Politecnico di Milano

SCHOOL OF INDUSTRIAL AND INFORMATION ENGINEERING
Master of Science in Aeronautical Engineering

Design and preliminary optimization of a supersonic turbine for Rotating Detonation Engine

Supervisor
Prof. Paolo GAETANI

Co-Supervisor
Prof. Giacomo PERSICO

Candidate
Noraiz MUSHTAQ - 913230

Academic Year 2019 – 2020

*A mio padre,
a cui devo tutto e di più.*

Acknowledgements

I would like to express my deepest gratitude to Prof. Paolo Gaetani, for giving me, first of all, the opportunity to work on such an interesting topic, for listening and advising me and for helping me overcoming the many difficulties encountered during this thesis.
I would like to extend my sincere thanks to Prof. Giacomo Persico not only for providing me an extraordinary optimization code, but also for dedicating me many hours. The work on the shape optimization would not have been possible without you.
I would like to express my sincere appreciation to my friend and colleague Gabriele Colella. We started this project together in March 2019 and it has been a great pleasure working with you on the first six chapters of this thesis. Thank you.

I am extremely grateful to my father and my mother for their unconditional love. You have supported me in every decision and always offered me the possibility to follow my passions. Thank you for everything you have done for me.
I cannot thank enough my brother Husnain, who has not only been an exceptional big brother, but also my mentor; whatever success I achieved during my studies cannot have been possible without you guiding me through it, starting from that astonishing fish we made together on the first day of elementary school.
I would like to thank my sister Iram, who had the "pleasure" of growing with three brothers. Thank you for always putting up with us and we will always be there for you.
I also wish to thank my brother Junaid for patiently reading and correcting all the reports I have written in these years and for broadening my knowledge by suggesting me books, articles and films.
Special thanks to my nephews Zoya, Haris, Ahmad and Alia. Playing and laughing with you it is always one of the best moments of my day.

I would like to thank my friends Awais, Usman M. and Usman Q. We have known each other for so many years and we have done so many things together. Thank you for always being by my side.
Many thanks to my friend Amedeo; we met on the first day of university and from that point on it has been an amazing five year journey together. I also wish to thank Federico, Filippo, Stefano and all the friends that I made at Politecnico di Milano: I came to PoliMi to become an engineer, but unexpectedly the most precious thing with which I am leaving is your friendship.

Finally to all the people that I did not mention above, but have significantly contributed to my life: I would like to thank all of you one by one, but unfortunately I cannot. Thank you!

Milano, 18/11/2020

Noraiz Mushtaq

Abstract

The aim of this work is the design of a supersonic turbine. To improve the efficiency of gas turbines, recently it was investigated the possibility to adopt, in place of a classical combustor, a Rotating Detonation Engine (RDE). This is a novel technology based on the detonation combustion mode and thermodynamic cycle studies have shown, at least theoretically, optimistic results in terms of performance, efficiency, emission of pollutants and CO_2. The flow coming out from the detonative combustor is supersonic, non-uniform and unsteady. Modern conventional turbines are not able to withstand this kind of flow as they generate a tremendous amount of losses. For this reason a pioneering supersonic turbine must be designed. In this thesis, the turbine design was approached with a mean-line code, whose goal is to find the optimal design parameters, to assure the starting of the blade passages and to create a suitable profile for the incoming supersonic flow. The design of the supersonic blade profile has been carried out employing the vortex-flow method, an implicit version of the Method of Characteristics that works with two dimensional, steady and inviscid flows. The resulting blade shapes have been thickened and properly rounded, inducing a detached bow shock wave in front of the leading edge. In the mean-line code, this shock has been modelled, along with its first and stronger reflection, and the resulting losses have been taken into account to design the turbine. Afterwards, the thesis work was focused on finding a proper loading criteria to define the optimum number of blades, but the classic loading criteria are not thought for supersonic rows, so an optimization procedure has been implemented. For both stator and rotor rows, the performance has been optimized by varying Zweifel's tangential force coefficient. The optimization algorithm adopted was the Golden-section search method and the entropy generated was evaluated by means of blade-to-blade simulations in Ansys CFX. The results have been used also to validate the predictions of the mean-line code. After that, by means of blade-to-blade simulations, off-design conditions have been studied for the stator row and for the full turbine stage by varying the inlet Mach number and the inlet flow angle. Positive incidence angles and lower inlet Mach numbers than the design value have been found to not excessively penalize the turbine stage performance. Finally, the design of the supersonic turbine is further enhanced through a high-fidelity shape optimization by surrogate-based evolutionary technique: this procedure is performed through FORMA, a code developed by the laboratory of fluid machines at Politecnico di Milano. This final optimization represents a logical step in the profile improvement as it modifies the shape considering also the boundary layer and its interaction with the impinging oblique shock wave. Two different optimal profiles are presented for the stator and each of them has its strengths and weaknesses, but most importantly the produced entropy is decreased by around 8%.

Keywords: Supersonic turbine design; Rotating Detonation Engine; Shock waves prediction models; Method of Characteristics; Solidity optimization; Off-design performance; Shape optimization; B-splines parametrization; Genetic algorithms; Kriging surrogate model.

Sommario

Lo scopo di questo lavoro è la progettazione di una turbina supersonica. Per migliorare l'efficienza delle turbine a gas, recenti ricerche hanno studiato la possibilità di adottare un Motore a Detonazione Rotante (RDE), al posto del classico combustore. Si tratta di una nuova tecnologia basata sulla modalità di combustione a detonazione e studi sul ciclo termodinamico hanno mostrato, almeno teoricamente, risultati ottimistici in termini di prestazioni, efficienza e riduzione delle emissioni inquinanti. Il flusso che esce dal combustore detonante è supersonico, non uniforme e instabile. I moderni stadi a turbina hanno dimostrato di non essere in grado di sopportare questo tipo di flusso senza un enorme aumento delle perdite. Per questo motivo è necessario progettare una innovativa turbina supersonica. In questa tesi, il progetto della turbina è stato affrontato con un mean-line code, il cui scopo è determinare i parametri di progetto ottimi al fine di assicurare l'avvio dei canali palari e di creare un profilo adatto al flusso supersonico entrante. Il design del profilo della pala supersonica è stato realizzato utilizzando il vortex-flow method, una versione implicita del Metodo delle Caratteristiche per flussi bidimensionali, stazionari e non viscosi. Le pale risultanti sono state ispessite e adeguatamente arrotondate, inducendo un'onda d'urto distaccata davanti al bordo d'attacco. Nel mean-line code, questo shock è stato opportunamente modellato, insieme alla sua prima e più forte riflessione, e le perdite risultanti sono state prese in considerazione per progettare la turbina. Successivamente, il lavoro di tesi si è concentrato sulla ricerca di un corretto criterio di carico per definire il numero ottimo di pale, ma i classici criteri di carico non sono pensati per le schiere supersoniche, quindi è stata implementata una procedura di ottimizzazione. Sia per la schiera statorica che per quella rotorica, le prestazioni sono state ottimizzate variando il coefficiente di forza tangenziale di Zweifel. L'algoritmo di ottimizzazione adottato è stato il metodo della sezione aurea e l'entropia generata è stata valutata tramite simulazioni blade-to-blade in Ansys CFX. I risultati sono stati utilizzati anche per la validazione del mean-line code. Dopo di che, mediante simulazioni blade-to-blade, è stato valutato il comportamento fuori progetto sia dello statore che dell'intero stadio di turbina variando in ingresso il numero di Mach e l'angolo di flusso. È stato riscontrato che angoli di incidenza positivi e numero di Mach in ingresso più bassi del valore di progetto non penalizzano eccessivamente le prestazioni dello stadio. Infine, il design della turbina supersonica è stato ulteriormente migliorato con un'ottimizzazione della forma ad alta fedeltà tramite metodi evoluzionistici basati su modelli surrogati: tale procedura è stata possibile grazie a FORMA, un codice sviluppato dal laboratorio di fluidodinamica delle macchine del Politecnico di Milano. Questa tipologia di ottimizzazione rappresenta il passaggio logico corretto per il miglioramento del profilo, poiché modifica la forma tenendo conto anche dello strato limite e della sua interazione con gli urti obliqui. Due diversi profili ottimi vengono proposti per lo statore e ciascuno ha i suoi punti di forza e le sue lacune, però l'aspetto di primaria importanza è una riduzione intorno all' 8% dell'entropia prodotta.

Parole chiave: Progettazione di turbina supersonica; Motore a Detonazione Rotante; Modelli di previsione delle onde d'urto; Metodo delle Caratteristiche; Ottimizzazione della solidity; Prestazioni fuori progetto; Ottimizzazione della forma; Algoritmi genetici; Modello surrogato di Kriging.

