new design space are added to the database as an extra set of samples. The region outside the new design space is removed.

Figure 4.7 shows the new design space after a re-scale. The size of the space reduction is specified by the user. With this Re-scaled RSM approach, the difference between the performance predicted by the response surface and the direct solution is reduced on each re-scale and convergence is enhanced**.** This convergence improvement behaviour can be seen in Figure 4.8 where the errors between the objective function, I, predicted by the response surface and those by the direct solution method are shown for two types of response surface modes: the standard RSM and the RRSM for one of the concurrent op-

timization cases of Valero et al. (2014). It can be observed in Figure 4.8 how the errors of the standard RSM oscillate around 6% (very high for the problems of the present interest) while the RRSM shows a starting error of 7.5% but it decreases well below 0.5% after a few design cycles (re-scaling is performed between iterations 2-3 and 4-5). The reduction of the error below 1% significantly increases speed of convergence and convergence is usually reached after 5-6 design cycles when the error is below 0.5%. It is important to note that with the RRSM approach, the definition of design cycle does not necessarily mean design iterations. Because a new design cycle is only started when improvement has been obtained, more than one sequential candidate evaluation (iteration) can be performed for the same design cycle. However, in order to show the convergence behaviour of both methods in Figure 4.8, the value of the error is shown as a function of iterations for both methods.

One of the potential disadvantages of this approach is that after performing a few re-scales, a big part of the original design space is no longer considered when looking for optimum points. Hence, chances of missing the global optimum may exist. These chances may be minimized by taking sensibly chosen re-scaled space. In practice, the present case studies to be presented later show that the proposed rescaled RSM model works very well, producing converged optimization solutions with significant performance gains which would be otherwise non-obtainable using a standard RSM.

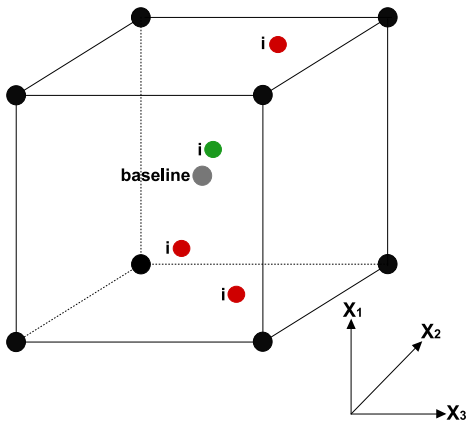


Figure 4.4: Design space at optimization cycle i with first set of candidates (red points)

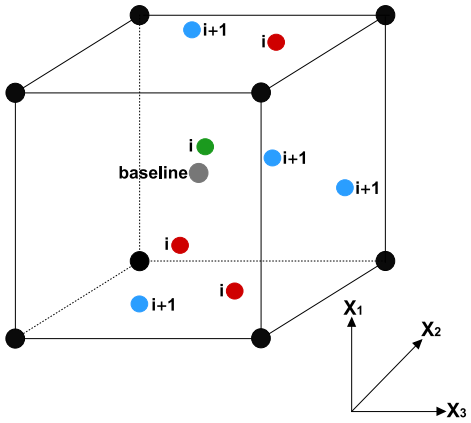


Figure 4.5: Design space at optimization cycle $i+1$ with new set of candidates (blue points)

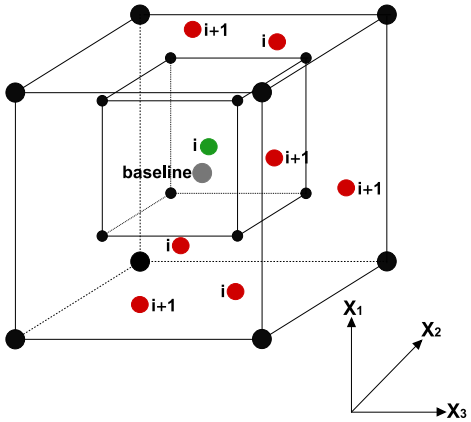


Figure 4.6: Design space at optimization cycle $i+1$ with a triggered re-scale

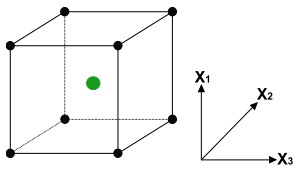


Figure 4.7: New design space after the re-scale at optimization cycle $i+1$

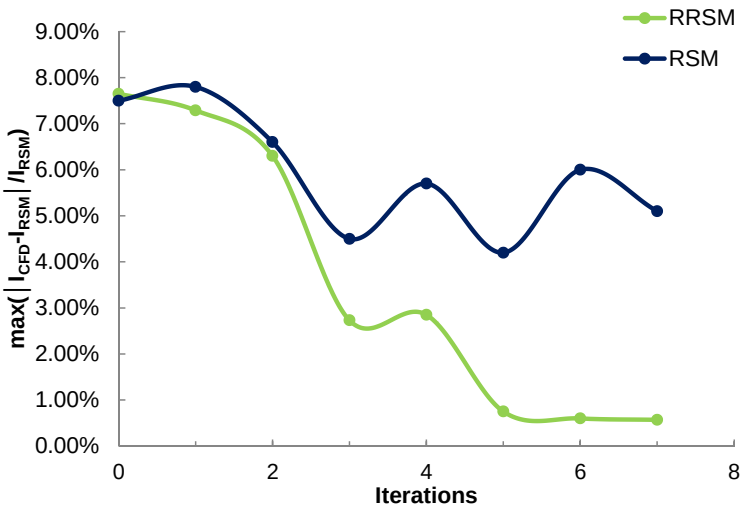


Figure 4.8: Relative error of objective function predictions. RSM and RRSM approaches

4.4 Method Validation

Two numerical case studies are carried out to evaluate the validity and effectiveness of the RRSM methodology by testing it against analytical functions. In the first case, the optimization system with the RRSM feature will predict the global optimum of a one-dimensional function with local minima and maxima. In the second case, the same approach will be tested against a two-dimensional function with the same topology as the one-dimensional function.

4.4.1 Optimization System

Figure 4.9 shows the main structure of the non-gradient optimization method that is used for these validations. The starting point for the optimization is the database creation which is done using Latin Hypercube Sampling. Once the samples are defined and evaluated, the RSM is defined and its coefficients are fitted. Next, a DE algorithm searches for the optimum candidate by mutation and crossover of the original database population. The target for the optimization method is to find the point where the function value $f(\mathbf{x})$ in Equation 4.7 is minimum.

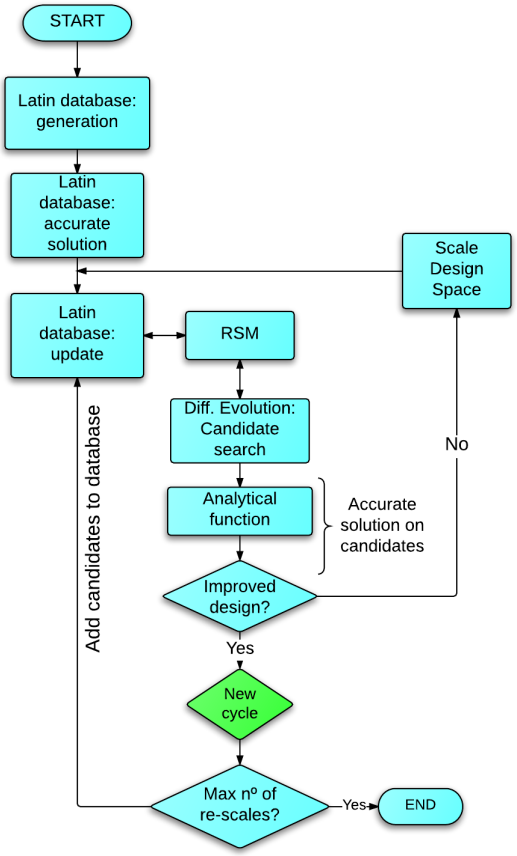


Figure 4.9: Flowchart of the Re-scaled RSM non-gradient method for validation purposes

1. The design space is defined and a set of database samples is generated from Latin hypercubes. For this validation, a specific set of uniformly distributed Latin samples is chosen. The objective function of the samples is calculated by direct evaluation.

2. Once the samples have been evaluated, the coefficients of the Response Surface Model are fitted.

3. The Differential Evolution algorithm calculates the best candidate.

4. The analytical value of these candidate is compared against the database samples and all the previous candidates that have been evaluated so far.

   (a) If there is improvement: the database is updated with the recently evaluated candidate and a new one is generated.

(b) If no improvement is found:

i. If the maximum number of re-scales has been reached: the optimization process ends.

ii. Otherwise: steps 1) to 5) are repeated with a smaller design space that is centered around the best point found so far.

4.4.2 Validation Against a One-dimensional Function

The 1D function chosen for the validation is:

$$f(x) = \cos[8(x-0.3)] + 3(x-0.2) + \sin\left(\frac{1}{x+0.2} + 2\right) \tag{4.9}$$

This function is shown in Figure 4.10 and it has a local minima on $x \backsimeq 0.65$ and a local maxima on $x \backsimeq 0.4$ while an absolute minima on $x \backsimeq 0.1$. The shape of this function is very different to any second-order polynomial curve and so it is a good case to validate the RRSM method. The target for the optimization method is to find an optimum close to the absolute minimum and this is expected to be achieved through a range of re-scales.

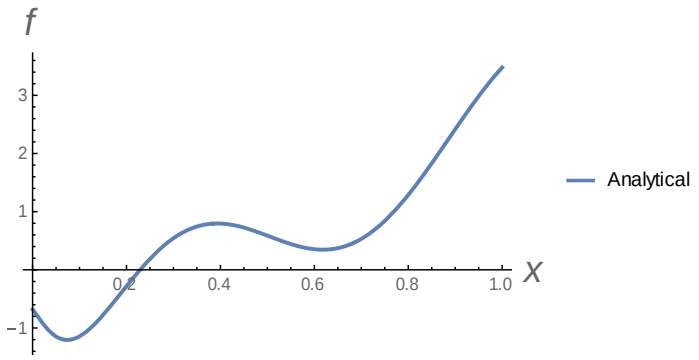


Figure 4.10: One-dimensional analytical function f(x)

Optimization cycle nº 1 (Re-scale nº 0):

The optimization process starts by building the starting database with samples that have been previously defined by Latin Hypercube Sampling. The analytical values of these

samples are obtained from Equation 4.9. Once the database has been built, the RSM coefficients are fitted. The curve defined by the RSM is shown in Figure 4.11. As expected from a second order polynomial, the curve is a wide parabola and its prediction of the complex features of the analytical function is expected to be poor. The Differential Evolution algorithm finds the best candidate by searching along the curve. This candidate, marked as a red point, is the best case by only looking at the RSM curve, but it is far from the absolute minimum of the analytical function. As no improvement has been found (the real value of the candidate point is the same as the sample from the database), the optimization system calls for a re-scaling and the design space is reduced to half of the original space.

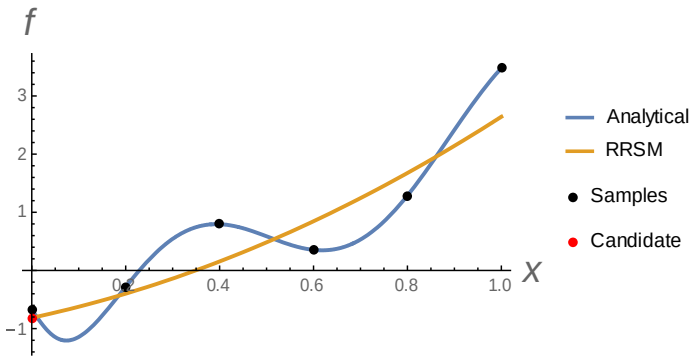


Figure 4.11: Design space of optimization cycle nº 1. Samples are shown in black while candidate point is red.

Optimization cycle nº 1 (Re-scale nº 1):

The design space is now smaller. A new RSM is fitted with a new set of samples from the database. The curve obtained from least squares by the RSM is shown in Figure 4.12. The limitations on the shape of a second order polynomial curve are still evident and the candidate found by the Differential Evolution is still not well predicted and not close enough to the real absolute minimum. Again, as no improvement has been found, a new re-scaling is performed.

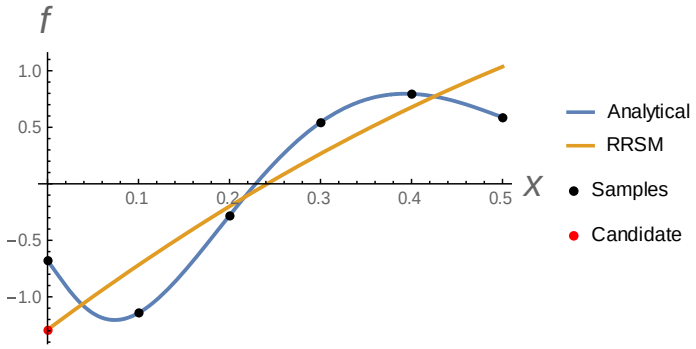


Figure 4.12: 1D f(x) and RRSM prediction at optimization cycle nº 1 after the first re-scale.

Optimization cycle nº 1 (Re-scale nº 2):

The design space is now a fourth of the original design space. However the absolute minimum is still inside the new design space. A new database is defined and the RSM is fitted accordingly. It can be seen now (Figure 4.13) how the re-scale process produces a better match of the shape of both curves in a smaller region. The real analytical value of the candidate generated by the Differential Evolution is now the best of among previous cases and close to the minimum. As improvement has been found, a new optimization cycle is started and a new search of a candidate is expected to close the gap and displace the best candidate location closer to the exact value of the absolute minimum.

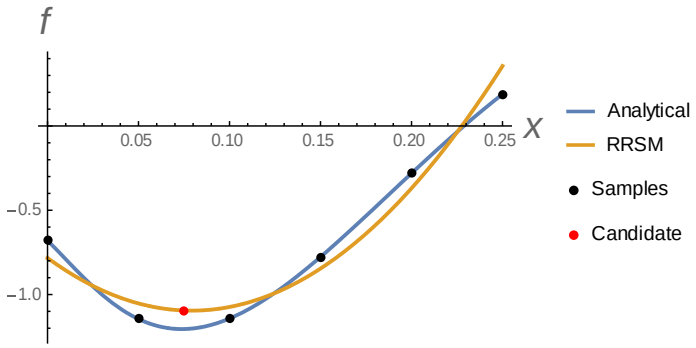


Figure 4.13: 1D f(x) and RRSM prediction at optimization cycle nº 1 after the second re-scale.

4.4.3 Validation Against a Two-dimensional Function

The process of validation against a 2D function is similar to what has been described in the above section. The 2D chosen function is:

$$f(x,y) = \sin\left[f_1(x)\right] + \sin\left[f_2(y)\right] \tag{4.10}$$

where

$$f_1(x) = \cos\left[8\,(x-0.3)\right] + 3\,(x-0.2) + \sin\left(\frac{1}{x+0.2} + 2\right)$$

$$f_1(y) = \cos\left[8\left(y-0.3\right)\right] - 3\left(y-0.2\right) - \sin\left(\frac{1}{y+0.2} + 2\right)$$

This function is an extension of the 1D function used for the previous validation. The surface of this function (Figure 4.14) has also a few local minima and maxima, an absolute maximum on $\left(x,y\right) \backsimeq (0.84, 0.11)$ and and absolute minimum on $\left(x,y\right) \backsimeq (0.09, 0.82)$. The target for the optimization method is still the same: to find an optimum close to the absolute minimum.

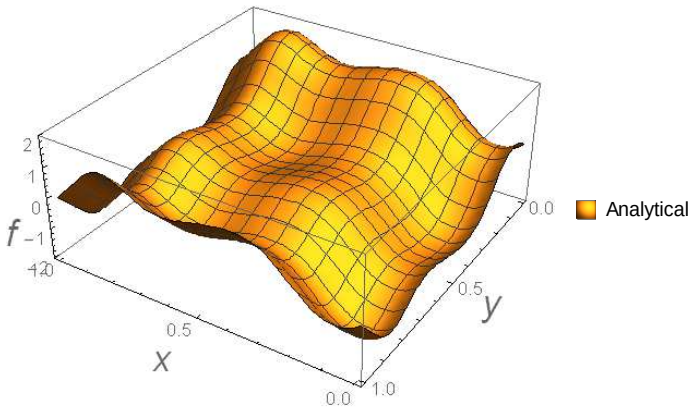


Figure 4.14: Two-dimensional analytical function f(x)

Optimization cycle nº 1 (Re-scale nº 0):

As with the 1D validation, the database of samples is built and the RSM coefficients are fitted. The surface defined by the RSM is shown in Figure 4.15. The RSM surface is very

steep but it is close to the analytical surface near the absolute minimum. The real value of the analytical function at the coordinates of the candidate predicted by the Differential Evolution algorithm is no better than the adjacent database sample. Hence, a re-scale around this region is triggered.

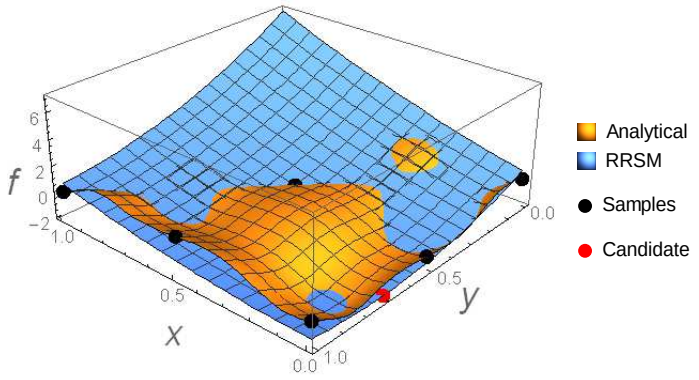


Figure 4.15: 2D f(x) and RRSM prediction at optimization cycle nº 1. Samples are shown in black while candidate point is red.

Optimization cycle nº 1 (Re-scale nº 1):

The new re-scaling reduces the design space to a half (Figure 4.16). The new surface built by the RSM is less steep and much closer to the analytical surface. However, near the absolute minimum, the predicted value of the candidate has been, again, overestimated. As a result, the adjacent sample from the database has a better real value. A new re-scaling is performed around this region.

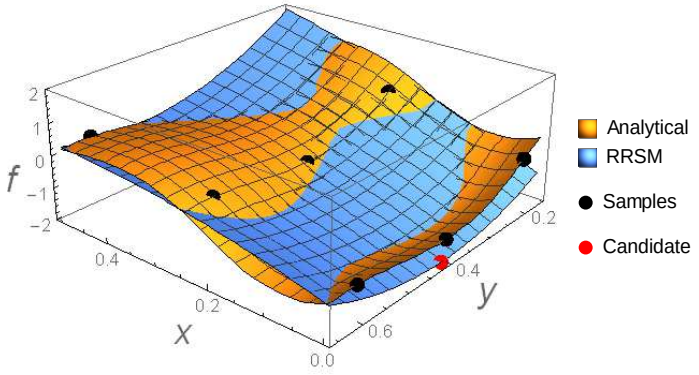


Figure 4.16: 2D f(x) and RRSM prediction at optimization cycle nº 1 after first re-scale.

Optimization cycle nº 1 (Re-scale nº 2):

The last re-scaling reduces again the design space (Figure 4.17). The new design space is now $\frac{1}{4}$th of the original design space. Near the absolute minimum region, the two surfaces coincide. Despite the slope of the surface built by the RSM being steep, the prediction of the candidate now returns the best value found so far (better than the database samples). Consequently, there is improvement, and a new optimization cycle starts with a new re-fitting of the coefficients from a database that has a new extra sample. The slope of this surface by the RSM will be adjusted with the re-fitted coefficients, until closing the gap between the predicted optimum and the real absolute minimum or until the maximum number of re-scales is reached.

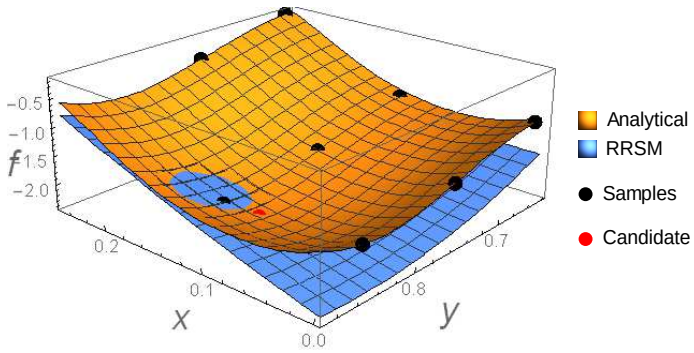


Figure 4.17: 2D f(x) and RRSM prediction at optimization cycle nº 1 after second re-scale.

4.5 Summary

The main elements of a non-gradient optimization method have been presented in this chapter, including the database sampling technique, candidate search and surrogate model technique. The distinctive component for this work is the RRSM. It has been validated against two analytical functions, and an improvement in accuracy over the standard method has been reported, finding the global optimum where a simple standard RSM may not be able to. This enhanced performance will be proven in the next chapter, where this method will be used to perform concurrent aeroelastic optimization with benchmark single-bladerow cases and real turbomachinery stages. This will also highlight the differ-

ences in complexity and global optimum search and its drawbacks in terms of computational cost when compared with the adjoint gradient-based method that is presented later.

Chapter 5

Optimization Case Studies using Re-scaled Response Surface Method

5.1 Introduction

The validation in section 4.4 showed some evidence for analytical cases that a non-gradient optimization method with a RRSM surrogate model provides better features than with a standard RSM. However, this is yet to be demonstrated for more realistic cases of practical interest. In this chapter, the RRSM is tested for concurrent optimization of turbomachinery stages.

5.2 Structural Modeling: FE-CFD interactions

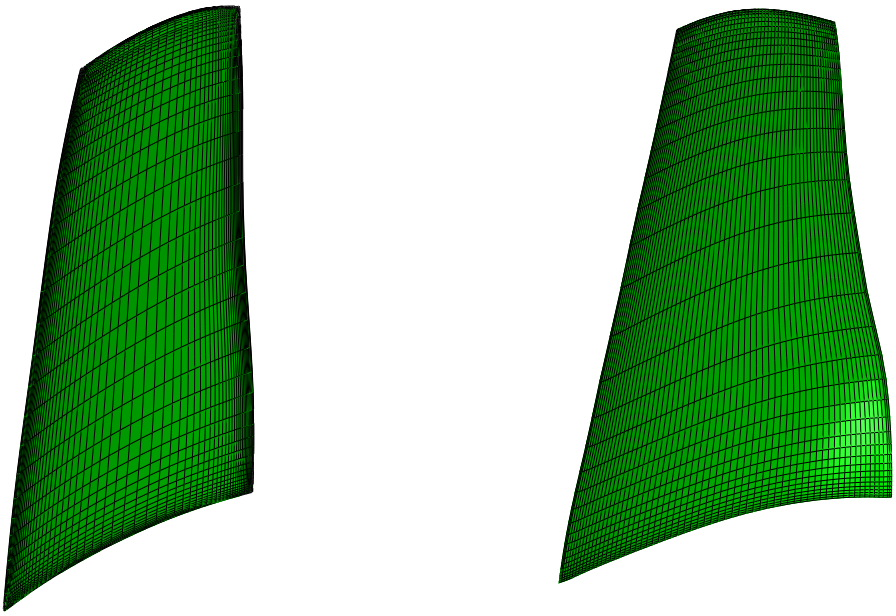
In an aeroelastic or concurrent design optimization, the blading performance is obtained from the unsteady flow solution in three phases. The flow field for blade flutter applications is largely linear (Moffat and He, 2003) and its influence on the vibrational frequency and structural mode shape is small and negligible. The aerodynamic performance is obtained from the time-averaged flow solution and the aeroelastic performance is obtained from the first harmonic component of the unsteady flow.

The blade flutter prediction is done in two sequential parts: first the structural dynamics (mode shape and vibration frequency of the first harmonic mode) is extracted using modal analysis and exported to the CFD solver to calculate the unsteady flow solution. As the time-averaged flow solution is independent of the structural dynamics, the major factor that changes the mode shape and vibration frequency is a change in the design variable (blade shape). The modal analysis is carried out using the finite element analysis software Abaqus. It is important to keep in mind that the FE and CFD spatial domains are different. The FE mesh discretizes the solid blade itself while the CFD mesh discretizes the space surrounding the blade. Moreover, the requirements of mesh nodal distribution is different due to the different gradients that each method has to resolve, the FE mesh being the least demanding in terms of number of nodes at the blade surface.

However, in this work the FE mesh is generated from the surface CFD mesh. The nodes of the blade surface are exported directly and the solid blade region is filled with tetrahedral elements (C3D6 and C3D8 elements in Abaqus) based on the nodes at the blade surface. Despite the resulting mesh being more expensive to evaluate than a conventional FE mesh, the cost of a linear frequency analysis is still cheap enough compared to the overall computational cost of the optimization process. This way the need for FE-CFD mesh interpolators is removed, allowing a direct export of the mode shape and frequency information back to the CFD solver. Figure 5.1 shows the FE mesh of the NASA rotor 67 blade and DLR rotor blade that is exported to Abaqus. After a linear frequency analysis, Figure 5.2 shows contours of the magnitude of the displacement of the first bending mode with the scaled modeshape.

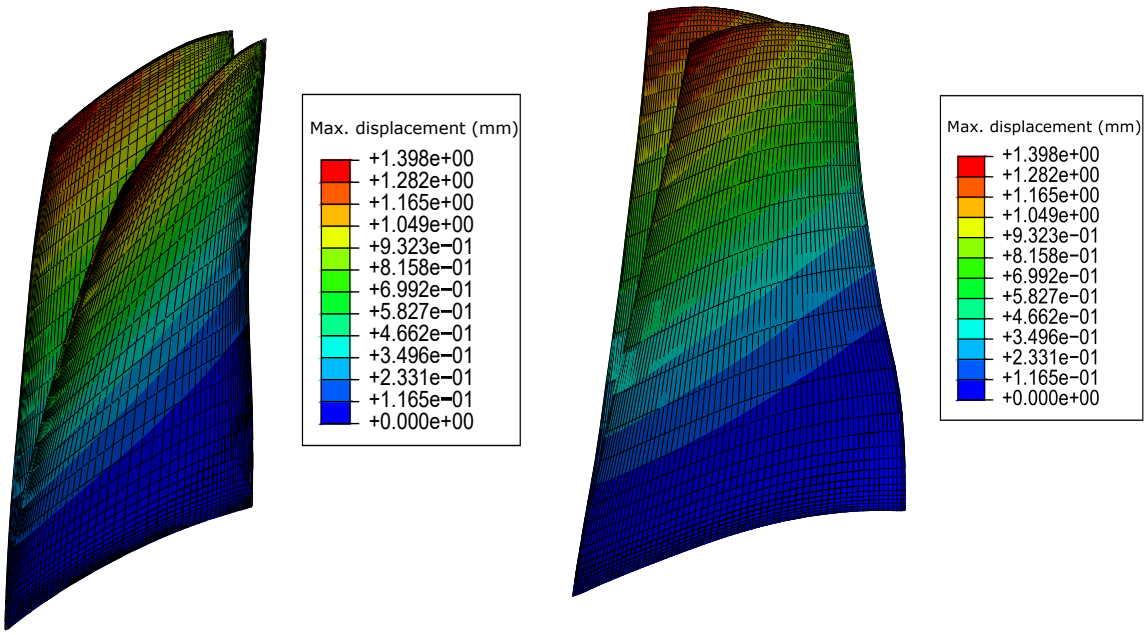
5.3 Blade Flutter Evaluation

In this work, damping is the target of improvement for aeroelastic design optimization. The magnitude that gives the aerodynamic damping is the work sum W. Aerodynamic



(a) DLR rotor blade (b) NASA rotor 67

Figure 5.1: Imported finite element mesh in Abaqus



(a) DLR rotor blade. First bending mode. Natural frequency = 746.54 Hz, scale factor = 9.77

(b) NASA rotor 67. First bending mode. Natural frequency = 526.96 Hz , scale factor = 15.43

Figure 5.2: Absolute displacements (mm) of first bending mode.

damping is calculated using the Energy Method (He, 2012) as follows:

$$\delta = -\frac{W}{E_s} \tag{5.1}$$

where the work sum W is defined as the aerodynamic work done by the fluid to the vibrating blade in one vibration cycle and E_s is the total strain energy of the vibrating solid blade, calculated as the sum of the vibrational kinetic energy. More details on work sum can be found in section 5.5. As it can be seen from Equation 5.1, positive work sum leads to a reduced aeroelastic stability in the blade, while a negative work sum means that energy is being dissipated from the blade into the working fluid, increasing aero-damping, thus aeroelastic stability.

After the structural dynamics information is exported to the CFD solver, the total work sum on the vibrating blade surface is calculated. Figure 5.3 shows the contours of local work on the NASA rotor 67 blade surface.

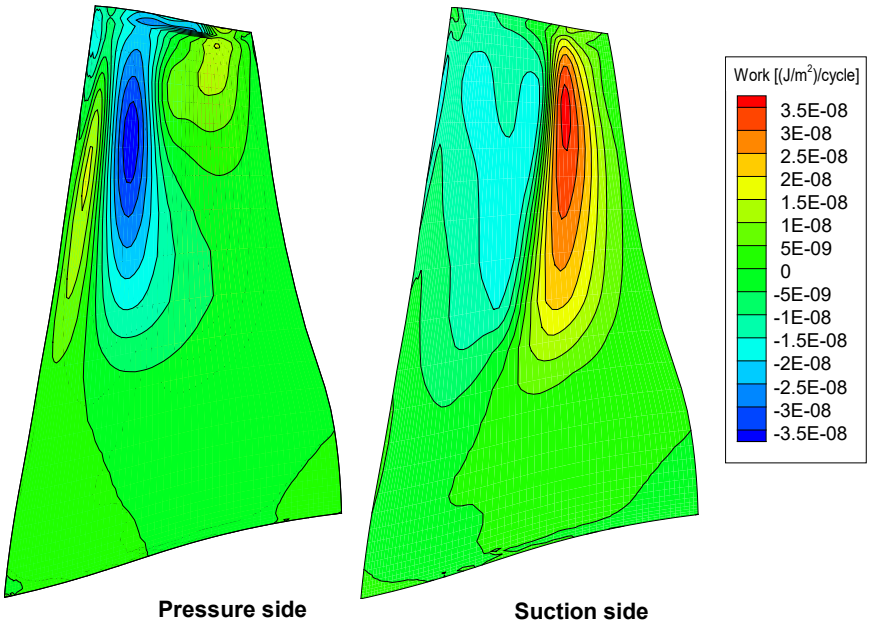


Figure 5.3: Local work contours (J/cycle) of NASA rotor 67 blade (first bending mode)

5.4 Shape Parameterization

One of the most important aspects in blade design optimization is shape parameterization. A good shape parameterization must be complex enough to include most of the geometrical features of modern blades while keeping the number of variables needed for the parameterization as low as possible. It also needs to be robust enough in order to produce realistic geometries without significant unwanted distortions.

There are mainly two approaches to parameterizing three-dimensional blades: blade shape parameterization and blade shape perturbation. The shape parameterization method in this work belongs to the latter and it is based on the Hicks-Henne functions implemented by Wang and He (2010a,b). The benefit of this approach is that the baseline geometry can be recovered by setting back the perturbation to zero. The same shape parameterization is implemented for both the non-gradient and gradient-based methods of this work.

These Hicks-Henne hump functions use a reduced system of coordinates by mapping the camber line into a rectangle, as shown in Figure 5.4. The reduced coordinates x and r take values between 0 and 1. The edges of the original blade geometry are translated to four straight edges of a rectangle. For any mesh point with nodal coordinates (j,k) in the axial and radial direction respectively, its reduced coordinates are:

$$x_d = \frac{X(j,k) - X(j_{le},k)}{X(j_{te},k) - X(j_{le},k)} \tag{5.2}$$

$$r_d = \frac{R(j,k) - R(j_{hub},k)}{R(j_{tip},k) - R(j_{hub},k)} \tag{5.3}$$

where the sub-indexes le, te, tip and hub denote the leading edge, trailing edge, tip and hub respectively. For a set of hump perturbation nodes, N, all the mesh points on the blade surface (both suction and pressure side) have their tangential coordinate changed

according to the next 2D Hicks-Henne function:

$$\delta_{r\theta} = \sum_{i=1}^{N} \left\{ a_i \cdot \frac{\cos^{b_r}\left[\frac{\pi}{2}(r_d - r_i) - y_r\right]}{1 - y_r} \cdot \frac{\cos^{b_x}\left[\frac{\pi}{2}(x_d - x_i) - y_x\right]}{1 - y_x} \right\} \tag{5.4}$$

where

$$y_r = \min\left\{\cos^{b_r}\left(r_i \frac{\pi}{2}\right), \cos^{b_r}\left[(1 - r_i)\frac{\pi}{2}\right]\right\}$$

$$y_x = \min\left\{\cos^{b_x}\left(x_i \frac{\pi}{2}\right), \cos^{b_x}\left[(1 - x_i)\frac{\pi}{2}\right]\right\}$$

The other two reduced coordinates x_i and r_i are referred to the location of each corresponding hump perturbation node, i. The a factor controls the size of the perturbation of the corresponding hump node. b_x and b_r are two factors that control the influence of the tangential perturbation of hump node in the mesh points. The bigger the value, the more local the perturbation will be. If the number of variables is large, it is better to keep the perturbations as local as possible. However, if the number of variables is small, the hump factors can have smaller values so the perturbation is more global and smooth. The blade tangential thickness is kept constant in order to avoid unrealistically thin blades. A single hump perturbation of a factor $a = 1x10^{-4}$ at coordinates $x_i = 0.5$ and $r_i = 0.5$ is illustrated in Figure 5.5. As it can be seen, one hump perturbation is able to create a smooth global change across the blade surface on its own.

5.5 Objective Function

The objective function is a weighted sum of performance parameters of interest, and it is the target value that the design optimization method aims to minimize. In terms of aeroelasticity, blade damping is quantified in the objective function with the work sum. On the other side, the aerodynamic performance is represented by the entropy generation rate. In addition, the objective function also has two constraints: mass flow rate and pressure ratio. There are two different forms of objective function depending on whether

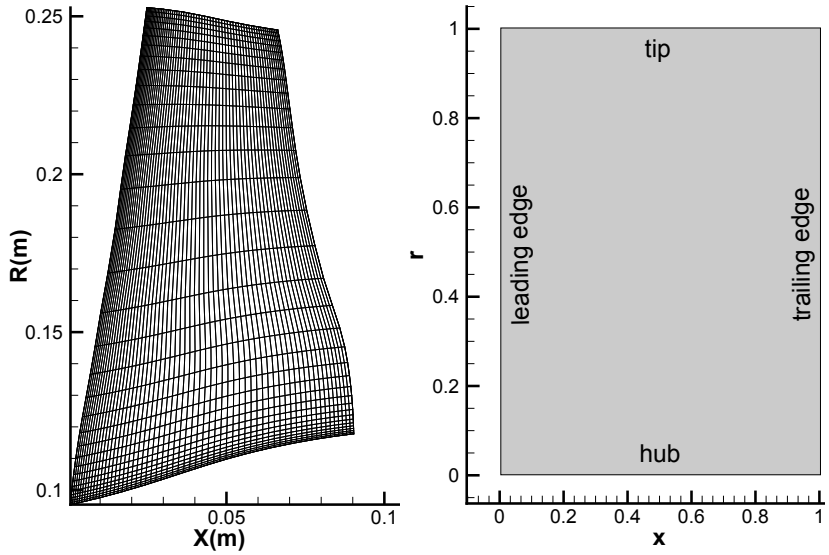


Figure 5.4: Coordinate transformation done by Hicks-Henne functions. NASA rotor 67

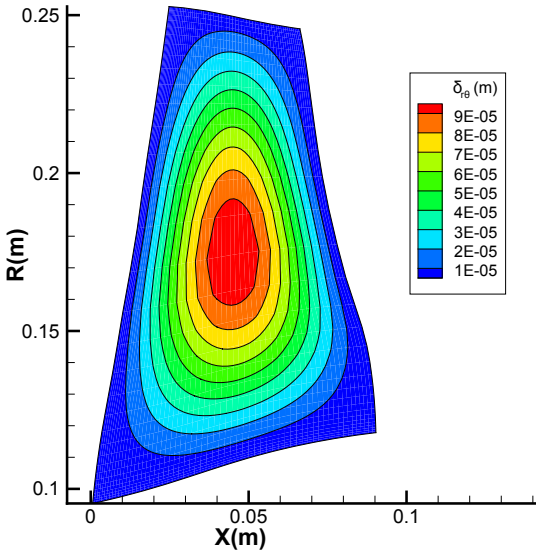


Figure 5.5: Local contours of tangential coordinate change, $\delta_{r\theta}$. 1 perturbation node at $x_i = 0.5$ and $r_i = 0.5$ with factor $a = 1x10^{-4}$

aerodynamics or aeroelasticity is the target of improvement:

$$I = \frac{\triangle \bar{s}}{\triangle \bar{s}_0} + \sigma_1 \left(\frac{\dot{m}}{m_0} - 1 \right)^2 + \sigma_2 \left(\frac{\dot{\pi}}{\pi_0} - 1 \right)^2 + \sigma_3 \cdot \max \left[\text{sign}\left(1, W - W_0\right), 0 \right] \cdot \left(\frac{W}{W_0} - 1 \right)^2 \quad (5.5)$$

$$I = \frac{W}{W_0} + \sigma_1 \left(\frac{\dot{m}}{m_0} - 1 \right)^2 + \sigma_2 \left(\frac{\dot{\pi}}{\pi_0} - 1 \right)^2 + \sigma_3 \cdot \max \left[\text{sign}\left(1, \triangle \bar{s} - \triangle \bar{s}_0\right), 0 \right] \cdot \left(\frac{\triangle \bar{s}}{\triangle \bar{s}_0} - 1 \right)^2 \quad (5.6)$$

where W_0 is the work sum value of the baseline design and $\triangle s_0$ is the mean flow entropy rise of the baseline design. σ_1, σ_2 are the weighting factors of the aero-thermal constraints. The third constraint σ_3, is either an entropy or work sum constraint. If $(\sigma_3 > 0)$, the constraint is activated when the terms $(W_0 - W)$ and $(\triangle \bar{s} - \triangle \bar{s}_0)$ are positive in Equations 5.5 and 5.6 respectively. Based on the above objective functions, there are four different types of design optimization that can be performed:

- **'Aerodynamic-only' design optimization:** In this case, the objective function is the one from Equation 5.5. The target now is to decrease the entropy generation rate and improve aerodynamic performance. The third constraint σ_3, is a work sum constraint and it is disabled and equal to zero. Hence, the aeroelastic performance is not taken into account.

- **'Concurrent aerodynamic' design optimization:** Based on the aerodynamic objective function (Equation 5.5), the work sum is constrained $(\sigma_3 > 0)$. Aerodynamic performance is still the main target of improvement. However, depending on how large the value of σ_3 is, the optimization method will aim to correct and keep the same or better aeroelastic performance as the baseline design in every iteration the aeroelastic performance decreases with respect to the baseline design.

- **'Aeroelastic-only' design optimization:** Based on Equation 5.6, the entropy gener-

ation rate constraint σ_3 is disabled and equal to zero. This means that the entropy generation is not being taken into account in the design optimization process and so, the aeroelastic improvement is not constrained by the aerodynamic performance.

- **'Concurrent aeroelastic' design optimization:** Based on the same aeroelastic objective function as above, the entropy generation rate constraint is now enabled and set to a specific value ($\sigma_3 > 0$). Similarly to the concurrent aerodynamic design optimization case, the optimization method will aim to correct and keep the same or better aerodynamic performance while improving the aeroelastic performance.

Entropy generation rate

The mass averaged entropy generation rate $\triangle \bar{s}$ is defined as the difference between the mass averaged entropy at the exit and at the inlet:

$$\triangle \bar{s} = \bar{s_e} - \bar{s_i} \tag{5.7}$$

where $v_n = (u \cdot n_x + v \cdot n_\theta + w \cdot n_r)$ is the normal velocity for the area over which the integration is performed.

Mass flow rate

The mass flow rate $\dot{m}$ is defined by:

$$\dot{m} = \frac{1}{2}\left(N_{bi}\int_{S_i} \rho v_n \cdot dS_i + N_{be}\int_{S_e} \rho v_n \cdot dS_e\right) \tag{5.8}$$

where N_{bi} and N_{be} are the number of blades of the first blade row (inlet) and last blade row (exit).

Stagnation pressure ratio

The stagnation pressure ratio is defined as follows:

$$\pi = \frac{\bar{P}_e}{\bar{P}_i} \tag{5.9}$$

$$\bar{P} = \frac{\int \bar{p}\rho v_n dS}{\int \rho v_n dS}$$

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

Where $\bar{P}$ is the mass averaged total pressure and $\bar{p}$ is the local total pressure. W is the local velocity magnitude and T the local static temperature.

Work sum

The work sum is a function of space and time and it is calculated on the vibrating blade with the integration of the unsteady pressure weighted by the mesh vibration velocities as follows:

$$W = \int_S \int_0^T p\left(v_x \cdot n_x + v_\theta \cdot n_\theta + v_r \cdot n_r\right) dt dS \tag{5.10}$$

Where T is the blade vibration period. p is the unsteady pressure, which is a function of space and time:

$$p = p(x, \theta, r, t)$$

v_x, v_θ and v_r represent the mesh vibration velocities:

$$v_x = v_x^A \sin\omega t + v_x^B \cos\omega t$$

$$v_\theta = v_\theta^A \sin\omega t + v_\theta^B \cos\omega t$$

$$v_r = v_r^A \sin\omega t + v_r^B \cos\omega t$$

5.6 Design Optimization Process

Figure 5.6 shows the overall structure of the design optimization system. It is a similar process to that described in the validation section 4.4 but now the FE and CFD solvers are included in the design process as direct solvers replacing the analytical function. In addition, the optimization program allows for a set of multiple candidates to be generated if requested. The design optimization process can be summarised in the next steps:

1. The design space is defined and a set of database samples is generated with Latin Hypercube Sampling.

2. The flow solution is obtained for each of the database samples and their objective function values are calculated.

3. The Response Surface Model coefficients are fitted by least squares.

4. The Differential Evolution algorithm generates the fittest candidate.

5. The new set of candidates is evaluated with direct solvers in a two-step process: First, the mode shape and vibration frequency information is obtained from the FE solver and then, the flow solution is calculated afterwards. The objective function of these candidates is compared against the best point found so far (i.e the baseline design for the first iteration) and the database cases:

   (a) If improvement is found: the database is updated with the latest candidates, RSM coefficients are re-fitted and a new set of candidates is generated.

   (b) If no improvement is found:

      i. If the maximum number of re-scales has been reached: the optimization process ends.

      ii. Otherwise: steps 1) to 5) are repeated with a smaller design space that is centered around the best point found so far.