Contents

Acknowledgements **v**

Abstract **vii**

Sommario **ix**

Contents **xiii**

List of Figures **xxii**

List of Tables **xxiii**

1 Pressure-Gain Combustion **1**
1.1 Detonation Propulsion 3
1.2 Cycle Analysis 4
1.3 Pulse Detonation Engine 7
1.4 Rotating Detonation Engine 8
1.4.1 Combustion Mechanism 8
1.4.2 Flow solver 10
1.4.3 Gas Turbine architecture with RDE 12
1.4.4 Future Challenges 14
1.5 Supersonic Turbine 16

2 Mean-line design **21**
2.1 List of symbols 22
2.2 Started or unstarted blade row 24
2.3 Evaluation and prediction of aerodynamic losses 26
2.3.1 Detached bow shock wave 27
2.3.2 First oblique shock wave reflection 31
2.3.3 Boundary layer and mixing losses 32
2.4 Blade profile design through the method of characteristics (MOC) . . 39
2.4.1 Circular arcs 43
2.4.2 Lower inlet transition arc 45
2.4.3 Upper inlet transition arc, lower outlet transition arc and upper outlet transition arc 47
2.4.4 Some examples and blade design limitations 48
2.4.5 From dimensionless to a finite thickness dimensional blade profile 52
2.5 Mean-line code: description and logic explanation 54
2.5.1 Variant A 57

2.5.2 Variant B 76
2.6 Mean-line code results validation through CFD simulations 82

3 Computation Fluid Dynamic 87
3.1 Navier-Stokes equations 87
3.2 Turbulence modelling 88
3.2.1 Reynolds Averaged Navier-Stokes equations 90
3.2.2 k-ω SST 94
3.3 Turbulent Boundary Layer 95
3.4 Finite Volume Method - Ansys CFX 97
3.4.1 Steps of a CFD simulation 98
3.4.2 Domain discretization 100
3.4.3 Integration and Discretization of Governing Equations 101
3.4.4 Linear Solver 104
3.4.5 Solving Procedures 106

4 Computational Models for Turbomachinery Flows 109
4.1 Spanwise model 109
4.2 Blade-to-blade simulation 112
4.3 Coupled stator-rotor simulation 113
4.3.1 Mixing-plane 114
4.3.2 Transient Blade Row Models 115

5 Research of optimal solidity by an automated optimization process 117
5.1 Solidity effect on rounded blade profile 118
5.2 Loading Criteria 118
5.2.1 Zweifel criterium 120
5.2.2 Upper limit for C_{Ft} 121
5.3 In-depth description of the automated optimization process 124
5.4 Optimal Solution 129
5.4.1 Stator Cascade 130
5.4.2 Rotor Cascade 132
5.5 Outlet flow angle analysis 134

6 2D turbine stage simulation 137
6.1 Full stage simulation in design condition 137
6.2 Frozen Gust Analysis 141
6.3 Off-design performances 144
6.3.1 Full stage: inlet flow angle variation 144
6.3.2 Stator row: inlet flow angle variation 148
6.3.3 Full stage: inlet Mach number variation 150
6.3.4 Stator row: inlet Mach number variation 153

7 High-fidelity shape optimization by surrogate-based evolutionary techniques 159
7.1 Detailed description of the FORMA package 160
7.1.1 Geometrical parametrization 160

7.1.2 Genetic algorithms 163
7.1.3 Surrogate model and design of experiments (DOE) 166
7.1.4 Surrogate-based optimization 167
7.1.5 Code implementation of FORMA 169
7.2 Stator profile shape optimization by FORMA 170
7.2.1 Case A 174
7.2.2 Case B 179
7.3 Some final considerations on the shape optimization 185

Conclusions **187**

A k-omega SST equations **191**

B Coupled Solver Flow Diagram **193**

C Mesh Independence Analysis **195**
C.1 Mesh refinement in Boundary Layer 196

D Coordinate conversion **199**

Acronyms **201**

Bibliography **209**

List of Figures

Figure 1.1 Idealized Brayton Cycle 1

Figure 1.2 Gas turbine evolution, 1985–2016 [1] 2

Figure 1.3 Gas turbine combined-cycle evolution, 1985–2016. [1] 3

Figure 1.4 (a) classical turbofan engine. b) turbofan engine equipped with Rotating Detonation Engine (RDE). [2] 4

Figure 1.5 Thermodynamic Brayton-Joule, Humphrey and RDE engine (Fickett- Jacobs) cycles [3] 6

Figure 1.6 Thermodynamic Brayton-Joule, Humphrey and RDE engine (Fickett- Jacobs) cycles [4] 6

Figure 1.7 First Pulse Detonation Engine (PDE) designed by J.A. Nicholls at the University of Michigan 7

Figure 1.8 (a) Three-dimensional Computational Fluid Dynamic (CFD) simulation of a stable Rotating Detonation Combustor (RDC). b) 3D flow features on a RDE with a clock-wise rotating detonation front. [5] 8

Figure 1.9 Numerical schlieren-like visualization of (top) temperature and (bottom) pressure of an "unrolled" hydrogen-air RDE solution [6]. . . 9

Figure 1.10 2D unfolded illustration of the flow features and identification of the corresponding angles of a rotating detonation wave front [5] . . 10

Figure 1.11 Comparison of the flow features contour obtained with the two-dimensional Unsteady Reynolds Averaged Navier–Stokes (URANS) solver (top) with the ones obtained with the Method of Characteristics (MOC) (bottom) [5] 11

Figure 1.12 from the left to the right: inlet static pressure; outlet static pressure; outlet Mach number [7]. 11

Figure 1.13 detonating combustion annulus [3]. 12

Figure 1.14 a) Combustor shape, no nozzle; b) straight duct nozzle; c) conical nozzle, with a constant radius inner wall; d) Bezier outer wall with a constant radius inner wall; e) Bezier inner wall with a constant radius outer wall; and f) two Bezier surfaces, combining the outer wall of d and inner wall of e. [7]. 13

Figure 1.15 Isocontours of static temperature along the inner and outer walls for the baseline and five different nozzle geometries: a) baseline (no nozzle), b) straight duct, c) conical nozzle, d) Bezier outer wall, e) Bezier inner wall, and f) two Bezier surfaces. [7]. 13

Figure 1.16 Basic engine geometry and stage designations (not to scale). [8]. 14

Figure 1.17 Possible injection conditions for fuel and oxidizer: - top, sonic injection ($p_w < p_{cr},$), premixed region expands almost linearly; - middle subsonic injection, ($p_o > p_w > p_{cr}$); - bottom, no injection ($p_w > p_{cr},$) (reverse flow and flashback risk). [3]. 15
Figure 1.18 Iso-Temperature contour on a three dimensional RDC. A – Detonation front, B – Oblique shock wave, C – contact surface, D – Blocked injection due to the higher post-detonation pressure, E – Refilling zone, F – slip-line. [9] 16
Figure 1.19 Supersonic turbine stage concepts [4] 17
Figure 1.20 Schlieren, iso-Mach number contours and static pressure distribution on the rotor passage at three different inlet flow angles (configuration (c)) [4] 18
Figure 1.21 (a) Illustration of the characteristic net and the empirically predicted shock waves. (b) Numerical Schlieren visualization obtained with 3D RANS simulations. [10] 19

Figure 2.1 Comparison between started and unstarted condition for a supersonic blade row (Figure from [11]). 24
Figure 2.2 Total pressure losses through a normal shock wave. 25
Figure 2.3 Contraction ratio for a started or unstarted blade row with varying inlet Mach number. 26
Figure 2.4 Detached shock wave and notation used in analysis (Figure from [12]). 27
Figure 2.5 Charts for the determination of supersonic air flow against inclined planes and axially symmetric cones [12]. 28
Figure 2.6 Comparison between the detached shock wave shape obtained with Moeckel theory and CFD. 29
Figure 2.7 Parametric analysis on the total pressure losses caused by bow shock wave. 30
Figure 2.8 Regular reflection from a solid boundary (Figure from [13]). . 31
Figure 2.9 Comparison between the reflected oblique shock wave determined analytically (in purple) and CFD. 32
Figure 2.10 Trend of the boundary layer quantities ratios with respect to Mach number. 34
Figure 2.11 Stations nomenclature for boundary and mixing losses calculation (Figure from [14]). 35
Figure 2.12 Flowchart to illustrate the iterative process used to calculate the boundary and mixing losses. 38
Figure 2.13 Design of ideal supersonic passage by method of characteristic (Figure from [15]). 39
Figure 2.14 Domain of dependence and region of influence (Figure from [13]). 40
Figure 2.15 Supersonic blade row design by the corner-flow method (Figure from [15]). 41
Figure 2.16 Velocity distribution comparison between corner-flow method and vortex-flow method. 41
Figure 2.17 Blade profile division in vortex-flow method. 42

Figure 2.18 Dimensionless supersonic blade profile geometry for vortex-flow method (Figure from [15]). 44
Figure 2.19 Lower inlet transition arc construction (Figure from [15]). . . . 45
Figure 2.20 Supersonic blade profiles produced by the vortex-flow method at varying upper and lower surface Mach number and total flow turning. Common inputs are inlet and outlet Mach number $M_i = M_o = 2.5$ and specific heat ratio $\gamma = 1.4$. 49
Figure 2.21 Flowchart to illustrate the procedure used to calculate the supersonic blade profile using the method of characteristics. 51
Figure 2.22 Finite thickness dimensional blade profile. 53
Figure 2.23 Assembly of a rotating detonation combustor with a diverging nozzle downstream (Figure from [16]). 55
Figure 2.24 Variant A turbine stage layout. 57
Figure 2.25 Flowchart to illustrate the input and output of the MATLAB function that generates the stator profile using the method of characteristics. 59
Figure 2.26 Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the detached bow-shock wave and its first reflections in the stator. 59
Figure 2.27 Detached bow-shock wave and its first reflection for the stator in variant A 60
Figure 2.28 Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the boundary layer and mixing in the stator. 60
Figure 2.29 While cycle convergence for stator outlet properties. 62
Figure 2.30 Absolute and relative velocity distribution at stator outlet obtained using constant nozzle-angle solution (variant A). 64
Figure 2.31 Absolute and relative angle distribution at stator outlet obtained using constant nozzle-angle solution (variant A). 64
Figure 2.32 Flowchart to illustrate the iterative process used to design the stator in variant A of the mean-line code. 65
Figure 2.33 Flowchart to illustrate the input and output of the MATLAB function that generates the rotor profile using the method of characteristics. 68
Figure 2.34 Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the detached bow-shock wave and its first reflections in the rotor. 68
Figure 2.35 Detached bow-shock wave and its first reflection for the rotor in variant A 69
Figure 2.36 Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the boundary layer and mixing in the rotor. 70
Figure 2.37 While cycle convergence for rotor outlet properties. 72
Figure 2.38 Velocity triangles at hub, mid and tip section of the rotor (variant A). 73
Figure 2.39 Flowchart to illustrate the iterative process used to design the rotor in variant A of the mean-line code. 74

Figure 2.40 Flowchart to illustrate the iterative process used to design the stator in variant B of the mean-line code. 78
Figure 2.41 Detached bow-shock wave and its first reflection for the stator in variant B. 79
Figure 2.42 Absolute and relative angle distribution at stator outlet obtained using constant nozzle-angle solution (variant B). 79
Figure 2.43 Detached bow-shock wave and its first reflection for the rotor in variant B 80
Figure 2.44 Absolute and relative angle distribution at rotor outlet obtained using constant nozzle-angle solution (variant B). 80
Figure 2.45 Variant B turbine stage layout. 81
Figure 2.46 Velocity triangles at hub, mid and tip section of the rotor (variant B). 81
Figure 2.47 Entropy production comparison between the mean-line code and the CFD simulations. 83
Figure 2.48 Total pressure losses comparison between the mean-line code and the CFD simulations for the stator. 83
Figure 2.49 Stator outlet total pressure comparison between the mean-line code and the CFD simulations. 84
Figure 2.50 Standard deviation of the stator outlet flow angle obtained from CFD simulations and calculated at a chord distance from the trailing edge. 85

Figure 3.1 Turbulent eddies scale 89
Figure 3.2 Favre averaged Navier-Stokes equations from [17] 91
Figure 3.3 Boundary layer - dimensionless velocity profile 96
Figure 3.4 Boundary layer behaviour along blade profile 96
Figure 3.5 Ansys CFX modules 98
Figure 3.6 Control Volume Definition 100
Figure 3.7 TurboGrid blade-to-blade coarse mesh 101
Figure 3.8 Hexahedral shape functions 104
Figure 3.9 Representation of a linear system in matrix form for a scalar variable 105
Figure 3.10 Algebraic Multigrid procedure 106
Figure 3.11 Schematization of different solver approaches: 1) pressure-based implicit segregated, 2) pressure-based implicit coupled, 3) density-based implicit coupled, 4) density-based explicit coupled 107

Figure 4.1 Stream flow surfaces: blade-to-blade surface S1; meridional surface S2; secondary flow surface S3 110
Figure 4.2 Meridional surface and quasi-normal traces 111
Figure 4.3 Comparison of the meridional flow-field of a mixed flow turbine compared with a 3D CFD calculation (ANSYS CFX) [18] 111
Figure 4.4 From left to right and top to the bottom: CAD turbine geometry reproduction; Blades replaced by mean surfaces; Mean surfaces projected on the meridional plane; Results in term of entropy production 112
Figure 4.5 Blade-to-blade surface evolution 113

Figure 4.6 Use of ’classical’ spatial periodicity to simulate equal sectors of rotor and stator, possible only when blade numbers are not mutual prime. [19] 114
Figure 4.7 three-dimensional mixing problems [20]. 114
Figure 4.8 Phase Shifted Periodicity. P_R and P_S are the rotor and stator pitches, respectively, V_R is the rotor velocity [19]. 115

Figure 5.1 Density gradient contours in arbitrary linear scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with solidity imposed by the Method of Characteristics 119
Figure 5.2 Mach contours of supersonic stator blade with solidity imposed by the Method of Characteristics 119
Figure 5.4 Evolution of the pressure losses with pitch to leading edge thickness from mean-line code parametric analysis. 121
Figure 5.5 Mach contours of supersonic stator blade with solidity imposed by the mean-line code $C_{Ft} = 0.3$; 26 blades 122
Figure 5.6 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with maximum solidity accepted by the mean-line code $C_{Ft} = 0.3$; 26 blades 123
Figure 5.7 Mach contours of supersonic stator blade with solidity imposed by the mean-line code $C_{Ft} = 0.44$; 17 blades 123
Figure 5.8 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with maximum solidity accepted by the mean-line code $C_{Ft} = 0.44$; 17 blades 124
Figure 5.9 Examples of unimodal function 124
Figure 5.10 Illustration of the golden section search method [21]. 125
Figure 5.11 Flowchart of the golden-section search method. Note that the function $f(x)$ corresponds to the entropy production Δs evaluated by means of CFD simulations as the difference between mass-flow averaged entropy at domain outlet and the entropy at the inlet. 126
Figure 5.12 Flowchart to illustrate the procedure used to calculate the entropy production for each value of C_{ft} given by the optimization algorithm. 127
Figure 5.13 Research of optimal solidity to obtain the minimum entropy increase across stator. 129
Figure 5.14 Research of optimal solidity to obtain the minimum entropy increase across rotor. 130
Figure 5.15 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with optimum solidity $C_{Ft} = 0.226$, 33 blades 131
Figure 5.16 Mach number contours of supersonic stator blade with optimum solidity $C_{Ft} = 0.226$, 33 blades 131
Figure 5.17 Blade loading chart of supersonic stator blade with various C_{Ft} (number of blades) 132
Figure 5.18 Density gradient contours in arbitrary linear scale (i.e. numerical-Schlieren visualization) of supersonic rotor blade with optimum solidity $C_{Ft} = 0.245$, 32 blades. Relative frame of reference 133

Figure 5.19 Mach number contours of supersonic rotor blade with optimum solidity $C_{Ft} = 0.245$, 32 blades. Relative frame of reference 133
Figure 5.20 Plot of outlet velocity flow angle. To compare the curves extracted from four different CFD simulations, the curvilinear (circumferential) coordinate have been normalized with respect to the pitch dimension for every stator configuration reported here. 135
Figure 5.21 Plot of deviation angle. To compare the curves extracted from four different CFD simulations, the curvilinear (circumferential) coordinate have been normalized with respect to the pitch dimension for every stator configuration reported here. 135

Figure 6.1 Mach number contour in design condition. Relative Mach number reported for the rotor domain 139
Figure 6.2 Static pressure contour in design condition 139
Figure 6.3 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) in design condition 140
Figure 6.4 Static temperature contour in design condition 140
Figure 6.5 Plot of velocity flow angle for rotor inlet and outlet vs time . . 142
Figure 6.6 Plot of entropy increase throughout the rotor passage vs time 142
Figure 6.7 Mach number contours at 3.7e-4 [s]. Arbitrary scale: low limit fixed to 1 143
Figure 6.8 Mach number contours at 7.8e-4 [s]. Wake, became again supersonic, hits the rotor leading edge. Arbitrary scale: low limit fixed to 1. Note the presence of larger subsonic zone behind the shock waves. 143
Figure 6.9 Plot of entropy increase vs inlet flow angle 145
Figure 6.10 Outlet flow angle averaged value vs inlet angle (left); Outlet flow angle standard deviation vs inlet angle (left) 145
Figure 6.11 Mach number contours. Inlet incidence angle: -12 [deg]. $\eta_{tt} =$ 18.86%. Subsonic pocket downstream the bow shock, close to the Bounday Layer (BL), in wake region and at the rotor inlet. This configuration is close to the unstarting. 146
Figure 6.12 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet incidence angle: -12 [deg]. $\eta_{tt} = 18.86\%$. The shock lattice confirms that this configuration is close to the Kantrowitz limit. Note that, because of BL separation, the useful stator passage section is reduced. 146
Figure 6.13 Mach number contours. Inlet incidence angle: +12 [deg]. $\eta_{tt} =$ 21.98% 147
Figure 6.14 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet incidence angle: +12 [deg]. $\eta_{tt} = 21.98\%$ 147
Figure 6.15 Entropy production (left) and Total pressure ratio (right) for the stator blade vs inlet flow angle 148
Figure 6.16 Mean value of stator outlet flow angles (left) and standard deviation (right) vs stator inlet flow angle 149
Figure 6.17 Mach number contours of the -16 [deg] inlet flow angle configuration. The scale is limited to a value of 1 for the upper bound to better evidence the supersonic regions (in red). 149