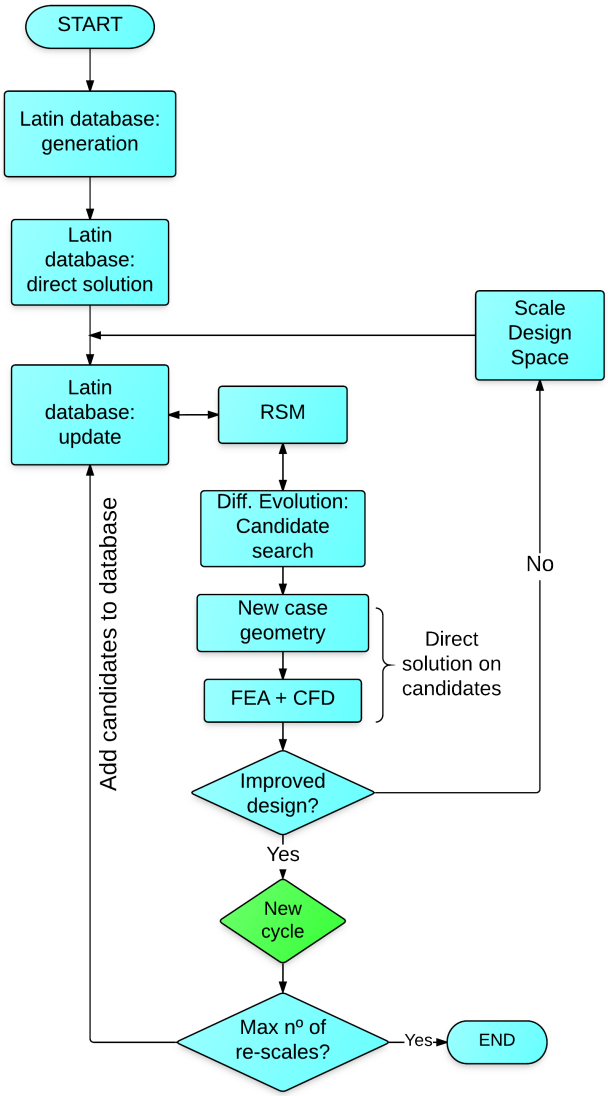
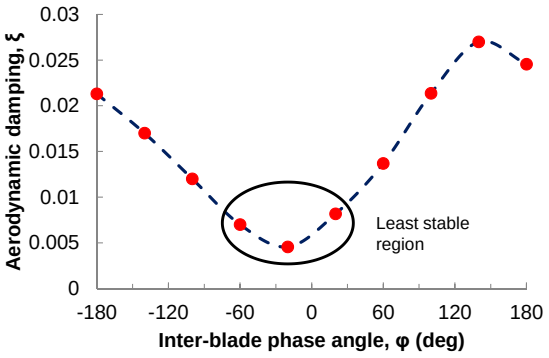


Figure 5.6: Flowchart of the non-gradient optimization method with the enhanced RRSM technique

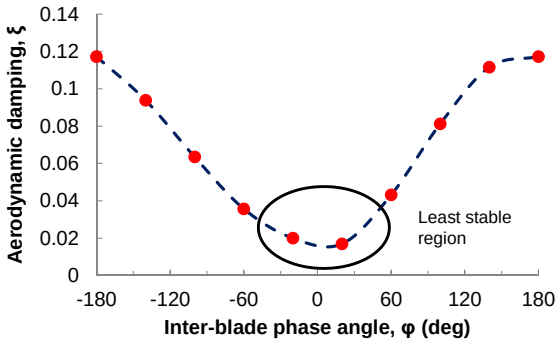
5.7 Design Optimization Cases and Sensitivity Analysis

Three different case studies are presented in this section. The first case is the transonic rotor NASA 67, the second case is a transonic compressor stage designed by DLR and the third case is an industrial low pressure turbine stage.

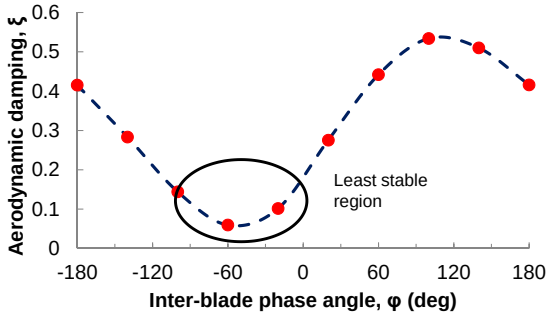
All three optimization cases are carried out at their peak efficiency operating points re-



(a) NASA rotor 67



(b) DLR compressor



(c) Industrial LP turbine

Figure 5.7: Aerodynamic damping coefficient at different inter-blade phase angles

spectively. In terms of unsteady disturbances, it is useful to evaluate the margin of blade damping improvement that an optimum solution can provide at the least stable aeroelastic condition. Because of this, the minimum aerodynamic damping across the IBPA range for each of the cases was calculated and shown in Figure 5.7. Despite none of these cases being unstable unstable ($\xi < 0$) at any IBPA value, each of the cases that are presented are evaluated at the least stable IBPA.

A moderate change in the blade shape will inevitably cause a change in the pressure ratio and mass flow rate of the row and or stage of interest. This change of operating condition can also produce a change in the two target parameters of the objective function: isentropic efficiency and aeroelastic log-dec. In order to value the margin of improvement linked to a blade shape change, it is important to quantify how much of this change is due only to a change in the operating conditions. Ideally, the best approach to com-

pare the sensitivities of the different cases to mass flow rate and pressure ratio changes would be directly changing these parameters in separated cases. Due to how the boundary conditions are specified in the flow solution method this is not possible. However, a similar alternative can be obtained by modifying inlet total pressure and exit static pressure boundary conditions. For each case study presented below, three different cases are evaluated with the same baseline geometries but different operating conditions as follows:

1. Baseline operating conditions
2. Baseline operating conditions with inlet total pressure and exit static pressure, both relatively increased by 0.2%
3. Baseline operating conditions with exit static pressure relatively increased by 0.2%

Changing simultaneously inlet total and exit static pressure values keeps the baseline design mass flow rate while only changing the pressure ratio of the domain. On the other side, a change in only exit static pressure will cause a small change in mass flow rate and a proportional change in pressure ratio.

The change in performance parameters of the objective function (isentropic efficiency and aeroelastic log-dec) is compared among the three cases.

5.7.1 NASA Rotor 67

The NASA rotor 67 is the first stage rotor of a two-stage fan. The computational domain consists of a H-type mesh with a density of $115x52x38$ in the axial, pitchwise and radial directions respectively. Figure 5.8 shows a blade-to-blade and meridional view of the computational domain. This rotor is a popular test case for automated design optimization methods (Li et al., 2006; Lian and Liou, 2006; Pierret, 2005).

The results of the mass flow rate and pressure ratio sensitivity analysis are shown in Table 5.1. Compared to the baseline case, an increase in exit static pressure and total inlet pres-

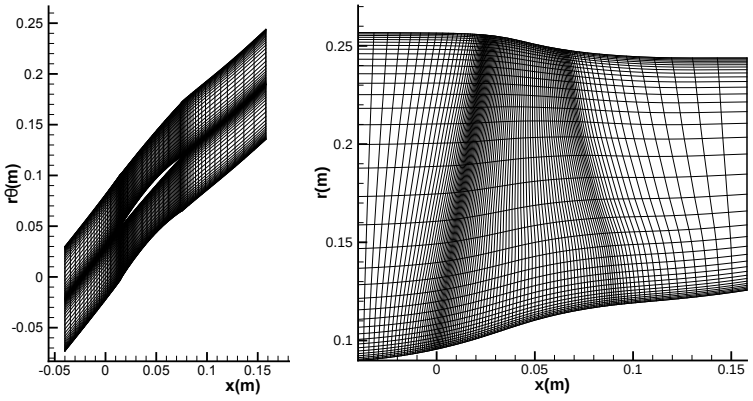


Figure 5.8: Computational domain of the NASA rotor 67. Blade to blade view (left) and meridional view (right)

Case	Baseline	Static and total pressure		Total pressure	
	Values	Value	Rel. change (%)	Values	Rel. change (%)
Mass flow rate (kg/s)	33.37	33.44	+0.21	33.36	-0.02
Pressure ratio	1.656	1.656	0	1.658	+0.15
Isentropic efficiency (%)	91.72	91.72	0	91.61	-0.12
Log-dec (%)	3.12	3.12	0	3.03	-2.88

Table 5.1: Sensitivity analysis NASA rotor 67. Values and relative change of obj. function parameters

sure of 0.2% increases the mass flow rate of the row by 0.2% and does not change the isentropic efficiency and aeroelastic log-dec. An increase in only exit static pressure changes the pressure ratio by 0.15% while slightly changing the mass flow rate. This change affects the isentropic efficiency and aeroelastic log-dec, but the magnitude of the changes is relatively small compared to the values that are obtained with an optimized design (see Section 5.8.1).

5.7.2 DLR Transonic Compressor Stage

The DLR axial transonic compressor stage was first designed by Dunker et al. (1984) and it has been previously used as a multi-bladerow case for design optimization (Goinis et al., 2013; He and Wang, 2011; Siller et al., 2009; Wang and He, 2010a,b). The computational

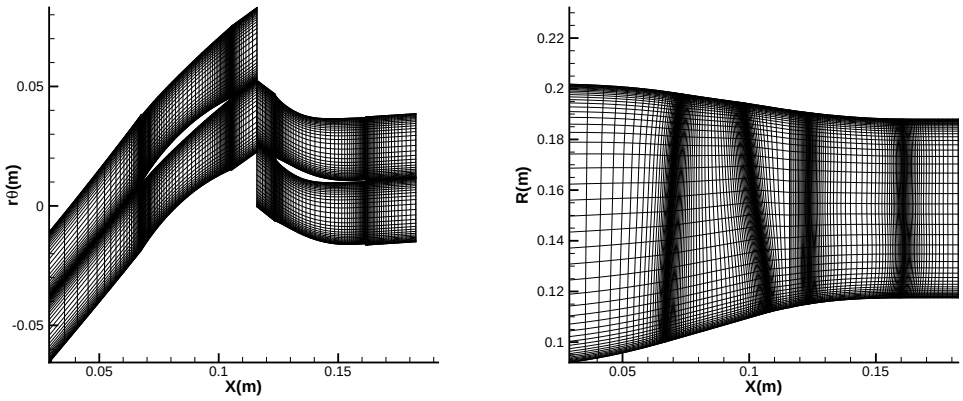


Figure 5.9: Computational domain of the DLR transonic compressor stage. Blade to blade view (left) and meridional view (right)

Case	Baseline	Static and total pressure		Total pressure	
	Values	Value	Rel. change (%)	Values	Rel. change (%)
Mass flow rate (kg/s)	16.95	16.98	+0.2	16.94	-0.06
Pressure ratio	1.594	1.594	0	1.597	+0.18
Isentropic efficiency (%)	86.79	86.79	0	86.77	-0.02
Log-dec (%)	2.84	2.84	0	2.77	-2.46

Table 5.2: Sensitivity analysis DLR stage. Values and relative change of obj. function parameters

domain (Figure 5.9) consists of a H-type mesh with a density of $115x51x40$ in the axial, pitchwise and radial directions respectively for the rotor domain and $90x32x35$ for the stator domain, with a mixing plane between both domains.

The results for the sensitivity analysis are shown in Table 5.2. The combined change in exit static and inlet total pressure changes the stage mass flow rate by 0.2% without any changes to isentropic efficiency and aeroelastic log-dec. A change in inlet total pressure slightly changes mass flow rate and increases pressure ratio by 0.18%. As with the NASA rotor 67 case, the changes in performance are very small compared to those presented next.

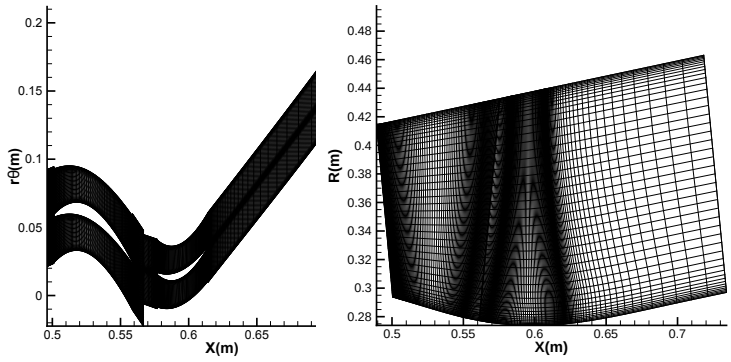


Figure 5.10: Computational domain of an industrial low-pressure turbine stage. Blade to blade view (left) and meridional view (right)

Case	Baseline	Static and total pressure		Total pressure	
	Values	Value	Rel. change (%)	Values	Rel. change (%)
Mass flow rate (kg/s)	39.91	39.99	+0.2	39.91	0
Pressure ratio	2.131	2.131	0	2.128	-0.13
Isentropic efficiency (%)	89.26	89.26	0	89.27	+0.01
Log-dec (%)	0.442	0.442	0	0.441	-0.23

Table 5.3: Sensitivity analysis LP turbine stage. Values and relative change of obj. function parameters

5.7.3 Industrial Low-pressure Turbine Stage

The last case is the last stage of a power turbine from a gas turbine for power generation. It has only been previously used for design optimization in Valero et al. (2014). The computational domain consists of a H-type mesh with a node density of $130x60x45$ in the stator domain and $140x45x45$ in the rotor domain. The relevant views of the mesh are shown in (Figure 5.10). The results for the sensitivity analysis are shown in Table 5.3. A change in static and total pressure increases the stage mass flow rate by 0.2% without any changes in performance while only a total pressure change causes slight changes to the performance parameters. Compared to the two previous compressor cases, this case is less sensitive to mass flow and pressure ratio changes. For this last case, the improvements obtained by the optimized designs can also be considered independent of mass flow rate and pressure ratio changes. .

Optimization case	NASA 67	DLR stage	LP turbine stage
Number of variables	4	6	6
Number of re-scales	3	3	3
Starting number of database samples	15	30	30
Number of candidates	15	30	30
Number of processors	15	30	30
CPU time per CFD run (per core)	0.5 h	2.5 h	3 h
Average CPU time per cycle	2 h	8.5 h	10 h

Table 5.4: Optimization settings and computational costs

5.8 Results

For all the three cases, an aeroelastic design optimization study is carried out based on Equation 5.6. First an aeroelastic-only optimization ($\sigma_3 = 0$) is performed. If the entropy generation rate is increased on the optimized design, an additional concurrent aeroelastic optimization ($\sigma_3 > 0$) is carried out so the difference between both cases can be illustrated. Table 5.4 shows some information of the optimization settings and computational costs of each of the three cases.

5.8.1 NASA Rotor 67 Aeroelastic-only Optimization

The design optimization is performed with 4 variables and 15 processors in parallel. Figure 5.11 shows the reduced coordinates of the 4 perturbation nodes corresponding to the 4 variables, which divide the blade surface into equally spaced intervals. Convergence is reached after 5 design cycles with 3 re-scales. It is useful to recall that a design optimization cycle may involve more than one direct evaluation of candidates. In fact, the average number of subsequent candidates evaluations (iterations) per cycle is 2. The convergence of the objective function and its three constraints is shown in Figure 5.12 and Table 5.5. The objective function converges smoothly and it is reduced by 20% at the end of the optimization, with an absolute increase of 0.85% of the aeroelastic log-dec. However, the optimum design also decreases the entropy generation rate, which leads to an isentropic efficiency absolute increase of 0.44%. The mass flow rate and pressure ratio

	Original	Optimized	Abs. change	Rel. change
Isentropic efficiency (%)	91.72	92.16	+0.44	+0.48%
Log-dec (%)	3.12	3.97	+0.85	+27.4%
Mass flow rate (kg/s)	33.37	33.41	+0.04	+0.12%
Pressure ratio	1.656	1.657	+0.001	+0.03%

Table 5.5: Final values, absolute and relative differences. NASA rotor 67 redesign

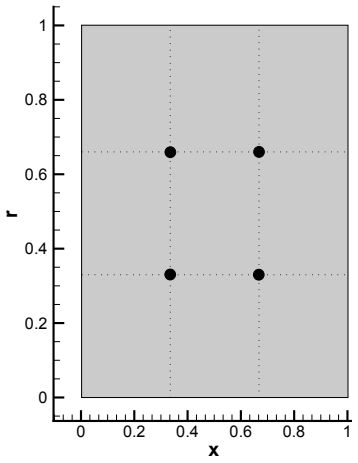


Figure 5.11: Hicks-Henne function perturbation nodes (reduced coordinates). NASA rotor 67

constraints keep the difference between the original and optimized values small.

The flow fields of the optimized and original design are compared. Figures 5.13 and 5.14 show the pressure contours on the blade surface of the NASA rotor 67 for both designs. There is a strong passage shock normal to the casing at the pressure side for the original flow field. This shock is weakened and moved downstream in the optimized blade. This is the main factor that leads to the efficiency gain.

In terms of aeroelasticity, Figures 5.15 and 5.16 shows the local work contours for the first mode in both sides of the blade surface for both original and optimized designs. As introduced in section 5.5, the stable areas are those with negative work (blue contours) while the unstable areas are those with positive values of work (yellow and red contours). The weakening of the shock wave reduces the small unstable areas around the pressure

side while increasing the strength of the stable areas around the pressure side of the blade surface.

The changes in the blade geometry are shown in Figure 5.17. The biggest changes can be found around the midspan region. Overall, there is a negative humping effect before midchord and an upward humping effect after midchord. Despite only 4 variables being used in this case, it can be seen that the global influence of the hump functions creates a global change in the rotor blade surface.

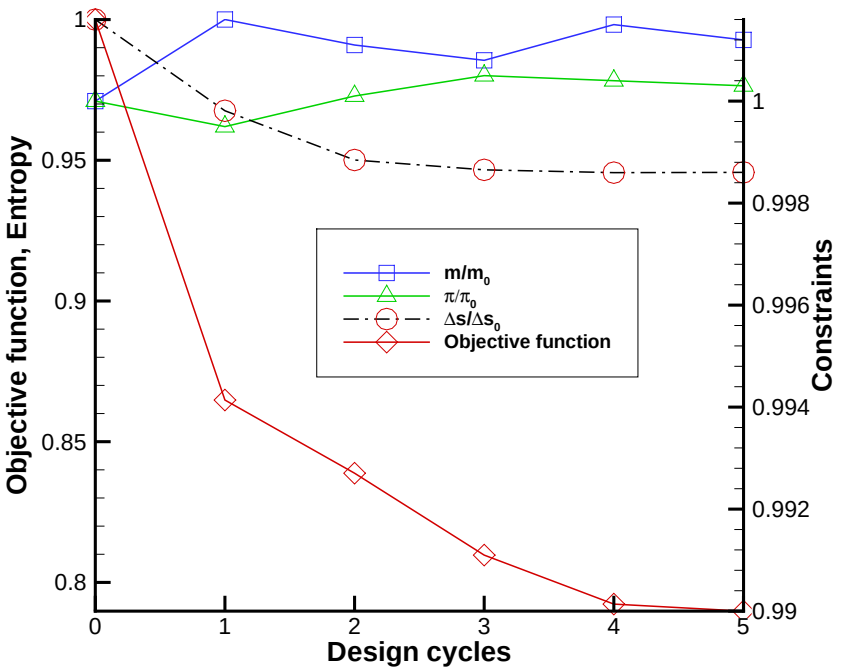


Figure 5.12: Convergence of objective function and constraints. NASA rotor 67

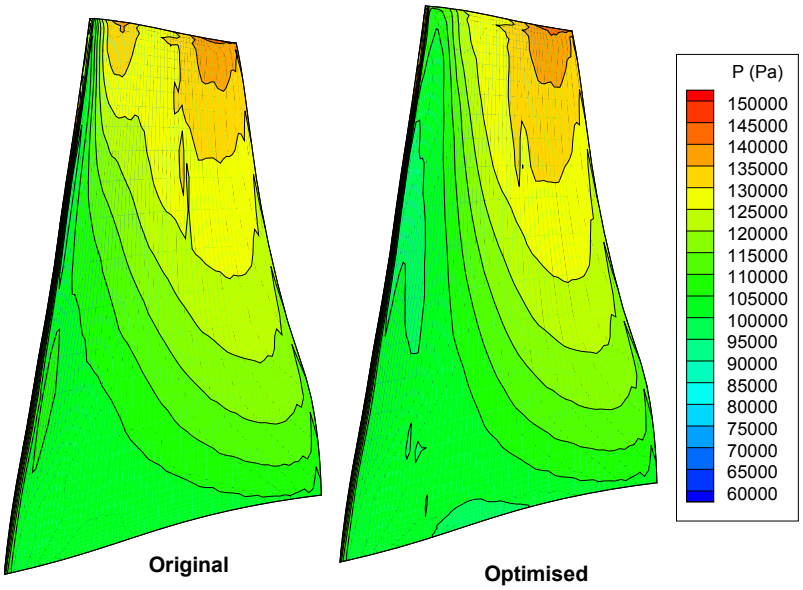


Figure 5.13: NASA rotor 67 pressure distribution contours. Pressure side.

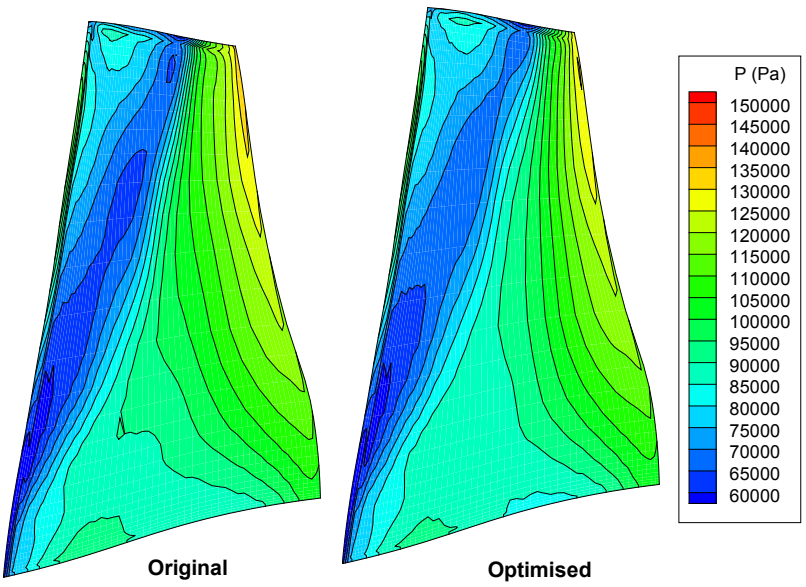


Figure 5.14: NASA rotor 67 pressure distribution contours. Suction side

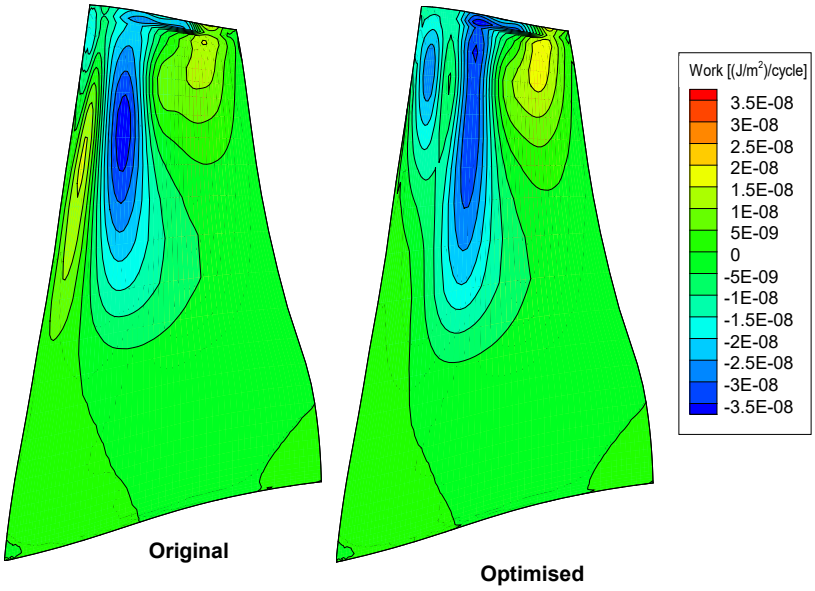


Figure 5.15: Local work contours on NASA rotor 67 (first bending mode). Pressure side.

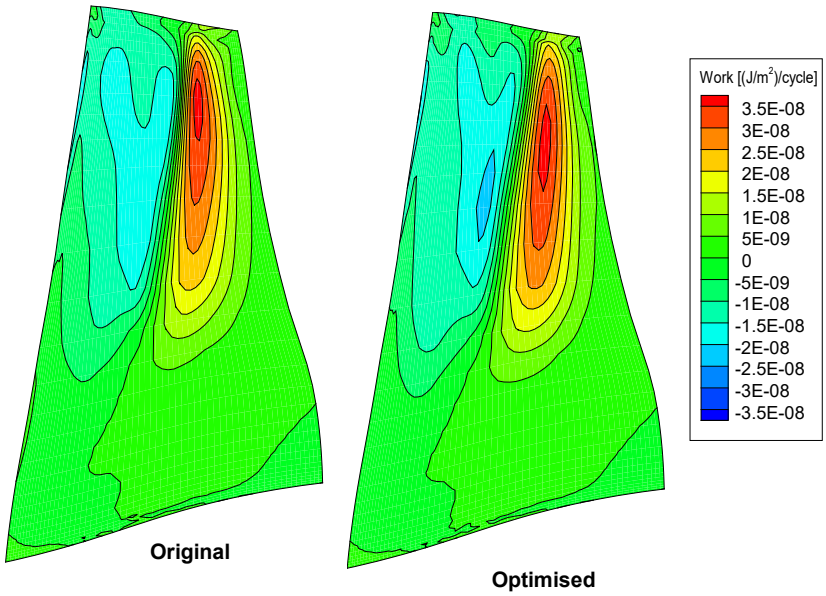
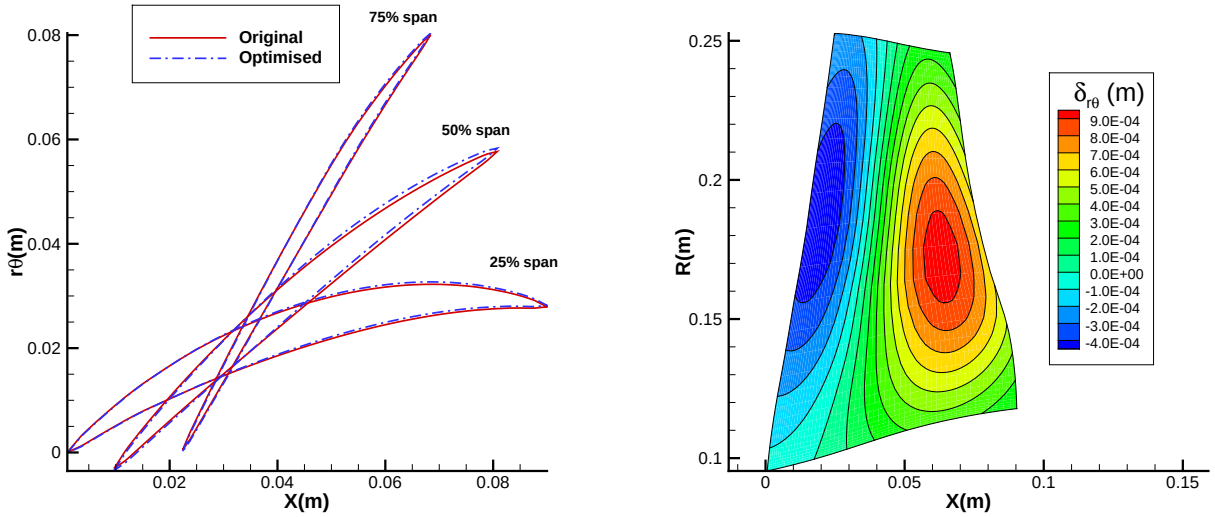


Figure 5.16: Local work contours on NASA rotor 67 (first bending mode). Suction side



(a) Rotor blade profiles at 25%, 50% and 75% span locations

(b) Contours of tangential coordinate change, $\delta_{r\theta}$

Figure 5.17: Blade geometry comparison. NASA rotor 67 optimization

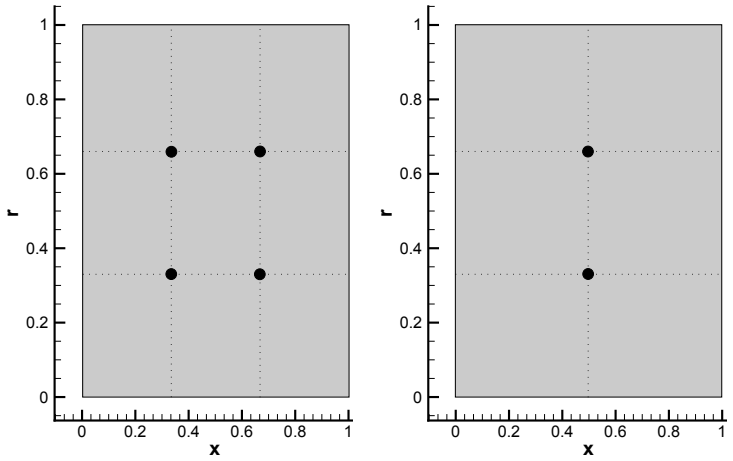
5.8.2 DLR Transonic Stage Aeroelastic-only Optimization

The optimization of the transonic compressor stage is carried out with 6 design variables, which are distributed between rotor and stator blades as shown in Figure 5.18. It takes nearly 2 days to complete 5 design cycles with an average cost of 9 hours per cycle. The convergence of the objective function and its constraints is shown in Figure 5.19. The objective function drops its value by 32% and the final values are shown in Table 5.6. The aeroelastic log-dec is increased an extra 0.75%. As with the NASA optimization case, a concurrent design optimization is not needed. The improvement of aeroelastic log-dec has also reduced the entropy generation rate of the stage, with an absolute efficiency increase of 0.32%. The mass flow rate and pressure ratio difference is small.

The local Mach contours at 50% and 75% span (Figures 5.20 and 5.21) show a weakened shock that is also pushed downstream. Figure 5.22 shows the pressure coefficient distributions at different span locations for the original and optimized cases. The weakening and displacement of the shock region improves the diffusion of the optimized design and

	Original	Optimized	Abs. change	Rel. change
Isentropic efficiency (%)	86.79	87.11	+0.32	+0.37%
Log-dec (%)	2.81	3.56	+0.75	+26.69%
Mass flow rate (kg/s)	16.95	16.954	+0.004	+0.02%
Pressure ratio	1.594	1.595	+0.001	+0.06%

Table 5.6: Final values, absolute and relative differences. DLR stage redesign



(a) Rotor blade. 4 variables (b) Stator blade. 2 variables

Figure 5.18: Hicks-Henne function perturbation nodes (reduced coordinates). DLR stage

increase its blade loading.

The local work distributions for the first mode are shown in Figures 5.23 and 5.24. Both suction and pressure surfaces reveal that the most stable areas are located upstream of the shock while the unstable areas are located downstream. The displacement of the shock increases the size of the stable area on the pressure side and, significantly, on the suction side. The unstable areas are also increased, but by a lesser extent.

The stage blade geometries of both the original and optimized designs at three span locations and the overall spanwise hump perturbation are shown in Figures 5.25 and 5.26. The rotor is mostly humped upwards from midchord in the axial direction and from midspan to the hub in the radial direction. The stator is changed differently, with a negative humping centered around midspan and midchord.

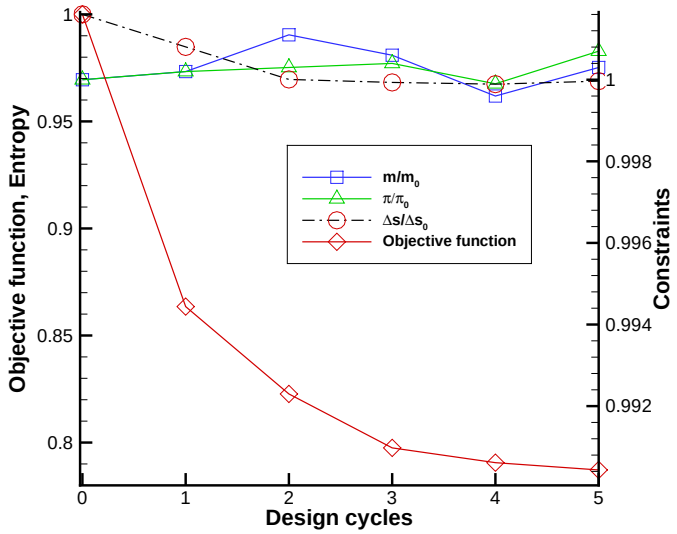


Figure 5.19: Convergence of objective function and constraints. DLR stage

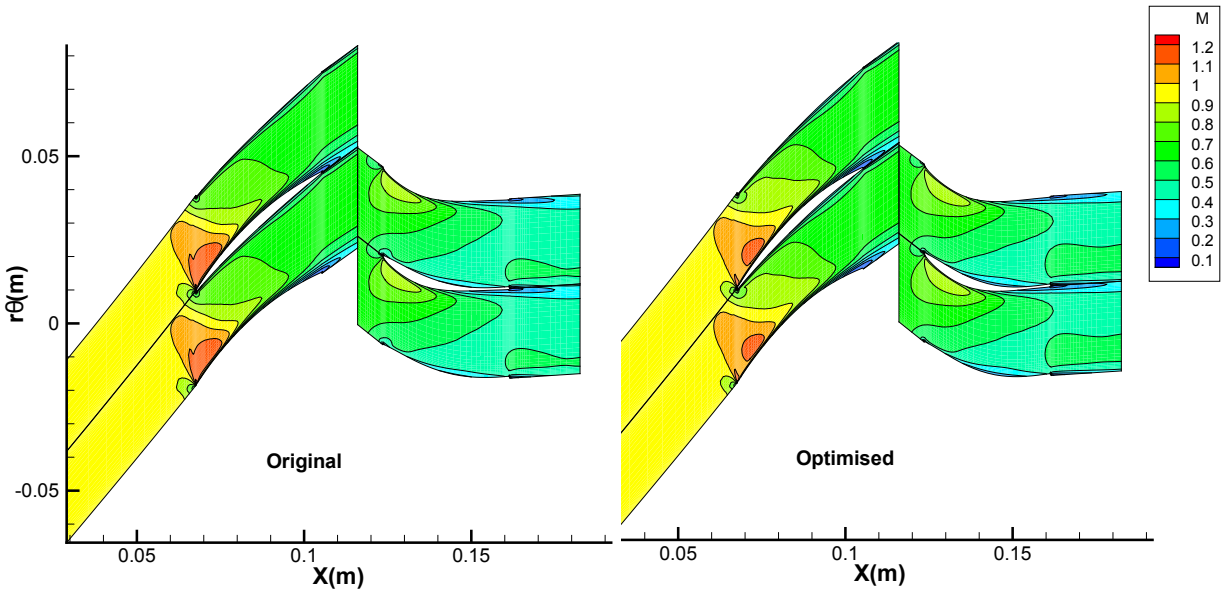


Figure 5.20: Local Mach number contours at 50% span. DLR stage

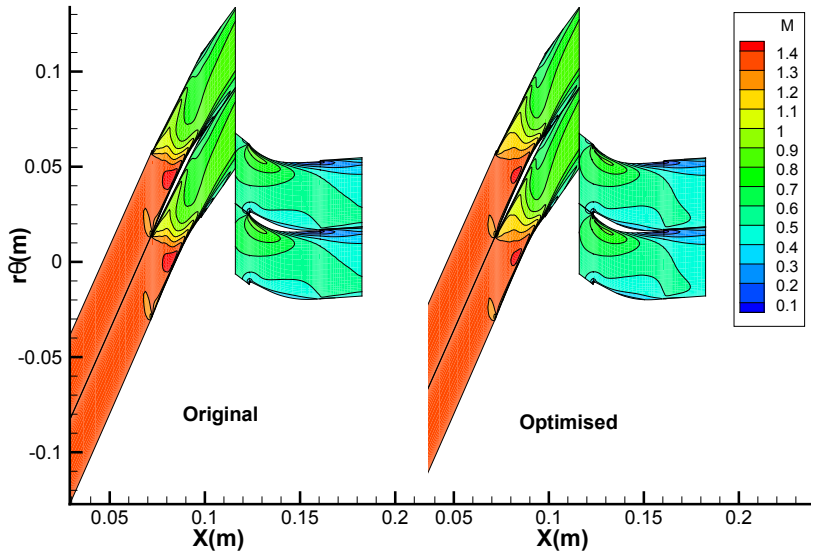
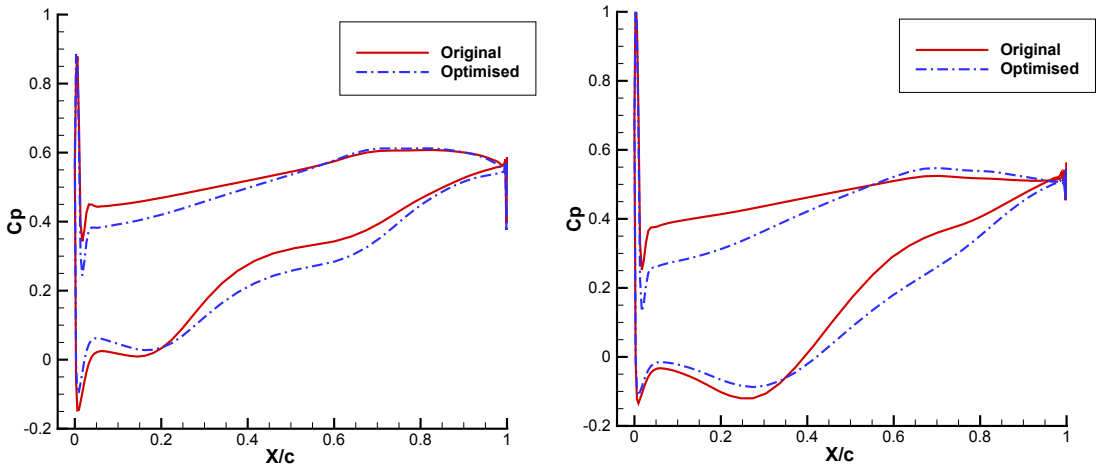
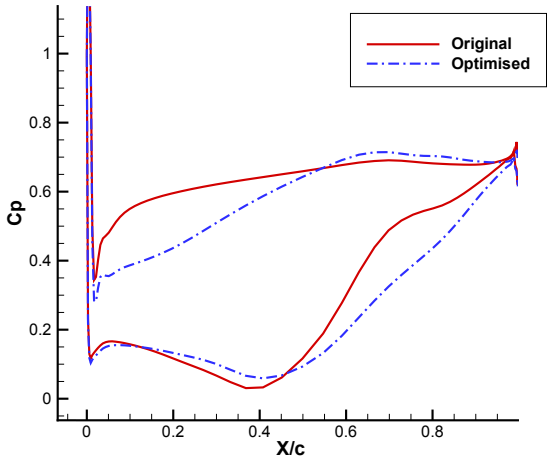


Figure 5.21: Local Mach number contours at 75% span. DLR stage



(a) 25% span

(b) 50% span



(c) 75% span

Figure 5.22: Pressure coefficient distributions of DLR rotor blade

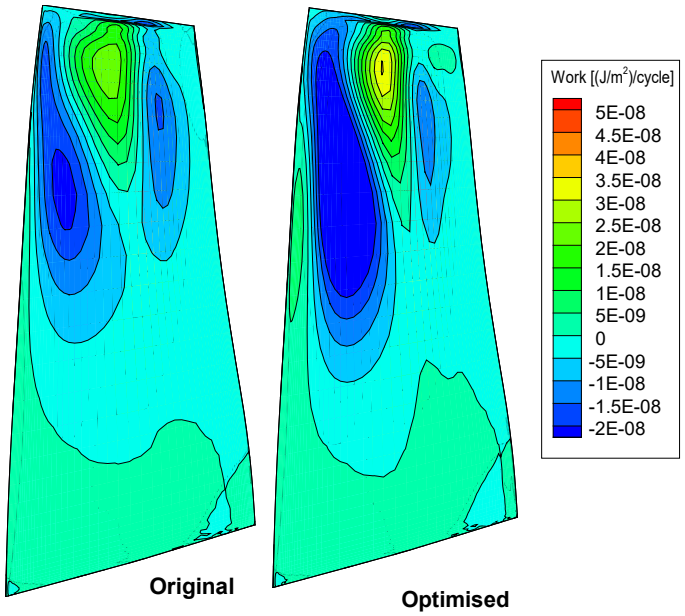


Figure 5.23: Local work contours on DLR blade surface (first bending mode). Pressure side

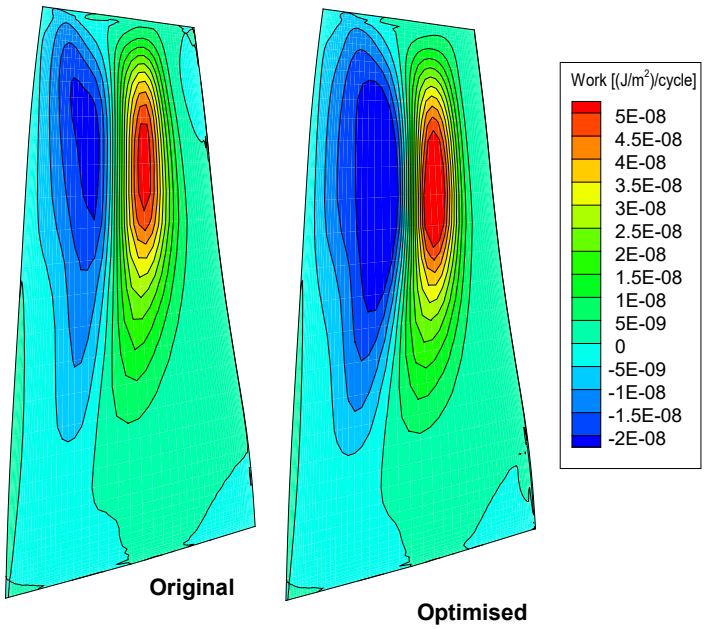


Figure 5.24: Local work contours on DLR blade surface (first bending mode). Suction side

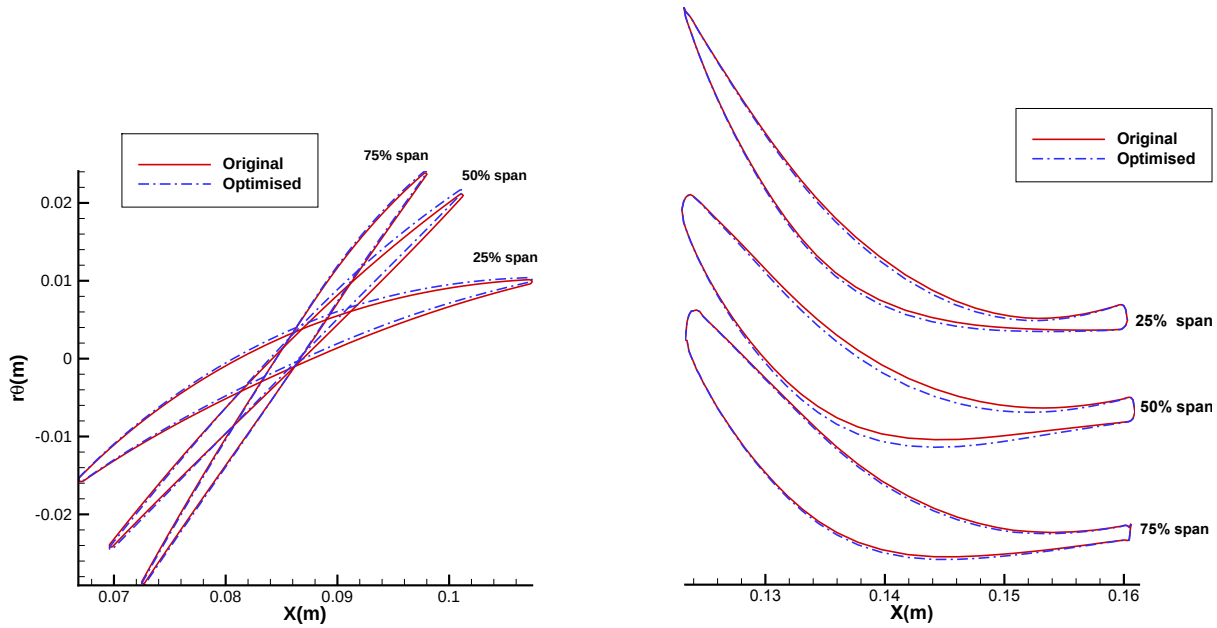


Figure 5.25: DLR stage profiles at 25%, 50% and 75% span locations

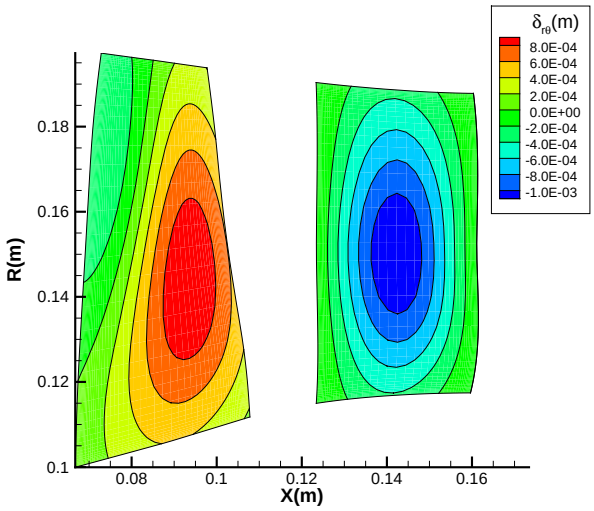
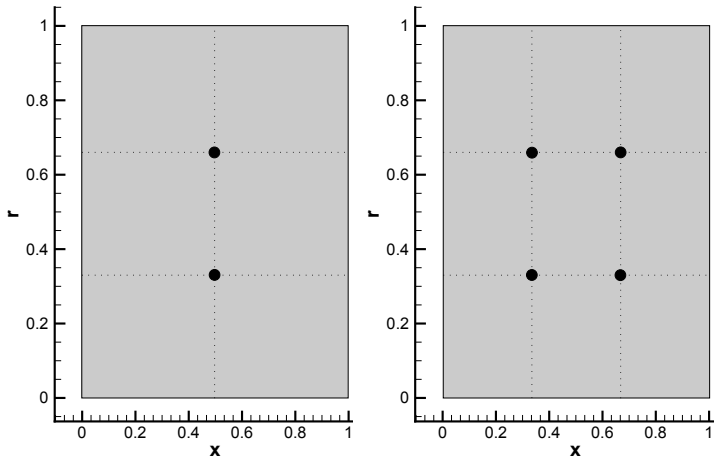


Figure 5.26: Contours of tangential coordinate change, $\delta_{r\theta}$. DLR stage

5.8.3 Industrial LP Stage Aeroelastic Optimization

Two design optimization cases are carried out next for the industrial LP turbine stage. The first case is an aeroelastic-only design optimization while the second case is a con-



(a) Stator blade. 2 variables (b) Rotor blade. 4 variables

Figure 5.27: Hicks-Henne function perturbation nodes (reduced coordinates). LP turbine stage

current aeroelastic design optimization. As with the DLR stage case, 6 design variables are used in total and the location of the perturbation nodes are shown in Figure 5.27. This case is the most expensive of all three and the design process has a total cost of 2.5 days through 5 design cycles.