Figure 6.18 Shear Stress axial component vs stream-wise coordinate. Where the value are positive, there is the presence of boundary layer separation. For the design condition, these regions are very limited and present only where shock waves hit the BL; the worst situation is in the -12[deg] configuration, where front part of the blade pressure side experience extensive reverse flow due to BL separation, because of the negative incidence. 150
Figure 6.19 Entropy values throughout the full stage domain 151
Figure 6.20 Mean value of rotor relative outlet flow angles (left) and standard deviation (right) vs stator inlet Mach number 151
Figure 6.21 Mach number contours. Inlet $M = 2.6$ 152
Figure 6.22 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 2.6$ 152
Figure 6.23 Mean value of stator outlet flow angles (left) and standard deviation (right) vs stator inlet Mach number 153
Figure 6.24 Stator entropy increase [J kg^{-}1 K^{-}1] vs inlet Mach number. . 154
Figure 6.25 Blade loading chart for different inlet mach number conditions. 154
Figure 6.26 Mach number contours. Inlet $M = 1.6$ 155
Figure 6.27 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 1.6$. Note that the shock lattice is *tighter* than the one resulting from simulation in nominal condition (Fig. 5.15) 155
Figure 6.28 Mach number contours. Inlet $M = 2.4$ 156
Figure 6.29 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 2.4$. Note that the shock lattice is more *stretched* than the one resulting from simulation in nominal condition (Fig. 5.15). This is the effect of the higher inlet Mach number, resulting in stronger and more oblique bow shock. 156
Figure 6.30 Mach number contours. Inlet $M = 2.8$ 157
Figure 6.31 Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 2.8$ 157

Figure 7.1 B-spline moving a control point (Figure from [22]). 160
Figure 7.2 Point by point error comparison between centripetal and rational parametrization (Figure from [22]). 162
Figure 7.3 Example of the control points and their associated normals used to perform a B-spline parametrization. 163
Figure 7.4 Genetic algorithms logic (Figure from [22]). 164
Figure 7.5 Two different methods of the crossover process. 165
Figure 7.6 Example of the mutation process. 166
Figure 7.7 Example of latin hypercube sampling in two dimensions (latin square). 167
Figure 7.8 Flowchart to illustrate the surrogate-based global optimization algorithm. 168
Figure 7.9 Block-structure implemented in Dakota for FORMA (Figure from [22]). 170
Figure 7.10 Folder structure for Ansys CFX (Figure from [22]). 171

Figure 7.11 Asynchronous local parallelism logic in Dakota. 173
Figure 7.12 Interpolation of the stator profile for case A. 174
Figure 7.13 Design space definition in case A. 175
Figure 7.14 Objective function evaluations along iterations for case A. . . 176
Figure 7.15 Comparison between the baseline profile and the optimal one for case A. 177
Figure 7.16 Mach contours and flow features comparison between the baseline and the optimal profile of case A. 178
Figure 7.17 Interpolation of the stator profile for case B. 180
Figure 7.18 Design space definition in case B. 181
Figure 7.19 Objective function evaluations along iterations for case B. . . . 182
Figure 7.20 Comparison between the baseline profile and the optimal one for case B. 183
Figure 7.21 Mach contours and flow features comparison between the baseline and the optimal profile of case B. 184
Figure 7.22 Comparison between a 100k mesh Mach contour in background with a 300k mesh density gradient contours in arbitrary linear scale (i.e. numerical-Schlieren visualization) transparent in foreground. . . 185

Figure B.1 Flow diagram of CFX coupled solver 193

Figure C.1 Stator mesh adopted 195
Figure C.2 Rotor mesh adopted 196
Figure C.3 Trend of monitored parameter for stator 196
Figure C.4 Trend of monitored parameter for rotor 197
Figure C.5 y plus at blade wall vs stream-wise coordinate for (a) stator and (b) rotor. 197

Figure D.1 Stator blade in Cartesian coordinate (red curve) and in cylindrical one (blue curve) 199

List of Tables

Table 1.1 Comparison of calculated thermodynamic efficiencies for various fuels under different cycles for initial compressor ratio equal to 5 [3]. . 6

Table 2.1 Inlet and outlet quantities for the stator and rotor designed with the variant B of the mean-line code. 77
Table 2.2 Stator, rotor and turbine stage main parameters (variant B). . 77

Table 4.1 Pros and Cons of mixing-plane technique 115

Table 5.1 Simulation settings for rotor and stator blade CFD simulation . 117
Table 5.2 Comparison of outlet flow angle: CFD simulation vs mean-line code results 134

Table 6.1 Relevant quantities, mass averaged, at stator inlet and rotor outlet.138
Table 6.2 Results quantities, mass averaged, at stator inlet and rotor outlet. Note that the 0D code provides a value of $\eta_{tt} = 56.4$ and of 172 [kJ kg^{-}1] for the specific (Eulerian) work. Discrepancies with respect to CFD simulation are due to the lack of a deviation angles model. . . . 138
Table 6.3 Relevant quantities, mass averaged, at stator inlet and rotor outlet.141
Table 6.4 *annular* mass flow rate passing through the thin layer which, in Ansys CFX, describes the blade-to-blade passage 144
Table 6.5 *annular* mass flow rate passing through the thin layer which, in Ansys CFX, describes the blade-to-blade passage. Fixing the static conditions, the mass flow rate varies linearly with the inlet Mach number. No presence of chocking. 153

Table 7.1 Lower and upper bounds for the normal direction motion of the adjustable control points in case A. 176
Table 7.2 Lower and upper bounds for the normal direction motion of the adjustable control points in case B. 180

Table A.1 Default Coefficient Values in Shear Stress Transport (SST) model191

Chapter 1

Pressure-Gain Combustion

Nowadays, gas turbine engines are playing a key role in several industry fields, most of them crucial for the world economy. Every technology and device they are equipped with is treated very carefully and developed with a lot of interest about the long-term efficiency and reliability gain, that can be transferred to the gas turbines themselves. By definition, gas turbines are continuous and internal combustion engines based on the Brayton cycle, from a theoretical thermodynamic point of view. Although only Diesel (compression ignition) engines and Otto (spark ignition) engines are commonly considered internal combustion engines, also the gas turbines are included into this classification because there is not a heat exchanger which transfer energy from the heat source to the working fluid. In an ideal gas turbine, gases undergo four thermodynamic processes: an isentropic compression, an isobaric (constant pressure) combustion, an isentropic expansion and heat rejection. Together, these make up the Brayton cycle (Fig. 1.1). These engines are mainly used for power production, in terms of electricity or mechanical power for industrial application, and for aeronautical and marine propulsion.

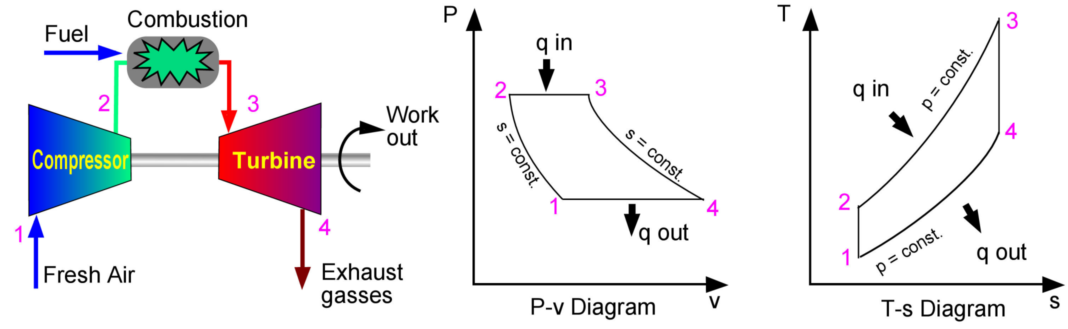


Figure 1.1. Idealized Brayton Cycle

The real thermodynamic cycle differs from the ideal one, for what concern compression and expansion in the turbomachines, due to aerodynamic losses, mechanical losses of various nature and so on. Regarding the combustion process, the constant pressure combustion shows a relevant increase in terms of entropy production, also in the ideal

Brayton cycle. This is the reason why a different combustion mode have been proposed and analysed during the last decades almost together with the development of gas turbines, started more than one century ago. Different way to overcome the efficiency penalty inherent the combustion process at constant pressure, have been proposed and a group of them, namely the Pressure Gain Combustion (PGC), is the leading and most promising approach according to the last researches and scientific publications. Two main type of Pressure-Gain Combustion engine are under development: the Pulse Detonation Engine (PDE) and the Rotating Detonation Engine (RDE), the latter object of interest for this thesis work.
It is important to notice again that this field of research is part of a technology area which undergoes huge economic interest worldwide. In this framework, the last 40 years in the gas turbine world were characterised by a relevant positive trend in terms of efficiency and any other performance indicator for these machines, as tested by every single Original Equipment Manufacturer (OEM). For the power production, for example, Gas Turbine Combined Cycle (GTCC) power plant is the most efficient and least-polluting fossil fuel-based electric power generation technology. This kind of power-plant will be crucial in the transition towards a full green-power generation. Largely driven by Turbine Inlet Temperature (TIT) and pressure ratio, the Brayton cycle based "J class" heavy-duty industrial gas turbines, with more than 40% net thermal efficiency (Fig. 1.2), have been the foundation of combined cycle power plants with field-proven 60+% net thermal efficiency (Fig. 1.3). Nevertheless, published ratings and field performances show that the 30+ years old steady gain in performance is rapidly approaching a plateau.

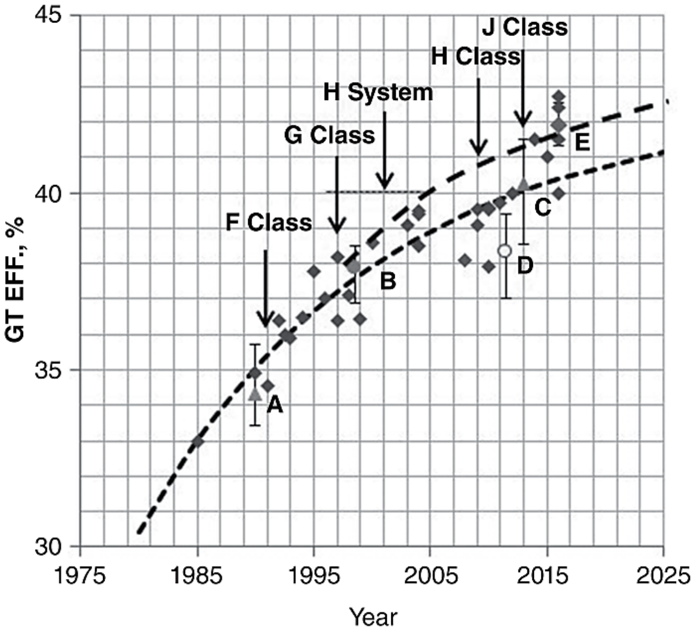


Figure 1.2. Gas turbine evolution, 1985–2016 [1]

Present state-of-the-art is 1600°C TIT with cycle pressure ratio of 23. The enabling technologies are advanced Hot Gas Path (HGP) materials (superalloys), Thermal Barrier Coatings (TBC), film cooling techniques and 3D-aero design software. The next class of machines are targeting 1700°C TIT, which is putting a big onus on Dry-Low-NOx combustor technology, superalloy capabilities and HGP parts manufacturability for commensurate film cooling effectiveness. Additive manufacturing (commonly known as “3D Printing”) and ceramic matrix composite Ceramic Matrix Composite (CMC) materials are two promising technologies to facilitate successful deployment of the K class in the next decade or so [23]. Similar trends and results regard also the other application of gas turbine engines and thermodynamic evaluations motivate the interest for the PGC.

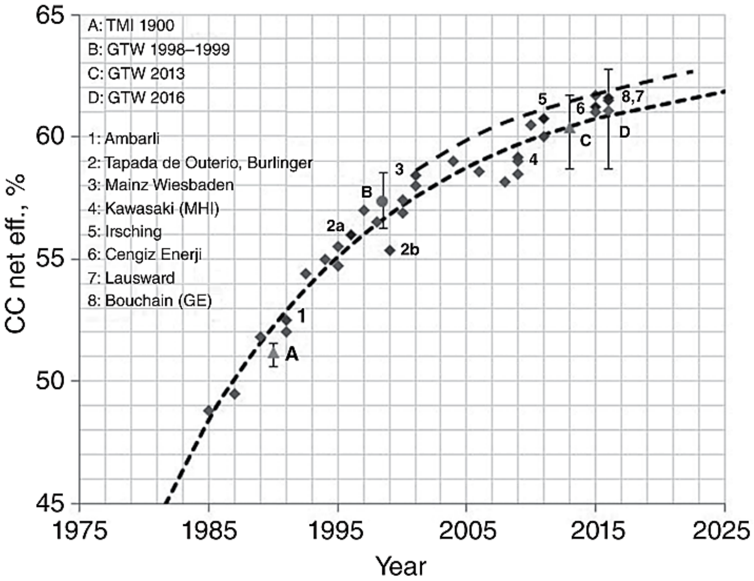


Figure 1.3. Gas turbine combined-cycle evolution, 1985–2016. [1]

1.1 Detonation Propulsion

Deflagration is the combustion mode currently exploited in gas turbine and, generally speaking, in every kind of internal combustion engine. Detonation, instead, is a type of combustion involving a supersonic exothermic front, which eventually drives a shock front propagating directly in front of it, whose power has been well recognized and analysed. It has been estimated that a 20 m^2 detonation wave operates at a power level equal to that received by the Earth from the sun [24]. Deflagration main difference with respect to detonation is the exothermic front speed, which is subsonic. It is intuitive to understand why, in the last century, scientists and engineers have always tried to manage detonations for power production purpose, at least since ’40, but, up to now, practical applications of detonative combustion have been very limited. In detonation, the reaction front in fuel-air mixture propagates with the velocity of order of km/s and produces a significant pressure increase. In contrast to detonation, deflagration flame velocity is of the order of dozens m/s, so combustion needs to be organized at the stoichiometric ratio (higher burning velocity), that results in a high combustion temperature and in production of high NOx concentration. Since temperature of burned products is very high in jet engine, it is necessary to mix extra air before turbine, which makes the design more complicated. Moreover, the pressure drops in combustion chamber, due to fluid dynamic losses. In contrast to deflagration, the combustion zone in detonation is very small. If the detonation propagates in a lean mixture, the combustion temperature is relatively low, so a low emission of NOx is expected (also due to a small residence time). Thus, there is no necessity to add extra air before the turbine (in turbojet engines). In addition, due to the detonative combustion, chamber pressure is increased [2] and a different and more efficient thermodynamic cycle is exploited by gas turbines. Briefly, the main advantages, at least theoretically, of the detonative combustion are the following:

- Detonation velocity of the order of km/s
- Short combustion zone, so more compact combustor
- Combustion (detonation) of ultra-lean or rich mixtures
- Lower temperature
- Low NOx emission
- No necessity to mix extra air
- Pressure increases due to detonation
- Continuous injection of the fresh mixture, which results in fewer fluctuating outlet conditions
- More efficient cycle for gas turbines

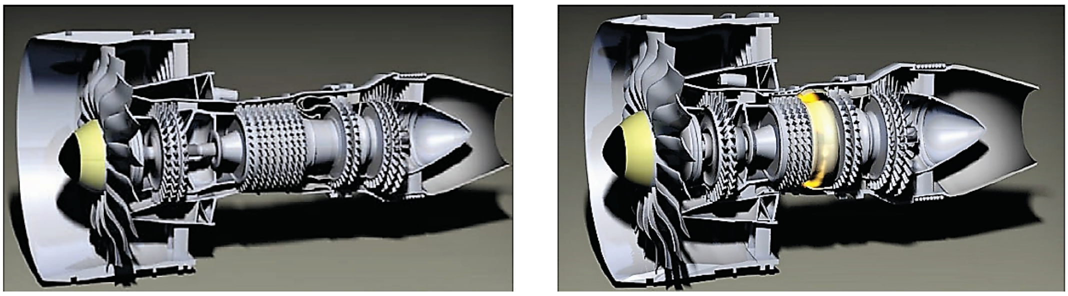


Figure 1.4. (a) classical turbofan engine. b) turbofan engine equipped with RDE. [2]

1.2 Cycle Analysis

In order to evaluate the advantages in adopting detonative combustion in gas turbine engines in terms of purely thermodynamic cycle efficiency, it is necessary to model the involved gas evolution. Whereas deflagration is well represented, with a reasonable level of approximation, by constant pressure heat addiction in the already explained Brayton cycle, detonation con be modelled with two different cycles: the Humphrey and the Fickett–Jacobs ones.

The Humphrey cycle is a thermodynamic cycle, which may be considered a modification of the Brayton cycle, in which the constant-pressure heat addition process is replaced by a constant-volume heat addition process. Hence, the ideal Humphrey cycle consists of 4 processes:

- Reversible, adiabatic (isentropic) compression of the incoming gas. Stagnation pressure and temperature increase because of the work done by the compressor. Entropy is unchanged.
- Constant-volume heat addition. In this step, heat is added, while the gas is kept at constant volume. Humphrey-cycle engines are considered open cycles in which air flows through continuously and this makes difficult to have a "constant volume" during the heat addition. Hence, the combustion mode is detonative and the heat addition happens only for a small volume of charge (in a combustor annulus) at a constant volume, while the remaining sections in the chamber are refilled with fresh premixed charge. This method allows for continuous flow in the system, while at the same time achieving the pseudo constant volume requirement for the heat addition process.
- Reversible, adiabatic (isentropic) expansion of the gas. Stagnation pressure and temperature decrease because of the work extracted by the turbine.
- Constant-pressure heat rejection. In open-cycle engines, this process represents expulsion of gases from the engine

The Fickett–Jacobs (FJ) cycle is similar to the Humphrey one, but relies on the Chapman–Jouguet (CJ) model for detonation. The working gas is first compressed by a turbomachine, than subjected to detonation, i.e. compressed by the shock wave and heated by the exothermic reaction, afterwards expanded isentropically to the initial pressure and finally the cycle is closed by a fictitious isobaric process of heat rejection to the initial state.
Experimental researches find that, among these cycles, FJ is the most representatives of PGC, although the Humphrey cycle is good enough for comparison purpose. Moreover, equations to evaluate the cycle efficiency are reported in (1.1) with reference to (Fig. 1.5) and used for calculation with different fuel in Table 1.1.The three cycle described so far are represented in the Pressure-Volume diagram (Fig. 1.5) and in the Temperature-Entropy diagrams (Fig. 1.6) shown below.

$$\eta_B = 1 - \frac{1}{\left(\frac{p_2}{p_1}\right)^{\frac{k-1}{k}}} \tag{1.1a}$$

$$\eta_H = 1 - k\frac{T_1}{T_2}\frac{\left(\frac{T_3'}{T_2'}\right)^{\frac{1}{k}}}{\left(\frac{T_3'}{T_2'}\right) - 1} \tag{1.1b}$$

$$\eta_{FJ} = 1 - k\frac{1}{\left(\frac{p_3}{p_1}\right)^{\frac{k-1}{k}}}\frac{\left(\frac{T_3''}{T_2'}\right)^{\frac{1}{k}} - 1}{\left(\frac{T_3''}{T_2'}\right) - 1} \tag{1.1c}$$

Table 1.1. Comparison of calculated thermodynamic efficiencies for various fuels under different cycles for initial compressor ratio equal to 5 [3].