5.8.3.1 Aeroelastic-only Stage Design Optimization

The convergence of the aeroelastic-only design optimization is shown in Figure 5.28a. The objective function drops its value by 20% as shown in Table 5.7 where the absolute and relative differences between the original and optimized performance parameter values are given. The aeroelastic log-dec is relatively increased by 26% after 5 design cycles, leading to an absolute 1.05% increase. The isentropic efficiency is significantly decreased, as opposed to the previous two cases where the aeroelastic optimization was also coincidently showing an improvement in aerodynamics. Regarding the two constraints, the mass flow rate difference is small. However, the pressure ratio change is more significant but the constraint keeps its relative change below 0.2%**.**

	Abs. change	Rel. change
Isentropic efficiency (%)	-0.28	-0.31%
Log-dec (%)	+1.05	+26.79%
Mass flow rate (kg/s)	+0.012	+0.03%
Pressure ratio	-0.004	-0.17%

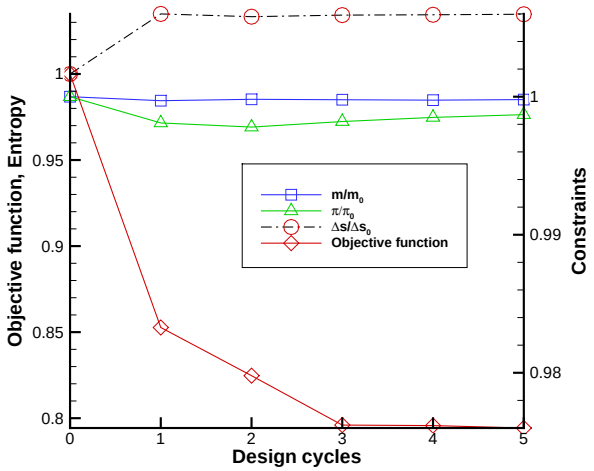
Table 5.7: Absolute and relative differences after convergence. LP turbine stage aeroelastic-only redesign

5.8.3.2 Concurrent Aeroelastic Stage Design Optimization

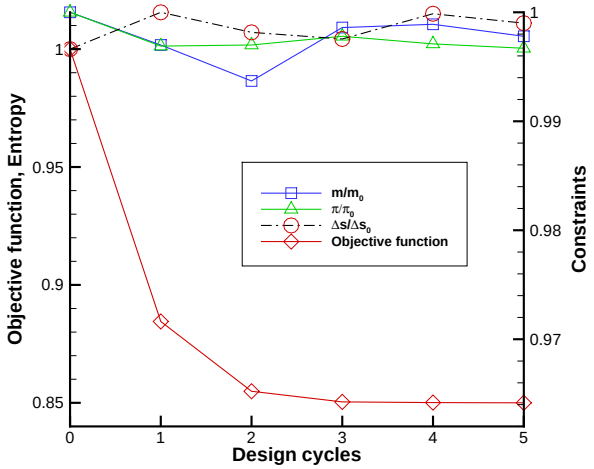
A concurrent aeroelastic design optimization is also performed for the low-pressure turbine stage in order to show the importance of concurrency. The objective function now includes the entropy generation rate as a constraint ($\sigma_3 > 0$). The final absolute and relative differences between original and optimized cases are given in Table 5.8 and its convergence is shown in Figure 5.28b. The objective function drops its value by 15% and the aeroelastic log-dec shows an absolute change of 0.76% after 5 design cycles. The margin of improvement of the aeroelastic log-dec is now smaller than the single-side aeroelastic design. However, the entropy generation rate constraint is now controlling that the efficiency is not changed excessively, with an absolute decrease of only 0.04%. This is where the benefits of a concurrent objective function can be observed. The mass flow rate and pressure ratio changes are not negligible but the constraints are keeping their relative differences below 0.2%.

	Abs. change	Rel. change
Isentropic efficiency (%)	-0.04	-0.04%
Log-dec (%)	+0.76	+19.4%
Mass flow rate (kg/s)	-0.072	-0.18%
Pressure ratio	-0.004	-0.18%

Table 5.8: Absolute and relative differences after convergence. LP turbine stage concurrent aeroelastic redesign



(a) Aeroelastic-only case



(b) Concurrent aeroelastic case

Figure 5.28: Convergence of objective function and constraints. LP turbine stage

5.8.3.3 Multi-disciplinary Optimization Comparison

The last two design optimization cases for the industrial LP turbine stage are compared. The local Mach number contours at 50% span are shown in Figure 5.29. The aeroelastic-only optimized design increases the peak speed at the stator passage, whereas the concurrent design shows almost no difference. This affects the incidence loading of the rotor

blade as it can be seen from Figure 5.30 where pressure coefficient distributions at different span locations of the rotor blade are shown. The concurrent design has a similar loading as the baseline design. However, the aeroelastic-only design shows an increased incidence loading and a slightly increased rear loading.

The local work contours are shown in Figures 5.31 and 5.32. The pressure side is completely stable and an increase of the stabilizing gradient can be observed for both the aeroelastic-only and concurrent optimized blades. The strength of the unstable area upstream of the supersonic region is decreased on both cases, while the downstream stable gradient is also slightly increased. Despite there are no big changes on both cases, the small improvements lead to an overall increase of the aeroelastic log-dec, which is more significant for the aeroelastic-only case.

The geometry perturbations on both blades and their blade profiles at three locations are compared in Figures 5.33 and 5.34. For the aeroelastic-only design, the stator is mostly unchanged, while the rotor shows an important change in the camber line of the blade centered at midspan and completely humped upwards. The concurrent design is quite different: the stator shows an upwards humping, with the biggest perturbation being located at midspan while the rotor is mostly humped down with the biggest perturbations being located from midspan to the hub of the blade.

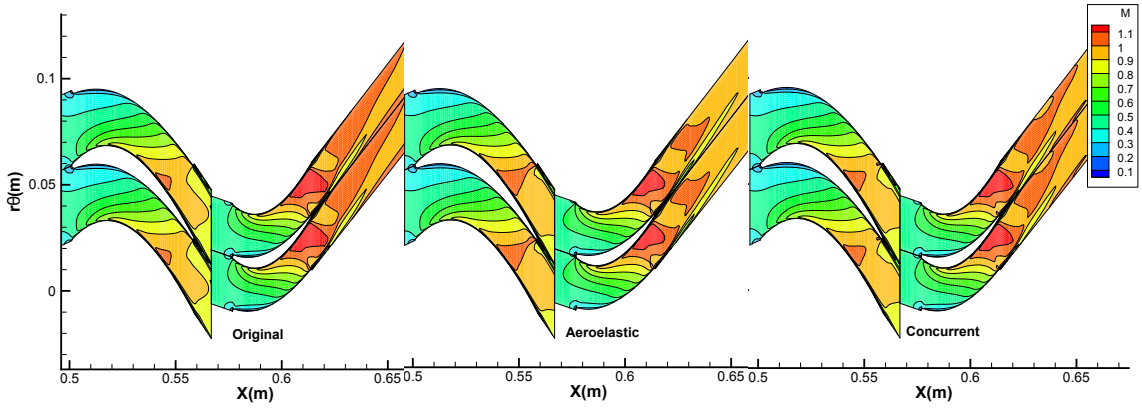
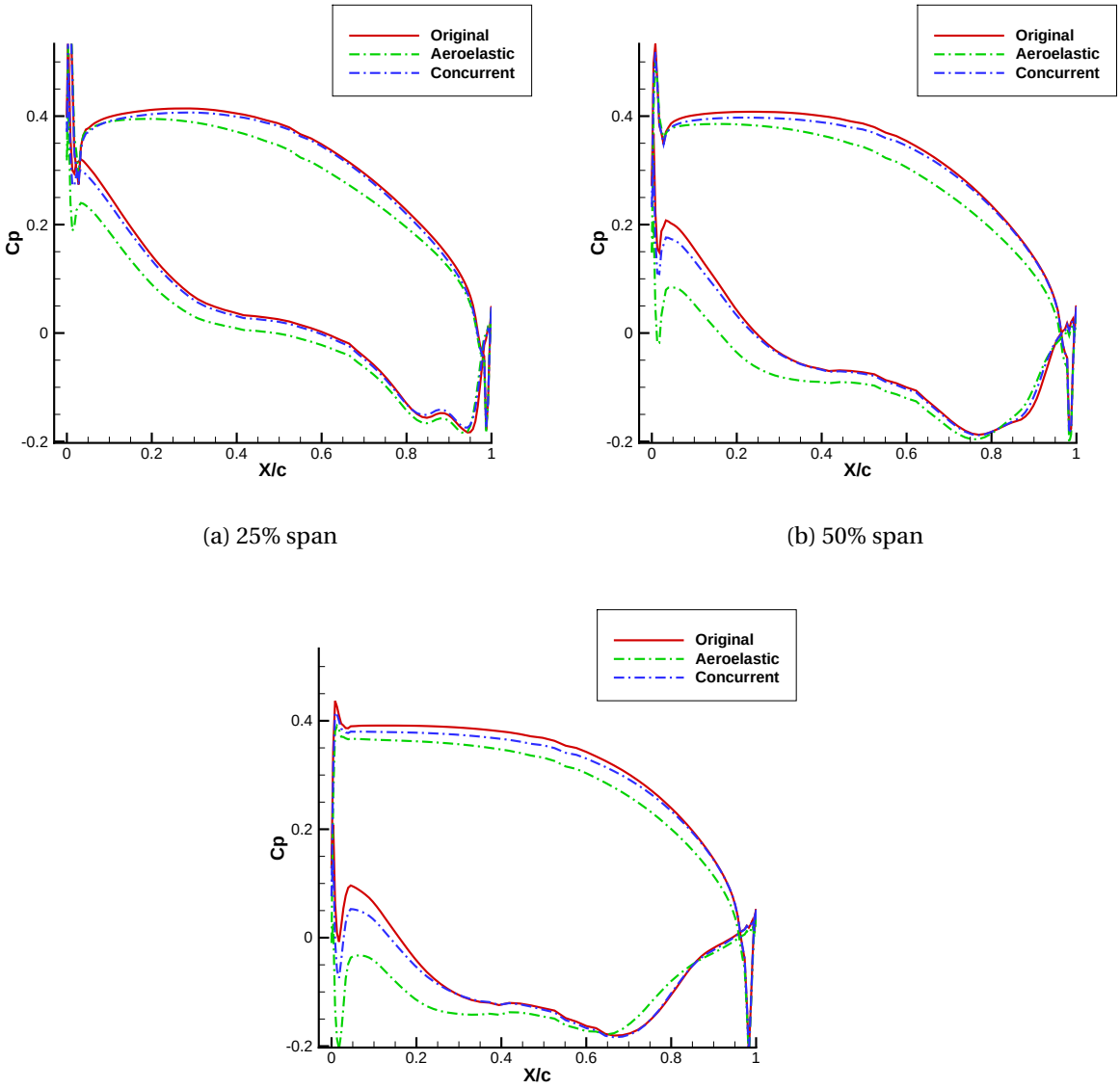


Figure 5.29: Local Mach number contours at 50% span. LP turbine stage



(a) 25% span

(b) 50% span

(c) 75% span

Figure 5.30: Pressure coefficient distributions of rotor blade. LP turbine aeroelastic redesign

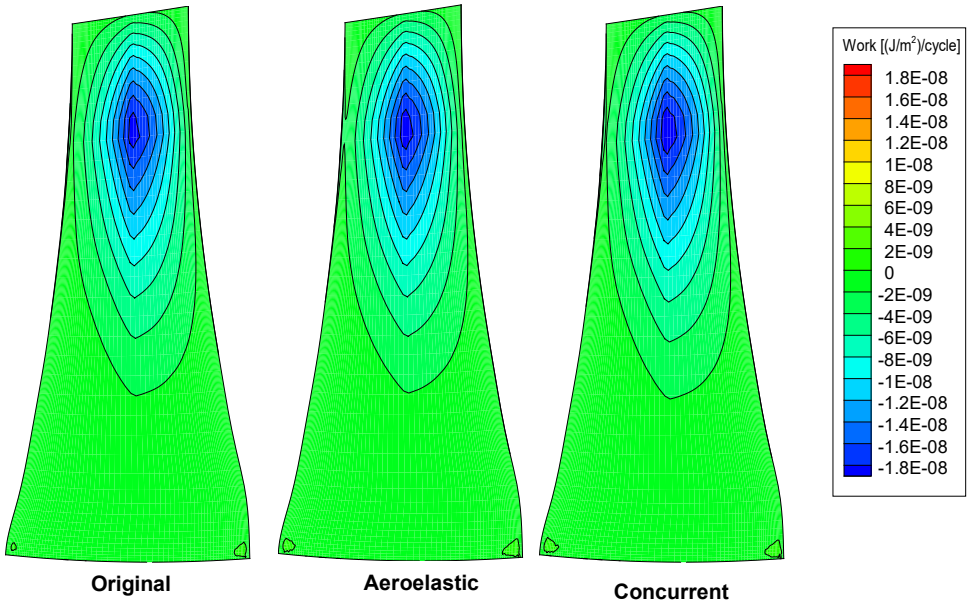


Figure 5.31: Local work contours on LP turbine blade (first bending mode). Pressure side

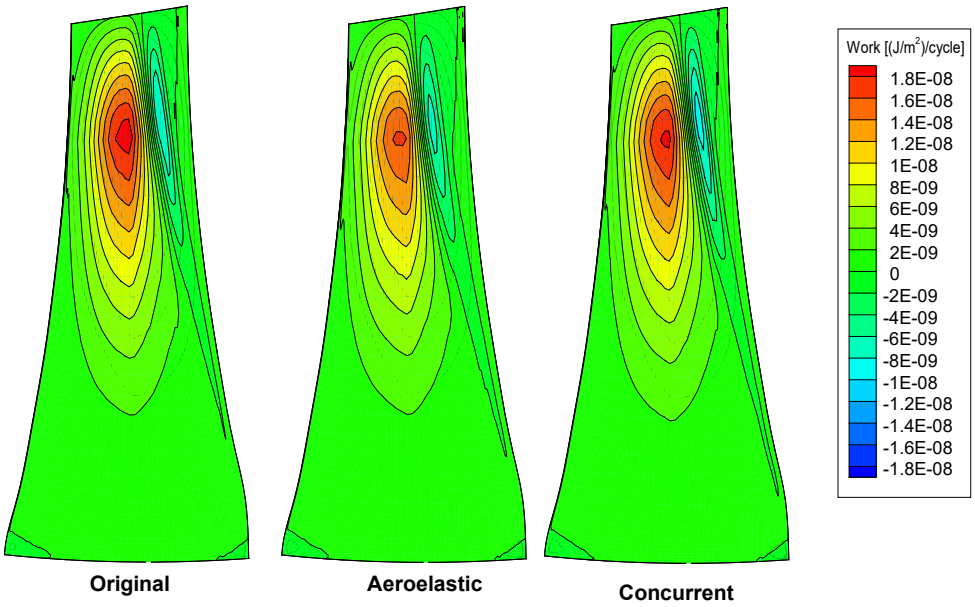


Figure 5.32: Local work contours on LP turbine blade (first bending mode). Suction side

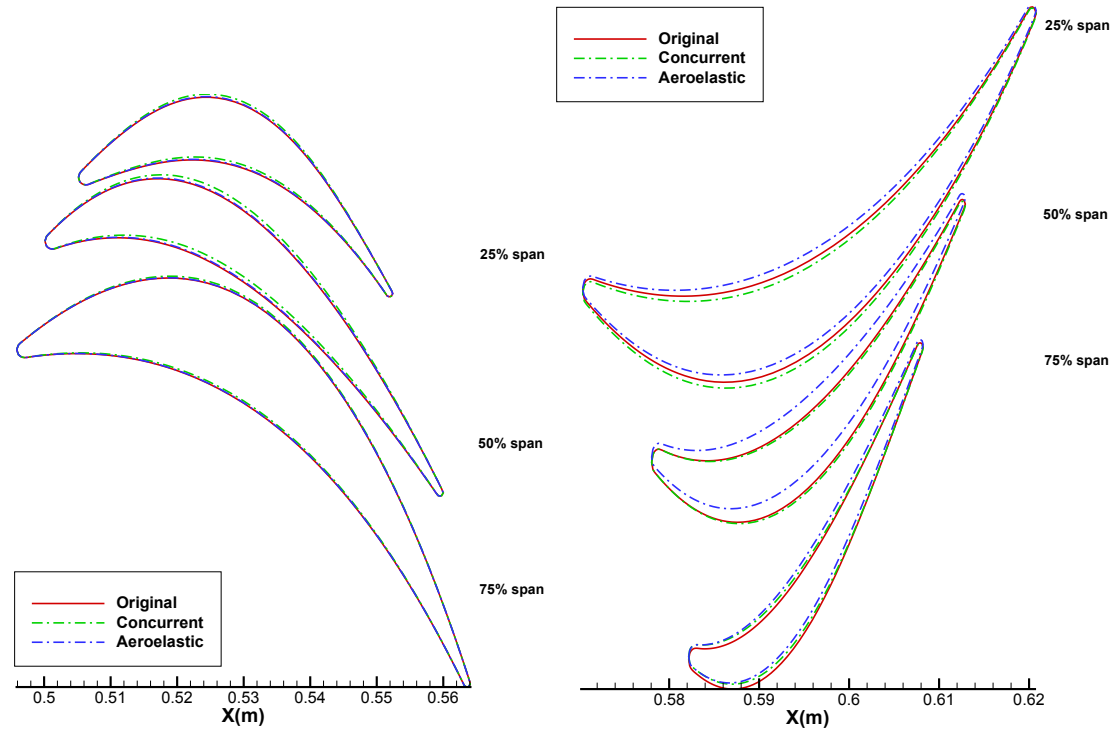
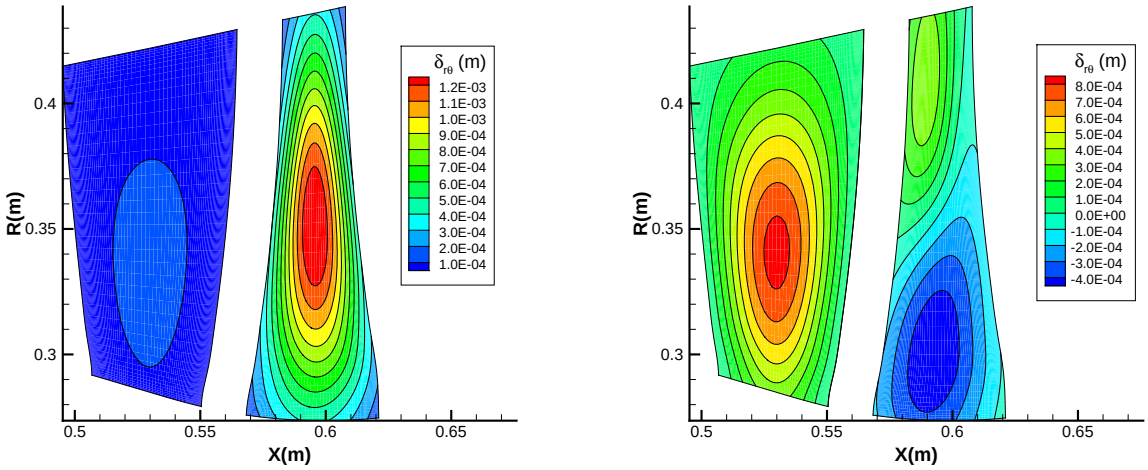


Figure 5.33: Stage profiles at 25%, 50% and 75% span. LP turbine aeroelastic redesign



(a) Aeroelastic-only case

(b) Concurrent aeroelastic case

Figure 5.34: Contours of tangential coordinate change, $\delta_{r\theta}$. LP turbine aeroelastic redesign

5.9 Summary

The main elements of the two design optimization systems that are implemented in this work have been presented first. Next, the non-gradient approach based on the RRSM surrogate model have been introduced with some results of its application to three different design cases. The combination of the RRSM and DE algorithms has demonstrated that convergence can be achieved for realistic multi-bladerow cases, even with databases that start with a relatively small number of samples. The shape parameterization method has also shown that a smooth global perturbation on the blade surfaces and good margins of improvements can be achieved with a small number of variables. These improvements have been shown to be considerably greater than those shown in the sensitivity analysis. One drawback of this method, however, is that its computational limitation makes it not ideal for complicated geometries, as it is often the case for real designs.

The NASA rotor 67 and DLR stage design optimizations have illustrated that an improvement in blade damping can also lead to a coincident improvement in aerodynamic performance. This is not the case for the industrial LP turbine, where a compromise has to be made between the two disciplines. A concurrent objective function plays an important role here and allows optimizing the damping of the rotor while ensuring that aerodynamic performance is not decreased.

Chapter 6

The Adjoint Method

The purpose of this chapter is to show the main features of the adjoint method and to introduce the adjoint method principle before using it for the derivation of the system of unsteady adjoint equations next chapter. The formulation principle of the adjoint is explained first, and secondly, a steady adjoint system based on the Euler flow equations is derived following the continuous approach.

6.1 Adjoint Method Principle

The objective of a gradient-based optimization method is to evaluate the gradient of the objective function with respect to the design variable in order to determine the best search direction to find the optimum. This function, I, is normally a scalar that is a function of a set of design variables and flow variables as follows:

$$I = I(\alpha, U) \tag{6.1}$$

where $\alpha = (\alpha_1, \alpha_2 ..., \alpha_n)$ and U denote the design variable and flow variable vectors respectively. The relation between the design and flow variable vectors is determined by the nonlinear flow equations, which can be represented in a residual operator as shown

below:

$$R[\alpha, U(\alpha)] = 0 \tag{6.2}$$

The equality above is true for any design variable. This means that the total derivative of the residual operator with respect to any design variable is equal to zero, leading to the linearised flow equations:

$$\frac{dR}{d\alpha} = \frac{\partial R}{\partial \alpha} + \frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} = 0 \tag{6.3}$$

Using the chain rule, the total derivative of the objective function I with respect to the design variable α can be written as:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} + \frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha} \tag{6.4}$$

I is an explicit function of α and U; therefore $\frac{\partial I}{\partial \alpha}$ and $\frac{\partial I}{\partial U}$ can be obtained analytically using the complex variable method by Vatsa (2004). The only term that is left in Equation 6.4 is $\frac{\partial U}{\partial \alpha}$. This gradient is called the flow variable sensitivity and can be approximated using the finite difference method:

$$\frac{\partial U}{\partial \alpha} = \frac{U(\alpha + \triangle\alpha) - U(\alpha)}{\triangle\alpha} \tag{6.5}$$

For a large number of design variables, a direct evaluation of the derivative $\frac{\partial U}{\partial \alpha}$ becomes very expensive, since it requires a solution of the flow equations for each design variable. The main feature of the adjoint gradient-based method is that it eliminates the flow variable sensitivity from the gradient equation and, consequently, the cost of calculating the gradient expression above (Equation 6.4) is no longer proportional to the number of variables. The demonstration starts by multiplying the nonlinear flow equations with a Lagrange multiplier variable λ called the adjoint variable:

$$\lambda^T \Big[R(\alpha, U) \Big] = 0 \tag{6.6}$$

The product of this is always zero, no matter the value of the adjoint variable. This still holds if the nonlinear flow equations are linearised:

$$\lambda^T \left(\frac{\partial R}{\partial \alpha} + \frac{\partial R}{\partial U} \frac{\partial U}{\partial \alpha} \right) = 0 \tag{6.7}$$

Subtracting this product from the gradient expression (6.4) yields:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial U} \frac{\partial U}{\partial \alpha} + \frac{\partial I}{\partial \alpha} - \lambda^T \left(\frac{\partial R}{\partial \alpha} + \frac{\partial R}{\partial U} \frac{\partial U}{\partial \alpha} \right) \tag{6.8}$$

According to Equation 6.3, the term $\frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} + \frac{\partial R}{\partial \alpha}$ is always zero and so its product, independently of the value of the adjoint variable λ. Equation 6.8 is regrouped by collecting the terms around $\frac{\partial U}{\partial \alpha}$ as follows:

$$\frac{dI}{d\alpha} = \left(\frac{\partial I}{\partial U} - \lambda^T \frac{\partial R}{\partial U} \right) \frac{\partial U}{\partial \alpha} + \frac{\partial I}{\partial \alpha} - \lambda^T \frac{\partial R}{\partial \alpha} \tag{6.9}$$

In order to cancel the flow variable sensitivity $\frac{\partial U}{\partial \alpha}$, the adjoint variable has to be chosen in a way that the following equation is satisfied:

$$\frac{\partial I}{\partial U} - \lambda^T \frac{\partial R}{\partial U} = 0 \tag{6.10}$$

This equation is called the adjoint equation. If the adjoint variable satisfies the equation, the gradient equation (6.9) is reduced to one independent of the flow variable sensitivity:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} - \lambda^T \frac{\partial R}{\partial \alpha} \tag{6.11}$$

It is important to note that, similar to the linearised flow equation (6.3), the adjoint equation (6.11) is also a linear system based on a given flow field, U. The important feature of this equation is that it is free from any design variable, α_i. Hence, only one adjoint equation of the same size (and similar computational cost) as the flow equation system is enough for all the design variables. The new gradient expression (6.11) is still linked to the design variable vector α but the computational cost of calculating $\frac{\partial R}{\partial \alpha}$ for a large

number of design variables is considerably smaller than the computational time of one flow solution.

6.2 Steady Adjoint Method: Euler Flow Equations

The derivation of the adjoint equation based on the Euler flow equations is included below to show the application of the adjoint method principle to derive a continuous adjoint system. The inviscid Euler flow equations are written down in cylindrical coordinates as follows:

$$\frac{\partial F}{\partial x}+\frac{\partial\left(G-Uv_g\right)}{r\partial\theta}+\frac{\partial rH}{r\partial r}-S=0 \tag{6.12}$$

The objective functions implemented in this work are a weighted sum of the mass flow rate and pressure ratio as aero-thermal constraints and the entropy generation rate and aeroelastic work sum as performance parameters. The objective function, I, defined in Equation 6.1 can be expressed as:

$$I=\int_{\partial V}M(\alpha,U)\cdot dS \tag{6.13}$$

where M is a function of the flow variable vector U and the design variable vector α. Based on the chain rule, the gradient of the objective function to a design variable is given by:

$$\frac{dI}{d\alpha}=\int_{\partial V}\left(\frac{\partial M}{\partial\alpha}+\frac{\partial M}{\partial U}\tilde{U}\right)\cdot dS+\int_{\partial V}M\cdot\tilde{dS} \tag{6.14}$$

where $\tilde{U}=\frac{\partial U}{\partial\alpha}$ is the flow variable sensitivity and $\tilde{dS}=\frac{d(dS)}{d\alpha}$ is the boundary geometry perturbation due to a change in any design variable. $\frac{\partial M}{\partial U}$ and $\frac{\partial M}{\partial\alpha}$ can be obtained analytically without solving any equations. The linearised Euler equations are shown below:

$$\frac{\partial\left(A\tilde{U}\right)}{\partial x}+\frac{\partial\left(B\tilde{U}-v_g\tilde{U}\right)}{r\partial\theta}+\frac{\partial r\left(C\tilde{U}\right)}{r\partial r}-D\tilde{U}=f \tag{6.15}$$

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 corresponds to $-\frac{\partial R}{\partial \alpha}$ in Equation 6.3, also referred as the flow residual perturbation, and it represents the linearisation of the geometry terms in the flow governing equations with respect to the design variable vector α. As previously stated, the calculation of the flow variable sensitivity $\tilde{U}$ involves solving the flow equations once for each design variable. Multiplying two sides of Equation 6.15 by the the adjoint variable λ and rearranging leads to:

$$\lambda^T \left[\frac{\partial \left(A\tilde{U}\right)}{\partial x} + \frac{\partial \left(B\tilde{U} - v_g\tilde{U}\right)}{r\partial\theta} + \frac{\partial r\left(C\tilde{U}\right)}{r\partial r} - D\tilde{U} - f \right] = 0 \tag{6.16}$$

where $\lambda^T = (\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5)^T$. These five λ_i scalar variables are linked to the continuity equation, the axial momentum equation, the moment of momentum equation, the radial momentum equation and the energy equation respectively. In order to remove the adjoint equation and calculate the adjoint variable, Equation 6.16 is integrated over the whole computational domain as follows:

$$\int_V \lambda^T \left[\frac{\partial \left(A\tilde{U}\right)}{\partial x} + \frac{\partial \left(B\tilde{U} - v_g\tilde{U}\right)}{r\partial\theta} + \frac{\partial r\left(C\tilde{U}\right)}{r\partial r} - D\tilde{U} - f \right] dv = 0 \tag{6.17}$$

Integration by parts is performed to give:

$$\begin{aligned} \int_{\partial V} \lambda^T \left[A\cdot n_x + \left(B - v_g I\right)\cdot n_\theta + C\cdot n_r \right] \tilde{U} dS \\ - \int_V \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial\theta}\left(B - v_g I\right) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv - \int_V \lambda^T f dv = 0 \end{aligned} \tag{6.18}$$

where I is a 5×5 identity matrix. The left hand side of Equation 6.18 is equal to zero and it is added to the gradient equation 6.14 as follows:

$$\begin{aligned} \frac{dI}{d\alpha} = \int_{\partial V} \left(\frac{\partial M}{\partial U}\tilde{U} + \frac{\partial M}{\partial \alpha} \right) dS + \int_S M\tilde{dS} - \int_{\partial V} \lambda^T \left[A\cdot n_x + \left(B - v_g I\right)\cdot n_\theta + C\cdot n_r \right] \tilde{U} dS \\ + \int_V \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial\theta}\left(B - v_g I\right) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv + \int_V \lambda^T f dv \end{aligned} \tag{6.19}$$

The expression above can be rearranged around the flow variable sensitivity $\tilde{U}$:

$$\frac{dI}{d\alpha} = \int_V \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial \theta} \left(B - v_g I \right) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv$$
$$- \int_{\partial V} \left[\lambda^T A \cdot n_x + \lambda^T \left(B - v_g I \right) \cdot n_\theta + \lambda^T C \cdot n_r - \frac{\partial M}{\partial U} \right] \tilde{U} dS + \int_{\partial V} \frac{\partial M}{\partial \alpha} dS + \int_{\partial V} M \tilde{dS} + \int_V \lambda^T f dv \tag{6.20}$$

The above equation is the equivalent of Equation 6.9 for the Euler flow equations. It can be seen that the influence of the flow variable in the objective function gradient comes from the whole domain and its boundaries. Hence, the gradient expression has a domain and boundary integral as follows:

$$\int_V \left[\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial \theta} \left(B - v_g I \right) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D \right] \tilde{U} dv \tag{6.21}$$

$$- \int_{\partial V} \left[\lambda^T A \cdot n_x + \lambda^T \left(B - v_g I \right) \cdot n_\theta + \lambda^T C \cdot n_r - \frac{\partial M}{\partial U} \right] \tilde{U} dS \tag{6.22}$$

The adjoint principle is based on eliminating the need for evaluating the objective function gradient (and solving the flow equations) n times for n design variables by eliminating the flow variable sensitivity $\tilde{U}$ from the gradient expression (Equation 6.20). This is effectively done in two steps:

1. Eliminate the flow sensitivity in the domain integral (Equation 6.21): Inside the computational domain, $\tilde{U}$ cannot be zero at all points and so, the term multiplying $\tilde{U}$ has to be vanished, leading to the adjoint Euler equations:

$$\frac{\partial \lambda^T}{\partial x} A + \frac{\partial \lambda^T}{r\partial \theta} \left(B - v_g I \right) + \frac{\partial \lambda^T}{\partial r} C + \lambda^T D = 0 \tag{6.23}$$

2. Eliminate the flow sensitivity in the boundary integral (Equation 6.22): Along the boundaries of the computational domain, $\tilde{U}$ can be zero at some points, but not in

every boundary. In order to do that, the term multiplying $\tilde{U}$ is set to zero as follows:

$$\lambda^T A \cdot n_x + \lambda^T \left(B - v_g I\right) \cdot n_\theta + \lambda^T C \cdot n_r - \frac{\partial M}{\partial U} = 0$$

This condition defines the derivation of the adjoint boundary conditions for the system of adjoint equations

From these two steps, the adjoint problem is well defined with its own system of equations and corresponding boundary conditions, where the adjoint variable values are calculated from. The time cost of solving the system of adjoint equations is roughly equivalent to the time cost of solving the nonlinear flow equations. The remaining terms in the gradient equation (6.20) are:

$$\frac{dI}{d\alpha} = \int_{\partial V} \frac{\partial M}{\partial \alpha} dS + \int_{\partial V} M \tilde{dS} + \int_V \lambda^T f dv \tag{6.24}$$

Once the adjoint variable is calculated, it is added to the expression above in order to calculate the gradient. The cost of post-processing all the remaining terms of the gradient equation is much smaller than the time needed for solving either the flow equations or adjoint equations. In summary, there are mainly three equations to solve:

- The nonlinear flow equations
- The linear adjoint equations (6.23)
- The gradient equation (6.24)

And the overall computational cost is roughly two times the cost of solving the baseline unsteady flow equations, independently of the number of design variables.

6.3 Summary

The adjoint method principle is first introduced in order to illustrate the strengths of the method. Then a continuous approach is followed to derive a system of steady adjoint

equations based on the Euler flow equations. This is expected to give a better understanding of the basics of the method and how it works before introducing a system of unsteady adjoint equations based on the URANS flow equations in the next chapter.

Chapter 7

Continuous Unsteady Adjoint System

7.1 Overview

The URANS equations in Chapter 3 are solved in the frequency domain using the Nonlinear Harmonic Phase Solution Method. Different approaches, both continuous and discrete, have been used to formulate a system of adjoint equations that could efficiently calculate the aeroelastic sensitivities of turbomachinery blades from the flow solutions obtained by frequency domain methods.

Duta et al. (2002) were the first to evaluate this approach with a discrete adjoint system based on the linear harmonic flow equations. In the discrete approach, the flow equations are fully discretized first and then linearisation is applied next to formulate the discrete system. Other discrete developments have been done by Thomas et al. (2003) and Huang and Ekici (2014) based on the Harmonic Balance Technique. These last two approaches made use of AD tools to obtain the adjoint equations.

The most relevant work for this thesis is the continuous discretization performed by He and Wang (2011). The discretization starts with the three semi-discretized flow equations at the three phase angles in one period, at which the nonlinear flow equations are linearised. Next, a system of three steady-like adjoint equations is derived based on the three

steady-like flow equations from the Nonlinear Harmonic Phase Solution Method. This offers an easy way to formulate the adjoint system (no time variable involved), making use of the flow discretization system and leading to a less complicated way of calculating the final gradients.

This continuous approach efficiently captures the aerodynamic damping and forcing sensitivities to blade geometry changes in a single-bladerow environment. However, while mirroring some of the features of the Nonlinear Harmonic Phase Solution Method, the derived adjoint periodic boundary conditions and source terms do not have the same harmonic form as their prime flow solver, and it is not intuitive to find a relation between them. While the flow variable is assumed to be harmonic, the same can not be easily demonstrated for the adjoint variable. This is important as it prevents the extension of the harmonic interface technique that is used on the flow solution method to enable multi-bladerow optimization with the unsteady adjoint method, as it is the case for the adjoint mixing-plane treatment for steady multi-bladerow optimization.

As an attempt to address the issues mentioned above, a system of unsteady adjoint equations is re-derived following a more continuous approach while still being based upon the Nonlinear Harmonic Phase Solution Method and the work done by He and Wang (2011). The URANS equations with the source terms from the Nonlinear Harmonic Phase Solution Method are linearised first in the time domain. Next, the adjoint equation is formulated and the resultant system of three steady-like adjoint equations is discretized in the frequency domain with the same harmonic assumption as the flow solution system. Following this approach, the resulting source terms of the final adjoint system and the new adjoint periodic boundary conditions share the same form as their equivalents from the prime flow equations. In addition, the harmonic multi-bladerow treatment can now be adopted as a basis to establish a new unsteady adjoint interface method to pass the unsteady adjoint disturbances across bladerows.

7.2 Continuous Unsteady Adjoint Method Principle

This section extends the adjoint principle formulation described in Chapter 6 to the derivation of a system of unsteady adjoint equations based on the Nonlinear Harmonic Phase Solution Method (Equation 3.5). The same solution method for solving the flow equations can be applied to calculate the adjoint variable vector λ^T. The URANS equation under the Nonlinear Harmonic Phase Solution Method assumption is written below:

$$\omega\left(U_A\cos(\omega t) - U_B\sin(\omega t)\right) + R'(U) = 0 \tag{7.1}$$

where R' represents the spatial differential operator of the flow equations (Equation 3.3). The time derivative term is substituted by a source term with the harmonic components of the flow variable. Following the adjoint method principle, the adjoint variable vector λ^T is added multiplying the flow equations as follows:

$$\lambda^T\left[\omega\left(U_A\cos\omega t - U_B\sin\omega t\right) + R'(U)\right] = 0 \tag{7.2}$$

This equation holds true, independently of the value of λ^T. As opposed to the steady flow equation, the above equation is a function that must be integrated over time:

$$\int_0^T \lambda^T\left[\omega\left(U_A\cos\omega t - U_B\sin\omega t\right) + R'(U)\right] dt = 0 \tag{7.3}$$

The equality above remains intact due to the periodicity of the flow. This allow us to assume that the adjoint variable has the same harmonic form as the flow variable:

$$\lambda = \overline{\lambda} + \lambda_A\sin\omega t + \lambda_B\cos\omega t \tag{7.4}$$

The next step is to linearise the flow equations. The source terms containing the unknowns $U_A$and U_B are rearranged using the properties of Fourier series for time-periodic

functions:

$$\frac{T}{2}\omega\left(\lambda_B^T U_A - \lambda_A^T U_B\right) + \int_0^T \lambda^T R'(U)\cdot dt = 0 \tag{7.5}$$

where:

$$\lambda_A^T = \frac{2}{T}\int_0^T \lambda^T \sin\omega t\cdot dt \tag{7.6a}$$

$$\lambda_B^T = \frac{2}{T}\int_0^T \lambda^T \cos\omega t\cdot dt \tag{7.6b}$$

$$U_A = \frac{2}{T}\int_0^T U \sin\omega t\cdot dt \tag{7.6c}$$

$$U_B = \frac{2}{T}\int_0^T U \cos\omega t\cdot dt \tag{7.6d}$$

Substituting U_A and U_B for Equations 7.6c and 7.6d in Equation 7.5 yields:

$$\int_0^T \left[\omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right)U + \lambda^T R'\right] dt = 0 \tag{7.7}$$

After this rearrangement, Equation 7.7 is linearised as follows:

$$\int_0^T \left[\omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) + \lambda^T \frac{\partial R'}{\partial U}\right]\frac{\partial U}{\partial \alpha}dt + \int_0^T \lambda^T \frac{\partial R}{\partial \alpha}dt = 0 \tag{7.8}$$

The equality holds again after linearisation if the adjoint variable is proven to be periodic. The objective function is also a function of time $I = I(\alpha, U)$, and so the gradient expression is written as follows:

$$\frac{dI}{d\alpha} = \int_0^T \left(\frac{\partial I}{\partial \alpha} + \frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha}\right) dt \tag{7.9}$$

Subtracting the product of Equation 7.8 from the gradient expression and rearranging around the flow variable sensitivity gives:

$$\frac{dI}{d\alpha} = \int_0^T \left[\frac{\partial I}{\partial U} - \omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) - \lambda^T \frac{\partial R'}{\partial U}\right]\frac{\partial U}{\partial \alpha}dt + \int_0^T \left(\frac{\partial I}{\partial \alpha} - \lambda^T \frac{\partial R}{\partial \alpha}\right) dt \tag{7.10}$$

The adjoint equation is defined so that the flow variable sensitivity term can be eliminated from the gradient expression:

$$\omega\left(\lambda_A^T \cos\omega t - \lambda_B^T \sin\omega t\right) + \frac{\partial I}{\partial U} - \lambda^T \frac{\partial R'}{\partial U} = 0 \tag{7.11}$$

The resulting unsteady adjoint equation has a source term with the same form as its equivalent from the flow equation but with a negative sign. This is in line to the opposite characteristics propagation in time that was first mentioned by Giles and Pierce (2000), which is called as 'anti-physics' adjoint information path and systematically shown by Wang and He (2010a). The fact that the adjoint variable is assumed to have the same harmonic form as the flow variable means that the adjoint equation can also be solved as a system of three steady-like equations following the same procedure as the Nonlinear Harmonic Phase Solution Method. The above equation becomes the steady adjoint equation 6.10 if steady flow, $R(U) = 0$, is considered instead. In this case, the adjoint variable becomes $\lambda = \overline{\lambda}$, and this leads to:

$$\frac{\partial I}{\partial U} - \lambda^T \frac{\partial R}{\partial U} = 0 \tag{7.12}$$

7.3 Unsteady Adjoint Equations

The principle explained above is now applied in detail to show the full system of unsteady adjoint equations based on the URANS equations and the Nonlinear Harmonic Phase Solution Method. The objective function in the context of unsteady flow design optimization is an integral over the whole computational domain and boundaries in space, and over one period of blade vibration in time.

$$I = \int_0^T \int_{\partial V} M\Big[\alpha, U(\alpha)\Big] dS dt \tag{7.13}$$

The gradient of the objective function to the design variables is given by:

$$\frac{dI}{d\alpha} = \int_0^T \int_{\partial V} \left(\frac{\partial M}{\partial \alpha} + \frac{\partial M}{\partial U} \tilde{U} \right) dS dt + \int_0^T \int_{\partial V} M \tilde{dS} dt \tag{7.14}$$

The URANS equation under the harmonic flow variable assumption (Equation 7.1) is expanded below:

$$\begin{aligned} &\omega \left(U_A \cos \omega t - U_B \sin \omega t \right) \\ &\qquad + \frac{\partial \left(F - U u_g - V_x \right)}{\partial x} + \frac{\partial \left(G - U v_g - V_\theta \right)}{r \partial \theta} + \frac{\partial r \left(H - U w_g - V_r \right)}{r \partial r} - S = 0 \end{aligned} \tag{7.15}$$

The above equation is multiplied by the adjoint variable and integrated in both time and space:

$$\begin{aligned} \int_0^T \int_V \lambda^T \Bigg[&\omega \left(U_A \cos \omega t - U_B \sin \omega t \right) + \frac{\partial \left(F - U u_g - V_x \right)}{\partial x} \\ &+ \frac{\partial \left(G - U v_g - V_\theta \right)}{r \partial \theta} + \frac{\partial r \left(H - U w_g - V_r \right)}{r \partial r} - S \Bigg] dv dt = 0 \end{aligned} \tag{7.16}$$

As with Equation 7.7, the properties of Fourier series for periodic time-functions can be used to rearrange the expression above as follows:

$$\begin{aligned} \int_0^T \int_V &\omega \left(\lambda_B^T \sin \omega t - \lambda_A^T \cos \omega t \right) U + \lambda^T \Bigg[\frac{\partial \left(F - U u_g - V_x \right)}{\partial x} \\ &+ \frac{\partial \left(G - U v_g - V_\theta \right)}{r \partial \theta} + \frac{\partial r \left(H - U w_g - V_r \right)}{r \partial r} - S \Bigg] dv dt = 0 \end{aligned} \tag{7.17}$$

Equation 7.17 can now be linearised. The process of linearisation and further subtraction from the gradient equation is detailed in Appendix A. The result of this process gives the

system of unsteady adjoint equations below:

$$\omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) + \frac{\partial \lambda^T}{\partial x}\left(A - Iu_g - A_v\right) + \frac{\partial \lambda^T}{r\partial\theta}\left(B - Iv_g - B_v\right) + \frac{\partial \lambda^T}{\partial r}\left(C - Iw_g - C_v\right) - N^T\left(\frac{\partial D_x}{\partial x} + \frac{\partial D_\theta}{r\partial\theta} + \frac{\partial D_r}{\partial r}\right) - \lambda^T D = 0 \quad (7.18)$$

This expression shows some similarities with the steady adjoint Euler equations but with extra terms that are attributable to the viscous terms of the URANS equations and the time derivative term being substituted for a source term. The spatial derivatives above are collected in a residual term $R_\lambda(\lambda)$, which simplifies Equation 7.18 as follows:

$$-\omega\left(\lambda_A \cos\omega t - \lambda_B \sin\omega t\right) + R_\lambda(\lambda) = 0 \quad (7.19)$$

The key to solving this equation is to assume that the adjoint variable has the same harmonic form as the flow variable:

$$\lambda = \overline{\lambda} + \lambda_A \sin\omega t + \lambda_B \cos\omega t \quad (7.20)$$

Hence, it seems natural to apply the same technique as for the flow equations. The adjoint variable can be defined at three phase angles as follows:

$$\lambda\left(\omega t = -\frac{\pi}{2}\right) = \lambda_{-1} = \bar{\lambda} - \lambda_A \quad (7.21a)$$

$$\lambda\left(\omega t = 0\right) = \lambda_0 = \bar{\lambda} + \lambda_B \quad (7.21b)$$

$$\lambda\left(\omega t = \frac{\pi}{2}\right) = \lambda_1 = \bar{\lambda} + \lambda_A \quad (7.21c)$$

The values of the three unknowns of the adjoint equation $\bar{\lambda}, \lambda_A$and λ_B can be written as

a function of the equations above, leading to:

$$\bar{\lambda} = \frac{\lambda_1 + \lambda_{-1}}{2} \tag{7.22a}$$

$$\lambda_A = \frac{\lambda_1 - \lambda_{-1}}{2} \tag{7.22b}$$

$$\lambda_B = \lambda_0 - \frac{\lambda_1 + \lambda_{-1}}{2} \tag{7.22c}$$

Substituting this back into Equation 7.19 gives the final system of three steady-like adjoint equations:

$$-\omega\left(\lambda_0 - \frac{\lambda_1 + \lambda_{-1}}{2}\right) + R_\lambda(\lambda_1) = 0 \tag{7.23a}$$

$$-\omega\left(\frac{\lambda_1 - \lambda_{-1}}{2}\right) + R_\lambda(\lambda_0) = 0 \tag{7.23b}$$

$$\omega\left(\lambda_0 - \frac{\lambda_1 + \lambda_{-1}}{2}\right) + R_\lambda(\lambda_1) = 0 \tag{7.23c}$$

The system of equations above has the same form as the prime flow equations, but the opposite sign of the source terms in Equation 7.19 effectively changes the sign of the source terms for the three phase angles. The same discretization scheme as the one used for the flow equations can be used to solve the system of adjoint equations. It is important to note that, as indicated in section 2.4, the eddy viscosity is frozen and it is only updated in the direct flow solution once in each design cycle.