Fuel	Brayton	Humphrey	Fickett-Jacobs
Hydrogen (H_2)	36.9	54.3	59.3
Methane (CH_4)	31.4	50.5	53.2
Acetylene (C_2H_2)	36.9	54.1	61.4

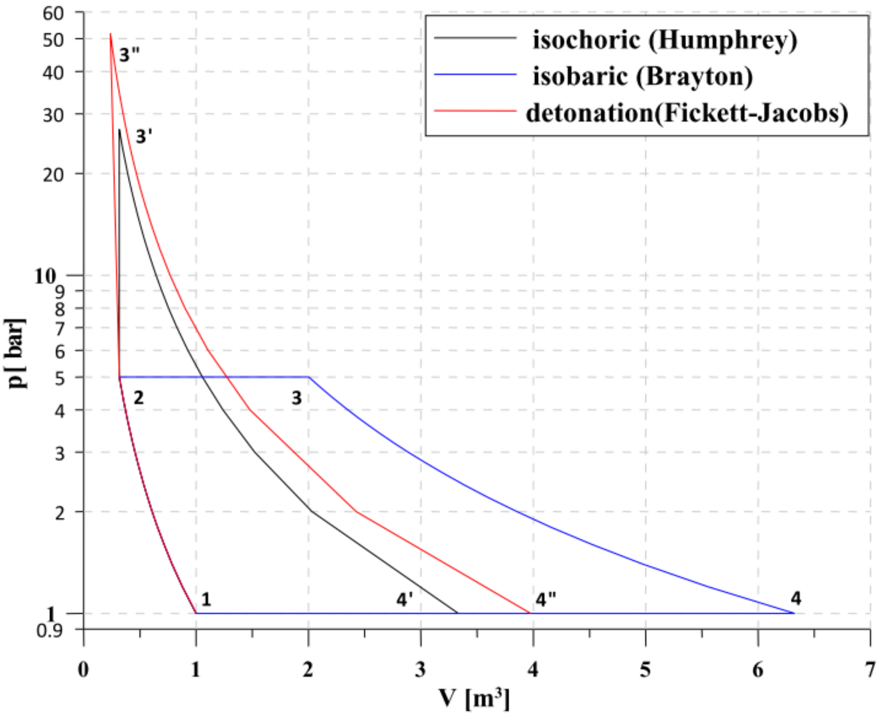


Figure 1.5. Thermodynamic Brayton-Joule, Humphrey and RDE engine (Fickett- Jacobs) cycles [3]

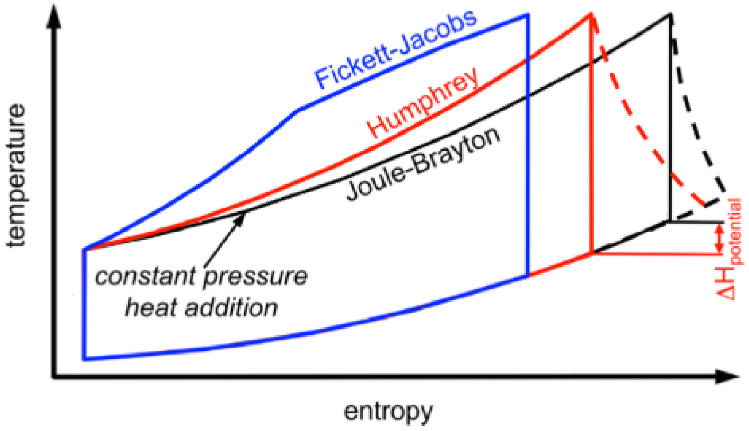


Figure 1.6. Thermodynamic Brayton-Joule, Humphrey and RDE engine (Fickett- Jacobs) cycles [4]

1.3 Pulse Detonation Engine

The most developed detonation engine concept, almost since the beginning of research activities in this propulsion field, is the Pulse Detonation Engine. PDE and RDE are the two most popular design concept to exploit the PGC and the only two exploitable for subsonic flight in aircraft propulsion. Other design concept have been proposed in literature with not remarkable success up to now or simply not exploitable for any application, except hypersonic flight.

PDE typically consists of a sufficiently long tube, which is filled with fresh fuel-oxidizer mixtures and ignited by sufficiently strong energy source. Flame initiated by ignition must accelerate to detonation velocity in relatively short time, so the transition from deflagration to detonation must happen in relatively small distance. Detonative combustion produces high pressure, which is converted to thrust. After all mixture is consumed by detonation, combustion products have to be evacuated from the tube and fresh mixture must be quickly resupplied, and the cycle is repeated. Typical frequency of such engine operation is usually in range of dozen Hertz, a too low value to obtain a smooth trust curve in time. The system is complicated because fast purging and refilling are required. Also the engine is operating in the stoichiometric condition, due to necessity of fast detonation initiation, requiring extra air to decrease temperature before the first turbine stage [2]. Other challenges are, again, related to initiation and repeatability of detonation by either reproduce in every cycle a combustion event sufficiently powerful, or by converting the downstream deflagration flame a supersonic wave in a process known as deflagration-detonation transition [25].

PDE have been extensively studied and several research Center are involved in numerical and experimental test all over the world, but the aspects listed previously represent too relevant weaknesses for the integration with turbomachinery in Gas Turbine engine.

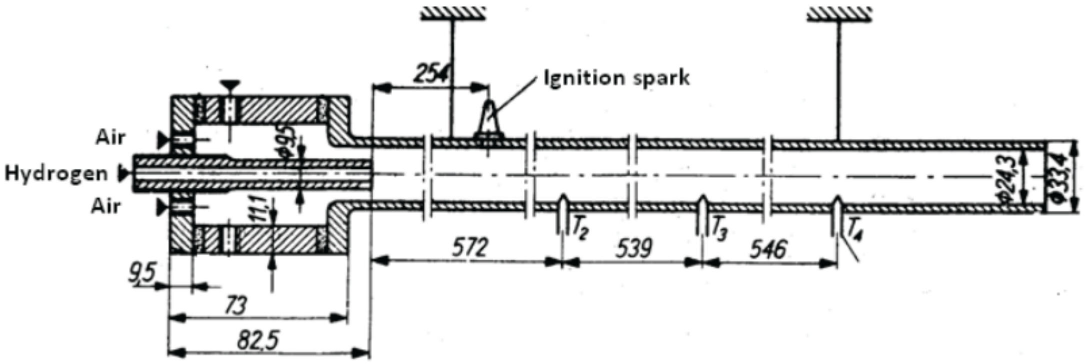


Figure 1.7. First PDE designed by J.A. Nicholls at the University of Michigan

1.4 Rotating Detonation Engine

1.4.1 Combustion Mechanism

Rotating Detonation Engine is the most promising technology able to enhance gas turbine performances. During the last two decades, this technology has been the object of growing interest. This engine type is characterised by one or more detonation waves contained within an open-ended annular chamber. A fuel/oxidiser mixture is fed into one end of the chamber, and the detonation wave consumes these reactants azimuthally, expelling them from the open end of the annulus. In some literature, this type of engine may also be referred to as a continuous detonation wave engine [26]. The detonation is continuously sustained by the injection of fresh reactants ahead of the wave. A fresh mixture is continuously injected along the axial direction with a certain velocity V_{inj} (creating the triangular refilling zone) while the transversal detonation fronts (dashed red lines) move in a clock-wise direction at a peripheral speed U, with an oblique shock wave anchored at a triple point [5]. The outlet flow is supersonic and with relevant spatial variability.

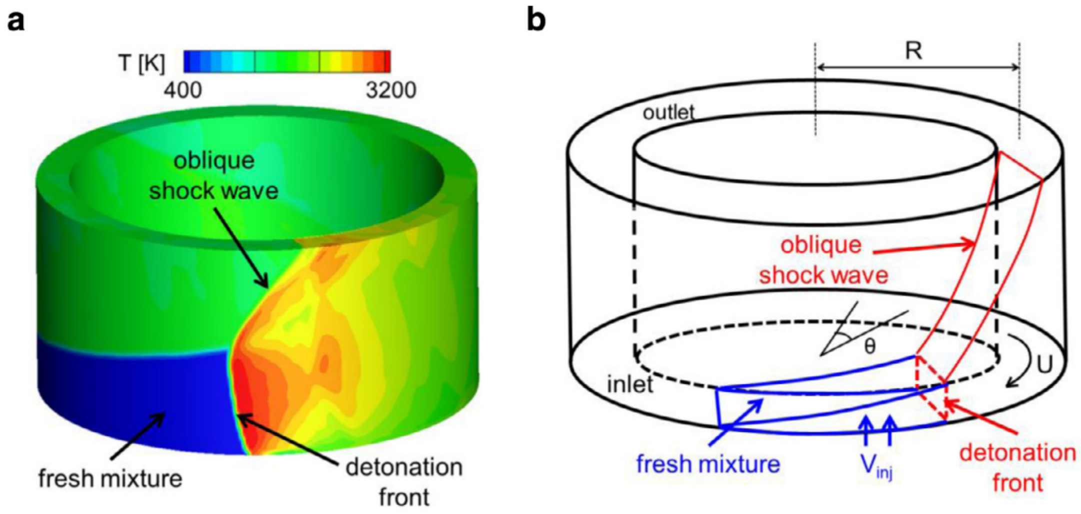


Figure 1.8. (a) Three-dimensional CFD simulation of a stable RDC. b) 3D flow features on a RDE with a clock-wise rotating detonation front. [5]

Moreover, both in numerical simulation and in experimental visualization the flow-field presents other relevant features. In Fig. 1.9 it is possible to observe:

(A) detonation wave

(B) trailing edge shock wave

(C) slip line between freshly detonated products and older products

(D) fill region

(E) non-detonated burned gas region

(F) expansion region with detonated products

(G) inlet region with blocked micro-nozzles

(H) inlet region with partial filling micro-nozzles

(I) inlet region with choked micro-nozzles

(J) secondary shock wave

Detonation wave is moving from left to right in Fig. 1.9, but the flow-field has been solved in the wave-fixed reference frame, so in steady state simulation.

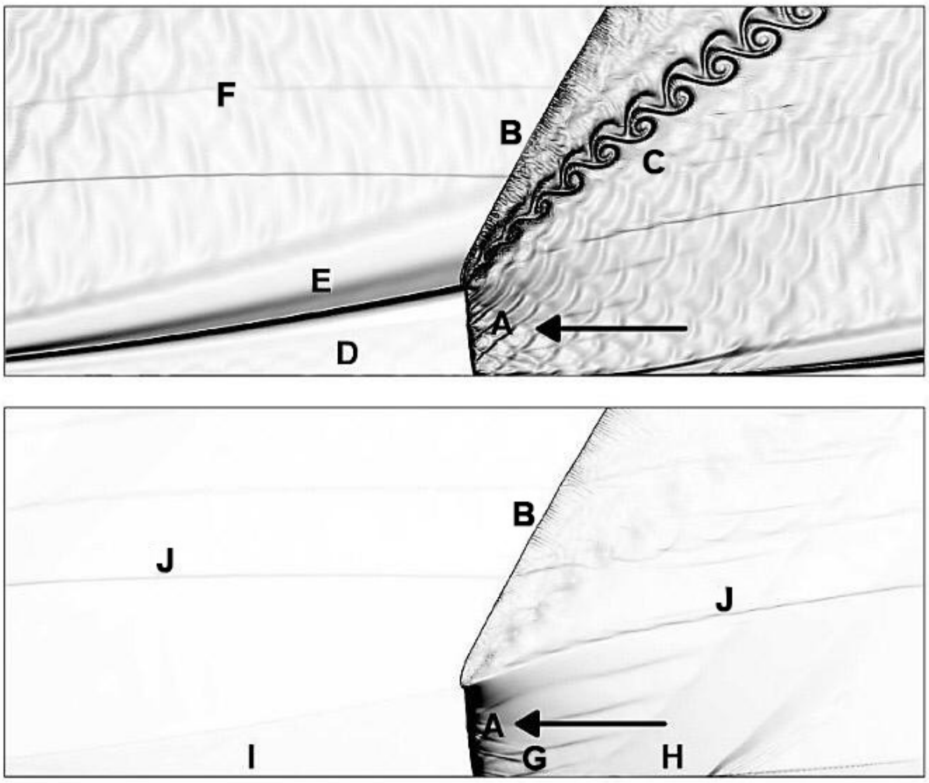


Figure 1.9. Numerical schlieren-like visualization of (top) temperature and (bottom) pressure of an "unrolled" hydrogen-air RDE solution [6].

In Fig. 1.10 it is reported a schematization of the unfolded 2D flow-field in the detonation front frame of reference, coming from a reduced order model. The fresh mixture moves from left to right at W_{cj} (Chapman–Jouguet velocity) towards the detonation front. At the triple point, a slip line and an oblique shock wave are generated and propagated towards the outlet. The slip line splits the burnt gases in two regions, by distinguishing those that just crossed the oblique shock wave. After the detonation wave, the wall pressure is higher than the injector static pressure which prevents the fresh mixture to refill the combustor [5]. Further away from the detonation front, a Prandtl-Meyer expansion imposes a flow acceleration and a progressive pressure decrease until it matches the injection pressure. At this point, the refilling re-starts and fresh mixture is injected axially into the combustor, resulting in a triangular refilling zone [9].

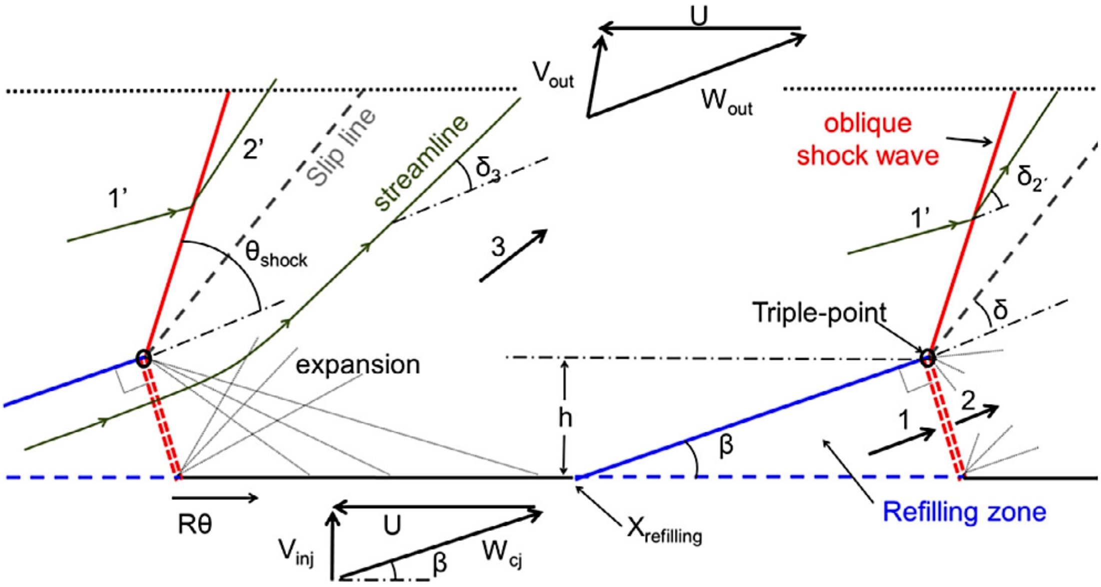


Figure 1.10. 2D unfolded illustration of the flow features and identification of the corresponding angles of a rotating detonation wave front [5]

1.4.2 Flow solver

Several numerical studies have been carried on to understand the flow structure in this complex environment and an increasing number of experimental tests confirm what was understood up to now. The flow nature itself is difficult to fully simulate not only due to the classical limitation imposed by the Navier-Stokes system of partial differential equations, but also because complex chemistry is involved and high gradient value regions are present.

Different kind of solvers have been used with various degrees of accuracy and computational cost. The full set of 3D Unsteady Reynolds Averaged Navier–Stokes (URANS) equations requires tremendous computational effort, but results are highly accurate and full of relevant information about the engine behaviour. To reduce this effort, different modelling strategies have been proposed in literature. A good trade-off between computational cost and solution accuracy can be easily found in literature with the radial thickness of the annulus chamber neglected and the fluid-dynamic domain unfolded and solved with a two dimensional solver. The detonation process can be modelled in several ways: using one-step reaction kinetics [27] with a limited number of species; or with two-step chemical reaction models [28]; or with an induction parameter which allows for larger grid size and time steps [7, 29]; or with detailed reaction kinetics which lead to expensive computations [30].

The faster solvers developed during the last 15 years are based on the Method of Characteristics (MOC) and include simple combustion models, generally the Zel'dovich - von Neumann - Döring (ZND) 1D detonation model, to solve RDE flow-fields in the wave-fixed reference frame. The MOC is a technique for solving partial differential equations and it is valid for any hyperbolic partial differential equation. When the flow can be considered fully supersonic, steady, adiabatic, inviscid and irrotational, the Navier-Stokes equations became hyperbolic and the problem can be addressed with this classical methodology. It consists in reducing a partial differential equation to a family of ordinary differential equations, along which the solution can be integrated from some initial data given on a suitable hypersurface. This method will implicitly exploited in this thesis work to generate the blade profile.