7.4 Discretization of the Equations

7.4.1 Spatial Discretization

The adjoint equations are solved in a finite volume framework similarly to the unsteady flow equations. In order to do so, they have to be written in a strong conservative form with the extra terms being taken as source terms:

$$\int_{\partial V}\Big[\lambda^T\left(A-Iu_g-A_v\right)\cdot n_x+\lambda^T\left(B-Iv_g-B_v\right)\cdot n_\theta + \lambda^T\left(C-Iw_g-C_v\right)\cdot n_r-N^T\left(D_x\cdot n_x+D_\theta\cdot n_\theta+D_r\cdot n_r\right)\Big]dS = -\int_V\left[\omega\left(\lambda_B^T\sin\omega t-\lambda_A^T\cos\omega t\right)+\lambda^T D\right]dv \quad (7.24)$$

The spatial discretization is performed using a central difference method with artificial viscosity. The discretized convective fluxes have also three components: inviscid, viscid and dissipative. The calculation of the inviscid and viscous fluxes of the adjoint system is explained below based on the same 2D control volume with primary and staggered meshes (Figures 3.4a and 3.4b) as those used in Chapter 3.

Inviscid Flux

The adjoint inviscid flux for a 2D control volume is defined as:

$$\lambda^T\left(A-Iu_g\right)\cdot n_x+\lambda^T\left(B-Iv_g\right)\cdot n_y \quad (7.25)$$

At the bounding surface indexed as $\left(i+\frac{1}{2},\,j\right)$, the inviscid adjoint flux is the average of the calculated adjoint fluxes using the flow and adjoint solution at the two adjacent cells $(i\,,j)$ and $\left(i+1,\,j\right)$ as follows:

$$Flux^i_{i+\frac{1}{2},\,j}=\lambda^T_{i+\frac{1}{2},\,j}\cdot\left(A-Iu_g\right)_{i+\frac{1}{2},\,j}S_x+\lambda^T_{i+\frac{1}{2},\,j}\left(B-Iv_g\right)_{i+\frac{1}{2},\,j}S_y \quad (7.26)$$

where

$$\lambda^T_{i+\frac{1}{2},j} = \frac{1}{2}\left[\lambda^T_{i,j} + \lambda^T_{i+1,j}\right]$$

$$\left(A - Iu_g\right)_{i+\frac{1}{2},j} = \frac{1}{2}\left[A\left(U_{i,j}\right) + A\left(U_{i+1,j}\right)\right] - \frac{1}{2}I\left[u_g\left(U_{i,j}\right) + u_g\left(U_{i+1,j}\right)\right]$$

$$\left(B - Iv_g\right)_{i+\frac{1}{2},j} = \frac{1}{2}\left[B\left(U_{i,j}\right) + B\left(U_{i+1,j}\right)\right] - \frac{1}{2}I\left[v_g\left(U_{i,j}\right) + v_g\left(U_{i+1,j}\right)\right]$$

Viscous Flux

The adjoint viscous flux for a 2D control volume is defined as:

$$Flux^v_{i+\frac{1}{2},j} = -\lambda^T A_v \cdot n_x - \lambda^T B_v \cdot n_y - N^T\left(D_x \cdot n_x + D_y \cdot n_y\right) \tag{7.27}$$

At the bounding surface indexed as $\left(i+\frac{1}{2},\ j\right)$, this is equal to:

$$Flux^v_{i+\frac{1}{2},j} = -\lambda^T_{i+\frac{1}{2},j} \cdot A_{v_{i+\frac{1}{2},j}} S_x - \lambda^T_{i+\frac{1}{2},j} B_{v_{i+\frac{1}{2},j}} S_y - N^T\left(D_{x_{i+\frac{1}{2},j}} \cdot S_x + D_{y_{i+\frac{1}{2},j}} \cdot S_y\right) \tag{7.28}$$

As with the viscous fluxes of the flow system, the spatial derivatives have to be discretized with the Green's theorem. Once all the spatial derivatives have been approximated, all the terms in the viscous flux expression can be resolved.

7.4.2 Temporal Discretization

The time-marching is performed using the same 4-steps Runge-Kutta method used for the flow equations in Chapter 3. Because the adjoint time derivative term has being substituted by the source terms due to the harmonic assumption, a pseudo time derivative term is added to the adjoint equations as follows:

$$\frac{\partial \lambda}{\partial \tau} + \omega\left(\lambda_A \cos\omega t - \lambda_B \sin\omega t\right) + R_\lambda(\lambda) = 0 \tag{7.29}$$

Same as the flow equations, a new residual operator can be defined so that it includes the source terms for each of the three phases of unsteadiness:

$$R'_{\lambda}(\lambda_i) = -\omega\left(\lambda_A \cos\omega t - \lambda_B \sin\omega t\right) + R_{\lambda}(\lambda_i) = 0 \quad i = -1, 0, 1 \tag{7.30}$$

Hence, the four-step time-marching method is summarised as follows:

$$\lambda_i^{n+\frac{1}{4}} = \lambda_i^n - \frac{1}{4}\frac{\triangle\tau}{\triangle V}\left[R'_{\lambda}\left(\lambda_i^n\right)\right] \tag{7.31a}$$

$$\lambda_i^{n+\frac{1}{3}} = \lambda_i^n - \frac{1}{3}\frac{\triangle\tau}{\triangle V}\left[R'_{\lambda}\left(\lambda_i^{n+\frac{1}{4}}\right)\right] \tag{7.31b}$$

$$\lambda_i^{n+\frac{1}{2}} = \lambda_i^n - \frac{1}{2}\frac{\triangle\tau}{\triangle V}\left[R'_{\lambda}\left(\lambda_i^{n+\frac{1}{3}}\right)\right] \tag{7.31c}$$

$$\lambda_i^{n+1} = \lambda_i^n - \frac{\triangle\tau}{\triangle V}\left[R'_{\lambda}\left(\lambda_i^{n+\frac{1}{2}}\right)\right] \tag{7.31d}$$

The multi-grid and local time-stepping techniques are also used to solve the adjoint equations and improve the convergence speed. All these subroutines can be directly applied to the adjoint solver without major modifications.

7.5 Boundary Conditions

The system of adjoint equations is also solved as a hyperbolic system due to the addition of the pseudo time derivative term. The compatibility requirement is that the number of inlet and exit boundary conditions must be the same as the number of characteristics running into the computational domain through the corresponding boundary. Giles and Pierce (2000) pointed out that the sign of each adjoint characteristic velocity is opposite to that of the corresponding flow characteristic. This approach was first applied by He and Wang (2011); Wang and He (2010a,b) on their continuous approach. For each flow characteristic, there is a corresponding adjoint flow characteristic travelling at a speed of

the same magnitude but in opposite direction. This forces the number of inlet and exit adjoint boundary conditions to be the opposite as for the flow inlet and exit boundary conditions. There are four types of adjoint boundary conditions based upon the four types of flow boundary conditions:

- Subsonic inlet boundary condition
- Subsonic exit boundary condition
- Viscous solid wall boundary condition
- Periodic boundary condition

The viscous terms are assumed to be negligible at the inlet and exit boundary. However the viscous effect at the solid wall boundary is significant and can not be neglected. The derivation of these three boundary conditions (inlet, exit and solid wall) is explained in a way that the similarities and differences with their flow counterparts can be better understood. Their implementation is the same as the steady and unsteady work of He and Wang (2011); Wang and He (2010a,b). However the continuous periodic boundary condition is unique to this work and will be presented in a separate section.

An important term in the boundary integral is $\frac{\partial M}{\partial U}$, which is an analytical term that relates the gradient of the objective function I to the flow variable vector U . This term is explained in detail in section 7.7 and its expression is different for each boundary condition.

Inlet boundary condition

The subsonic inlet boundary condition of the flow equations has four incoming and one outgoing flow characteristics. The static pressure at the inlet boundary is extrapolated from the interior domain while the rest of the flow variables (density and velocity components) are obtained through the isentropic relation and specified velocity angles. Pressure is the only independent flow variable at the inlet boundary. The opposite charac-

teristics of the adjoint system compared to that of the physical flow system means that there is only one incoming and four outgoing adjoint characteristics. If the inlet plane is assumed to be normal to the axial axis, only axial directional flux goes through the boundary. The boundary integral (Equation. A.12) can be rewritten as follows.

$$\int_{\partial V}\left[\lambda^T\left(A-Iu_g\right)\cdot n_x-\frac{\partial M}{\partial U}\right]\tilde{U}dS=\int_{\partial V}\left[\lambda^T\left(\frac{\partial F}{\partial U}-Iu_g\right)\cdot n_x-\frac{\partial M}{\partial U}\right]\tilde{U}dS=0 \tag{7.32}$$

The flow variable sensitivity is a function of the primitive flow variables: $\tilde{U}=\frac{\partial}{\partial\alpha}\{\rho,u,v,w,p\}^T$. Based on these variables, the boundary integral above can be expanded to give:

$$\begin{aligned}\int_{\partial V}\Bigg[&\left(\lambda^T\frac{\partial F}{\partial\rho}\cdot n_x-\frac{\partial M}{\partial\rho}\right)\tilde{\rho}+\left(\lambda^T\frac{\partial F}{\partial u}\cdot n_x-\frac{\partial M}{\partial u}\right)\tilde{u}\\&+\left(\lambda^T\frac{\partial F}{\partial v}\cdot n_x-\frac{\partial M}{\partial v}\right)\tilde{v}+\left(\lambda^T\frac{\partial F}{\partial w}\cdot n_x-\frac{\partial M}{\partial w}\right)\tilde{w}+\left(\lambda^T\frac{\partial F}{\partial p}\cdot n_x-\frac{\partial M}{\partial p}\right)\tilde{p}\Bigg]dS=0\end{aligned} \tag{7.33}$$

Similar to the flow variable U, there are also 5 primitive adjoint variables $\lambda=\{\lambda_1,\lambda_2,\lambda_3,\lambda_4,\lambda_5\}$. Each of these adjoint variables is linked to its corresponding flow variable. Hence, λ_1 could be considered as a pseudo adjoint density and so the rest of the terms of λ. Because there is only one independent flow variable at the inlet, the only product of the equation above that is not equal to zero is the one corresponding to the pressure sensitivity $\tilde{p}$. Because $\tilde{p}$ can not be zero in the whole boundary, the sufficient and necessary condition to satisfy the above equation is expressed as the derivative of the boundary integral to the pressure times the pressure perturbation:

$$\lambda^T\frac{\partial F}{\partial p}\cdot n_x-\frac{\partial M}{\partial p}=0 \tag{7.34}$$

This scalar equation defines the corresponding adjoint inlet boundary condition. As opposed to the flow solution method, four adjoint variables $\lambda_{1,2,3,4}$ are extrapolated from the interior domain, while one adjoint variable λ_5 is worked out according to Equation 7.34 as follows:

$$\lambda_5=\frac{\partial M}{\partial p}-\frac{\partial p}{\partial F_5}\left[\sum_{i=1}^{n=4}\left(\lambda_i\frac{\partial F_i}{\partial p}\right)\right] \tag{7.35}$$

The elements of the Jacobian matrix regarding the derivative of the flux vector to the pressure $\left(\frac{\partial F_i}{\partial p}\right)$ can be obtained via the complex variable method (Vatsa, 2004), while $\left(\frac{\partial M}{\partial p}\right)$ is obtained as shown in section 7.7.

Exit boundary condition

At the subsonic exit boundary, the number of characteristics and boundary conditions is reversed compared to that for the inlet. There is one incoming and four outgoing flow characteristics. The static pressure is now specified at the exit boundary condition while the density and velocity components are interpolated from the interior domain. Hence, there are four independent flow variables at the exit boundary and Equation 7.33 is reduced to the following expression:

$$\int_{\partial V}\left(\lambda^T\frac{\partial F}{\partial \rho}\cdot n_x-\frac{\partial M}{\partial \rho}\right)\tilde{\rho}+\left(\lambda^T\frac{\partial F}{\partial u}\cdot n_x-\frac{\partial M}{\partial u}\right)\tilde{u} + \left(\lambda^T\frac{\partial F}{\partial v}\cdot n_x-\frac{\partial M}{\partial v}\right)\tilde{v}+\left(\lambda^T\frac{\partial F}{\partial w}\cdot n_x-\frac{\partial M}{\partial w}\right)\tilde{w}\cdot dS=0 \quad (7.36)$$

The sufficient and necessary condition to satisfy the above equation is:

$$\left(\lambda^T\frac{\partial F}{\partial q}\cdot n_x-\frac{\partial M}{\partial q}\right)\tilde{q}=0 \;\; q=\rho,u,v,w \quad (7.37)$$

Equation 7.37 represents four scalar equations, corresponding to the four incoming adjoint exit boundary conditions. The adjoint variable λ_5 is extrapolated from the interior domain, while the other four adjoint variables $\lambda_1,\lambda_2,\lambda_3$ and λ_4 are calculated from the next system of four equations and four variables:

$$\lambda_j=\frac{\partial M}{\partial U_j}-\frac{\partial p}{\partial F_j}\left[\sum_{i=1}^{n=4}\left(\lambda_i\frac{\partial F_i}{\partial p}\right)\right] \quad j=1,..,4 \quad (7.38)$$

where the calculation of the elements of the Jacobian matrices and $\left(\frac{\partial M}{\partial U}\right)$ is performed with the same method as the adjoint inlet boundary condition.

Solid-wall boundary condition

Similar to the inlet and exit boundaries, the dependence of the flow variable sensitivity in the boundary integral also needs to be eliminated. The boundary integral is given below (Appendix A):

$$\int_0^T \int_{\partial V} \Big[\lambda^T \left(A - Iu_g - A_v\right) n_x + \lambda^T \left(B - Iv_g - B_v\right) + \lambda^T \left(C - Iw_g - C_v\right) n_r - \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x - \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta - \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r - \frac{\partial M}{\partial U} \Big] \tilde{U} \cdot dS dt = 0 \quad (7.39)$$

This integral can be separated into two parts: the inviscid part and the viscous part as follows:

$$\int_0^T \int_{\partial V} \left[\lambda^T \left(A - Iu_g\right) n_x + \lambda^T \left(B - Iv_g\right) n_\theta + \lambda^T \left(C - Iw_g\right) n_r - \frac{\partial M}{\partial U} \right] \tilde{U} \cdot dS dt = 0 \quad (7.40)$$

$$\int_0^T \int_{\partial V} \left[\lambda^T \left(A_v n_x + B_v n_\theta + C_v n_r\right) \right] \tilde{U} \cdot dS dt + \int_0^T \int_{\partial V} \Big[\left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x + \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta + \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r\partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r \Big] \tilde{U} \cdot dS dt = 0 \quad (7.41)$$

These two parts have to be vanished separately. The derivation of the corresponding solid wall boundary conditions is explained in detail in Appendix B. In summary, the necessary

conditions to vanish the above equalities are:

$$\lambda_2 \cdot n_x + r\lambda_3 \cdot n_\theta + \lambda_4 \cdot n_r + \lambda_5 \left(u_g \cdot n_x + v_g \cdot n_\theta + w_g \cdot n_r \right) = \frac{\partial M}{\partial p} \tag{7.42}$$

In addition, for a viscous solid wall, the next conditions also need to be satisfied:

$$\sigma_x = \frac{\tau_w}{W} \lambda_2 \tag{7.43a}$$

$$\sigma_\theta = \frac{\tau_w}{W} \left(\lambda_3 + \Omega \lambda_5 \right) \tag{7.43b}$$

$$\sigma_r = \frac{\tau_w}{W} \lambda_4 \tag{7.43c}$$

If the wall is adiabatic:

$$\frac{\partial \lambda_5}{\partial n} = 0 \tag{7.44}$$

If the wall is non-adiabatic:

$$\lambda_5 = 0 \tag{7.45}$$

7.6 Periodic Boundary Condition: Inter-blade Phase Angle

The "anti-physics" path of the adjoint characteristics, as articulated by Wang and He (2010a,b), is fundamental for the derivation of the inlet and exit adjoint boundary conditions. The propagation of the information is reversed from the physical domain and so the number of inlet and exit boundary conditions. This approach, suggested by He and Wang (2011) helped to clarify the meaning of the adjoint boundary conditions and its implementation. However, it is not possible to conclude a similar relation when contrasting unsteady flow and adjoint periodic boundary conditions.

In a physical time domain, the flow field at the initial time is defined by the boundary

conditions. The opposite is true with respect to the adjoint field: it is null at the final time (as imposed by the adjoint method and its boundary conditions), while it is unknown at the initial time. From this continuous approach, the adjoint variable has been assumed to have the same harmonic form as the flow variable. This is true for the physical time domain t_p, where:

$$\lambda = \overline{\lambda} + \lambda_A \sin \omega t_p + \lambda_B \cos \omega t_p \tag{7.46}$$

An adjoint time domain can be defined as being the opposite of the physical time domain $\left(t_p = -t_a\right)$. In this time domain, the adjoint variable has the next form:

$$\lambda = \overline{\lambda} - \lambda_A \sin \omega t_a + \lambda_B \cos \omega t_a \tag{7.47}$$

The switch between the physical time and the adjoint time changes the sign of the harmonic terms. The cosine term remains the same but the sine term has its sign changed. The harmonic form of the adjoint variable allows the assumption that the upper and lower periodic boundaries (Figure 7.1) are phase-shifted in a similar way to the flow field upper and lower periodic boundary conditions (Equation 3.29 and 3.30). In a physical time domain, the upper boundary condition has a phase lead over the lower boundary as follows:

$$\lambda^U = \bar{\lambda^L} + \lambda_A^L \sin\left(\omega t_p + \sigma\right) + \lambda_B^L \cos\left(\omega t_p + \sigma\right) \tag{7.48}$$

$$\lambda^L = \bar{\lambda^U} + \lambda_A^U \sin\left(\omega t_p - \sigma\right) + \lambda_B^U \cos\left(\omega t_p - \sigma\right) \tag{7.49}$$

In the adjoint time domain, the relation between the upper and lower periodic boundary is completely reversed. Changing to the adjoint time domain, the above equation at the upper boundary can be rewritten as:

$$\lambda^U = \bar{\lambda^L} + \lambda_A^L \sin\left(-\omega t_a + \sigma\right) + \lambda_B^L \cos\left(-\omega t_a + \sigma\right) \tag{7.50}$$

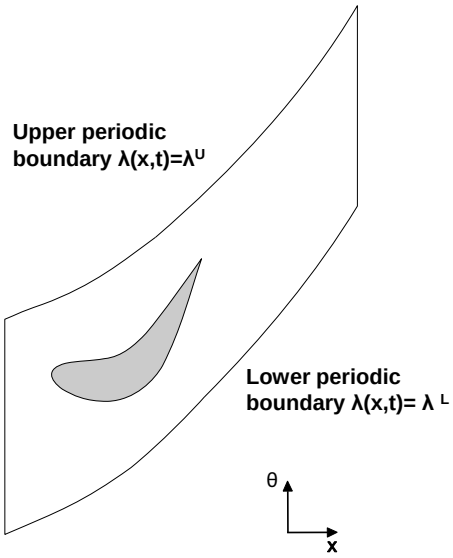


Figure 7.1: Single-passage with adjoint periodic boundaries

This leads to

$$\lambda^U = \bar{\lambda^L} - \lambda_A^L \sin(\omega t_a - \sigma) + \lambda_B^L \cos(\omega t_a - \sigma) \tag{7.51}$$

In the adjoint time domain, the upper boundary is lagged respect to the lower boundary. As a result, the lower boundary is now the boundary that has the phase lead:

$$\lambda^L = \bar{\lambda^U} - \lambda_A^U \sin(\omega t_a + \sigma) + \lambda_B^U \cos(\omega t_a + \sigma) \tag{7.52}$$

7.7 Gradients of the Objective Function

The objective function has been introduced in section 5.5 as a function M of the flow variable vector U and the design variable vector α as follows:

$$I = M(U, \alpha) \tag{7.53}$$

The performance parameters included in the objective function are: entropy generation rate $\triangle\bar{s}$, work sum W, mass flow rate $\dot{m}$ and pressure ratio π. The relation between

the objective function and the adjoint boundary conditions is explicitly included in the boundary integral (A.12) with the term $\frac{\partial M}{\partial U}$, that has to be evaluated at the inlet, exit and solid wall boundary conditions. This term establishes the sensitivity of the four performance parameters of the objective function to the primitive flow variables and it can be written as:

$$\frac{\partial M}{\partial U} = \frac{\partial M_{\triangle s}}{\partial U} + \frac{\partial M_{\dot{m}}}{\partial U} + \frac{\partial M_{\dot{\pi}}}{\partial U} + \frac{\partial M_W}{\partial U} \tag{7.54}$$

For each of the two main (aerodynamic or aeroelastic) objective functions, the equation above can be expanded as follows:

$$\frac{\partial M}{\partial U} = \frac{1}{\triangle \bar{s}_0}\frac{\partial \triangle \bar{s}}{\partial U} + 2\sigma_1\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0}\frac{\partial \dot{m}}{\partial U} + 2\sigma_2\left(\frac{\pi}{\pi_0} - 1\right)\frac{1}{\pi_0}\frac{\partial \pi}{\partial U} + 2\sigma_3\left(\frac{W}{W_0} - 1\right)\frac{1}{W_0}\frac{\partial W}{\partial U} \tag{7.55}$$

$$\frac{\partial M}{\partial U} = \frac{1}{W_0}\frac{\partial W}{\partial U} + 2\sigma_1\left(\frac{\dot{m}}{\dot{m}_0} - 1\right)\frac{1}{\dot{m}_0}\frac{\partial \dot{m}}{\partial U} + 2\sigma_2\left(\frac{\pi}{\pi_0} - 1\right)\frac{1}{\pi_0}\frac{\partial \pi}{\partial U} + 2\sigma_3\left(\frac{\triangle \bar{s}}{\triangle \bar{s}_0} - 1\right)\frac{1}{\triangle \bar{s}_0}\frac{\partial \triangle \bar{s}}{\partial U} \tag{7.56}$$

Where $\frac{\partial \triangle \bar{s}}{\partial U}$, $\frac{\partial W}{\partial U}$, $\frac{\partial \dot{m}}{\partial U}$ and $\frac{\partial \pi}{\partial U}$ are the gradients of entropy generation rate, work sum, mass flow rate and pressure ratio to the primitive flow variables respectively. The gradients above are shown for both concurrent objective functions. For aerodynamic-only and aeroelastic-only cases, the last term is equal to zero ($\sigma_3 = 0$). Each of these gradients are not calculated for every boundary condition. For example, the work sum is only calculated at the blade surface, and hence, $\frac{\partial M_W}{\partial U}$ is only calculated in the solid wall adjoint boundary condition. Moreover, each parameter is only a function of certain primitive flow variables $\left(\frac{\partial M_W}{\partial U} = \frac{\partial M_W}{\partial p}\right)$. The number of characteristics and independent primitive flow variables determine the number of $\frac{\partial M}{\partial U}$ terms that are evaluated at each boundary. These can be summarised as follows:

1. Inlet boundary condition: $\frac{\partial M}{\partial p}$: entropy generation, mass flow rate and pressure ratio

2. Exit boundary condition: $\frac{\partial M}{\partial u}, \frac{\partial M}{\partial v}, \frac{\partial M}{\partial w}$ and $\frac{\partial M}{\partial \rho}$: entropy generation, mass flow rate and pressure ratio

3. Wall boundary condition: $\frac{\partial M}{\partial p}$: work sum

7.7.1 Entropy generation rate gradient

The entropy generation rate gradient in Equations 7.55 and 7.56 is defined as the difference between the inlet and exit gradient as follows:

$$\frac{\partial(\triangle \bar{s})}{\partial U} = \frac{\partial(\bar{s_e} - \bar{s_i})}{\partial U} \tag{7.57}$$

Based on the definition of entropy generation in section 5.5, the derivative of $M_{\triangle s}$ with respect to pressure at the inlet boundary is given by:

$$\frac{\partial M_{\triangle s}}{\partial p} = -C_{\triangle s} \cdot \frac{1}{\dot{m}} \left[\left(\frac{1}{\rho^r} - \frac{\frac{p}{\rho^\gamma} \cdot \gamma}{\rho} \frac{d\rho}{dp} \right) \rho v_n + \left(\frac{p}{\rho^\gamma} - \bar{s} \right) \frac{d\rho v_n}{dp} \right] \tag{7.58}$$

where $\dot{m} = \int_i \rho v_n ds$ and $v_n = (u \cdot n_x + v \cdot n_\theta + w \cdot n_r)$. $\frac{d\rho v_n}{dp}$, and $\frac{d\rho}{p}$ are calculated using the complex variable method (Vatsa, 2004) and $C_{\triangle s}$ is defined depending on whether the type of objective function is aerodynamic or aeroelastic (Equation 7.55 or 7.56).

$$C_{\triangle s} = \begin{cases} \frac{1}{\triangle \bar{s}_0} & (7.55) \\ 2\sigma_3 \left(\frac{\triangle \bar{s}}{\triangle \bar{s}_0} - 1 \right) \frac{1}{\triangle \bar{s}_0} & (7.56) \end{cases}$$

At the exit boundary, the derivatives are:

$$\frac{\partial M_{\triangle s}}{\partial \rho} = C_{\triangle s} \cdot \frac{1}{\dot{m}} \left[\left(-\frac{p}{\rho^\gamma} \cdot \gamma \right) v_n + \left(\frac{p}{\rho^\gamma} - \bar{s} \right) \cdot v_n \right] \tag{7.59a}$$

$$\frac{\partial M_{\triangle s}}{\partial u} = C_{\triangle s} \cdot \frac{1}{\dot{m}} \left[\left(\frac{p}{\rho^\gamma} - \bar{s} \right) \cdot \rho n_x \right] \tag{7.59b}$$

$$\frac{\partial M_{\triangle s}}{\partial v} = C_{\triangle s} \cdot \frac{1}{\dot{m}} \left[\left(\frac{p}{\rho^{\gamma}} - \bar{s} \right) \cdot \rho n_{\theta} \right] \tag{7.59c}$$

$$\frac{\partial M_{\triangle s}}{\partial w} = C_{\triangle s} \cdot \frac{1}{\dot{m}} \left[\left(\frac{p}{\rho^{\gamma}} - \bar{s} \right) \cdot \rho n_{r} \right] \tag{7.59d}$$

where $\dot{m} = \int_e \rho v_n ds$

7.7.2 Mass flow rate gradient

The mass flow rate gradient can be defined as follows:

$$\frac{\partial \dot{m}}{\partial U} = \frac{1}{2} \left(N_{bi} \int_i \frac{\partial \rho v_n}{\partial U} dS + N_{be} \int_e \frac{\partial \rho v_n}{\partial U} dS \right) \tag{7.60}$$

At the inlet boundary, the derivative with respect to the static pressure is:

$$\frac{\partial M_{\dot{m}}}{\partial p} = 2\sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right) \frac{1}{\dot{m}_0} \cdot 0.5 \frac{\partial \left(\rho v_n \right)}{\partial p} N_{bi} \tag{7.61}$$

At the exit boundary, the gradients with respect to the density and velocity components are:

$$\frac{\partial M_{\dot{m}}}{\partial \rho} = 2\sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right) \frac{1}{\dot{m}_0} \cdot 0.5 v_n N_{re} \tag{7.62a}$$

$$\frac{\partial M_{\dot{m}}}{\partial u} = 2\sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right) \frac{1}{\dot{m}_0} \cdot 0.5 \rho n_x N_{re} \tag{7.62b}$$

$$\frac{\partial M_{\dot{m}}}{\partial v} = 2\sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right) \frac{1}{\dot{m}_0} \cdot 0.5 \rho n_{\theta} N_{re} \tag{7.62c}$$

$$\frac{\partial M_{\dot{m}}}{\partial w} = 2\sigma_1 \left(\frac{\dot{m}}{\dot{m}_0} - 1 \right) \frac{1}{\dot{m}_0} \cdot 0.5 \rho n_r N_{re} \tag{7.62d}$$

7.7.3 Pressure ratio gradient

The inlet mass averaged total pressure is assumed to be constant due to the specified total pressure profile at the inlet. Its value at the inlet boundary is given below:

$$\frac{\partial M_{\dot{\pi}}}{\partial p} = -\pi \cdot 2\sigma_2 \left(\frac{\pi}{\pi_0} - 1\right) \frac{1}{\pi_0} \frac{1}{\bar{P}_1} \frac{1}{\dot{m}} \left(\bar{p}_1 - \bar{P}_1\right) \cdot \frac{\partial \rho v_n}{\partial p} \tag{7.63}$$

At the exit boundary:

$$\frac{\partial M_{\dot{\pi}}}{\partial \rho} = 2\sigma_2 \left(\frac{\pi}{\pi_0} - 1\right) \frac{1}{\pi_0} \frac{1}{\bar{P}_2} \frac{1}{\dot{m}} \left[\frac{1}{2} a \left(u^2 + v^2 + w^2\right) \rho v_n + \left(\bar{p}_2 - \bar{P}_2\right) v_n\right] \tag{7.64a}$$

$$\frac{\partial M_{\dot{\pi}}}{\partial u} = 2\sigma_2 \left(\frac{\pi}{\pi_0} - 1\right) \frac{1}{\pi_0} \frac{1}{\bar{P}_2} \frac{1}{\dot{m}} \left[a\rho^2 u v_n + \left(\bar{p}_2 - \bar{P}_2\right) \rho n_x\right] \tag{7.64b}$$

$$\frac{\partial M_{\dot{\pi}}}{\partial v} = 2\sigma_2 \left(\frac{\pi}{\pi_0} - 1\right) \frac{1}{\pi_0} \frac{1}{\bar{P}_2} \frac{1}{\dot{m}} \left[a\rho^2 v v_n + \left(\bar{p}_2 - \bar{P}_2\right) \rho n_\theta\right] \tag{7.64c}$$

$$\frac{\partial M_{\dot{\pi}}}{\partial w} = 2\sigma_2 \left(\frac{\pi}{\pi_0} - 1\right) \frac{1}{\pi_0} \frac{1}{\bar{P}_2} \frac{1}{\dot{m}} \left[a\rho^2 w v_n + \left(\bar{p}_2 - \bar{P}_2\right) \rho n_r\right] \tag{7.64d}$$

Where $a = \left[1 + \frac{W^2}{2 \cdot cp \cdot T}\right]^{\frac{1}{\gamma - 1}}$ being W^2 the local velocity magnitude and T the static temperature at the exit.

7.7.4 Work sum gradient

The gradient of work sum to the design variables is defined as:

$$\frac{dW}{d\alpha} = \int_{\partial V} \int_0^T p \left(v_x \cdot n_x + v_\theta \cdot n_\theta + v_r \cdot n_r\right) dt dS + \frac{\partial W}{\partial \alpha} \tag{7.65}$$

where $\frac{\partial W}{\partial \alpha}$ represents the sensitivity of work sum to changes in blade geometry (n_x, n_θ, n_r). This work sum gradient has to be linked to the solid wall boundary condition, as the work sum is a product of the integration of the unsteady pressure over the blade surface (Equa-

tion 5.10). The work sum gradient to the flow variable vector is defined as:

$$\frac{\partial W}{\partial U} = \frac{\partial W}{\partial p} = (v_x \cdot n_x + v_\theta \cdot n_\theta + v_r \cdot n_r) \tag{7.66}$$

The contribution of the work sum to $\frac{\partial M}{\partial U}$ is:

$$\frac{\partial M_W}{\partial p} = C_{\triangle W} \cdot (v_x \cdot n_x + v_\theta \cdot n_\theta + v_r \cdot n_r) \tag{7.67}$$

Similarly to the entropy generation rate, $C_{\triangle W}$ changes depending on the type of objective function (Equation 7.55 or 7.56), so that:

$$C_{\triangle W} = \begin{cases} \frac{1}{W_0} & (7.56) \\ 2\sigma_3 \left(\frac{W}{W_0} - 1\right) \frac{1}{W_0} & (7.55) \end{cases}$$

7.8 Summary

A system of unsteady adjoint equations is formulated based upon the Nonlinear Harmonic Phase Solution Method and the continuous approach. Compared with those obtained from previous approaches, the new system of adjoint equations illustrates better and more consistently the relation between the flow field and adjoint field. The key step of this approach is to assume that the adjoint variable has the same harmonic form as the flow variable. As a consequence, the obtained source terms are symmetric but opposite in sign to the flow equation ones, in accordance with the "anti-physics" path of information.

The new periodic boundary conditions also show how the phase-shifted periodicity can be applied to the adjoint equations and the relation between the physical and adjoint time domains is determined by a simple switch of the inter-blade phase angle.

Chapter 8

Multi-bladerow Treatments

The design optimization using the adjoint method is carried out in a multi-bladerow environment to include the multi-row interaction effects of interest for aeroelastic and aerodynamic optimization. Both, the time-averaged and the harmonic components of the adjoint variable are successfully exchanged among bladerows. This is one of the distinctive features of this work, and it is made possible thanks to the re-derivation of the unsteady adjoint system based upon the harmonic form assumption of the adjoint variable. The clear corresponding between the nonlinear harmonic phase solution formulation and the re-derived adjoint harmonic formulation (as presented in Chapter 7) gives the basis for extending the harmonic interfacing treatment to the adjoint one.

In the following, the adjoint mixing-plane method for the base steady flow is described first, before introducing the new adjoint harmonic interface method for the unsteady flow.

8.1 Adjoint Mixing-plane

The mixing-plane treatment used in this work for the adjoint time-averaged part was developed by Wang and He (2010a,b) and it is illustrated here for convenience. At the interface, the boundary integral of both domains is assumed to satisfy the following equality:

$$\frac{1}{Y_1}\int_{y_1}\left(\lambda^T A\right)\tilde{U}dy = \frac{1}{Y_2}\int_{y_2}\left(\lambda^T A\right)\tilde{U}dy \tag{8.1}$$

Where $A = \frac{\partial F}{\partial U}$ is a Jacobian matrix. The left hand side term belongs to domain 1 and the right hand side term belongs to domain 2, being $Y_1$and Y_2 the length of each domain. The way this equality is satisfied across the interface is by mutual cancellation. Equation 8.1 can be rewritten as follows:

$$\frac{1}{Y_1}\int_{y_1}\lambda^T\tilde{F}dy = \frac{1}{Y_2}\int_{y_2}\lambda^T\tilde{F}dy \tag{8.2}$$

The interface geometry is assumed to be kept intact when a design variable is perturbed as the adjoint variable vector λ^T is independent of any design variable. Hence, the above equation can be further rearranged and the differentiation operator $\widetilde{F} = \frac{\partial F}{\partial \alpha}$ can be removed, which gives:

$$\frac{1}{Y_1}\int_{y_1}\lambda^T F dy = \frac{1}{Y_2}\int_{y_2}\lambda^T F dy \tag{8.3}$$

This simplifies the equality to a system of five independent equations with one adjoint variable for each equation as follows:

$$\frac{1}{Y_1}\int_{y_1}\lambda_1\left(\rho u\right)dy = \frac{1}{Y_2}\int_{y_2}\lambda_1\left(\rho u\right)dy \tag{8.4a}$$

$$\frac{1}{Y_1}\int_{y_1}\lambda_2\left(\rho u^2+p\right)dy = \frac{1}{Y_2}\int_{y_2}\lambda_2\left(\rho u^2+p\right)dy \tag{8.4b}$$

$$\frac{1}{Y_1}\int_{y_1}\lambda_3\left(\rho uv\right)dy = \frac{1}{Y_2}\int_{y_2}\lambda_3\left(\rho uv\right)dy \tag{8.4c}$$

$$\frac{1}{Y_1}\int_{y_1}\lambda_4\left(\rho uw\right)dy = \frac{1}{Y_2}\int_{y_2}\lambda_4\left(\rho uw\right)dy \tag{8.4d}$$

$$\frac{1}{Y_1}\int_{y_1}\lambda_5\left(\rho Hu\right)dy = \frac{1}{Y_2}\int_{y_2}\lambda_5\left(\rho Hu\right)dy \tag{8.4e}$$

These five equations describe the circumferentially averaged fluxes of mass, momentum and energy that have to be conserved across the interface. The conservation of the weighted fluxes is achieved by applying an extension of the non-reflective treatment developed by Giles (1991). A set of mean (mixed-out) variables can be obtained by applying the mean value theorem to each side of the above equalities as follows:

$$\widehat{\lambda_{1,i}} = \frac{\int_{y_i} \lambda_1 (\rho u)\, dy}{\int_{y_i} \rho u dy} \tag{8.5a}$$

$$\widehat{\lambda_{2,i}} = \frac{\int_{y_i} \lambda_i (\rho u^2 + p)\, dy}{\int_{y_i} (\rho u^2 + p)\, dy} \tag{8.5b}$$

$$\widehat{(\lambda_3 v)_i} = \frac{\int_{y_i} \lambda_3 (\rho u v)\, dy}{\int_{y_i} \rho u dy} \tag{8.5c}$$

$$\widehat{(\lambda_4 w)_i} = \frac{\int_{y_i} \lambda_4 (\rho u w)\, dy}{\int_{y_i} \rho u dy} \tag{8.5d}$$

$$\widehat{\lambda_{5,i}} = \frac{\int_{y_i} \lambda_5 (\rho u h)\, dy}{\int_{y_i} \rho u h dy} \tag{8.5e}$$

where $(i = 1,2)$ denotes the index of the corresponding interface. The mixed-out variables must be equal across an interface. If there is a difference on the value of these mixed-out variables across the interface, this difference is taken as a characteristic jump disturbance. The non-reflective treatment will pass the disturbance across the interface according to the adjoint characteristic propagation and the local adjoint solution will be corrected until driving this difference to zero. The adjoint characteristic jumps, in a vector form, are defined based on the primitive adjoint variables as follows:

$$\tilde{\lambda_c} = S^T \lambda \tag{8.6}$$

where the subindex c denotes the characteristic jump and S is the matrix that diagonalizes the Jacobian matrix $A = \frac{\partial F}{\partial U} = S \Lambda S^{-1}$, being Λ a diagonal matrix which diagonal

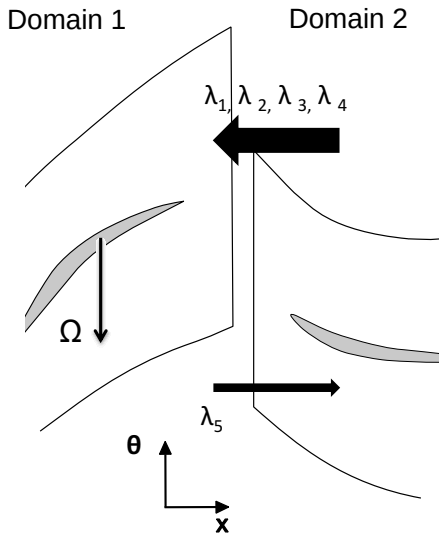


Figure 8.1: Rotor-stator interface with adjoint characteristics $\widetilde{\lambda}_{ci}$.

elements are eigenvalues of the matrix A. The scalar form of the above equation is given below:

$$\widetilde{\lambda}_{c1} = \frac{\widetilde{\lambda_5}}{u} + \widetilde{\lambda}_4 + \left[u - \frac{\left(u^2+v^2+w^2\right)^2}{2u} \right] \widetilde{\lambda}_1 \tag{8.7a}$$

$$\widetilde{\lambda}_{c2} = \widetilde{\lambda_2} + \frac{v}{r}\widetilde{\lambda}_1 \tag{8.7b}$$

$$\widetilde{\lambda}_{c3} = \widetilde{\lambda_3} + w\widetilde{\lambda}_1 \tag{8.7c}$$

$$\widetilde{\lambda}_{c4} = \widetilde{\lambda_5} + (u+c)\,\widetilde{\lambda}_4 + vr\widetilde{\lambda}_2 + w\widetilde{\lambda}_3 + (h+uc)\,\widetilde{\lambda}_1 \tag{8.7d}$$

$$\widetilde{\lambda}_{c5} = \widetilde{\lambda_5} + (u-c)\,\widetilde{\lambda}_4 + vr\widetilde{\lambda}_2 + w\widetilde{\lambda}_3 + (h-uc)\,\widetilde{\lambda}_1 \tag{8.7e}$$

where c is the speed of sound, h is the total enthalpy, r is the radial coordinate and u, v, w are the velocity components in the axial, circumferential and radial direction respectively. The five characteristic jumps, illustrated in Figure 8.1, are defined as follows:

- $\widetilde{\lambda}_{c1}$: one adjoint entropy disturbance propagating upstream at a speed of $-u$.
- $\widetilde{\lambda}_{c2}$ and $\widetilde{\lambda}_{c3}$: two adjoint vorticity disturbances propagating upstream at a speed of $-u$

- $\widetilde{\lambda}_{c4}$: one adjoint acoustic disturbance propagating upstream at a speed of $-u-c$.
- $\widetilde{\lambda}_{c5}$: one adjoint acoustic disturbance propagating downstream at a speed of $-u+c$

The exit of domain 1 (Figure 8.1) has four incoming adjoint characteristics $\left(\widetilde{\lambda}_{c2},\widetilde{\lambda}_{c3},\widetilde{\lambda}_{c4},\widetilde{\lambda}_{c5}\right)$ and the inlet of domain 2 has only one incoming characteristic $\widetilde{\lambda}_{c1}$. The values of these adjoint characteristics are calculated from the differences in the mixed-out quantities across the interface using the equations below:

$$\widetilde{\lambda_1} = \widehat{\lambda_{1,1}} - \widehat{\lambda_{1,2}} \tag{8.8a}$$

$$\widetilde{\lambda_2} = \frac{\widehat{(\lambda_2 v)_1} - \widehat{(\lambda_2 v)_2}}{v} \tag{8.8b}$$

$$\widetilde{\lambda_3} = \frac{\widehat{(\lambda_3 w)_1} - \widehat{(\lambda_3 w)_2}}{w} \tag{8.8c}$$

$$\widetilde{\lambda_4} = \widehat{\lambda_{4,1}} - \widehat{\lambda_{4,2}} \tag{8.8d}$$

$$\widetilde{\lambda_5} = \widehat{\lambda_{5,1}} - \widehat{\lambda_{5,2}} \tag{8.8e}$$

The values of the outgoing adjoint characteristics are calculated from their local values. Once the characteristic jumps are obtained for each domain, they can be converted back to primitive adjoint variables and be used to update the local adjoint solution at each side of the interface as follows:

$$\lambda = \left(S^T\right)^{-1} \tilde{\lambda_c} \tag{8.9}$$

In a scalar form:

$$\lambda_1 = \frac{k-1}{c^2}\left(\frac{\widetilde{\lambda}_{c5}}{2} + \frac{\widetilde{\lambda}_{c4}}{2} - u\widetilde{\lambda}_{c2} - vr\widetilde{\lambda}_{c3} - w\widetilde{\lambda}_{c1}\right) \tag{8.10a}$$

$$\lambda_2 = -\frac{v}{r}\widetilde{\lambda}_1 + \widetilde{\lambda}_{c3} \tag{8.10b}$$

$$\lambda_3 = -w\widetilde{\lambda}_1 + \widetilde{\lambda}_{c1} \tag{8.10c}$$

$$\lambda_4 = -u\widetilde{\lambda_1} + \frac{1}{2c}\left(-\widetilde{\lambda}_{c5} + \widetilde{\lambda}_{c4}\right) \tag{8.10d}$$

$$\lambda_5 = \frac{W^2}{2}\widetilde{\lambda}_1 + \frac{u}{2c}\left(\widetilde{\lambda}_{c5} - \widetilde{\lambda}_{c4} + 2c\widetilde{\lambda}_{c2}\right) \tag{8.10e}$$

8.2 Adjoint Harmonic Interface

With the continuous approach, the adjoint variable has the same harmonic form as the flow variable in the physical time domain. This allows the development of an adjoint multi-bladerow treatment that has the same working method as the flow harmonic interface treatment, in a similar way to the adjoint mixing-plane is based on the flow mixing-plane treatment. The adjoint harmonic interface treatment implemented in this work is an extension of the flow harmonic interface developed by Rahmati et al. (2013) and it follows the same two-steps strategy:

1. A time-trace signal of the adjoint variable is built at both sides of the interface (Time-trace reconstruction)

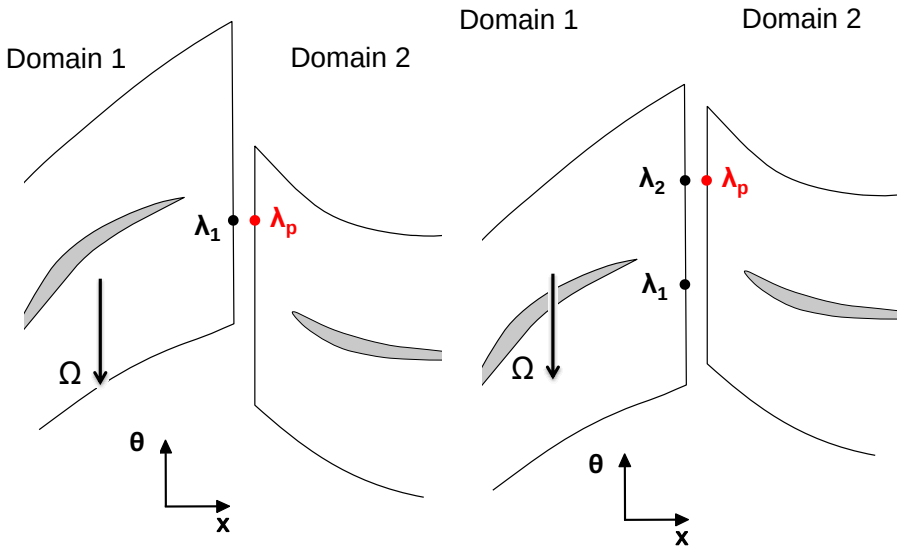
2. The different harmonic terms of the adjoint variable are separated and only those corresponding to the first harmonic are retained (Fourier filtering)

Time-trace reconstruction

A time trace of the adjoint variable is to be created. First, the relative position of a point p from domain 2 at the interface with respect to domain 1 is calculated at different time instances based on its initial position, rotational speed, relative motion and time instant (Figure 8.2). The time-trace for point p can be created either in the physical time domain or the adjoint time domain. The value of the adjoint variable at the point p, at different time instances can be given as follows:

$$\lambda_p\left(t = t_{p_i}\right) = \lambda_i \ \ (i = 0, 1, 2, ..., n) \tag{8.11}$$

where λ_i is the instant value of the adjoint variable in domain 2 located at the point of the interface that is coincident with p at the time instance i. If the same period of



(a) Time instance $t = t_1$ (b) Time instance $t = t_2$

Figure 8.2: Relative position between domains at two time instances t_1 and t_2 (physical time domain)

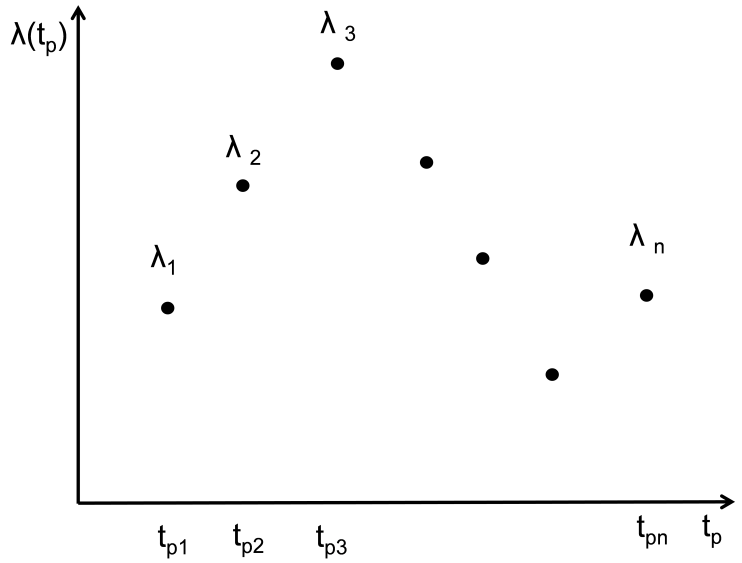
unsteadiness is considered, a time-trace of the adjoint variable for the same point in the adjoint time domain is given by:

$$\lambda_p\left(t = t_{a_i}\right) = \lambda_{n-i} \ (i = 0, 1, 2, ..., n) \tag{8.12}$$

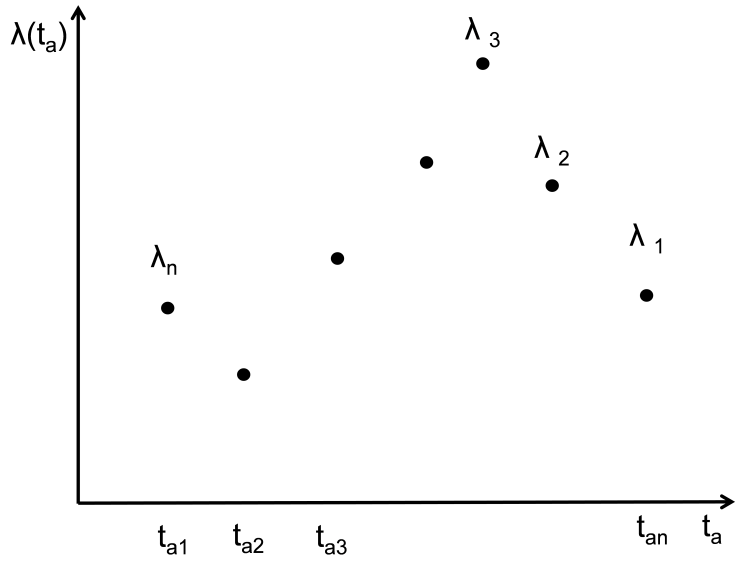
Both time traces will show a mirrored shape over a period of unsteadiness, due to the periodicity of the adjoint variable. This is shown in Figure 8.3.

Fourier filtering

Once a time trace is created for all the points at the interface of each domain, the components of the adjoint variable that correspond to the first harmonic are extracted while the rest of the terms are filtered out. This is effectively done by Fourier transformation. Irrespectively of the time domain used, the same coefficients will be obtained. As an example, the harmonic and time-averaged components of the adjoint variable can be



(a) Physical time domain



(b) Adjoint time domain

Figure 8.3: Time-traces of the adjoint variable $\lambda(t_a)$ and $\lambda(t_p)$ over a period of unsteadiness

calculated based on the physical time domain as follows:

$$\lambda_A = \frac{\omega}{\pi} \sum_{i=1}^{n} (\lambda_i) \sin\left(\omega t_{p_i}\right) \triangle t_{p_i} \tag{8.13a}$$

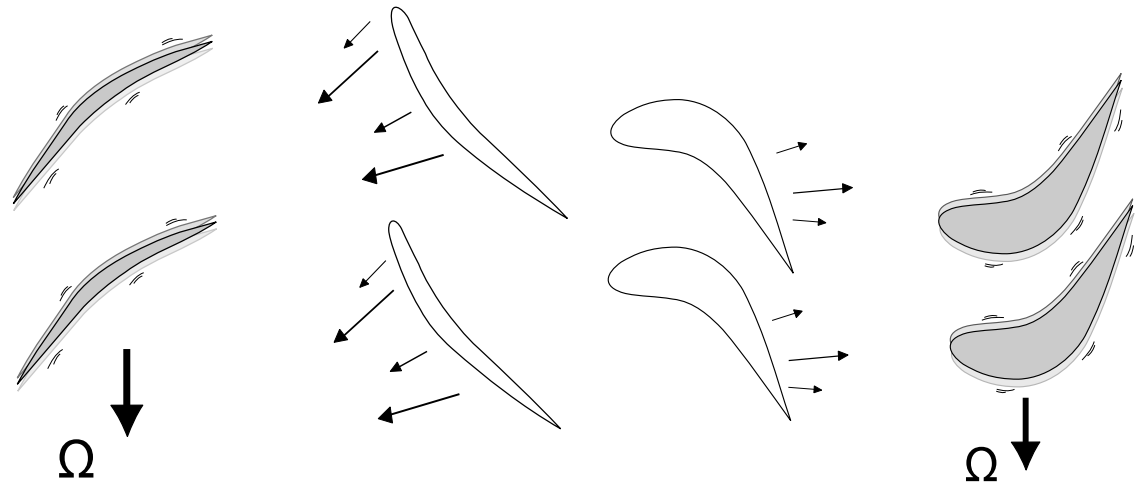
$$\lambda_B = \frac{\omega}{\pi} \sum_{i=1}^{n} (\lambda_i) \cos\left(\omega t_{p_i}\right) \triangle t_{p_i} \tag{8.13b}$$

$$\bar{\lambda} = \frac{\omega}{2\pi} \sum_{i=1}^{n} (\lambda_i) \triangle t_{p_i} \tag{8.13c}$$

The harmonic form of the adjoint variable leads to a very simple interface treatment that is essentially an extension of the unsteady flow harmonic interface method. The procedure steps shown above are directly and equally applicable to a physical harmonic flow variable as well as to an adjoint harmonic variable in both time domains.

8.3 Validation of the Adjoint Harmonic Interface

This section aims to demonstrate the validity of the adjoint harmonic interface treatment. The effectiveness of the harmonic interface treatment is very important as adjacent bladerows can have an impact in the damping of a vibrating blade (Rahmati et al., 2013). The only parameter in the objective function that is directly linked to the harmonic component of the adjoint variable is the work sum. A small perturbation in an adjacent bladerow may cause an unsteady reflection that will affect the work sum of the vibrating blade. In a compressor or turbine single stage, strong reflections are expected from downstream stator blade for a vibrating compressor rotor blade and from upstream stator blade for a turbine rotor blade as discussed by He (2012). In a multi stage configuration strong reflections are also expected from adjacent upstream or downstream stage. These reflections (Figure 8.4) are captured and exchanged between adjacent bladerows by the flow solver through the flow harmonic interface treatment. In order to perform a valid shape optimization of adjacent rows using the adjoint method, the adjoint harmonic interface treatment should be able to capture the work sum sensitivity of the vibrating blade to changes in the geometry of adjacent bladerows.



(a) Compressor stage (b) Turbine stage

Figure 8.4: Unsteady reflections from adjacent stators

For this validation, the work sum gradient at the surface of the vibrating blade is calculated using the direct finite difference method and its values are compared with those predicted by the adjoint method. The objective function for validation purposes is taken to be the work sum:

$$I = W \tag{8.14}$$

And the gradient of the objective function by a direct finite difference is:

$$\frac{dI}{d\alpha} = \frac{I(\alpha + h) - I(\alpha)}{h} \tag{8.15}$$

The validations are performed for two case studies: the DLR compressor stage and the industrial LP turbine stage previously presented in Chapter 5 (Figures 5.9 and 5.10). Three validations are carried out for each of the two cases. In the first case, a small hump perturbation is applied to the rotor blade. This will indicate the overall accuracy of the adjoint method on predicting the work sum gradient to rotor geometry changes. In the second and third case, a small hump perturbation is only applied to the stator blade. The adjoint harmonic interface treatment is only enabled on the third case. This means that there are no acoustic reflections going through the stator-rotor interface on the second

Gradients	Adjoint	Finite difference	% error
Rotor	-2.12×10^{-3}	-2.27×10^{-3}	-6.6%
Stator (without harmonic interface)	-1.91×10^{-5}	-1.98×10^{-5}	-3.53%
Stator (with harmonic interface)	-3.41×10^{-4}	-3.68×10^{-4}	-7.34%

Table 8.1: Work sum gradients. DLR compressor stage

Gradients	Adjoint	Finite difference	% error
Rotor	-5.12×10^{-3}	-5.27×10^{-3}	-2.84%
Stator (without harmonic interface)	-3.01×10^{-5}	-3.1×10^{-5}	-2.91%
Stator (with harmonic interface)	-2.21×10^{-3}	-2.38×10^{-3}	-7.14%

Table 8.2: Work sum gradients. LP turbine stage

case and that any change in the stator geometry that could affect the vibrating blade is only due to the changes in average flow field through the steady mixing plane. Table 8.1 and 8.2 show the work sum gradients obtained by the adjoint and direct finite difference for the three cases. The relative errors between these two values are also given.

By looking at both tables, it can be observed that the adjoint and finite difference methods are in good agreement, as their differences are always less than 10%. Another important point is that there is a disparity of 1-2 orders of magnitude between the calculated gradients of the second and third case. The smallest of both values corresponds to the case where no harmonic interface is enabled. This demonstrates that the adjoint harmonic interface is capturing the sensitivities from the aeroelastic reflections. Otherwise, these values would have at least the same order of magnitude

8.4 Summary

The multi-bladerow capability for unsteady design optimization is one of the distinctive features of this work, as current adjoint methods for turbomachinery were limited to a mixing-plane treatment for steady flows. The new adjoint principle based upon the harmonic form of the adjoint variable illustrated in Chapter 7 enables the development of a new and consistent (the first of its kind) adjoint interface method that works as an extension of the one implemented for the flow solution system.

The work sum gradient is used to validate and demonstrate the accuracy of the adjoint harmonic interface treatment, capturing the sensitivities of the work sum on the vibrating blade to changes in adjacent bladerows.

Chapter 9

Adjoint based Concurrent Design Optimization System

9.1 Introduction

This chapter shows the integration of the system of unsteady adjoint equations derived in Chapter 7 in an automated optimization method that performs concurrent aeroelastic and aerodynamic shape optimization for turbomachinery stages. The objective function, FE-CFD interaction and shape parameterization methods are the same as those implemented in the non-gradient method presented in Chapter 5. The remaining elements of the adjoint optimization system are explained next.

9.2 Gradient Calculation

The gradient expression defines the sensitivity parameters of the objective function due to geometry changes by the design variables. For concurrent aerodynamic and aeroelastic design optimization, these sensitivities are obtained by solving the set of three steady-like equations, which gives the time-averaged and the two harmonic in-phase and out-phase sensitivities. This principle for gradient based concurrent design optimization was first

proposed by He and Wang (2011). Particularly for the adjoint method, the gradient equation (A.7) can be written free from the flow variable sensitivity as follows:

$$\frac{dI}{d\alpha} = \int_0^T \int_{\partial V} \left(\frac{\partial M}{\partial \alpha}\right) dS + \int_0^T \int_{\partial V} M\tilde{d}s + \int_0^T \int_V \lambda^T f dv \tag{9.1}$$

The geometric gradients $\int_0^T \int_{\partial V} \left(\frac{\partial M}{\partial \alpha}\right) ds$ and $\int_0^T \int_{\partial V} M\tilde{d}s$ are calculated using the complex variable method (Vatsa, 2004). The f term of the last integral $\int_0^T \int_V \lambda^T f dv$ is defined as the negative of the flow residual perturbation due to a change in the design variables:

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

The calculation of this term is done by perturbing the mesh from a change to the design variable, calculating the flux perturbations for each bounding surface and collecting them for each control volume to form the flow residual perturbation $\frac{\partial R}{\partial \alpha}$. Based on the Nonlinear Harmonic Phase Solution Method, the negative of the flow residual perturbation, f, can be decomposed in the same manner as the unsteady flow variable, thus assuming its harmonic form as follows:

$$f = \bar{f} + f_A \sin \omega t + f_B \cos \omega t \tag{9.2}$$

Hence, at the three phase angles:

$$f\left(\omega t = -\frac{\pi}{2}\right) = f_{-1} = \bar{f} - f_A \tag{9.3a}$$

$$f\left(\omega t = 0\right) = f_0 = \bar{f} + f_B \tag{9.3b}$$

$$f\left(\omega t = \frac{\pi}{2}\right) = f_1 = \bar{f} + f_A \tag{9.3c}$$

And the time-averaged and harmonic terms of Equation 9.2 are obtained from above as follows:

$$f_A = \frac{f_1 - f_{-1}}{2} \tag{9.4a}$$

$$f_B = f_0 - \frac{f_1 + f_{-1}}{2} \tag{9.4b}$$

$$\bar{f} = \frac{f_1 + f_{-1}}{2} \tag{9.4c}$$

Based on this and the harmonic assumption of the adjoint variable (Equation 7.4), the term $\int_0^T \int_V \lambda^T f \, dv$ in Equation 9.1 can be expanded as:

$$\int_0^T \int_V \lambda^T f \, dv = \int_0^T \int_V \left[\bar{\lambda} + \lambda_A \sin\omega t + \lambda_B \cos\omega t\right] \left[\bar{f} + f_A \sin\omega t + f_B \cos\omega t\right] dv \tag{9.5}$$

Integration of the above in time over one period of blade vibration gives:

$$\int_V T \cdot \left(\bar{\lambda}\bar{f} + \frac{\lambda_A f_A}{2} + \frac{\lambda_B f_B}{2}\right) dv \tag{9.6}$$

where the harmonic and time-averaged components of both the adjoint and flow residual perturbation are calculated based on Equations 7.22 and 9.4. Overall, the calculation of the gradient of the objective function can be summarised in the following steps:

1. The overall mesh perturbation is calculated collecting the hump perturbation values for each design variable α_i.

2. The geometric gradients $\int_0^T \int_{\partial V} \left(\frac{\partial M}{\partial \alpha}\right) dS$ and $\int_0^T \int_{\partial V} M d\tilde{S}$ are obtained using the complex variable method.

3. The flux residual perturbations that result from the changes to the mesh are calculated for each control volume, yielding $\frac{\partial R_i}{\partial \alpha_i}$. This process is repeated for each of the three phase angles of unsteadiness.

4. For each phase, the flow residual perturbation due to a change in the design variable α_i at every control volume, j, is multiplied by the adjoint variable vector λ.

Next, the product is summed up for n control volumes to calculate the opposite of the flow residual perturbation, f_i as follows:

$$f_i = -\sum_{j}^{N} \lambda \frac{\partial R_j}{\partial \alpha_i} \tag{9.7}$$

5. The total gradient (Equation 9.1) is calculated summing up all the terms from steps 2-4.

The overall cost of this process is proportional to the number of design variables. However the time cost is still much smaller than the time needed to solve the flow equations or adjoint equations, even for thousands of design variables, due to the non-iterative nature of the gradient calculation process.

9.3 Search Method: Steep Descent

Having calculated the gradient of the objective function to a design variable, in order to find the minimum of a function, a search step has to be applied. This is done with the steep descent method for which a small step is taken in the negative direction of the gradient as follows:

$$\triangle \alpha = -s \cdot \frac{dI}{d\alpha} \tag{9.8}$$

where s is the step size. The process is repeated for each design cycle until the gradient is zero and the optimization has converged. The gradient of the objective function can be approximated to finite differences if $\triangle \alpha$ is small enough:

$$\frac{\triangle I}{\triangle \alpha} = \frac{dI}{d\alpha} \tag{9.9}$$

The change to the objective function, $\triangle I$, can be written as follows:

$$\triangle I = \frac{dI}{d\alpha} \triangle \alpha \tag{9.10}$$

Combining Equation 9.8 and Equation 9.10 allows eliminating the $\triangle\alpha$ term and expression of the change to the objective function, $\triangle I$, as a function of the step size and gradient as follows:

$$\triangle I = -s\frac{dI}{d\alpha}\cdot\frac{dI}{d\alpha} \tag{9.11}$$

If the gradient is not zero, the negative step will always decrease the objective function $\triangle I < 0$. The optimum size of the step is normally calculated with a line search approach. However, this requires extra computational cost for solving the flow equations. In this work, an arbitrary step size, s, is used first to calculate the perturbations to the coordinates of all mesh points. A change in the design variable vector will change the camber line due to the 2D Hicks Henne function to a maximum of $\delta_{r\theta_{max}}$ in a point of the blade surface. This value is scaled to a maximum value (e.g 0.05 mm) specified by the user. Based on this, the final step size s that is applied is scaled to match the specified $\delta_{r\theta_{max}}$.

There is another factor, k, that controls the step size over iterations. This is implemented in order to avoid persistent oscillations in cases where strong constraints may cause the search to go back and forth between two regions in the design space. A controlled step size can help the optimization method to find a compromise solution by taking smaller steps progressively. This factor, k, reduces the step size (after being scaled according to $\delta_{r\theta_{max}}$) of the next iteration $i+1$ as follows:

$$s_{i+1} = k \cdot s_i \tag{9.12}$$

where i denotes the optimization cycle index. Typical values of this factor range from 0.85 to 0.95 in order to avoid a slow convergence while ensuring that the optimization will find the absolute minimum smoothly.

9.4 Design Optimization Process

Figure 9.1 shows the overall structure of a gradient-based design optimization system using the unsteady adjoint method. The overall process is summarised in the following

steps:

1. The optimization settings are read from the input file. The baseline design geometry and flow field are initialised.

2. The flow solutions at the three phases are calculated for the baseline design.

3. The unsteady adjoint flow field is initialised and the unsteady adjoint equations at the three phases are solved.

4. A mesh perturbation is applied for each design variable and the gradient of the objective function to each design variable is calculated.

5. A step is taken in the negative direction of the gradient calculated in step 4.

6. The geometry perturbations are scaled to the specified step size and the blade geometry is updated.

7. The mode shape and vibration frequency of the new blade geometry are obtained by FEA.

8. The flow solution at the three phases is calculated for the new geometry. Its objective function is also calculated.

Steps 3-8 form an adjoint optimization cycle. This process is repeated until a maximum number of iterations (specified by the user) is reached or a convergence criteria is met. The computational cost of each cycle rarely exceeds three times the cost of solving the unsteady flow equations. In order to save some computational time, the adjoint and flow field of the previous design cycle can be used to initialise the adjoint and flow field of the next optimization cycle.

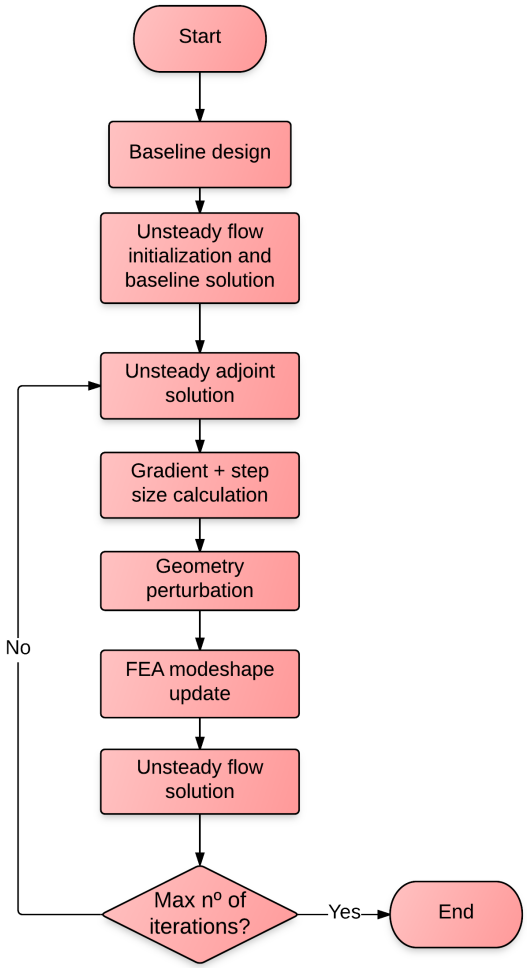


Figure 9.1: Flowchart of the unsteady adjoint optimization system

9.5 Summary

The specific elements of an optimization system based on the adjoint method are presented in this chapter. The gradient of the objective function and its sensitivities for aerodynamics and aeroelastic design optimization is calculated from the set of three steady-like flow and adjoint equations. Once the gradient is calculated, the steepest descent method is used to determine the search direction towards an optimum design. Overall, the computational cost of this process is proportional to the number of design variables but it is generally small compared to the cost of solving the system of flow and adjoint equations.

Chapter 10

Results and Optimization Case Studies using Adjoint

The geometry configurations of the cases that are going to be used for the demonstration of the unsteady adjoint optimization system are the same as those presented in Chapter 5: The NASA rotor 67, the DLR transonic compressor stage and the industrial LP turbine stage.

Table 10.1 shows the optimization and computational settings for the three case studies. For each blade that is optimized, 81 variables or hump perturbation nodes are defined, distributed over the blade span and chord in 10 equally spaced intervals, as shown in Figure 10.1. That means that for a full stage design optimization case, 162 design variables are used in total. Based on the objective functions (Equations 5.5 and 5.6) defined in Chapter 5, the next results are aimed to show the multi-component and multi-disciplinary capabilities of the optimization method built upon the new adjoint approach.

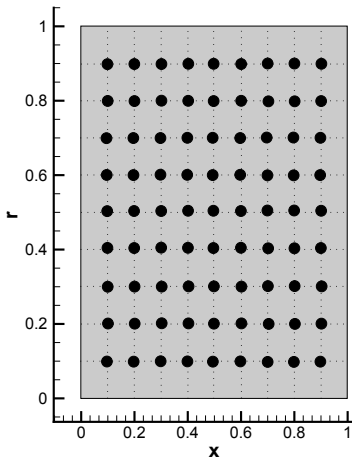


Figure 10.1: Hicks-Henne function perturbation nodes (reduced coordinates). 81 variables

Optimization case	NASA 67	DLR stage	LP turbine stage
Number of variables per bladerow	81	81	81
Number of processors	1	1	1
CPU time per cycle	1 h	4.5 h	5.5h
Mass flow rate constraint σ_1	1000	1000	100
Pressure ratio constraint σ_2	1000	1000	100
Entropy constraint (if applicable) σ_3	-	-	100
Work sum constraint (if applicable) σ_3	-	-	100

Table 10.1: Optimization settings and computational costs

10.1 Aeroelastic-only Optimization of Stator Blades

The first set of design optimizations to be presented are for two multi-bladerow cases: the DLR compressor stage and the industrial LP stage. This study involves the optimization of the stator blades only, so the performance of the adjoint harmonic interface treatment can be evaluated to see the effect that adjacent blades can have in the aeroelastic performance (section 8.3) of the rotor blade. The stator redesign is only aimed to

improve rotor blade log-dec without taking into account the aerodynamic performance. Hence, the aeroelastic-only objective function (Equation 5.6) with the entropy constraint disabled ($\sigma_3 = 0$) is used for both cases.

10.1.1 DLR transonic compressor stage

The computational domain of the DLR compressor stage is shown in Figure 5.9. The design optimization is carried out with 81 design variables on a stator blade and convergence is reached after 30 design cycles (Figure 10.2). The final values of the objective function parameters are shown in Table 10.2. The rotor aeroelastic log-dec is slightly increased from 2.81% to 2.92% (i.e 4% relative increase). The margin is small and not as large as what can be achieved with a rotor redesign. This is expected as the rotor work sum gradient with respect to a change of stator geometry for the DLR case was reported to be two orders of magnitude lower than the rotor's gradient in the validation test (Table 8.1). Despite there being no aerodynamic constraint, the aerodynamic performance is not decreased by the new redesign and the efficiency of the stage barely changes.

	Original	Optimized	Abs. change	Rel. change
Isentropic efficiency (%)	86.79	86.8	+0.01	+0.01%
Log-dec (%)	2.81	2.92	+0.11	+3.9%
Mass flow rate (kg/s)	16.95	16.97	+0.02	+0.12%
Pressure ratio	1.594	1.5935	-0.005	-0.03%

Table 10.2: Final values, absolute and relative differences. DLR stator redesign

The new stator blade's influence on the local work contours at the rotor blade surface for the first vibration mode is as shown in Figures 10.3 and 10.4. It can be seen how the two stable areas located before the shock wave on both the pressure and suction side are increased.

The changes to the shape of the stator blade are shown in Figure 10.5. The leading edge is slightly humped up while the rest of the profile is more significantly humped down. The

midspan area shows the biggest changes.

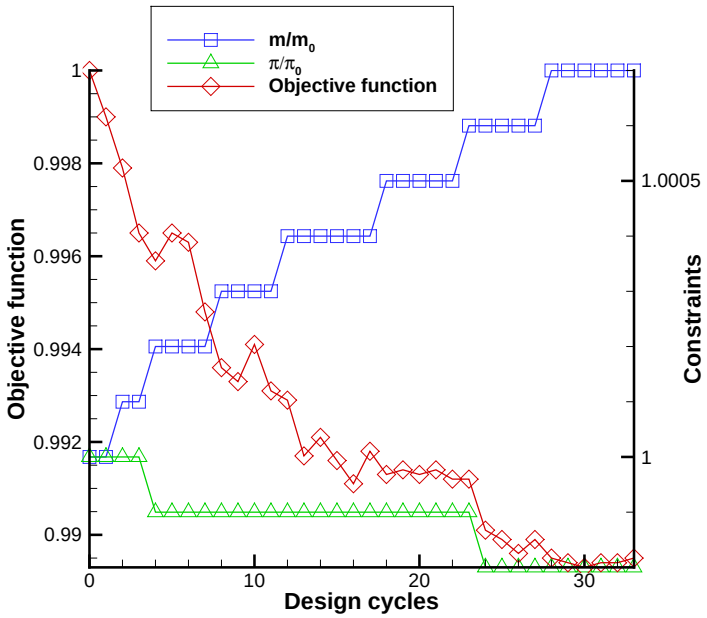


Figure 10.2: Convergence of objective function and constraints. DLR stator redesign

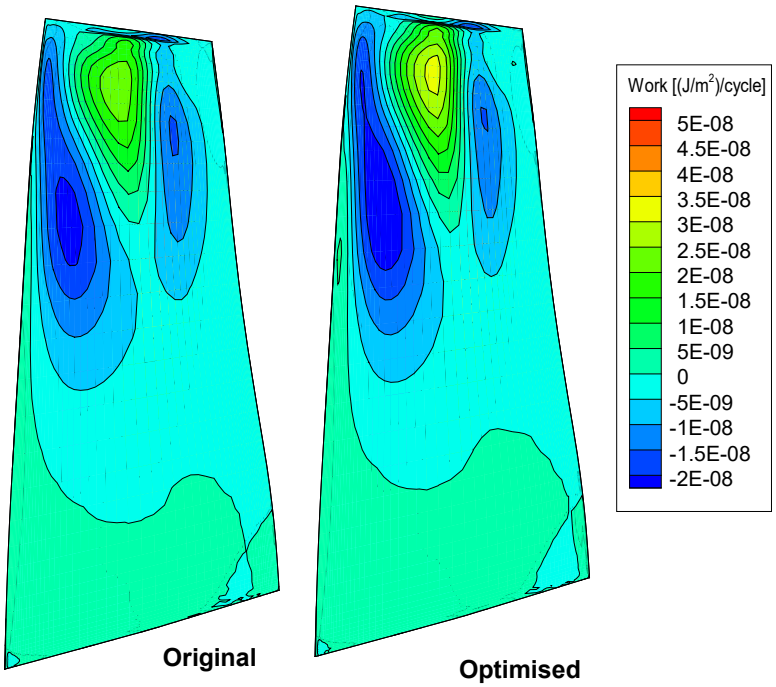


Figure 10.3: Local work contours on DLR blade (first bending mode). Pressure side

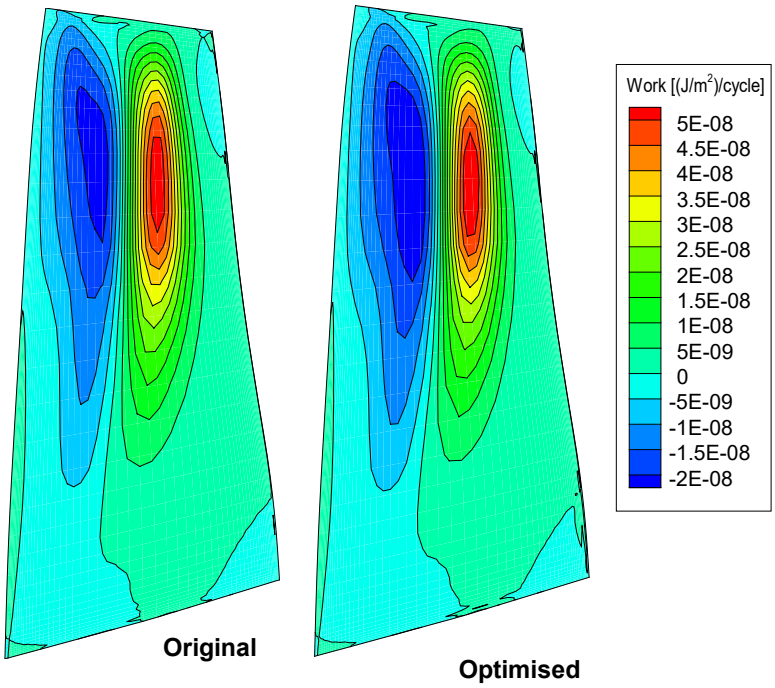
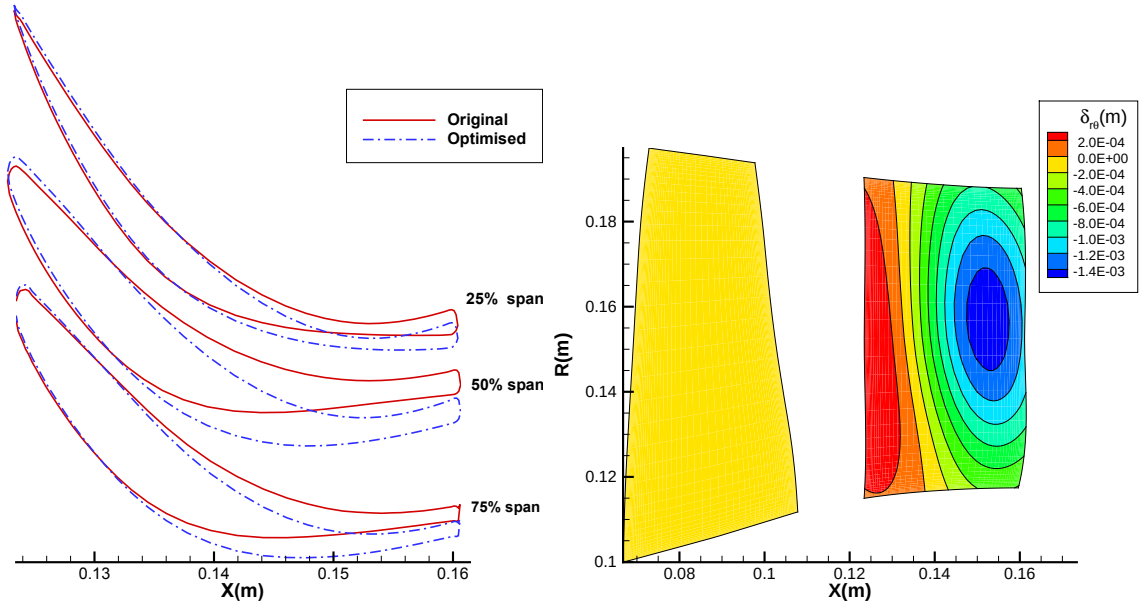


Figure 10.4: Local work contours on DLR blade (first bending mode). Suction side



(a) Stator profiles at 25%, 50% and 75% span

(b) Contours of tangential coordinate change, $\delta_{r\theta}$

Figure 10.5: Blade geometry comparison. DLR stator redesign

10.1.2 Industrial LP turbine stage

The second redesign of a stator blade is performed for the industrial LP turbine stage. Again, the design optimization is carried out with 81 design variables. Figure 10.6 shows that convergence is reached after 40 design cycles and the absolute and relative differences between original and optimized cases are given in Table 10.3. The relative improvement in rotor blade log-dec from the stator blade geometry changes is nearly 5% with a total increase of 0.19%. The entropy, however is increased for this case, thus reducing isentropic efficiency 0.16%.

The changes to the local work contours on the rotor blade surface for the first mode are shown in Figures 10.7 and 10.8. The work sum contours at the pressure side remain mostly unchanged. However the unstable gradient at the suction side is decreased before the shock. The stable area is also increased.

In terms of the changes to the stator blade, a small positive humping before midchord is applied (Figure 10.9) followed by a negative humping after midchord. The biggest values of humping are found around the midspan region.

	Abs. change	Rel. change
Isentropic efficiency (%)	-0.16	-0.18%
Log-dec (%)	+0.19	+4.85%
Mass flow rate (kg/s)	-0.044	-0.11%
Pressure ratio	-0.002	-0.08%

Table 10.3: Absolute and relative differences after convergence. LP turbine stator redesign

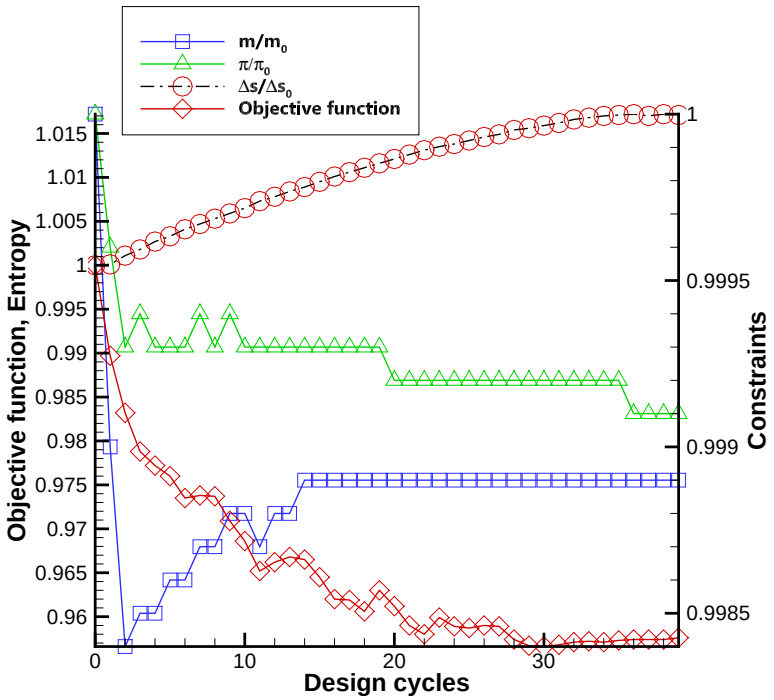


Figure 10.6: Convergence of objective function and constraints. LP turbine stator redesign

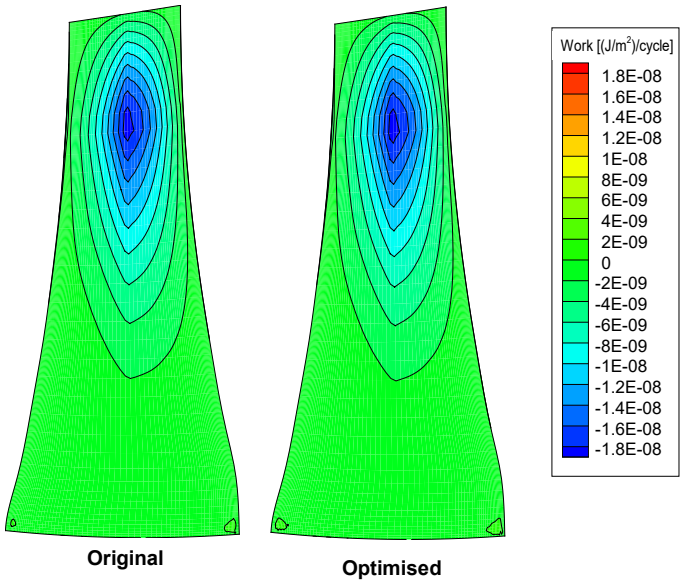


Figure 10.7: Local work contours on LP turbine blade (first bending mode). Pressure side

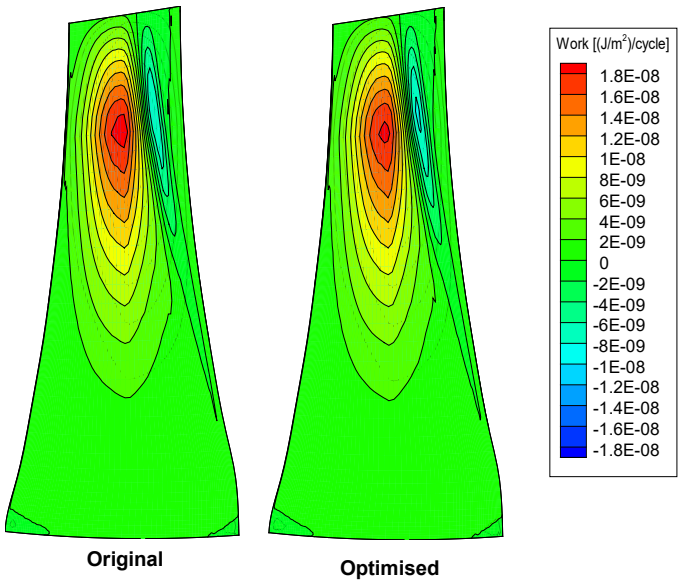
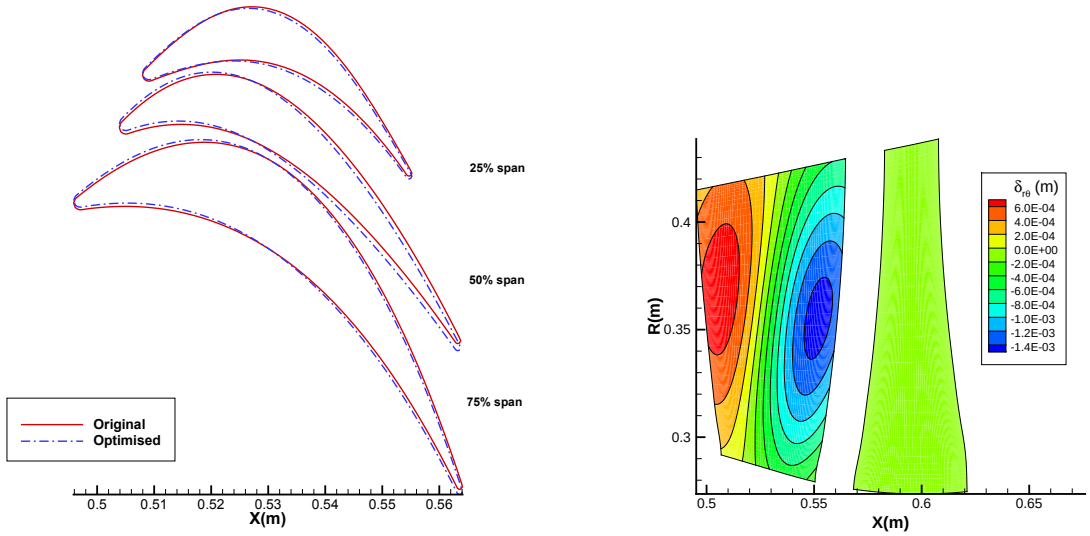


Figure 10.8: Local work contours on LP turbine blade (first bending mode). Suction side



(a) Stator blade profiles at 25%, 50% and 75% span

(b) Contours of tangential coordinate change, $\delta_{r\theta}$

Figure 10.9: Blade geometry comparison. LP turbine stator redesign

10.2 NASA Rotor 67 Aeroelastic-only Optimization

The next design optimization is performed for the NASA rotor 67 with an aeroelastic-only ($\sigma_3 = 0$) objective function based on Equation 5.6. 81 design variables are used to redesign the rotor and convergence is reached after 60 design cycles (Figure 10.10). The final values are given in Table 10.4. The aeroelastic improvement is achieved by weakening the shock wave and pushing it downstream as shown by the contours of static pressure at the blade surface (Figures 10.11 and 10.12). This is the same redesign strategy followed by the non-gradient method (section 5.8.1). However the adjoint method allows parameterizing the blade with much more variables which consequently leads to a further displacement of the shock wave. This translates to a better margin of improvement, where the log-dec is now increased from 3.12% to 4.44% (i.e nearly 43% relative improvement). The entropy generation rate after convergence is also decreased, increasing the isentropic an extra 0.5%, from 91.72% to 91.22%.

The local work contours on the blade surface for the first mode show an increase in stability before the shock region (Figures 10.13 and 10.14). The unstable area after the shock at the suction side is also increased but by a lesser extent.

In terms of the changes to the rotor blade (Figure 10.15), the largest changes are located in the midspan area, with a negative humping effect before midchord and an upward hump after midchord.

	Original	Optimized	Abs. change	Rel. change
Isentropic efficiency (%)	91.72	91.22	+0.5	+0.54%
Log-dec (%)	3.12	4.44	+1.32	+42.3%
Mass flow rate (kg/s)	33.37	33.35	-0.02	-0.06%
Pressure ratio	1.656	1.655	-0.001	-0.07%

Table 10.4: Final values, absolute and relative differences. NASA rotor 67

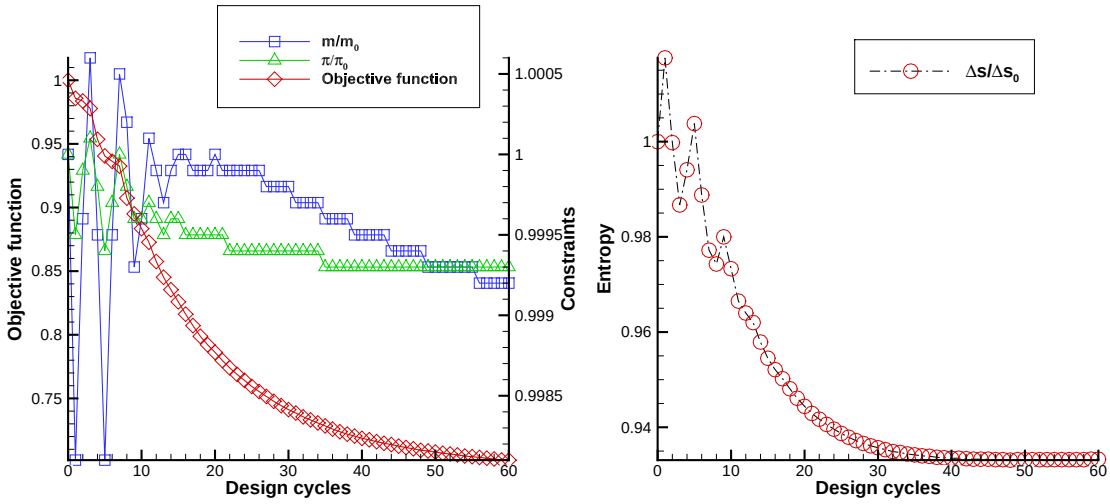


Figure 10.10: Convergence of objective function and constraints. NASA rotor 67

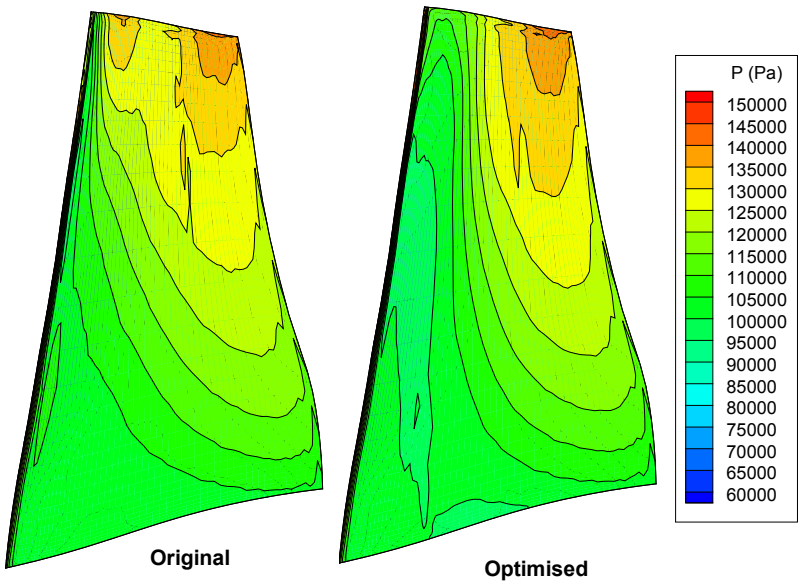


Figure 10.11: NASA rotor 67 pressure distribution contours. Pressure side

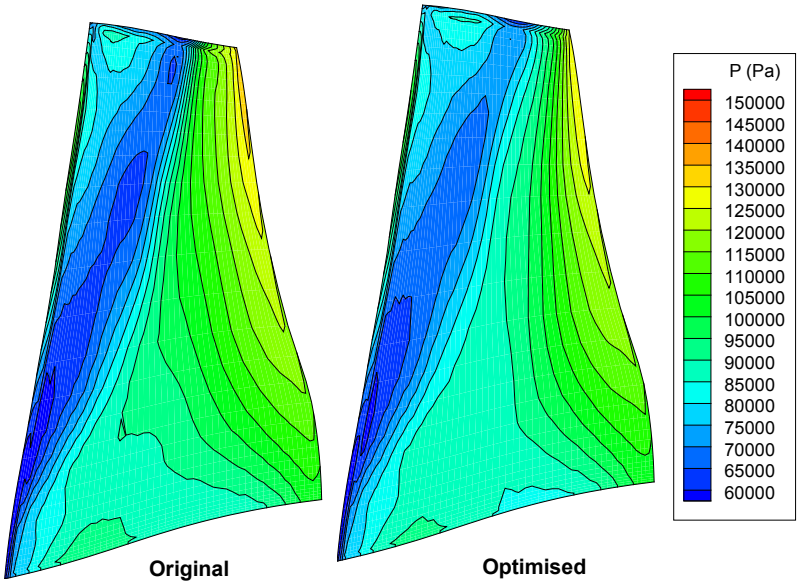


Figure 10.12: NASA rotor 67 pressure distribution contours. Suction side

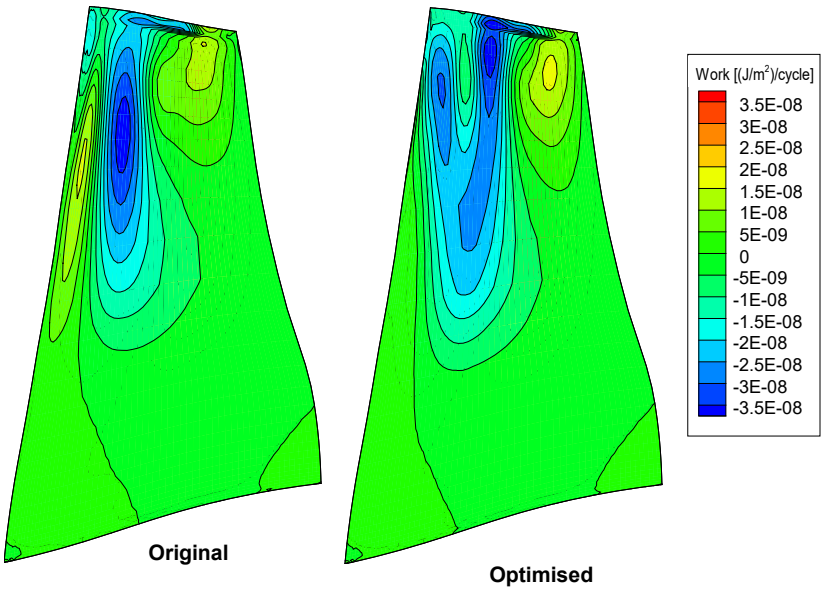


Figure 10.13: Local work contours on NASA rotor 67 (first bending mode). Pressure side

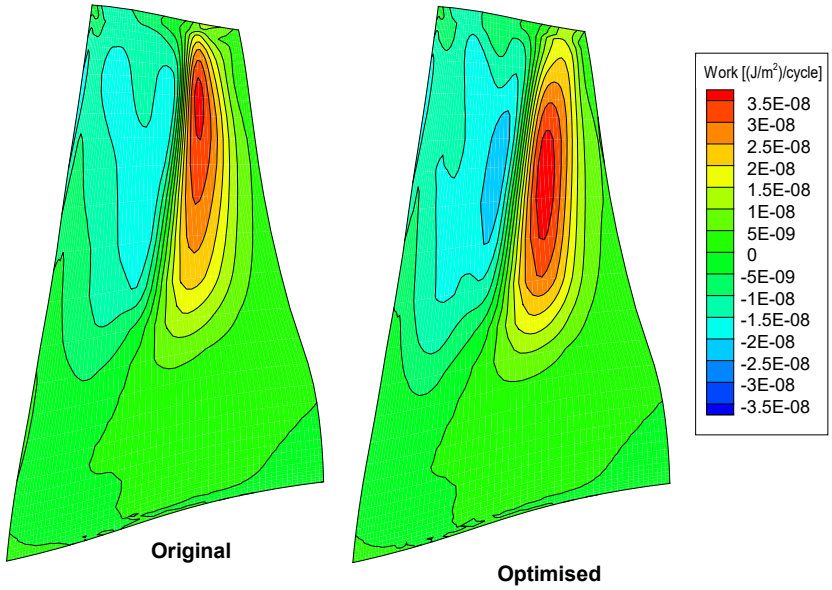
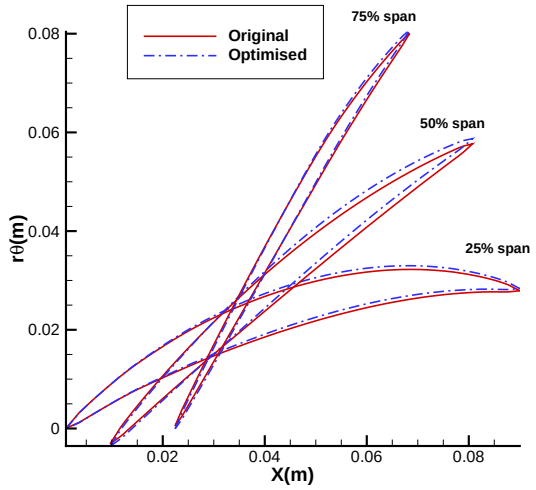
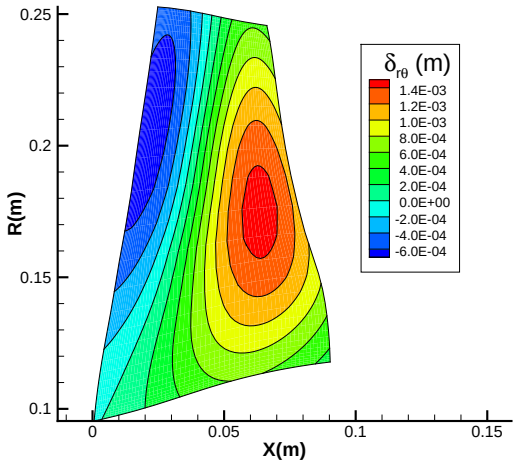


Figure 10.14: Local work contours on NASA rotor 67 (first bending mode). Suction side



(a) Rotor profile at 25%, 50% and 75% span



(b) Contours of tangential coordinate change, $\delta_{r\theta}$

Figure 10.15: Blade geometry comparison. NASA rotor 67

10.3 DLR Transonic Stage Aeroelastic-only Optimization

The redesign of the DLR transonic stage is also carried out with an aeroelastic-only objective function as simultaneous improvement of isentropic efficiency and rotor aeroelastic log-dec is expected based on the results from Chapter 5. The next two cases and their comparison aim to demonstrate the benefits of multi-component design optimization; adding the stator to the shape optimization loop and having a better control on the reflections from adjacent blades to the rotor blade. The first case is a rotor blade only design optimizations while the last case is a complete stage (stator and rotor) design optimization.

10.3.1 Aeroelastic-only Rotor Blade Design Optimization

The convergence of this design optimization case is shown in Figure 10.16a. The objective function is decreased by 30% and rotor blade log-dec is increased up to a final value of 4.23%, as shown in Table 10.5. As expected, the improvement in rotor blade log-dec also brings an improvement in aerodynamics, decreasing the entropy generation rate and increasing the optimized isentropic efficiency up to 87.2%.

	Original	Optimized	Abs. change	Rel. change
Isentropic efficiency (%)	86.79	87.2	+0.41	+0.47%
Log-dec (%)	2.81	4.23	+1.42	+50.5%
Mass flow rate (kg/s)	16.95	16.89	-0.06	-0.35%
Pressure ratio	1.594	1.592	-0.002	-0.13%

Table 10.5: Final values, absolute and relative differences. DLR rotor redesign

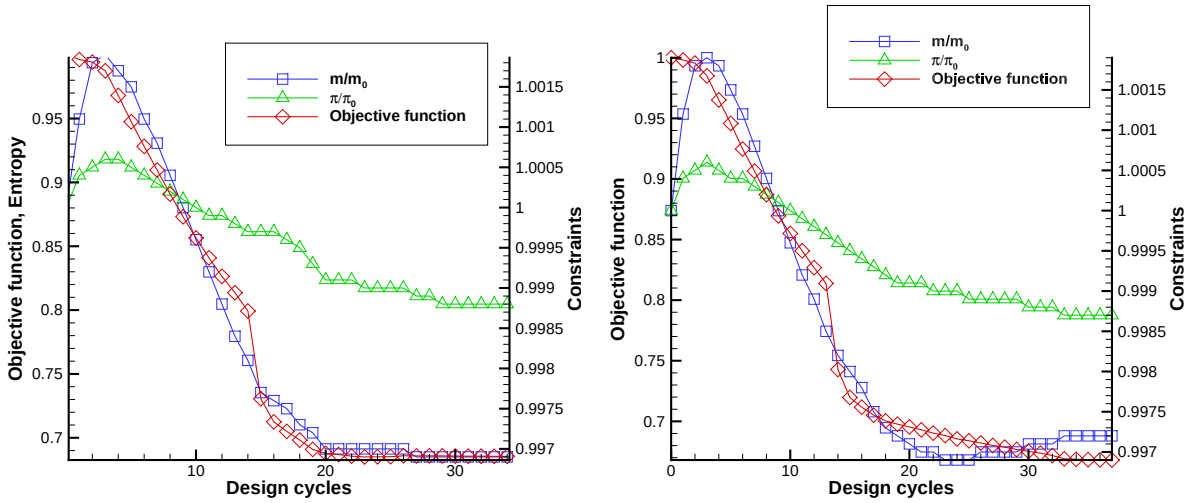
10.3.2 Aeroelastic-only Stage Design Optimization

Table 10.6 shows the final values of the design optimization, where convergence is reached after 45 design cycles (Figure 10.16b). The objective function drops its value below 0.7,

showing a relative improvement of 52% for log-dec, which increases its final value up to 4.28%. As expected, the isentropic efficiency is also increased up to 87.21%. The mass flow and pressure ratio changes are kept at reasonable levels by the constraints.

	Original	Optimized	Abs. change	Rel. change
Isentropic efficiency (%)	86.79	87.21	+0.42	+0.48%
Log-dec (%)	2.81	4.28	+1.47	+52.3%
Mass flow rate (kg/s)	16.95	16.9	-0.05	-0.29%
Pressure ratio	1.594	1.592	-0.002	-0.13%

Table 10.6: Final values, absolute and relative differences. DLR stage redesign



(a) "Rotor-only" optimization case

(b) "Rotor and stator" optimization case

Figure 10.16: Convergence of objective function and constraints. DLR stage redesign

10.3.3 Multi-component Optimization Comparison

Comparing both cases, an extra 0.05% (2% relative increase) of rotor blade log-dec improvement is obtained by slightly shaping the stator as it can be seen when comparing the work sum convergence of both cases in Figure 10.17. This extra amount is not significant but it is important to note that the work sum gradient for the stator blade was

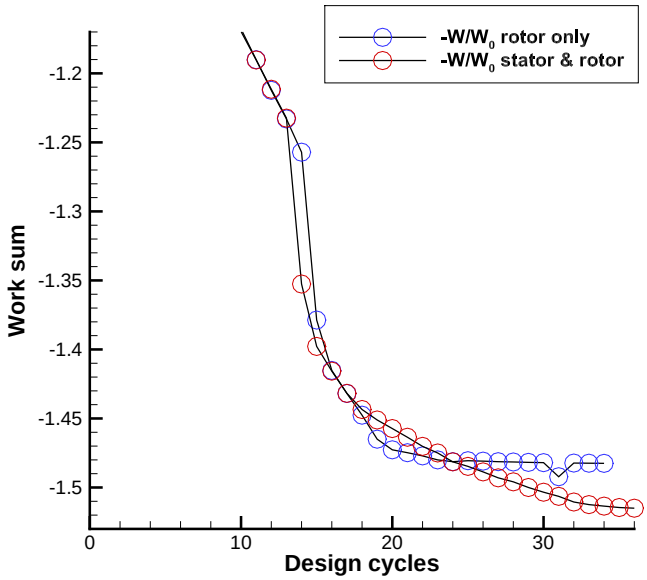


Figure 10.17: Work sum convergence comparison

reported two orders of magnitude lower than the rotor work sum gradient (Table 8.1). A bigger margin of improvement would be obtained in cases where the stator work sum gradient is closer in value to the rotor one (i.e the industrial LP turbine case). The local work contours at the rotor blade surface corresponding to the first mode of vibration are shown for both cases in Figures 10.18 and 10.19. The log-dec improvement comes from the stable gradients being significantly increased (specially on the pressure side) while the unstable area on the suction side is also decreased. The improvement can be observed to be slightly better for the full stage design optimization case.

The geometry changes to both stator and rotor blades are shown in Figures 10.20 and 10.21. The rotor is the blade that shows the largest changes and these are practically the same for both cases. The optimized rotor profile is humped up after midchord and slightly humped down before midchord. The largest changes are located around 25% span and 75% chord. For the full stage design optimization case, the stator blade only shows a small negative (down) humping spanwise, near the trailing edge

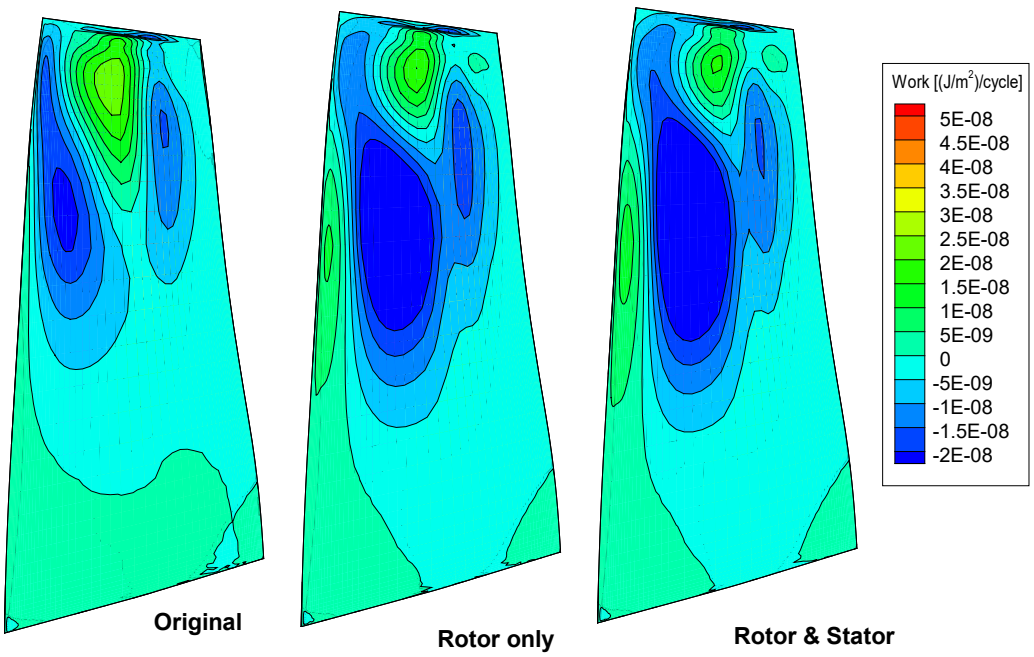


Figure 10.18: Local work contours on DLR blade (first bending mode). Pressure side

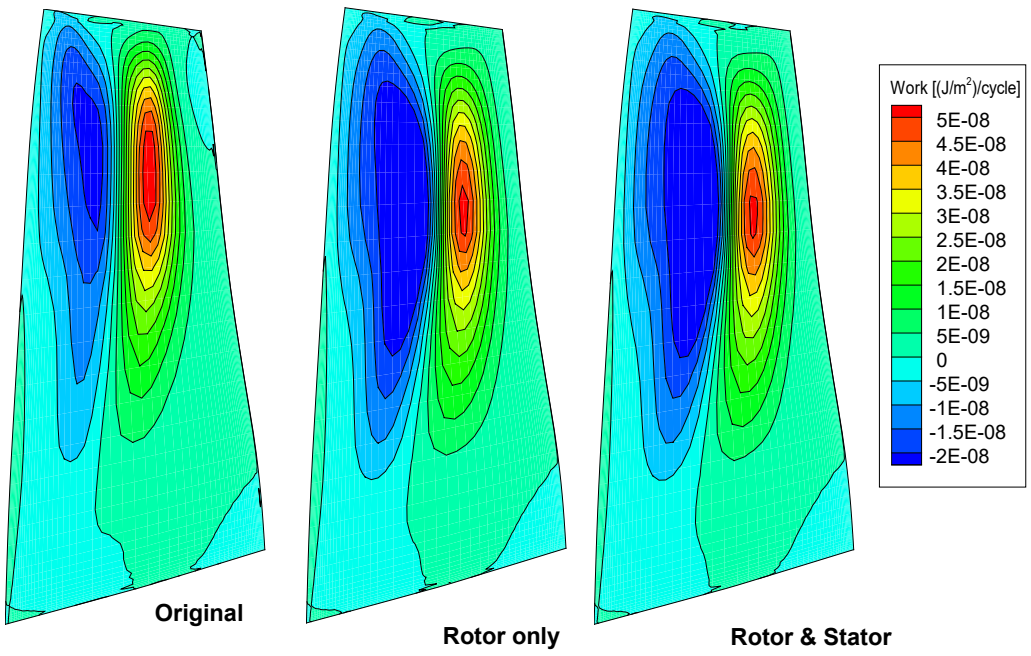


Figure 10.19: Local work contours on DLR blade (first bending mode). Suction side

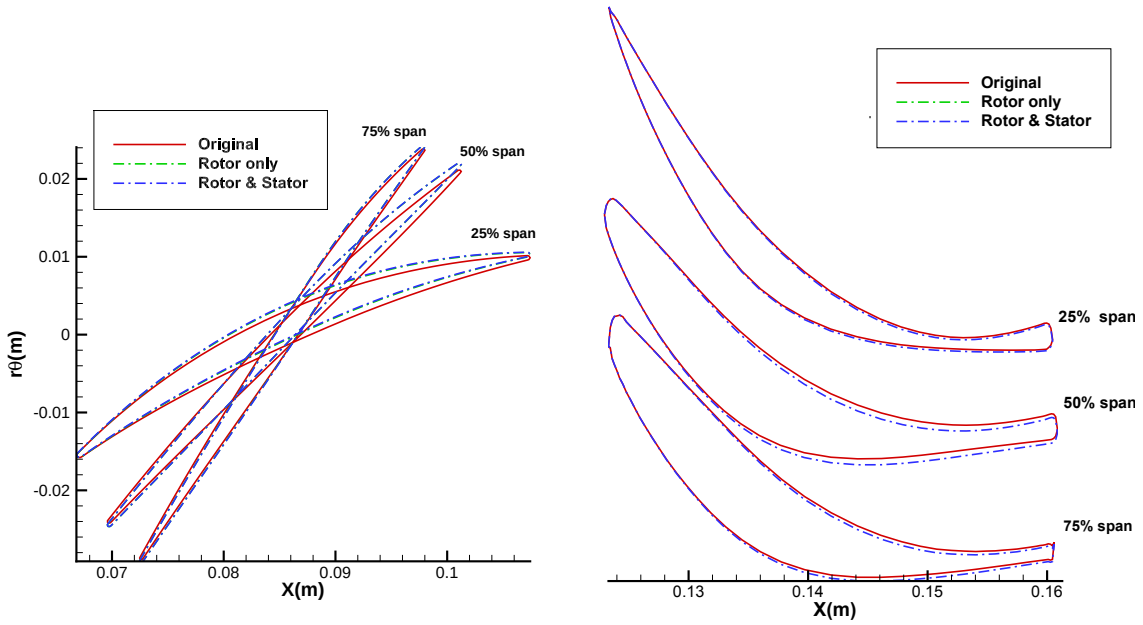
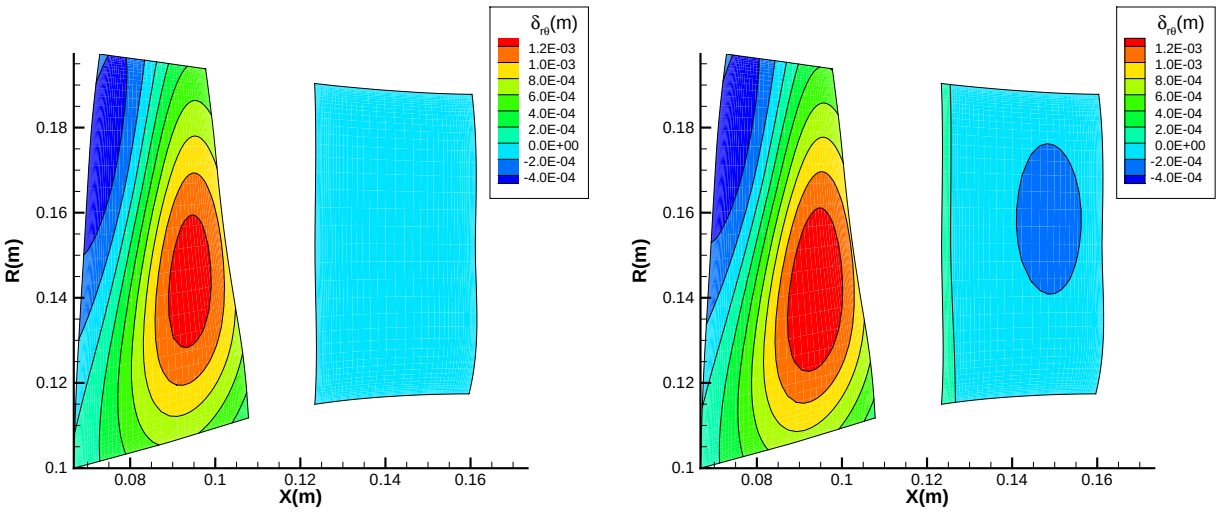


Figure 10.20: Stage profiles at 25%, 50% and 75% span. DLR stage redesign



(a) "Rotor-only" optimization case

(b) "Rotor and stator" optimization case

Figure 10.21: Contours of tangential coordinate change, $\delta_{r\theta}$. DLR stage redesign

10.3.4 Overall Aerodynamic Performance

In terms of aerodynamic performance, both cases show almost the same improvement in efficiency. The local Mach number contours for both the baseline design and the optimized cases are shown in Figures 10.22 and 10.23 at two different span locations. The redesign strategy shown by the adjoint method is to weaken the shock wave and push it downstream, similarly to what has been done with the NASA rotor 67. The pressure coefficient distribution (Figure 10.24) shows a new blade loading distribution with a better diffusion rate.

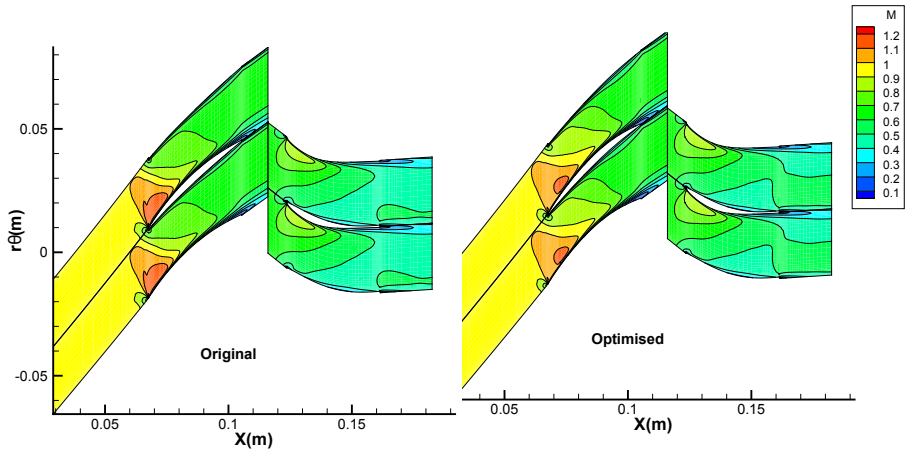


Figure 10.22: Local Mach number contours at 50% span. DLR stage redesign

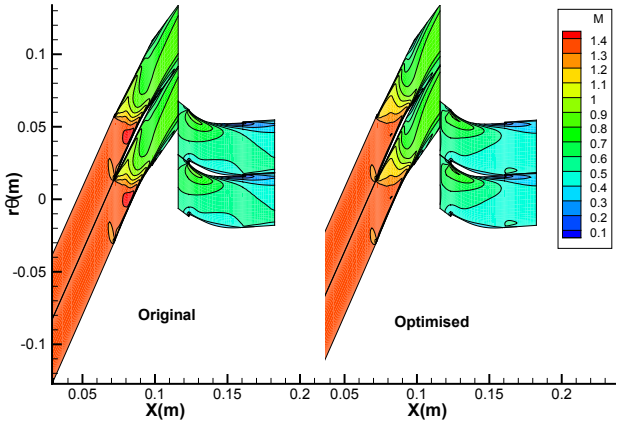
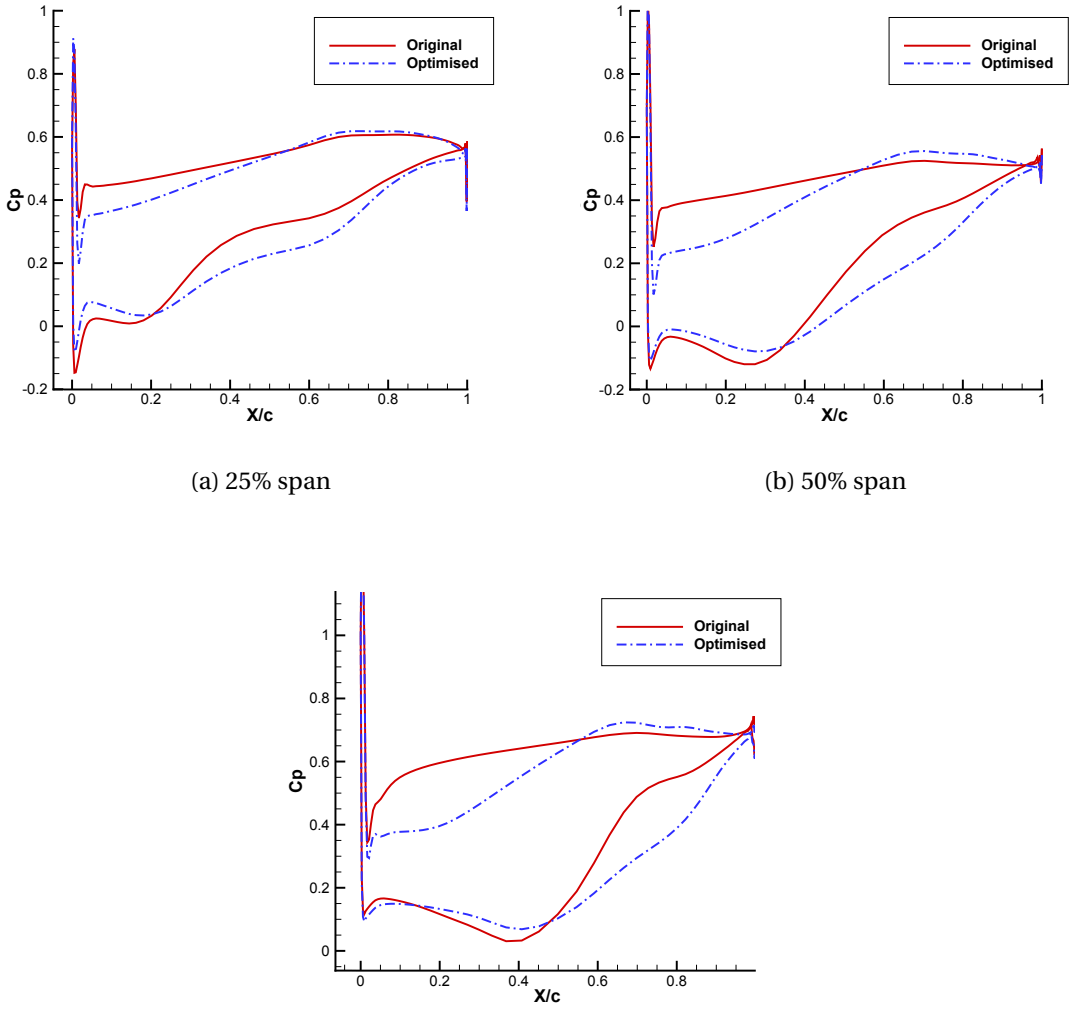


Figure 10.23: Local Mach number contours at 75% span. DLR stage redesign



(a) 25% span

(b) 50% span

(c) 75% span

Figure 10.24: Pressure coefficient distributions of rotor blade. DLR stage redesign

10.4 Industrial LP Stage Aeroelastic Optimization

The next set of cases is intended to show not only a multi-component comparison for the industrial LP turbine stage but also the multi-disciplinary capabilities of the optimization method. As shown in section 5.8.3, an aeroelastic-only redesign for this turbine case does not simultaneously improve aerodynamic efficiency, as it is the case for the NASA rotor 67 and DLR stage cases. Hence, in order to keep the aerodynamic perform-

ance of the baseline design, the entropy generation rate constraint has to be enabled in the objective function (Equation 5.6). Three cases are presented next. The first two are aeroelastic-only ($\sigma_3 = 0$) design optimizations of the rotor blade only and the whole stage respectively. In order to illustrate the strengths of a concurrent design optimization, a concurrent aeroelastic ($\sigma_3 > 0$) stage design optimization is presented as the third case study.

10.4.1 Aeroelastic-only Rotor Blade Design Optimization

This first case only optimizes the shape of the rotor blade. A final converged design is reached after 30 design cycles (Figure 10.25a). Table 10.7 has the absolute and relative differences between optimized and baseline designs written down. The rotor blade log-dec is relatively increased by 26%. However, the entropy generation rate is increased and the isentropic efficiency is decreased by 0.22%. The mass flow rate and pressure ratio both increase but the relative difference to baseline design values are below 0.20%.

	Abs. change	Rel. change
Isentropic efficiency (%)	-0.22	-0.25%
Log-dec (%)	+1.04	+26.53%
Mass flow rate (kg/s)	+0.06	+0.15%
Pressure ratio	-0.003	-0.15%

Table 10.7: Absolute and relative differences after convergence. LP turbine rotor aeroelastic-only redesign

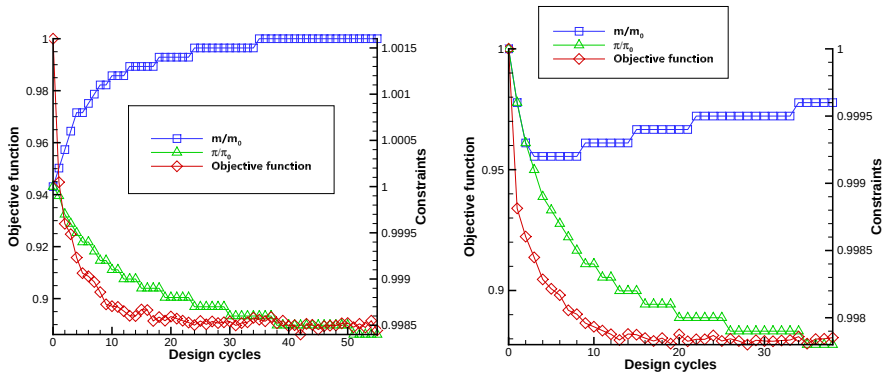
10.4.2 Aeroelastic-only Stage Design Optimization

For this case, both stator and rotor have their shapes optimized. The aeroelastic-only design optimization converges after 35 design cycles as shown in Figure 10.25b and the objective function decreases at a similar rate as the “rotor-only” case. Table 10.8 shows an absolute rotor blade log-dec increase of 1.15%, which is nearly a 30% increase of the

baseline design value. The isentropic efficiency is also decreased to a similar value as the case above. The mass flow is now kept constant but the pressure ratio decreases to a maximum relative decrease of 0.22%.

	Abs. change	Rel. change
Isentropic efficiency (%)	-0.23	-0.26%
Log-dec (%)	+1.15	+29.34%
Mass flow rate (kg/s)	-0.02	-0.04%
Pressure ratio	-0.005	-0.22%

Table 10.8: Absolute and relative differences after convergence. LP turbine stage aeroelastic-only redesign



(a) "Rotor-only" optimization case case (b) "Stator and rotor" optimization case

Figure 10.25: Convergence of objective function and constraints. LP turbine stage aeroelastic-only redesign

10.4.3 Multi-component Optimization Comparison

Adding the stator blade to the shape optimization leads to an additional increase of nearly 4% in the rotor blade log-dec. The extra margin of improvement is considerably smaller than the main improvement obtained by shaping the rotor, but it is important to note from Table 8.2 that the stator work sum gradient is half the value of the rotor work sum gradient. Hence, more improvement can be expected in cases where unsteady reflec-

tions are more significant. The local work contours for the first mode on the rotor blade surface of both cases are shown in Figures 10.26 and 10.27. At the pressure surface, both cases show decreased values of the stable gradient. However the values of the unstable gradient at the suction side are significantly decreased before the shock while the small stable gradient values after the shock are increased. The overall improvement is shown to be better for the full stage (stator and rotor) case. On the other side, both cases show a significant decrease in isentropic efficiency due to the lack of entropy generation rate constraint in the objective function. This compromises the aerodynamic performance of the blade and highlights the importance of concurrent design optimization. The changes to the blade geometry for both cases are shown at three different span locations in Figure 10.28. The full stage design optimization case shows an upwards hump before midchord and a downwards hump after midchord on the stator blade. In terms of the rotor changes both cases show similar geometries, with a more significant positive humping on the leading edge of the rotor blade and negative humping at the trailing edge, which changes the camber line of the blade significantly.

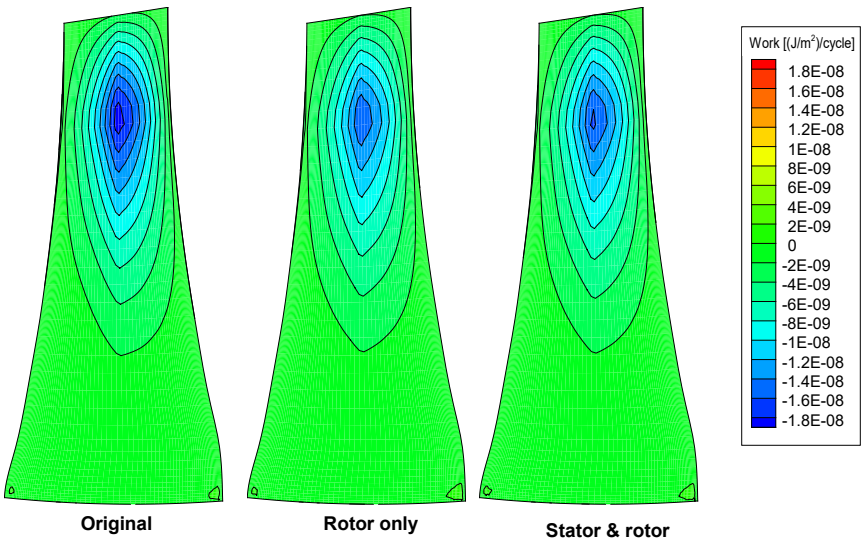


Figure 10.26: Local work contours on blade (first bending mode). Pressure side. LP turbine stage aeroelastic-only redesign

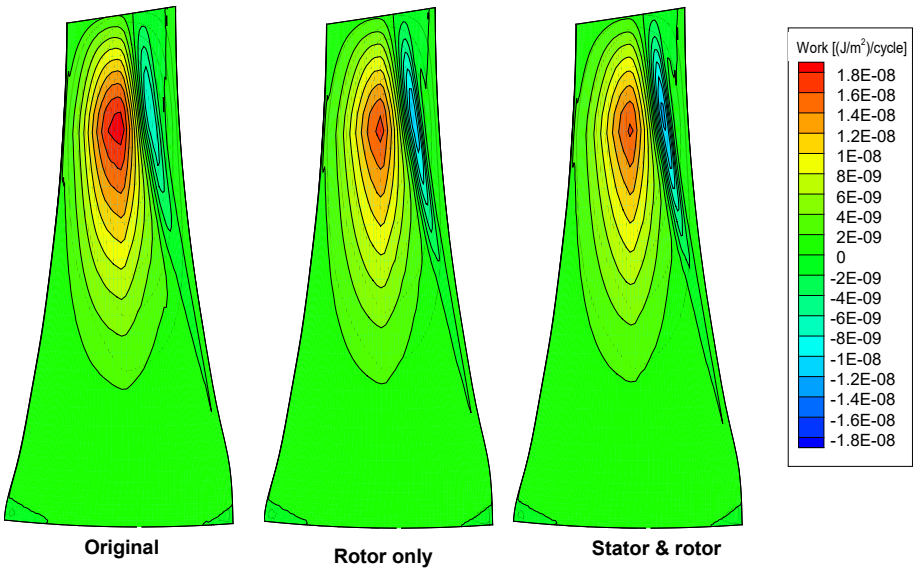


Figure 10.27: Local work contours on blade (first bending mode). Suction side. LP turbine stage aeroelastic-only redesign

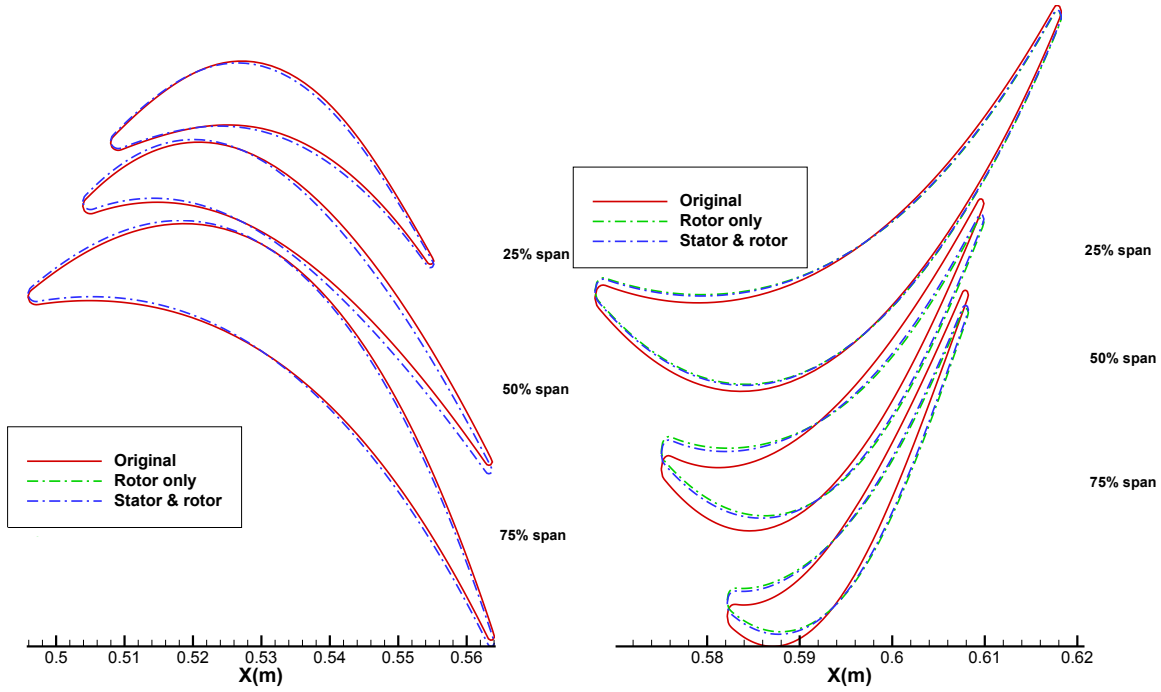


Figure 10.28: Stage profiles at 25%, 50% and 75% span. LP turbine stage aeroelastic-only redesign

10.4.4 Concurrent Aeroelastic Stage Design Optimization

The concurrent design optimization takes a similar time to converge as the aeroelastic-only cases as shown in Figure 10.29. The absolute and relative differences are given in Table 10.9. The rotor blade log-dec is now only increased 0.84%, which leads to 8% less relative improvement compared to the aeroelastic-only case. However, the entropy constraint is now able to control the entropy generation rate and as a result, the isentropic efficiency is almost unchanged compared to the original design. The other two constraints keep both the mass flow rate and pressure ratio at similar levels as the original design.

	Abs. change	Rel. change
Isentropic efficiency (%)	-0.02	-0.02%
Log-dec (%)	+0.84	+21.43%
Mass flow rate (kg/s)	-0.04	-0.11%
Pressure ratio	-0.001	-0.05%

Table 10.9: Absolute and relative differences after convergence. LP turbine concurrent aeroelastic redesign

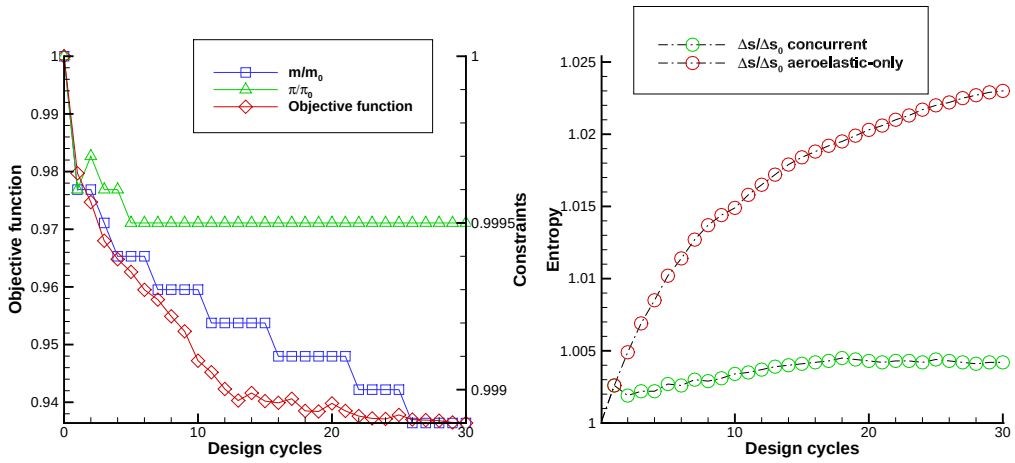


Figure 10.29: Convergence of objective function and constraints. LP turbine stage concurrent aeroelastic redesign

10.4.5 Multi-disciplinary Optimization Comparison

A concurrent design optimization improves performance in rotor blade log-dec with a much better control of the entropy generation rate (Figure 10.29) and its impact in the overall efficiency of the stage. In order to illustrate this better, the last two aeroelastic-only and concurrent aeroelastic stage design optimization cases are compared next. The local work contours for the first mode on both blade surfaces are shown in Figures 10.30 and 10.31. On the pressure surface, the stable gradient for both cases seems to be weaker with respect to the original contour. However on the suction surface, it can be seen how the unstable gradient for the two new designs is weakened and the stable gradient increases its strength immediately after the shock. The margin of aeroelastic improvement is greater for the aeroelastic-only design.

In terms of the aerodynamic performance, the local Mach contours at midspan are shown in Figure 10.32. The aeroelastic-only case increases the local peak Mach number at the stator passage, which has a negative effect on the passage flow losses. However, the concurrent case keeps the local Mach numbers to almost the same level at the stator passage. Regarding the rotor passage, the pressure coefficient distributions at three span locations are shown in Figure 10.33. Near the throat, the two new designs show a higher loading and higher peak velocity with a much steeper deceleration until the trailing edge; which increases the flow losses in the passage. However, the concurrent case reduces the loading at the leading edge which reduces the secondary flows and incidence losses and compensates for the increased peak velocity after the throat.

In terms of blade shapes (Figures 10.34 and 10.35), both designs show a rotor redesign which is humped upwards before midchord while humped downwards after midchord, with the biggest changes on the camber being located around the midspan region. The magnitude of the shape perturbation is twice as large for the aeroelastic-only design as for the concurrent design. The changes in the stator, however, are very similar for both cases and smaller in magnitude when compared to the rotor blade shape changes.

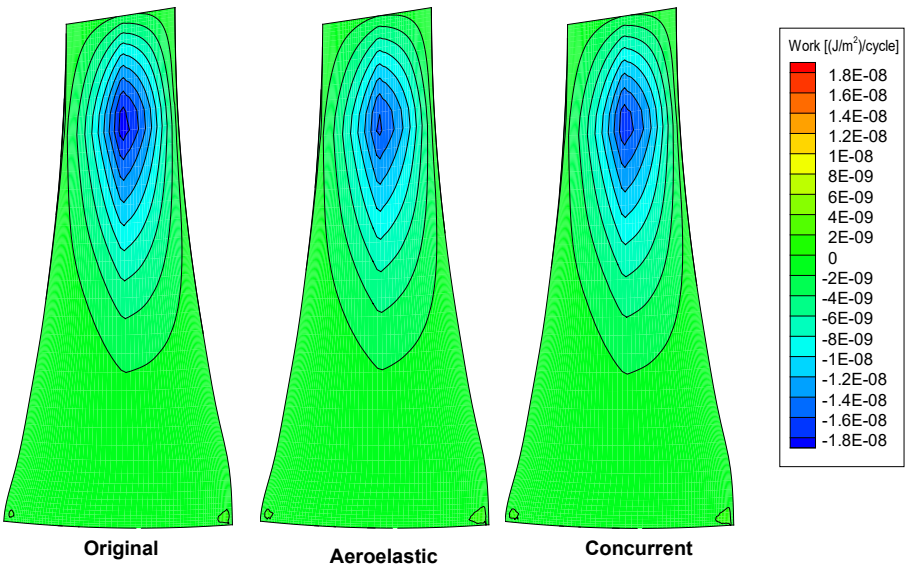


Figure 10.30: Local work contours on blade (first bending mode). Pressure side. LP turbine stage aeroelastic redesign

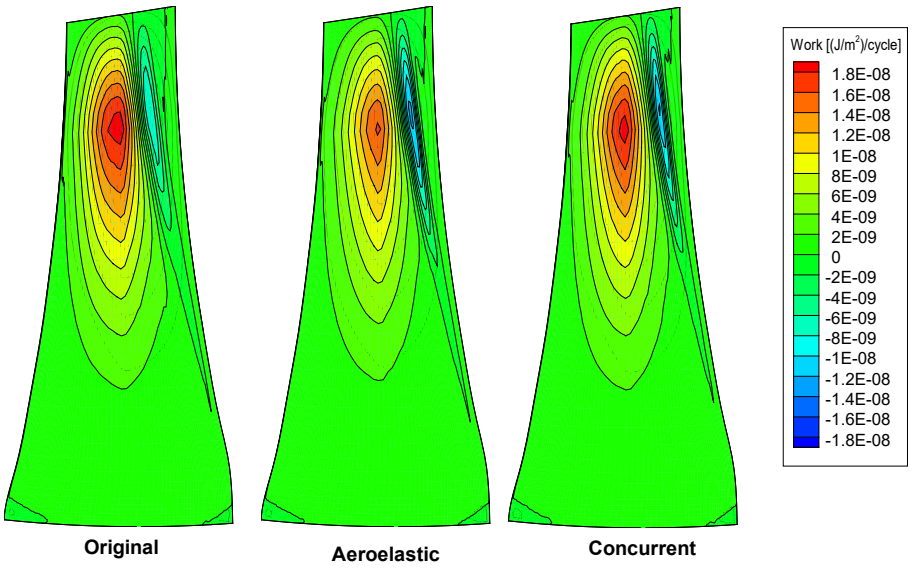


Figure 10.31: Local work contours on blade (first bending mode). Suction side. LP turbine stage aeroelastic redesign

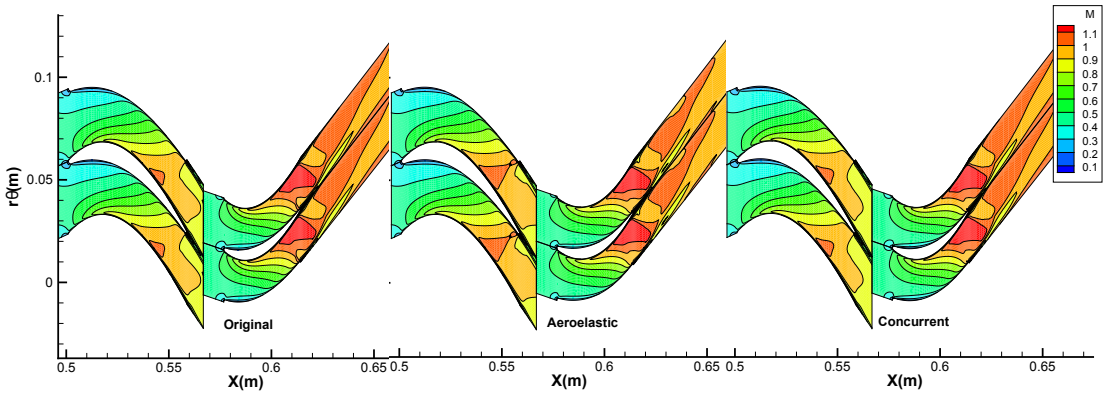
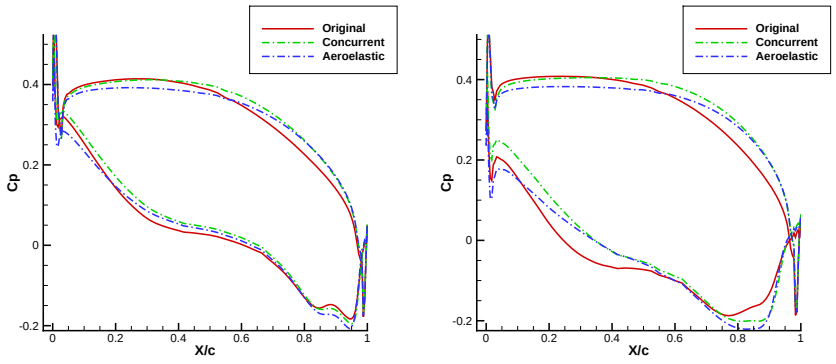
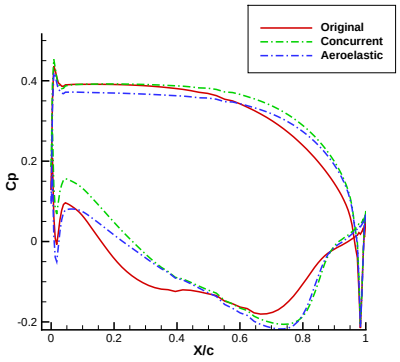


Figure 10.32: Local Mach number contours at 50% span. LP turbine stage aeroelastic redesign



(a) 25% span

(b) 50% span



(c) 75% span

Figure 10.33: Pressure coefficient distributions of rotor blade. LP turbine aeroelastic design

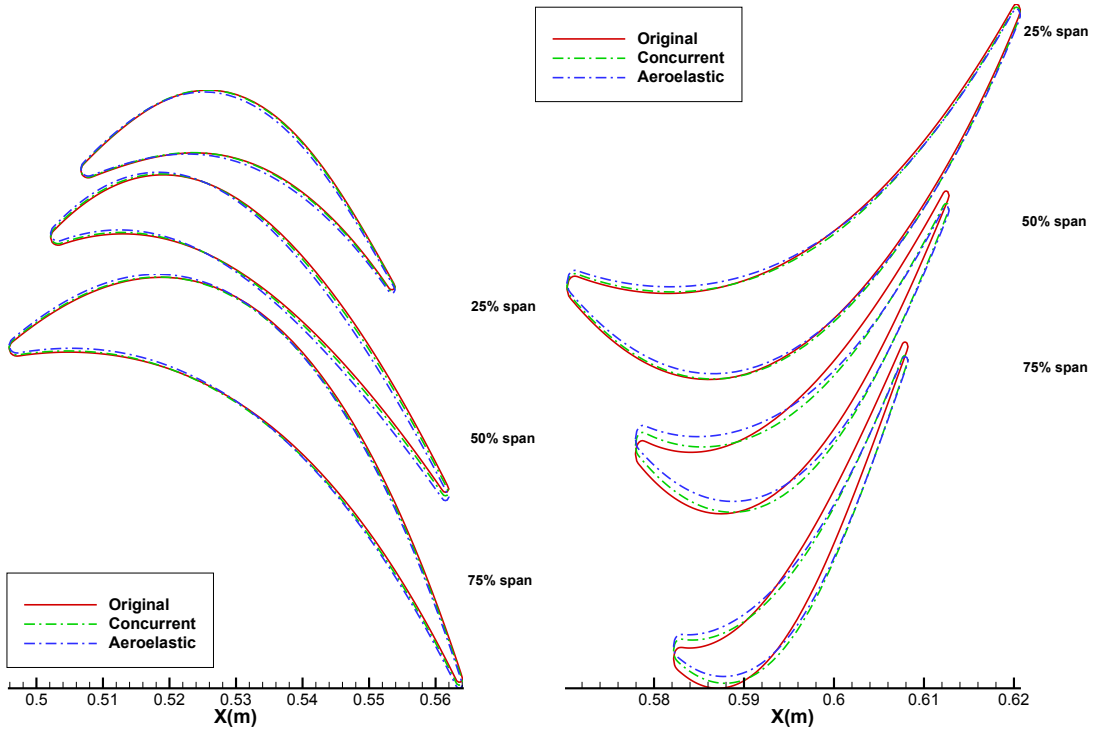
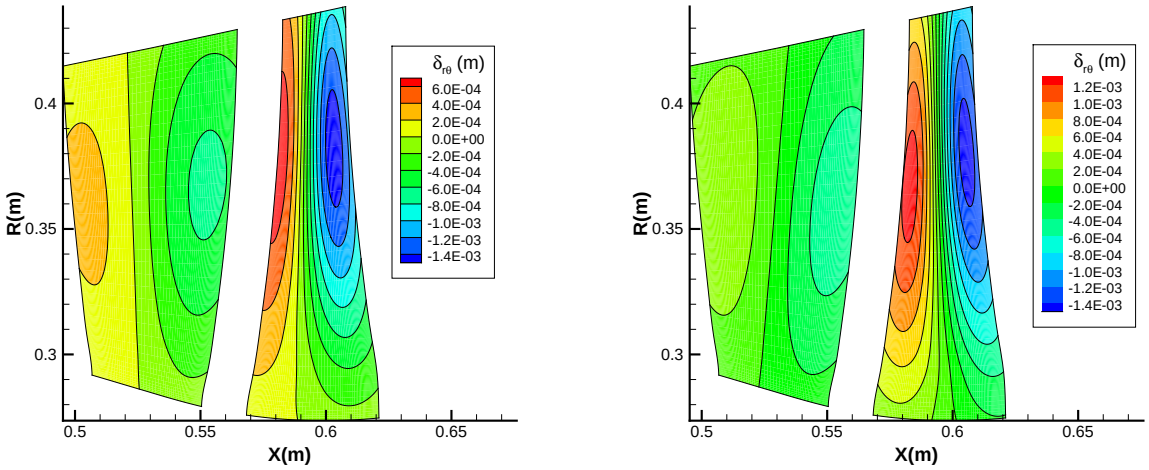


Figure 10.34: Stage profiles at 25%, 50% and 75% span. LP turbine stage aeroelastic redesign



(a) Aeroelastic-only stage

(b) Concurrent aeroelastic stage

Figure 10.35: Contours of tangential coordinate change, $\delta_{r\theta}$. LP turbine stage aeroelastic redesign

10.5 Industrial LP Stage Aerodynamic Optimization

The adjoint method can also be used to improve the aerodynamic performance of a stage while also keeping the same aeroelastic performance. A comparison between an aerodynamics-only and concurrent aerodynamic optimization is presented next.

10.5.1 Aerodynamics-only Stage Design Optimization

The aerodynamics-only optimization is mostly converged after 40 design cycles as shown in Figure 10.36. The differences between baseline and optimized case are given in Table 10.10. The isentropic efficiency is relatively increased 0.31%. However the log-dec is decreased 0.91%. The pressure ratio shows the biggest change with a relative 0.32% increase.

	Abs. change	Rel. change
Isentropic efficiency (%)	+0.31	+0.35%
Log-dec (%)	-0.91	-23.21%
Mass flow rate (kg/s)	-0.032	-0.08%
Pressure ratio	+0.007	+0.32%

Table 10.10: Absolute and relative differences after convergence. LP turbine stage aerodynamic-only redesign

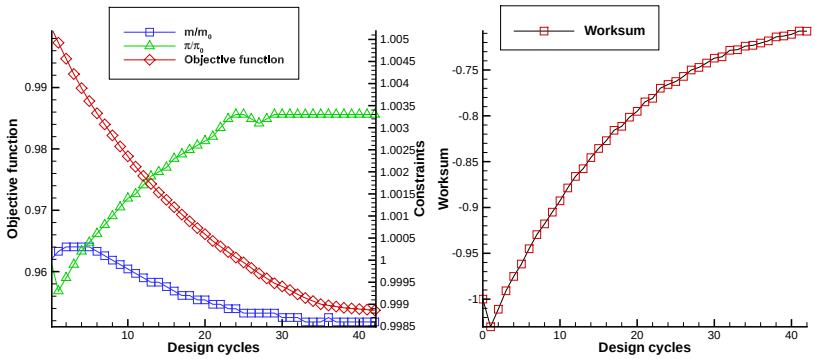


Figure 10.36: Convergence of objective function and constraints. LP turbine stage aerodynamic-only redesign

10.5.2 Concurrent Aerodynamic Stage Design Optimization

The concurrent design optimization also converges after 40 design cycles but it can be seen from Figure 10.37 that the convergence trend is different to its single discipline counterpart. This case illustrates very well how the optimization method tries to revert the work sum value back to original design levels on every iteration that the aeroelastic performance is compromised. The final values (Table 10.11) show a smaller improvement of the isentropic efficiency, almost half the improvement obtained with the aerodynamics-only optimization. The negative impact on the aeroelastic performance, however, is considerably smaller and the log-dec shows only a total decrease of 0.176%. Both the mass flow rate and pressure ratio values are almost unchanged.

	Abs. change	Rel. change
Isentropic efficiency (%)	+0.16	+0.18%
Log-dec (%)	-0.176	-4.34%
Mass flow rate (kg/s)	-0.044	-0.11%
Pressure ratio	+0.0013	+0.06%

Table 10.11: Absolute and relative differences after convergence. LP turbine stage concurrent aerodynamic redesign

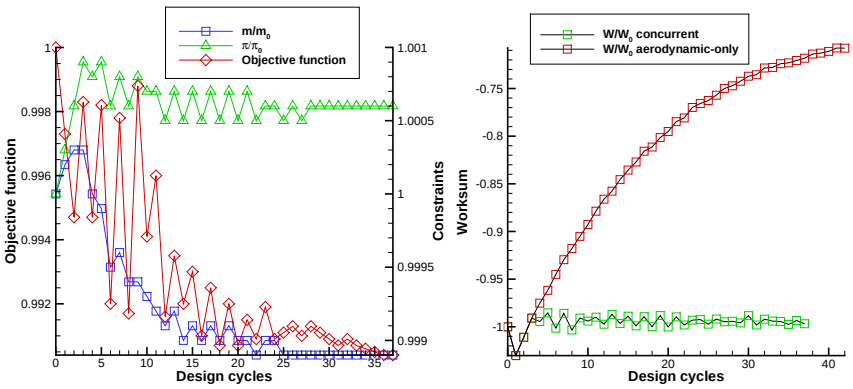


Figure 10.37: Convergence of objective function and constraints. LP turbine stage concurrent aerodynamic redesign

10.5.3 Multi-disciplinary Optimization Comparison

The two cases, aerodynamic-only and concurrent, are compared between them. The local Mach contours at midspan are shown in Figure 10.38. Both cases show an stator passage with a reduced peak Mach number. This translates to less flow losses in the stator passage and an overall increase in the isentropic efficiency of the stage. Looking at the surface pressure coefficient distributions in Figure 10.39, it can be observed that the loading levels at the throat of the rotor are almost the same for the three cases. The most important change is observed at the leading edge with a smaller loading that reduces secondary flow and incidence losses. These two improvements account for the increase in isentropic efficiency.

The local work contours are shown in Figures 10.40 and 10.41. While the stable gradient at the pressure side is almost the same for the three cases, the biggest changes can be observed on the unstable gradient at the suction side. The aerodynamic-only design increases significantly the strength of the unstable area while the concurrent design shows a smaller increase.

In terms of the changes to the blade shapes (Figures 10.42 and 10.43), both the aerodynamic-only and concurrent cases show different perturbations. Overall, both stator and rotor blades are mostly humped downwards with very small changes on the stator blade. The optimized rotor blade from the aerodynamic-only design shows bigger changes before midchord and midspan. For the optimized design of the concurrent case the biggest changes on the redesigned rotor blade are located after midchord and closer to the tip region.

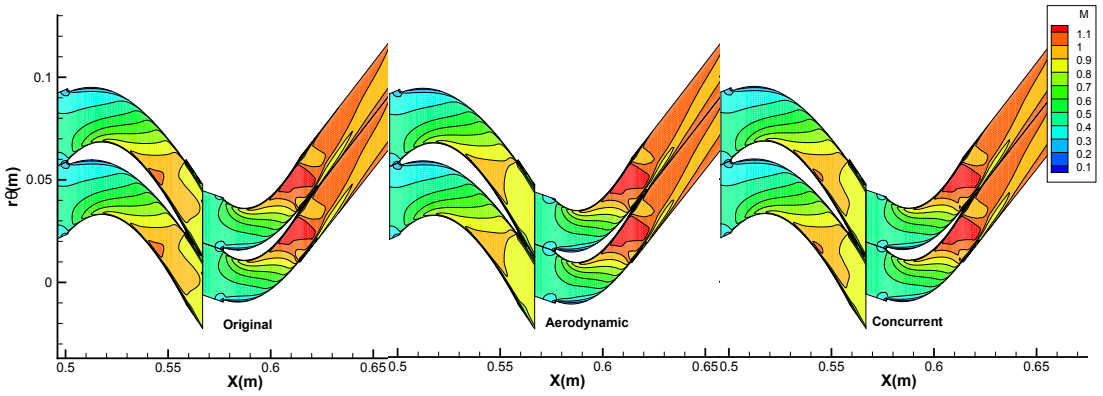
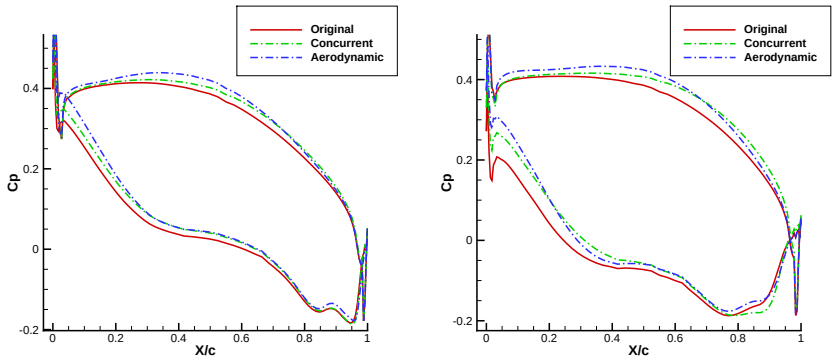
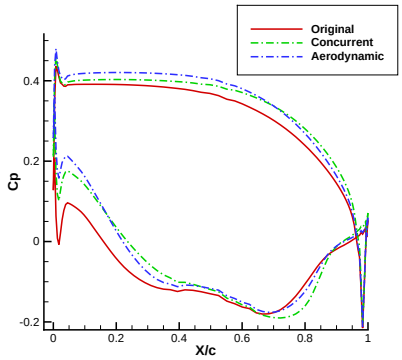


Figure 10.38: Local Mach number contours at 50% span. LP turbine stage aerodynamic redesign



(a) 25% span

(b) 50% span



(c) 75% span

Figure 10.39: Pressure coefficient distributions of rotor blade. LP turbine stage aerodynamic redesign

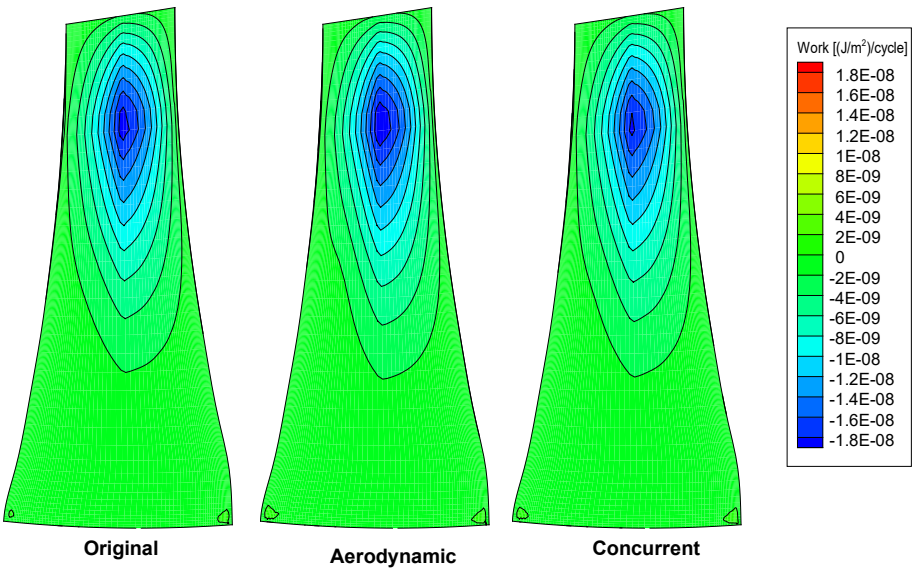


Figure 10.40: Local work contours on blade (first bending mode). Pressure side. LP turbine stage aerodynamic redesign

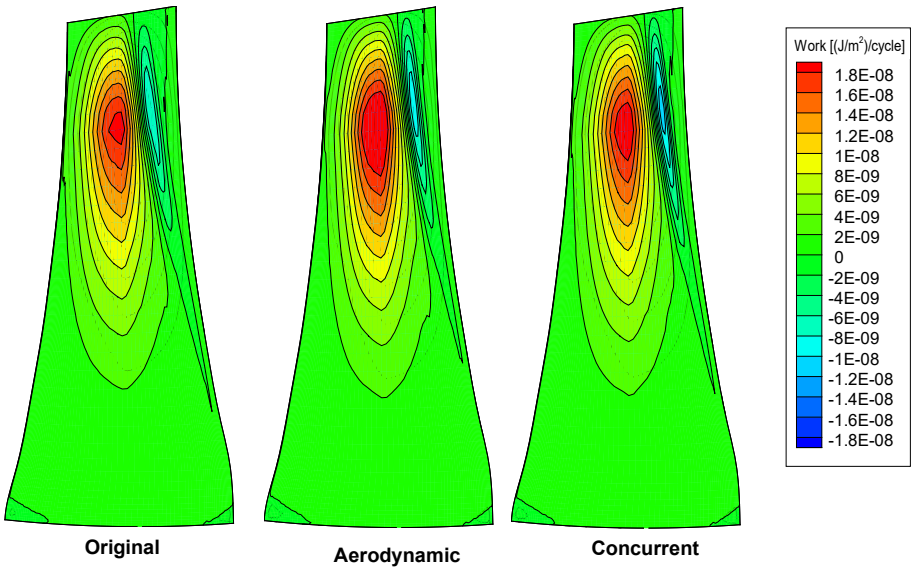


Figure 10.41: Local work contours on blade (first bending mode). Suction side. LP turbine stage aerodynamic redesign

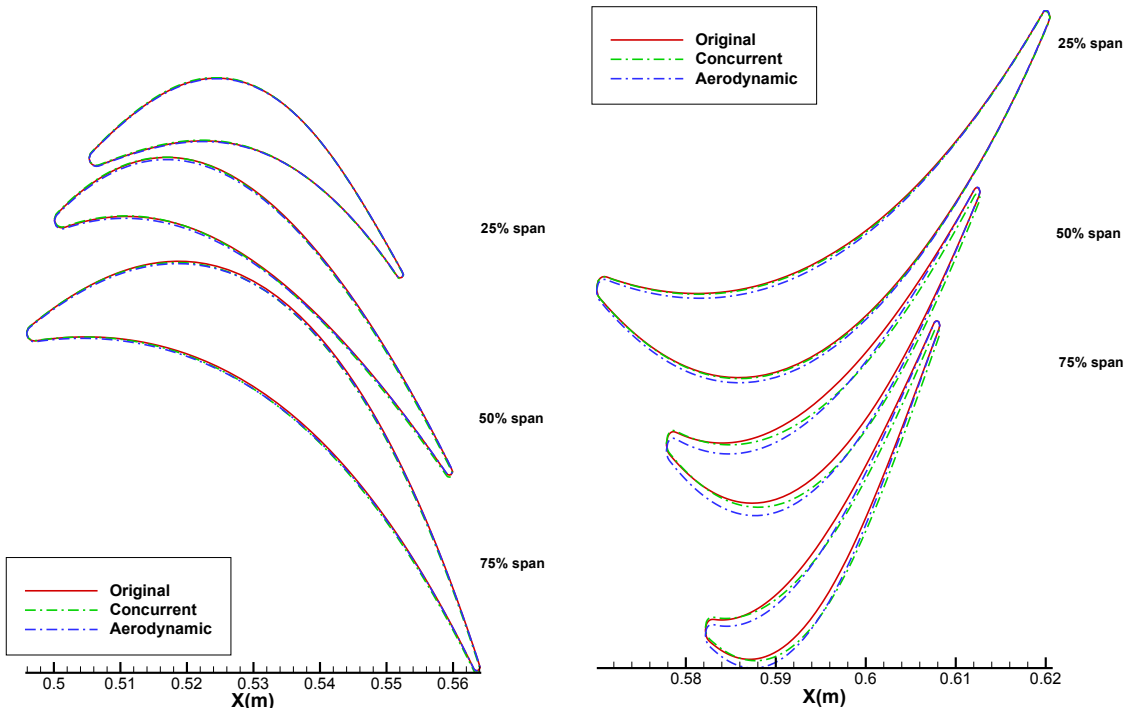
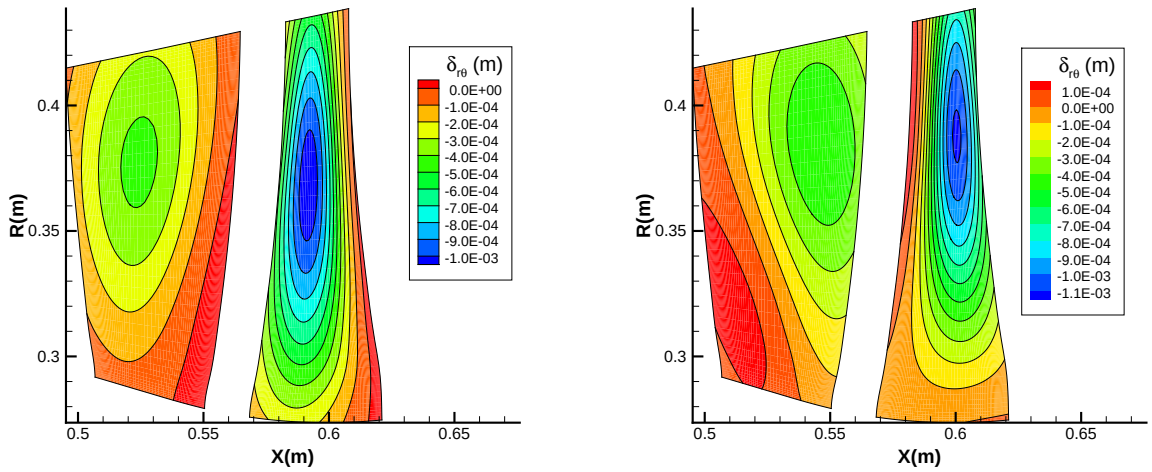


Figure 10.42: Stage profiles at 25%, 50% and 75% span. LP turbine stage aerodynamic redesign



(a) Aerodynamic-only stage

(b) Concurrent stage

Figure 10.43: Contours of tangential coordinate change, $\delta_{r\theta}$. LP turbine stage aerodynamic redesign

10.6 Summary

The current adjoint system is able to perform aeroelastic and aerodynamic design optimization of turbomachinery blades in a single-bladerow and multi-bladerow environment. The first set of cases has demonstrated the effectiveness of the new adjoint harmonic interface treatment and the potential of the stator to influence the aeroelastic performance of adjacent rotor blades. The next two redesigns of the NASA rotor 67 and the DLR transonic stage have showed the capabilities of the adjoint method to improve aeroelastic performance for both single-bladerow and multi-bladerow cases. The multi-component capabilities for aeroelastic optimization have also been shown. Adding the stator to the design optimization process allowed increasing the margin of improvement of the rotor blade log-dec on the DLR and LP turbine stage cases. This improvement, despite small, is proportional to the work sum gradient of the stator blade and better margins can be obtained on cases where the adjacent reflections from the stator are more important.

Last but not least important, the strengths of a concurrent objective function have been demonstrated with the optimization of the industrial LP turbine stage. Concurrent aeroelastic and aerodynamic redesigns have been carried out and compared against the results obtained from their single-sided counterparts. The concurrent optimized designs are able to show improvement in performance in their respective discipline (aerodynamics or aeroelasticity) while controlling the negative impact on the other discipline's performance.

Chapter 11

Conclusions and Future Work

11.1 Conclusions

The main objective of this work has been to develop a multi-component and multi-disciplinary optimization method that could improve the performance of real turbomachinery blades within a reasonable timescale.

Re-scaled Response Surface Model

The non-gradient Re-scaled Response Surface Method has proven its effectiveness for improving the aeroelasticity performance of three case studies in a concurrent way, controlling the impact of the rotor blade log-dec improvement in the overall aerodynamic efficiency of the blade or stage. However, this method has a strong limitation in the number of variables that can be used.

Nonlinear Harmonic Phase Solution Method

The Nonlinear Harmonic Phase Solution Method calculates efficiently time-periodic unsteady flows for blade flutter applications and it is also a key element to provide a basis for the development of the multi-bladerow unsteady adjoint system presented in this work

Unsteady Adjoint Method

The new unsteady adjoint formulation based upon the flow solution is the most important part of this work. The re-derived system of adjoint equations illustrates very well the relation between the flow field and its corresponding adjoint field. The "anti-physics" path of information is consistently shown in the formulation of the source terms of the adjoint equation and the phase-shifted periodicity differences between the physical and adjoint time domains.

The assumption of a harmonic adjoint variable has also allowed use of the flow interface treatment as a basis to create a new adjoint interface treatment. This has been validated by a quantitative comparison with the finite difference method, but also showing qualitatively how the adjoint system is able to improve the log-dec of a rotor blade by only shaping the adjacent stator blade.

Efficient Multi-disciplinary and Multi-component Design Optimization

The adjoint method presented here is the first of its kind to perform multi-disciplinary and multi-component design optimization for turbomachinery blades. This has been demonstrated with the design optimization of a transonic compressor stage and a low-pressure turbine stage. An optimized adjacent stator has been shown to have a positive effect on the rotor blade log-dec, showing a better margin of improvement when compared to a rotor-only design optimization case. When an improvement in aeroelasticity or aerodynamics negatively impacts other discipline´s performance, a concurrent design optimization has been demonstrated to show improvements on one discipline while controlling the negative impact on the other discipline.

11.2 Future Work

Based on the findings presented in this thesis, below are some suggestions that can be followed for future work.

Forced-response Capability

In this work, blade flutter has been frequently linked to aeroelastic performance. However, forced response is also a problem that is caused by unsteady aerodynamic forces generated by adjacent rows and circumferential distortion. In order to get a whole picture of the aeroelastic performance, this should also be taken into account in the design optimization process.

Multi-objective Optimization

The design optimization results that have been presented here are all for single operating points. That is something that can be definitely improved and it will be particular useful to either fully improve the aeroelastic performance of a rotor blade across the whole spectrum of inter-blade phase angles and nodal diameters or to ensure good aerodynamic performance of a turbomachine at off-design conditions. The fact that the adjoint method is computationally cheap is an advantage for this.

Combination of Different Optimization Methods

The adjoint method on its own is very efficient, but as a gradient-based method, it can get trapped at local optimum solutions. This drawback can be relieved if some sort of supportive method like a non-gradient based method (e.g the RRSM method) can be used to build an initial response surface model that could ideally lead to a faster convergence or a guaranteed convergence towards the global optimum.

References

Adamczyk, J. (1985). Model equations for simulating flows in multistage-turbomachinery. *ASME Paper*, 85-GT-226.

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.

Astrua, P., Piola, S., Silingardi, A., and Bonzani, F. (2012). Multi-objective constrained aero-mechanical optimization of an axial compressor transonic blade. *ASME Paper*, GT2012-68993.

Box, G. E. P. and Wilson, K. B. (1951). On the experimental attainment of optimium conditions (with discussion). *Journal of the Royal Statistical Society*, 13(1):1–45.

Chahine, C., Seume, J. R., and Verstraete, T. (2012). The influence of metamodeling techniques on the multidisciplinary design optimization of a radial compressor impeller. *ASME Paper*, GT2012-68358.

Chen, N., Zhang, H., Huang, W., and Xu, Y. (2005). Study on aerodynamic design optimization of turbomachinery blades. *Journal of Thermal Science*, 14(4).

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).

Cumpsty, N. and Greitzer, E. (2004). Ideas and methods of turbomachinery aerodynamics: A historical view. *Journal of Propulsion and Power*, 20(1):15–26.

Dawes, W. N. (1986). A numerical analysis of three-dimensional viscous flow in a transonic compressor rotor and comparison with experimental data. *ASME Paper*, 86-GT-16.

Deng, X., Guo, F., Liu, Y., and Han, P. (2013). Aero-mechanical optimization design of a transonic fan blade. *ASME Paper*, GT2013-95357.

Denton, J. D. (1979). Extension of the vinite volume time marching method to three dimensions. *VKI Lecture Series*, 1979-7.

Denton, J. D. (1985). Calculation of three dimensional flow through any type of turbomachine. *AGARD Lecture Series*, 140.

Denton, J. D. (1992). Calculation of 3D viscous flow through multi-stage turbomachines. *ASME Journal of Turbomachinery*, 114(1).

Denton, J. D. and Singh, U. (1979). Time-marching methods for turbomachinery flow calculations.

Denton, J. D. and Xu, L. (1999). The exploitation of three-dimensional flow in turbomachinery design. *Journal of Mechanical and Engineering Science*, 213(C2):125–137.

Dewhurst, S. and He, L. (2000). Unsteady flow calculations through turbomachinery stages using single-passage domain with shape-correction method. *Proceedings of 9th ISUAAT*.

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.

Duta, M. (2002). *The use of the adjoint method for the minimization of forced vibration in turbomachinery*. PhD thesis, University of Oxford.

Duta, M., Shahpar, S., and Giles, M. B. (2007). Turbomachinery design optimization using automatic differentiated adjoint code. *ASME Paper*, GT2007-28329.

Duta, M. C., Giles, M. B., and Campobasso, M. S. (2002). The harmonic adjoint approach to unsteady turbomachinery design. *International Journal for Numerical Methods in Fluids*, (40):323–332.

Ekici, K. and Hall, K. C. (2007). Nonlinear analysis of unsteady flows in multistage turbomachines using harmonic balance. *AIAA Journal*, 45(5):1047–57.

Ekici, K. and Hall, K. C. (2008). Nonlinear frequency-domain analysis of unsteady flows in turbomachinery with multiple excitation frequencies. *AIAA Journal*, 46(8):1912–20.

Erdos, J. I., Alzner, E., and McNally, W. (1977). Numerical solution of periodic transonic flow through a fan stage. *AIAA Journal*, 15(11).

Giles, M. B. (1988). Calculation of unsteady wake rotor interaction. *Journal of Propulsion and Power*, 4(4):356–362.

Giles, M. B. (1991). A numerical method for the calculation of unsteady flow in turbomachinery. 2.1.1, Massachusetts institute of Technology.

Giles, M. B. (1992). An approach for multi-stage calculations incorporating unsteadiness. *ASME Paper*, 92-GT-282.

Giles, M. B., Ghate, D., and Duta, M. (2005). Using automatic differentiation for adjoint CFD code development. *SAROD Workshop*. Tata McGraw-Hill.

Giles, M. B. and Pierce, N. (2000). Adjoint equations in CFD: duality, boundary conditions and solution behaviour. *Flow, Turbulence and Combustion*, 65(3/4):393–415.

Giles, M. B. and Pierce, N. (2001). Analytic adjoint solution for the quasi-one-dimensional Euler equations. *Journal of Fluid Mechanics*, 426:327–345.

Goinis, G., Vob, C., and Aulich, M. (2013). Automated optimization of an axial-slot type casing treatment for a transonic compressor. *ASME Paper*, (GT2013-94765).

Graf, M. B., Greitzer, E., Marble, F. E., and Sharma, O. P. (1999). Effects of stator pressure field on upstream rotor performance. *ASME Paper*, 99-GT-99.

Hall, K. and Lorence, C. (1993). Calculation of three-dimensional unsteady flows in turbomachinery using the linear harmonic Euler equations. *ASME Journal of Turbomachinery*, 115(4):800–809.

Hall, K., Thomas, J., and Clark, W. (2002). Computation of unsteady nonlinear flows in cascade using a harmonic balance technique. *AIAA Journal*, 40(5):879–886.

He, L. (1989). An Euler solution for unsteady flows around oscillating blades. *ASME Journal of Turbomachinery*, 112:714–722.

He, L. (2008). Harmonic solution of unsteady flow around blade with separation. *AIAA Journal*, 46(6):1299–1307.

He, L. (2010). Fourier methods for turbomachinery applications. *Progress in Aerospace Sciences*, 46(8):329–341.

He, L. (2012). Cambridge turbomachinery course notes.

He, L. and Denton, J. D. (1994). Three-dimensional time-marching inviscid and viscous solutions for unsteady flows around vibrating blades. *ASME Journal of Turbomachinery*, 116:469–476.

He, L. and Ning, W. (1998). An efficient approach for analysis of unsteady viscous flows in turbomachines. *AIAA Journal*, 36(11):2005–2012.

He, L. and Sato, K. (2000). Numerical solution of incompressible unsteady flows in turbomachinery. *Journal of Fluids Engineering*, 123(3):680–685.

He, L. and Wang, D. X. (2011). Concurrent blade aerodynamic-aero-elastic design optimization using adjoint method. *ASME Journal of Turbomachinery*, 133(1):011021–011021–10.

Hicks, R. M. and Henne, P. A. (1978). Wing design by numerical optimization. *Journal of Aircraft*, (15):407–412.

Hicks, R. M., Murman, E. M., and Vanderplaats, G. N. (1974). An assessment of airfoil design by numerical optimization. *NASA Technical Report*, TM X-3092.

Holland, J. H. (1975). *Adaption in natural and artificial systems*. University of Michigan Press.

Huang, H. and Ekici, K. (2014). A discrete adjoint harmonic balance method for turbomachinery shape optimization. *Aerospace Science and Technology*, 39:481–490.

Huang, X. (2006). *Three-dimensional unsteady flow in oscillating turbine blade row*. PhD thesis, School of Engineering, University of Durham.

Huang, X., He, L., and Bell, D. L. (2009). Experimental and computational study of oscillating turbine cascade and influence of part-span shrouds. *ASME Journal of Fluids Engineering*, 131(5).

Jameson, A. (1988). Aerodynamic design via control theory. *Journal of Scientific Computing*, 3(3):233–260.

Jameson, A. (1989). Computational aerodynamics for aircraft design. *Journal of Science*, 245:361–371.

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.

Joly, M., Verstraete, T., and Paniagua, G. (2012). Full design of a highly loaded fan by multi-objective optimization of through-flow and high-fidelity aero-mechanical performances. *ASME Paper*, GT2012-69686.

Kim, C. (2007). Essential components for high-fidelity aerodynamic analysis and design. *In proceedings of the 7th Asian CFD Conference*. Bangalore, India.

Kim, H. and Nakahashi, K. (2005). Discrete adjoint method for unstructured Navier-Stokes solver. *AIAA Paper*, 05-449.

Larosiliere, L., Wood, J., Hathaway, M., Medd, A., and Dang, T. (2002). Aerodynamic design study of advanced multistage axial compressor. Tp2002-211568, NASA.

Lee, S., Lee, S., Kim, K., Lee, D., Kang, Y., and Rhee, D. (2014). Optimization framework using surrogate model for aerodinamically improved 3D turbine blade design. *ASME Paper*, GT2014-26571.

Li, H. D. and He, L. (2002). Single-passage analysis of unsteady flows around vibrating blades of a transonic fan under inlet distortion. *ASME Journal of Turbomachinery*, 124(2):285–292.

Li, H. D. and He, L. (2005). Blade aerodynamic damping variation with rotor-stator gap: a computational study using single-passage approach. *ASME Journal of Turbomachinery*, 127(3):573–579.

Li, H. D., He, L., Li, Y. S., and Wells, R. G. (2006). Blading aerodynamics design optimization with mechanical constraints. *ASME Paper*, GT2006-90503.

Lian, Y. and Liou, M. (2006). Aerostructural optimization of a transonic compressor rotor. *Journal of Propulsion and Power*, 22(4).

Luo, J., Xiong, J., Liu, F., and McBean, I. (2010). Three-dimensional aerodynamic design optimization of a turbine blade by using an adjoint method. *ASME Journal of Turbomachinery*, 133(1):011026.

Marta, A. C. and Shankaran, S. (2013). On the handling of turbulence equations in RANS adjoint solvers. *Computers & Fluids*, 74:102–113.

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245.

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.

Mishra, S. K. (2007). Performance of differential evolution method in least squares fitting of some typical nonlinear curves. *Social Science Research Network*.

Moffat, S. and He, L. (2003). Blade forced response prediction for industrial gas turbine. part i: Methodologies. *ASME Paper*, (GT2003-38640).

Mohammadi, B. (1998). Shape optimization for 3D turbulent flows using automatic differentiation. *International Journal of Computational Fluid Dynamics*, 11:27–50.

Mohammadi, B. and Pironneau, O. (2004). Shape optimization in fluid mechanics. *Annual Review of Fluid Mechanics*, 36:255–279.

Mousavi, A. and Nadarajah, S. (2011). Adjoint-based multidisciplinary design optimization of cooled gas turbine blades. *AIAA Paper*, 2011-1131.

Mueller, L., Alsalihi, Z., and Verstraete, T. (2012). Multidisciplinary optimization of a turbocharger radial turbine. *ASME Journal of Turbomachinery*, 135(2):021022.

Nadarajah, S. and Jameson, A. (2000). A comparison of continuous and discrete adjoint approach to automatic aerodynamic optimization. *AIAA Paper*, 00-0667.

Nielsen, E. and Anderson, W. (1999). Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. *AIAA Journal*, 37(11):1411–1419.

Nielsen, E. and Kleb, B. (2005). Efficient construction of discrete adjoint operators on unstructured grids by using complex variables. *AIAA Paper*, 05-0324.

Okui, H., Verstraete, T., Van den Braembussche, R. A., and Alsalihi, Z. (2013). Three-dimensional design and optimization of a transonic rotor in axial flow compressors. *ASME Journal of Turbomachinery*, 135(3):031009.

Papadimitriou, D. and Giannakoglou, K. (2006). Compressor blade optimization using a continuous adjoint formulation. *ASME Paper*, GT2006-90466.

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*.

Price, K. and Storn, N. (1997). Differential evolution. *Dr. Dobb's Journal*, pages 18–24.

Rahmati, M. T., He, L., Wang, D. X., Li, Y. S., Wells, R. G., and Krishnababu, S. K. (2013). Nonlinear time and frequency domain methods for multirow aeromechanical analysis. *ASME Journal of Turbomachinery*, 136(4):041010.

Reddy, T. and Srivastava, R. (1999). Flutter and forced response analysis of cascade using a two-dimensional linear Euler solver. Tm1999-209633, NASA.

Samad, A., Kim, K., Goel, T., Haftka, R. T., and Shyy, W. (2006). Shape optimization of turbomachinery blade using multiple surrogate models. *ASME Paper*, FEDSM2006-98368:827–836.

Saxer, A. P. and Giles, M. B. (1993). Quasi-three dimensional nonreflecting boundary conditions for Euler equations calculations. *AIAA Journal*, 9(2):263–271.

Siller, U., Vob, C., and E, N. (2009). Automated multidisciplinary optimization of a transonic axial compressor. *AIAA Paper*, 2009-863.

Spalart, P. R. and Allmaras, S. R. (1992). A one-equation turbulence model for aerodynamic flows. *AIAA Paper*, 92-0439.

Srivastava, R., Bakhle, M., Keith, T., and Stefko, G. (2002). Flutter analysis of a transonic fan. Tm2002-211818, NASA.

Taheri, R. and Mazaheri, K. (2012). Blade shape optimization of marine propeller via genetic algorithm for efficiency improvement. *ASME Paper*, GT2012-68183.

Thomas, J., Hall, K., and Dowell, E. (2003). A discrete adjoint approach for modelling unsteady aerodynamic design sensitivities. *AIAA Paper*, 03-0041.

Valero, O. R., He, L., and Li, Y. S. (2014). Concurrent blade aerodynamic-aeroelastic design optimization with re-scaled response surface. *ASME Paper*, GT2014-25854.

Van den Braembussche, R. A., Alsalihi, Z., Verstraete, T., Matsuo, A., Ibaraki, S., Sugimoto, K., and Tomita, I. (2012). Multidisciplinary multipoint optimization of a transonic turbocharger compressor. *ASME Paper*, GT2012-69645.

Vatsa, V. (2004). Computation of sensitivity derivatives of Navier-Stokes equations using complex variables. *NASA Technical Report*, (2004-086675).

Verdon, J. M. and Caspar, J. R. (1982). A linearized unsteady aerodynamic analysis for transonic cascades. *Journal of Fluid Mechanics*, 149:403–429.

Verstraete, T., Alsalihi, Z., and Van den Braembussche, R. A. (2010). Multidisciplinary optimization of a radial compressor for microgras turbine applications. *ASME Journal of Turbomachinery*, 132(3):031004.

Verstraete, T., Prinsier, J., and Cosi, L. (2014). Design and off-design optimization of a low pressure steam turbine radial diffuser using an evolutionary algorithm and 3D CFD. *ASME Paper*, GT2014-26850.

Verstraete, T., Prinsier, J., Di Sante, A., Della Gata, S., and Cosi, L. (2012). Design optimization of a low pressure steam turbine radial diffuser using an evolutionary algorithm and 3D CFD. *ASME Paper*, GT2012-69515.

Walther, B. and Nadarajah, S. (2012). Constrained adjoint-based aerodynamic shape optimization of a single-stage transonic compressor. *ASME Journal of Turbomachinery*, 135(2):021017.

Walther, B. and Nadarajah, S. (2015). Optimum shape design for multirow turbomachinery configurations using a discrete adjoint approach and an efficient radial basis

function deformation scheme for complex multiblock grids. *ASME Journal of Turbomachinery*, 137(8):081006.

Wang, D. X. and He, L. (2010a). Adjoint aerodynamic design optimization for blades in multi-stage turbomachines: Part i - methodology and verification. *ASME Journal of Turbomachinery*, 132(2):021011.

Wang, D. X. and He, L. (2010b). Adjoint aerodynamic design optimization for blades in multi-stage turbomachines: Part ii - validation and application. *ASME Journal of Turbomachinery*, 132(2):021012.

Wu, H., Liu, F., and Tsai, H. (2005). Aerodynamic design of turbine blades using an adjoint equation method. *AIAA Paper*, 05-1006.

Wu, H., Yang, S., and Liu, F. (2003). Comparison of three geometric representation of airfoils for aerodynamic optimization. *AIAA Paper*, 2003-4095.

Yang, S., Wu, H., Liu, F., and Tsai, H. (2003). Aerodynamic design of cascade by using an adjoint equation method. *AIAA Paper*, 03-1068.

Yi, W., Huang, H., and Han, W. (2006). Design optimization of transonic compressor rotor using CFD and genetic algorithm. *ASME Paper*, GT2006-90155.

Appendix A

Unsteady Adjoint Navier-Stokes Equations

This appendix elaborates the linearisation of the URANS equation in order to derive the system of unsteady adjoint equations. The linearisation starts from Equation 7.17, which has been already discretized based on the Nonlinear Harmonic Phase Solution Method and rearranged by Fourier series. Linearising Equation 7.17 with respect to a design variable yields:

$$\begin{aligned}
&\int_0^T \int_V \omega \left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) \tilde{U} \\
+\lambda^T \Bigg[&\frac{\partial \left(A\tilde{U} - A_v\tilde{U} - D_{xx}\tilde{U}_x - D_{x\theta}\tilde{U}_\theta - D_{xr}\tilde{U}_r\right) - \tilde{U}u_g}{\partial x} \\
&+ \frac{\partial \left(B\tilde{U} - B_v\tilde{U} - D_{\theta x}\tilde{U}_x - D_{\theta\theta}\tilde{U}_\theta - D_{\theta r}\tilde{U}_r\right) - \tilde{U}v_g}{r\partial\theta} \\
&+ \frac{\partial r \left(C\tilde{U} - C_v\tilde{U} - D_{rx}\tilde{U}_x - D_{r\theta}\tilde{U}_\theta - D_{rr}\tilde{U}_r\right) - \tilde{U}w_g}{r\partial r} \\
&- D\tilde{U} - f \Bigg] dv dt = 0
\end{aligned} \tag{A.1}$$

where:

$$A = \frac{\partial F}{\partial U},\ A_v = \frac{\partial V_x}{\partial U},\ B = \frac{\partial G}{\partial U},\ B_v = \frac{\partial V_\theta}{\partial U},\ C = \frac{\partial H}{\partial U},$$
$$C_v = \frac{\partial V_r}{\partial U},\ D = \frac{\partial S}{\partial U},\ D_{ij} = \frac{\partial V_i}{\partial U_j}$$
$$\tilde{U}_x = \frac{\partial \tilde{U}}{\partial x},\ \tilde{U}_\theta = \frac{\partial \tilde{U}}{r\partial\theta},\ \tilde{U}_r = \frac{\partial r\tilde{U}}{r\partial r},\ f = -\frac{\partial R}{\partial \alpha},\quad i,j = x,\theta,r$$

A, B and C are the same Jacobian matrices as defined in Chapter 6 and f is the flow residual perturbation. The linearisation of the viscous terms creates a new set of Jacobian matrices A_v, B_v, C_v and a new set of stress tensors and linearised viscous terms. Integration by parts is performed to give:

$$\begin{aligned}
\int_0^T \int_{\partial V} \left[\lambda^T \left(A - Iu_g - A_v\right) n_x + \lambda^T \left(B - Iv_g - B_v\right) n_\theta + \lambda^T \left(C - Iw_g - C_v\right) n_r\right] \tilde{U} dS dt \\
+ \int_0^T \int_V \Bigg[\omega \left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) \\
+ \frac{\partial \lambda^T}{\partial x}\left(A - Iu_g - A_v\right) + \frac{\partial \lambda^T}{r\partial\theta}\left(B - Iv_g - B_v\right) + \frac{\partial \lambda^T}{\partial r}\left(C - Iw_g - C_v\right) - \lambda^T D \Bigg] \tilde{U} dv dt \\
- \int_0^T \int_V \Bigg[\frac{\partial \lambda^T}{\partial x}\left(D_{xx}\tilde{U}_x + D_{x\theta}\tilde{U}_{x\theta} + D_{xr}\tilde{U}_r\right) + \frac{\partial \lambda^T}{r\partial\theta}\left(D_{\theta x}\tilde{U}_x + D_{\theta\theta}\tilde{U}_\theta + D_{\theta r}\tilde{U}_r\right) \\
+ \frac{\partial \lambda^T}{\partial r}\left(D_{rx}\tilde{U}_x + D_{rx}\tilde{U}_x + D_{rx}\tilde{U}_x\right) \Bigg] dv dt - \int_0^T \int_V \lambda^T f dv dt = 0 \quad \text{(A.2)}
\end{aligned}$$

The objective of the adjoint equation is to eliminate the flow variable sensitivity out of the gradient equation. Because of the viscous terms, more integration by parts is needed in order to eliminate the flow variable sensitivity out of the spatial derivative operators. The second volume integral is rearranged around the spatial derivatives of $\tilde{U}$: $\tilde{U}_x, \tilde{U}_\theta$ and

$\tilde{U_r}$ as follows:

$$\begin{aligned}\int_0^T \int_V \Bigg[& \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \tilde{U_x} \\ & + \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \tilde{U_\theta} \\ & + \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \tilde{U_r} \Bigg] d\nu dt \end{aligned} \tag{A.3}$$

Integration by parts is performed to the above equation:

$$\begin{aligned}\int_0^T \int_{\partial V} \Bigg[& \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x \\ & + \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta \\ & + \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r \Bigg] \tilde{U} \cdot ds dt \\ & + \int_0^T \int_V \Bigg[\frac{\partial}{\partial x} \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \\ & + \frac{\partial}{r \partial \theta} \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \\ & + \frac{\partial}{\partial r} \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \Bigg] \tilde{U} \cdot d\nu dt \end{aligned} \tag{A.4}$$

Substituting the second volume integral in A.2 for A.4 and rearranging around the flow variable sensitivity gives:

$$\int_0^T \int_{\partial V} \Big[\lambda^T \left(A - Iu_g - A_v\right) n_x + \lambda^T \left(B - Iv_g - B_v\right) n_\theta + \lambda^T \left(C - Iw_g - C_v\right) n_r$$
$$- \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x$$
$$- \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta$$
$$- \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r \Big] \tilde{U} \cdot dS dt$$
$$+ \int_0^T \int_V \Big[\omega \left(\lambda_B^T \sin \omega t - \lambda_A^T \cos \omega t \right)$$
$$+ \frac{\partial \lambda^T}{\partial x} \left(A - Iu_g - A_v\right) + \frac{\partial \lambda^T}{r \partial \theta} \left(B - Iv_g - B_v\right) + \frac{\partial \lambda^T}{\partial r} \left(C - Iw_g - C_v\right) - \lambda^T D$$
$$- \frac{\partial}{\partial x} \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right)$$
$$- \frac{\partial}{r \partial \theta} \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right)$$
$$- \frac{\partial}{\partial r} \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \Big] \tilde{U} \cdot dv dt$$
$$- \int_V \lambda^T f dv = 0 \quad \text{(A.5)}$$

The left hand side of the above equation is substracted from the gradient equation 7.14. This operation does not change the original value of the gradient and gives:

$$\begin{aligned}
\frac{dI}{d\alpha} = &\int_0^T \int_{\partial V} \left(\frac{\partial M}{\partial U} \tilde{U} + \frac{\partial M}{\partial \alpha} \right) dS dt + \int_0^T \int_{\partial V} M \tilde{d} S dt \\
&\int_0^T \int_{\partial V} \Big[-\lambda^T \left(A - I u_g - A_v \right) n_x - \lambda^T \left(B - I v_g - B_v \right) n_\theta - \lambda^T \left(C - I w_g - C_v \right) n_r \\
&\qquad + \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) \cdot n_x \\
&\qquad + \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) \cdot n_\theta \\
&\qquad + \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) \cdot n_r \Big] \tilde{U} \cdot dS dt \\
&\qquad + \int_0^T \int_V \Big[-\omega \left(\lambda_B^T \sin \omega t - \lambda_A^T \cos \omega t \right) \\
&\quad - \frac{\partial \lambda^T}{\partial x} \left(A - I u_g - A_v \right) - \frac{\partial \lambda^T}{r \partial \theta} \left(B - I v_g - B_v \right) - \frac{\partial \lambda^T}{\partial r} \left(C - I w_g - C_v \right) + \lambda^T D \\
&\qquad + \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) \\
&\qquad + \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) \\
&\qquad + \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) \Big] \tilde{U} \cdot dv dt \\
&\qquad + \int_0^T \int_V \lambda^T f \cdot dv dt = 0 \qquad \text{(A.6)}
\end{aligned}$$

Rearranging the surface and volume integral terms around the flow variable sensitivity $\tilde{U}$:

$$\begin{aligned}
\int_0^T \int_{\partial V} \Big[& -\lambda^T \left(A - Iu_g - A_v\right) n_x - \lambda^T \left(B - Iv_g - B_v\right) n_\theta - \lambda^T \left(C - Iw_g - C_v\right) n_r \\
& + \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x \\
& + \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta \\
& + \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r + \frac{\partial M}{\partial U} \Big] \tilde{U} \cdot dS dt \\
& \int_0^T \int_V \Big[-\omega \left(\lambda_B^T \sin \omega t - \lambda_A^T \cos \omega t \right) \\
& - \frac{\partial \lambda^T}{\partial x} \left(A - Iu_g - A_v\right) - \frac{\partial \lambda^T}{r \partial \theta} \left(B - Iv_g - B_v\right) - \frac{\partial \lambda^T}{\partial r} \left(C - Iw_g - C_v\right) + \lambda^T D \\
& + \frac{\partial}{\partial x} \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \\
& + \frac{\partial}{r \partial \theta} \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \\
& + \frac{\partial}{\partial r} \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \Big] \tilde{U} \cdot dv dt \\
& + \int_0^T \int_{\partial V} \frac{\partial M}{\partial \alpha} dS dt + \int_0^T \int_{\partial V} M d\tilde{S} dt + \int_0^T \int_V \lambda^T f \cdot dv dt = 0
\end{aligned} \tag{A.7}$$

The flow variable sensitivity $\tilde{U}$ has to be eliminated out of the equation. This defines a domain (volume) integral and a boundary (surface) integral as follows:

$$\begin{aligned}
\int_0^T \int_{\partial V} - \Big[& \lambda^T \left(A - Iu_g - A_v\right) n_x + \lambda^T \left(B - Iv_g - B_v\right) n_\theta + \lambda^T \left(C - Iw_g - C_v\right) n_r \Big] \tilde{U} \\
& + \Big[\left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x \\
& + \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta \\
& + \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r + \frac{\partial M}{\partial U} \Big] \tilde{U} \cdot dS dt = 0
\end{aligned} \tag{A.8}$$

$$\int_0^T \int_V -\Big[\omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) + \frac{\partial \lambda^T}{\partial x}\left(A - Iu_g - A_v\right) + \frac{\partial \lambda^T}{r\partial\theta}\left(B - Iv_g - B_v\right) + \frac{\partial \lambda^T}{\partial r}\left(C - Iw_g - C_v\right)\Big]\tilde{U} + \Big[\lambda^T D + \frac{\partial}{\partial x}\left(\frac{\partial \lambda^T}{\partial x}D_{xx} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta x} + \frac{\partial \lambda^T}{\partial r}D_{rx}\right) + \frac{\partial}{r\partial\theta}\left(\frac{\partial \lambda^T}{\partial x}D_{x\theta} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r}D_{r\theta}\right) + \frac{\partial}{\partial r}\left(\frac{\partial \lambda^T}{\partial x}D_{xr} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta r} + \frac{\partial \lambda^T}{\partial r}D_{rr}\right)\Big]\tilde{U} \cdot dv\,dt = 0 \quad \text{(A.9)}$$

A.1 Domain Integral

The adjoint formulation requires both the domain and boundary integral to be vanished. Inside the computational domain, the sufficient and necessary condition is to vanish the term multiplying $\tilde{U}$. This defines the adjoint Navier-Stokes equations as follows:

$$-\Big[\omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) + \frac{\partial \lambda^T}{\partial x}\left(A - Iu_g - A_v\right) + \frac{\partial \lambda^T}{r\partial\theta}\left(B - Iv_g - B_v\right) + \frac{\partial \lambda^T}{\partial r}\left(C - Iw_g - C_v\right)\Big] + \lambda^T D + \frac{\partial}{\partial x}\left(\frac{\partial \lambda^T}{\partial x}D_{xx} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta x} + \frac{\partial \lambda^T}{\partial r}D_{rx}\right) + \frac{\partial}{r\partial\theta}\left(\frac{\partial \lambda^T}{\partial x}D_{x\theta} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r}D_{r\theta}\right) + \frac{\partial}{\partial r}\left(\frac{\partial \lambda^T}{\partial x}D_{xr} + \frac{\partial \lambda^T}{r\partial\theta}D_{\theta r} + \frac{\partial \lambda^T}{\partial r}D_{rr}\right) = 0 \quad \text{(A.10)}$$

In order to simplify the spatial discretization, the above equation can be rearranged around the spatial derivatives:

$$\omega\left(\lambda_B^T \sin\omega t - \lambda_A^T \cos\omega t\right) + \frac{\partial \lambda^T}{\partial x}\left(A - Iu_g - A_v\right) + \frac{\partial \lambda^T}{r\partial\theta}\left(B - Iv_g - B_v\right) + \frac{\partial \lambda^T}{\partial r}\left(C - Iw_g - C_v\right) - N^T\left(\frac{\partial D_x}{\partial x} + \frac{\partial D_\theta}{r\partial\theta} + \frac{\partial D_r}{\partial r}\right) - \lambda^T D = 0 \quad \text{(A.11)}$$

where:

$$N = \frac{\partial U}{\partial U_v},\ U_v = \left(p, u, v, w, T\right)^T$$

$$D_x = \left(0, \sigma_{xx}, \sigma_{\theta x}/r, \sigma_{rx}, \phi_x\right)^T$$

$$D_\theta = \left(0, \sigma_{x\theta}, \sigma_{\theta\theta}/r, \sigma_{r\theta}, \phi_\theta\right)^T$$

$$D_r = \left(0, \sigma_{xr}, \sigma_{\theta r}/r, \sigma_{rr}, \phi_r\right)^T$$

$$\sigma_{xx} = \frac{2}{3}\mu\left[2\left(\frac{\partial\lambda_2}{\partial x} + u\frac{\partial\lambda_5}{\partial x}\right) - \left(\frac{\partial\lambda_3}{r\partial\theta} + v\frac{\partial\lambda_5}{r\partial\theta}\right) - \left(\frac{\partial\lambda}{\partial r} + w\frac{\partial\lambda}{\partial r}\right)\right]$$

$$\sigma_{\theta\theta} = \frac{2}{3}\mu\left[2\left(\frac{\partial\lambda_3}{\partial\theta} + v\frac{\partial\lambda_5}{r\partial\theta}\right) - \left(\frac{\partial\lambda_2}{\partial x} + u\frac{\partial\lambda_5}{\partial x}\right) - \left(\frac{\partial\lambda_4}{\partial r} + w\frac{\partial\lambda_5}{\partial r}\right)\right]$$

$$\sigma_{rr} = \frac{2}{3}\mu\left[2\left(\frac{\partial\lambda_4}{\partial r} + w\frac{\partial\lambda_5}{\partial r}\right) - \left(\frac{\partial\lambda_3}{r\partial\theta} + v\frac{\partial\lambda_5}{r\partial\theta}\right) - \left(\frac{\partial\lambda_2}{\partial x} + u\frac{\partial\lambda_5}{\partial x}\right)\right]$$

$$\sigma_{x\theta} = \sigma_{\theta x} = \mu\left[\left(\frac{\partial\lambda_2}{r\partial\theta} + u\frac{\partial\lambda_5}{r\partial\theta}\right) + \left(r\frac{\partial\lambda_3}{\partial x} + v\frac{\partial\lambda_5}{\partial x}\right)\right]$$

$$\sigma_{xr} = \sigma_{rx} = \mu\left[\left(\frac{\partial\lambda_2}{\partial r} + u\frac{\partial\lambda_5}{\partial r}\right) + \left(\frac{\partial\lambda_4}{\partial x} + w\frac{\partial\lambda_5}{\partial x}\right)\right]$$

$$\sigma_{r\theta} = \sigma_{\theta r} = \mu\left[\left(\frac{\partial\lambda_4}{r\partial\theta} + w\frac{\partial\lambda_5}{r\partial\theta}\right) + \left(r\frac{\partial\lambda_3}{\partial r} + v\frac{\partial\lambda_5}{\partial r}\right)\right]$$

$$\phi_x = k\frac{\partial\lambda_5}{\partial x},\ \phi_\theta = k\frac{\partial\lambda_5}{r\partial\theta},\ \phi_r = k\frac{\partial\lambda_5}{\partial r}$$

A.2 Boundary Integral

In order to solve the boundary integral, the corresponding flow boundary conditions need to be derived in a way that the term multiplying $\tilde{U}$ is vanished in each boundary

condition:

$$\int_0^T \int_{\partial V} \Big[\lambda^T \left(A - I u_g - A_v \right) n_x + \lambda^T \left(B - I v_g - B_v \right) n_\theta + \lambda^T \left(C - I w_g - C_v \right) n_r$$
$$- \left(\frac{\partial \lambda^T}{\partial x} D_{xx} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta x} + \frac{\partial \lambda^T}{\partial r} D_{rx} \right) \cdot n_x$$
$$- \left(\frac{\partial \lambda^T}{\partial x} D_{x\theta} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r} D_{r\theta} \right) \cdot n_\theta$$
$$- \left(\frac{\partial \lambda^T}{\partial x} D_{xr} + \frac{\partial \lambda^T}{r \partial \theta} D_{\theta r} + \frac{\partial \lambda^T}{\partial r} D_{rr} \right) \cdot n_r - \frac{\partial M}{\partial U} \Big] \tilde{U} \cdot dS dt = 0 \quad \text{(A.12)}$$

This is done in Chapter 7 for the inlet and exit adjoint boundary conditions and in Appendix B for the solid wall boundary condition.

Appendix B

Unsteady Adjoint Navier-Stokes Solid Wall Boundary Condition

This appendix shows in detail the derivation of each of the adjoint solid wall boundary integral terms derived from the URANS equations.

B.1 Inviscid terms

The inviscid convective flux of the URANS equations through a control face can be written as:

$$\left(F - U u_g\right) \cdot n_x + \left(G - U v_g\right) \cdot n_\theta + \left(H - U w_g\right) \cdot n_r = U \cdot v_n + \begin{pmatrix} 0 \\ p \cdot n_x \\ p \cdot r \cdot n_\theta \\ p \cdot n_r \\ p \cdot \left(u_g \cdot n_x + v_g \cdot n_\theta + w_g \cdot n_r\right) \end{pmatrix} \tag{B.1}$$

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

$$U \cdot v_n = 0 \tag{B.2}$$

Hence:

$$\left(F-Uu_g\right)\cdot n_x+\left(G-Uv_g\right)\cdot n_\theta+\left(H-Uw_g\right)\cdot n_r=\begin{pmatrix}0\\ p\cdot n_x\\ p\cdot r\cdot n_\theta\\ p\cdot n_r\\ p\cdot\left(u_g\cdot n_x+v_g\cdot n_\theta+w_g\cdot n_r\right)\end{pmatrix} \tag{B.3}$$

From Appendix A, the inviscid boundary integral is defined as:

$$\int_0^T\int_{\partial V}\left[\lambda^T\left(A-Iu_g\right)n_x+\lambda^T\left(B-Iv_g\right)n_\theta+\lambda^T\left(C-Iw_g\right)n_r-\frac{\partial M}{\partial U}\right]\tilde{U}\cdot dSdt=0 \tag{B.4}$$

The only independent primitive variable in the convective flux above is the pressure. Hence, only the derivative of the flux vector to the pressure yields a value. The equation below can be written as:

$$\int_0^T\int_{\partial V}\left[\lambda^T\left(\frac{\partial F}{\partial p}-Iu_g\right)n_x+\lambda^T\left(\frac{\partial G}{\partial p}-Iv_g\right)n_\theta+\lambda^T\left(\frac{\partial H}{\partial p}-Iw_g\right)n_r-\frac{\partial M}{\partial p}\right]\tilde{p}\cdot dSdt=0 \tag{B.5}$$

The convective flux vector (Equation B.3) can be added to the equation above, which gives:

$$\int_S\left[\lambda_2\cdot n_x+r\lambda_3\cdot n_\theta+\lambda_4\cdot n_r+\lambda_5\left(u_g\cdot n_x+v_g\cdot n_\theta+w_g\cdot n_r\right)-\frac{\partial M}{\partial p}\right]\tilde{p}dS=0 \tag{B.6}$$

Same as with the inlet and exit boundary conditions, $\tilde{p}$ can not be zero in the whole wall. Hence, the sufficient and necessary condition is to vanish the term in the left hand side

of the equation as follows:

$$\lambda_2 \cdot n_x + r\lambda_3 \cdot n_\theta + \lambda_4 \cdot n_r + \lambda_5 \left(u_g \cdot n_x + v_g \cdot n_\theta + w_g \cdot n_r\right) = \frac{\partial M}{\partial p} \tag{B.7}$$

The slip boundary condition for a viscous solid wall also needs to be implemented. The next equalities based on the log-law (Equation 3.26) and wall shear stress are also added:

$$\sigma_x = \frac{\tau_w}{W}\lambda_2 \tag{B.8a}$$

$$\sigma_\theta = \frac{\tau_w}{W}\left(\lambda_3 + \Omega\lambda_5\right) \tag{B.8b}$$

$$\sigma_r = \frac{\tau_w}{W}\lambda_4 \tag{B.8c}$$

If the wall is adiabatic:

$$\frac{\partial \lambda_5}{\partial n} = 0 \tag{B.9}$$

If the wall is non-adiabatic:

$$\lambda_5 = 0 \tag{B.10}$$

B.2 Viscous terms

The viscous part of the adjoint boundary integral (Equation 7.41) is divided in two integral terms: one with the Jacobian matrices from the viscous flow vector and the second term with the additional spatial derivative terms. These two parts are developed separately.

Adjoint viscous flux vector

The adjoint viscous flow vector from Equation 7.41 can be written as:

$$\int_0^T \int_{\partial V} \left[\lambda^T \left(\frac{\partial V_x}{\partial U} n_x + \frac{\partial V_\theta}{\partial U} n_\theta + \frac{\partial V_r}{\partial U} n_r\right)\right] \tilde{U} \cdot dS dt \tag{B.11}$$

From the flow equation, the viscous flow vectors can be approximated with the log-law for shear stress as follows:

$$V_x n_x + V_\theta n_\theta + V_r n_r = \begin{pmatrix} 0 \\ \tau_x \\ \tau_\theta r \\ \tau_r \\ \tau_\theta \Omega r + k\frac{\partial T}{\partial n} \end{pmatrix} \tag{B.12}$$

where $\tau_x, \tau_\theta, \tau_r$ are the components of the wall shear stress τ_w in the x, θ and r directions respectively. These components can be written in function of the wall shear stress τ_w as follows:

$$\tau_x = \tau_w \cdot \frac{u}{W} \tag{B.13a}$$

$$\tau_\theta = \tau_w \cdot \frac{v_{rel}}{W} \tag{B.13b}$$

$$\tau_r = \tau_w \cdot \frac{w}{W} \tag{B.13c}$$

Substituting the approximation done in Equation B.12 for Equation B.11 yields:

$$\left[\lambda^T \left(\frac{\partial V_x}{\partial U} n_x + \frac{\partial V_\theta}{\partial U} n_\theta + \frac{\partial V_r}{\partial U} n_r\right)\right] \tilde{U} = \lambda_2 \tilde{\tau_x} + \lambda_3 \tilde{\tau_\theta} r + \lambda_4 \tilde{\tau_r} + \lambda_5 \left(\tilde{\tau_\theta} \Omega r + k\frac{\partial \tilde{T}}{\partial n}\right) = 0 \tag{B.14}$$

The stress terms in Equation B.14 are linked to the wall shear stress τ_w (Equation 3.26), which is a function of u, v, w and ρ. The sensitivity of these stress terms in terms of the

flow variable sensitivity is calculated as follows:

$$\tilde{\tau_x} = \frac{\partial \tau_x}{\partial u}\tilde{u} + \frac{\partial \tau_x}{\partial v}\tilde{v} + \frac{\partial \tau_x}{\partial w}\tilde{w} + \frac{\partial \tau_x}{\partial \rho}\tilde{\rho} \tag{B.15a}$$

$$\tilde{\tau_\theta} = \frac{\partial \tau_\theta}{\partial u}\tilde{u} + \frac{\partial \tau_\theta}{\partial v}\tilde{v} + \frac{\partial \tau_\theta}{\partial w}\tilde{w} + \frac{\partial \tau_\theta}{\partial \rho}\tilde{\rho} \tag{B.15b}$$

$$\tilde{\tau_r} = \frac{\partial \tau_r}{\partial u}\tilde{u} + \frac{\partial \tau_r}{\partial v}\tilde{v} + \frac{\partial \tau_r}{\partial w}\tilde{w} + \frac{\partial \tau_r}{\partial \rho}\tilde{\rho} \tag{B.15c}$$

Adding this to Equation B.14 gives:

$$\begin{aligned}
&\lambda_2\tilde{\tau_x} + \lambda_3\tilde{\tau_\theta}r + \lambda_4\tilde{\tau_r} + \lambda_5\left(\tilde{\tau_\theta}\Omega r + k\frac{\partial \tilde{T}}{\partial n}\right) = \\
&\left[\lambda\frac{\partial \tau_x}{\partial u} + (\lambda_3 + \lambda_5\Omega)\, r\frac{\partial \tau\theta}{\partial u} + \lambda_4\frac{\partial \tau_r}{\partial u}\right]\tilde{u} + \\
&\left[\lambda\frac{\partial \tau_x}{\partial v} + (\lambda_3 + \lambda_5\Omega)\, r\frac{\partial \tau\theta}{\partial v} + \lambda_4\frac{\partial \tau_r}{\partial v}\right]\tilde{v} + \\
&\left[\lambda\frac{\partial \tau_x}{\partial w} + (\lambda_3 + \lambda_5\Omega)\, r\frac{\partial \tau\theta}{\partial w} + \lambda_4\frac{\partial \tau_r}{\partial w}\right]\tilde{w} + \\
&\left[\lambda\frac{\partial \tau_x}{\partial \rho} + (\lambda_3 + \lambda_5\Omega)\, r\frac{\partial \tau\theta}{\partial \rho} + \lambda_4\frac{\partial \tau_r}{\partial \rho}\right]\tilde{\rho} + \\
&\qquad + \lambda_5 k\frac{\partial \tilde{T}}{\partial n}
\end{aligned} \tag{B.16}$$

Adjoint spatial derivative terms

The remaining elements of 7.41 are the spatial derivative terms, which are written below:

$$\begin{aligned}
&\left[\left(\frac{\partial \lambda^T}{\partial x}D_{xx} + \frac{\partial \lambda^T}{r\partial \theta}D_{\theta x} + \frac{\partial \lambda^T}{\partial r}D_{rx}\right)\cdot n_x\right. \\
&+\left(\frac{\partial \lambda^T}{\partial x}D_{x\theta} + \frac{\partial \lambda^T}{r\partial \theta}D_{\theta\theta} + \frac{\partial \lambda^T}{\partial r}D_{r\theta}\right)\cdot n_\theta \\
&\left.+\left(\frac{\partial \lambda^T}{\partial x}D_{xr} + \frac{\partial \lambda^T}{r\partial \theta}D_{\theta r} + \frac{\partial \lambda^T}{\partial r}D_{rr}\right)\cdot n_r\right]\tilde{U}
\end{aligned} \tag{B.17}$$

The viscous stress terms above can also be written in the next form:

$$X_1\cdot\tilde{\rho} + X_2\cdot\tilde{u} + X_3\cdot r\cdot\tilde{v} + X_4\cdot\tilde{w} + X_5 \tag{B.18}$$

These terms are spatial derivatives of velocity components only. Consequently**:**

$$X_1 = X_5 = 0 \tag{B.19}$$

Cancellation of the viscous terms in the boundary integral

Both the adjoint viscous flux vector and the spatial derivative terms have been rewritten in terms of the independent flow variables $\tilde{\rho}, \tilde{u}, \tilde{v}, \tilde{w}$ and $\tilde{p}$. Because these variables can not be zero in the whole wall, the sufficient and necessary condition is to vanish all the terms multiplying the independent flow variables. Hence, Equations B.16 and B.18 must satisfy the next equalities:

$$\lambda_2 \frac{\partial \tau_x}{\partial \rho} + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial \tau_\theta}{\partial \rho} + \lambda_4 \frac{\partial \tau_r}{\partial \rho} = 0 \tag{B.20a}$$

$$\lambda_2 \frac{\partial \tau_x}{\partial u} + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial \tau_\theta}{\partial u} + \lambda_4 \frac{\partial \tau_r}{\partial u} = X_2 \tag{B.20b}$$

$$\lambda_2 \frac{\partial \tau_x}{\partial v} + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial \tau_\theta}{\partial v} + \lambda_4 \frac{\partial \tau_r}{\partial v} = X_3 r \tag{B.20c}$$

$$\lambda_2 \frac{\partial \tau_x}{\partial w} + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial \tau_\theta}{\partial w} + \lambda_4 \frac{\partial \tau_r}{\partial w} = X_4 \tag{B.20d}$$

In order to obtain the values of X_2, X_3 and X_4 the shear stresses can be rewritten in terms of the wall shear stresses as follows:

$$f = \frac{\tau_w}{W} \rightarrow \tau_x = f\,u,\ \tau_\theta = f\,v_{rel},\ \tau_r = f\,w \tag{B.21}$$

The stress terms sensitivities are now defined as:

$$\begin{array}{lll}
\frac{\partial \tau_x}{\partial u} = \frac{\partial f}{\partial u} u + f & \frac{\partial \tau_\theta}{\partial u} = \frac{\partial f}{\partial u} v_{rel} & \frac{\partial \tau_r}{\partial u} = \frac{\partial f}{\partial u} w \\
\frac{\partial \tau_x}{\partial v} = \frac{\partial f}{\partial v} u & \frac{\partial \tau_\theta}{\partial v} = \frac{\partial f}{\partial v} v_{rel} + f & \frac{\partial \tau_r}{\partial v} = \frac{\partial f}{\partial v} w \\
\frac{\partial \tau_x}{\partial w} = \frac{\partial f}{\partial w} u & \frac{\partial \tau_\theta}{\partial w} = \frac{\partial f}{\partial w} v_{rel} & \frac{\partial \tau_r}{\partial w} = \frac{\partial f}{\partial w} w + f \\
\frac{\partial \tau_x}{\partial \rho} = \frac{\partial f}{\partial \rho} u & \frac{\partial \tau_\theta}{\partial \rho} = \frac{\partial f}{\partial \rho} v_{rel} & \frac{\partial \tau_r}{\partial \rho} = \frac{\partial f}{\partial \rho} w
\end{array} \tag{B.22}$$

As a result, the system of Equations B.20 can be written again as follows:

$$\lambda_2 \frac{\partial f}{\partial \rho} u + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial f}{\partial \rho} v_{rel} + \lambda_4 \frac{\partial f}{\partial \rho} w = 0 \tag{B.23a}$$

$$\lambda_2 \left(\frac{\partial f}{\partial u} u + f \right) + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial f}{\partial u} v_{rel} + \lambda_4 \frac{\partial f}{\partial u} w = X_2 \tag{B.23b}$$

$$\lambda_2 \frac{\partial f}{\partial v} u + (\lambda_3 + \lambda_5 \Omega)\, r \left(\frac{\partial f}{\partial v} v_{rel} + f \right) + \lambda_4 \frac{\partial f}{\partial v} w = X_3 r \tag{B.23c}$$

$$\lambda_2 \frac{\partial f}{\partial w} + (\lambda_3 + \lambda_5 \Omega)\, r \frac{\partial f}{\partial w} v_{rel} + \lambda_4 \left(\frac{\partial f}{\partial w} + f \right) = X_4 \tag{B.23d}$$

The three unknowns X_2, X_3 and X_4 can now be solved:

$$X_2 = f \cdot \lambda_2 \tag{B.24a}$$

$$X_3 = f \cdot (\lambda_3 + \lambda_5 \Omega) \tag{B.24b}$$

$$X_4 = f \cdot \lambda_4 \tag{B.24c}$$

In fact, X_2, X_3 and X_4 are the terms σ_x, σ_θ and σ_r respectively, due to the implementation of the slip boundary condition (Equation 3.26):

$$\sigma_x = \frac{\tau_w}{W} \lambda_2 \tag{B.25a}$$

$$\sigma_\theta = \frac{\tau_w}{W} (\lambda_3 + \Omega \lambda_5) \tag{B.25b}$$

$$\sigma_r = \frac{\tau_w}{W} \lambda_4 \tag{B.25c}$$

The last unknown X_5, is defined so it can cancel the last term in Equation B.16:

$$X_5 = k \frac{\partial \lambda_5}{\partial n} \tag{B.26}$$

The last condition depends whether if the viscous solid wall is assumed to be adiabatic.

If it is adiabatic, then $\frac{\partial T}{\partial n} = 0$ and $\tilde{T} \neq 0$ and:

$$\frac{\partial \lambda_5}{\partial n} = 0 \tag{B.27}$$

If a temperature field or value is specified at any point of the viscous solid wall, then $\tilde{T} = 0$, and $\frac{\partial T}{\partial n} \neq 0$, which leads to:

$$\lambda_5 = 0 \tag{B.28}$$