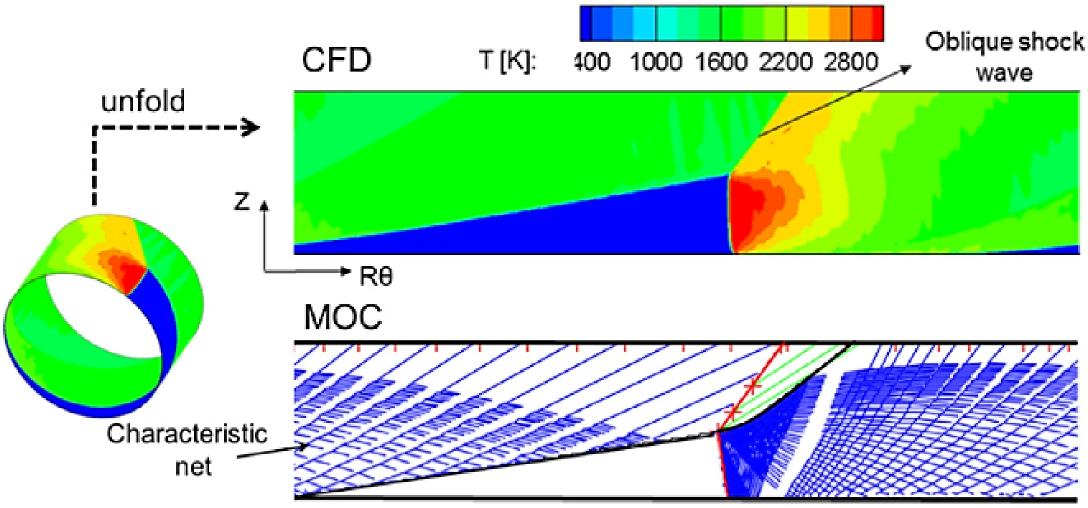


Figure 1.11. Comparison of the flow features contour obtained with the two-dimensional URANS solver (top) with the ones obtained with the MOC (bottom) [5]

Since this tool is the less computational expensive, it is the most suitable for parametric analysis to design the RDE [5, 9, 31]. In Fig. 1.11 agreement between the reduced order model based on the MOC and the 2D URANS solver can be appreciated even qualitatively, comparing the contours. Typical pressure profile are reported in Fig. 1.12 below and agreed with experimental measurement.

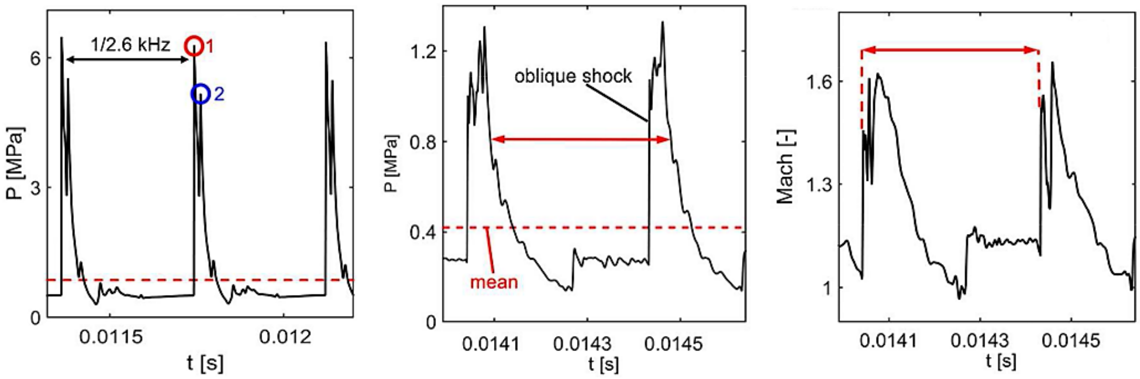


Figure 1.12. from the left to the right: inlet static pressure; outlet static pressure; outlet Mach number [7].

1.4.3 Gas Turbine architecture with RDE

Some concepts of RDE have been proposed and design solution reported in literature are quite similar to each other, presenting minor differences for the combustor itself (Fig. 1.13), but also for the entire Gas Turbine architecture.
For what concern the combustor annulus, it is possible to have different positions and number of detonation initiators, related to how many combustion fronts is more convenient to establish [32, 33]. Other differences in technical design are related to material choice, thermal analysis and mechanical assessment, due to the presence of important heat flux caused by rotating detonations. The first step in this sense is to evaluate the heat flux and how it is influenced by design choices and operation events [31]; then the engineering task consists mainly in cooling system sizing and assessment.

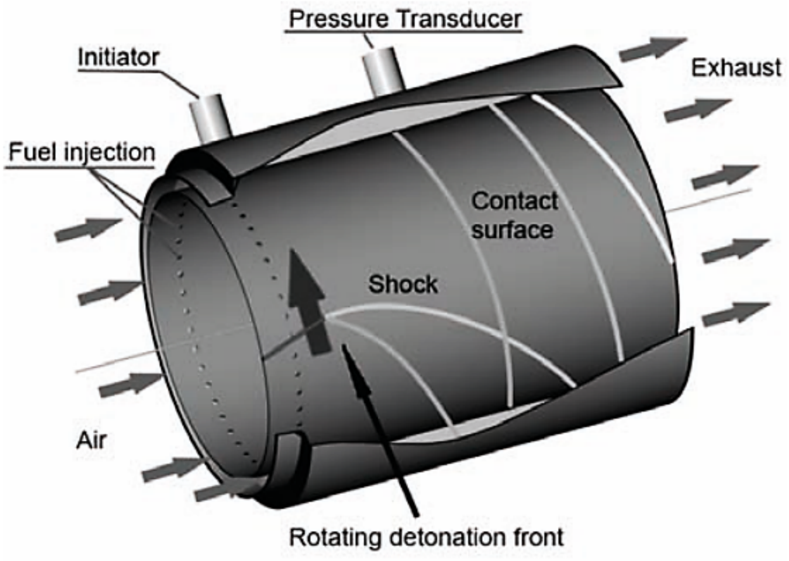


Figure 1.13. detonating combustion annulus [3].

Whereas adoption of the Rotating Detonation Engine will not heavily affect the cold side of the Gas Turbine, except the important advantage of a lower pressure ratio required to the axial compressor, several adjustments would be needed for the hot side. First of all, in literature there is agreement about the necessity to use a divergent nozzle placed at combustor outlet. The aim is to reduce non-uniformities and accelerate the flow to prevent the unstarting issues within the supersonic flow-passage in turbine stator [10]. Many studies have been performed to compare different revolution profile (Fig. 1.13) for the annular diffuser [7] or outlet-to-inlet area ratio and other geometrical features [16, 34] and their influence on the flow-field properties. In [7] five different nozzles were investigated, each one with three distinct combustor inlet total pressures: straight duct, conical, Bezier outer wall, Bezier inner wall, and two Bezier surfaces (Fig. 1.14 and 1.15). Every divergent nozzle expands the flow from sonic conditions to $Ma = 2.25$, beneficial for downstream turbine starting. For identical outlet-to-combustor-inlet-area ratios, the Bezier-outer-wall nozzle is more adequate at higher inlet pressures in terms of total pressure gain [7]. It offers the best performance in terms of flow fluctuation mitigation, achieving a global damping of 42.8% with a mass-flow averaged Mach number of 2. Moreover, this damping capacity varies with the severity of combustor inlet quantity fluctuation, in such a way that the downstream elements will be subjected to a constant level of oscillation regardless of the combustor exhaust flow root mean square deviations [35].

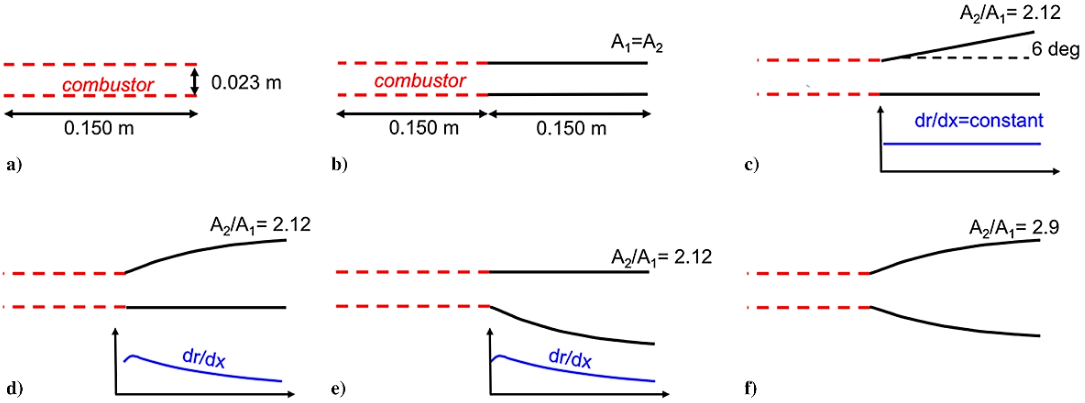


Figure 1.14. a) Combustor shape, no nozzle; b) straight duct nozzle; c) conical nozzle, with a constant radius inner wall; d) Bezier outer wall with a constant radius inner wall; e) Bezier inner wall with a constant radius outer wall; and f) two Bezier surfaces, combining the outer wall of d and inner wall of e. [7].

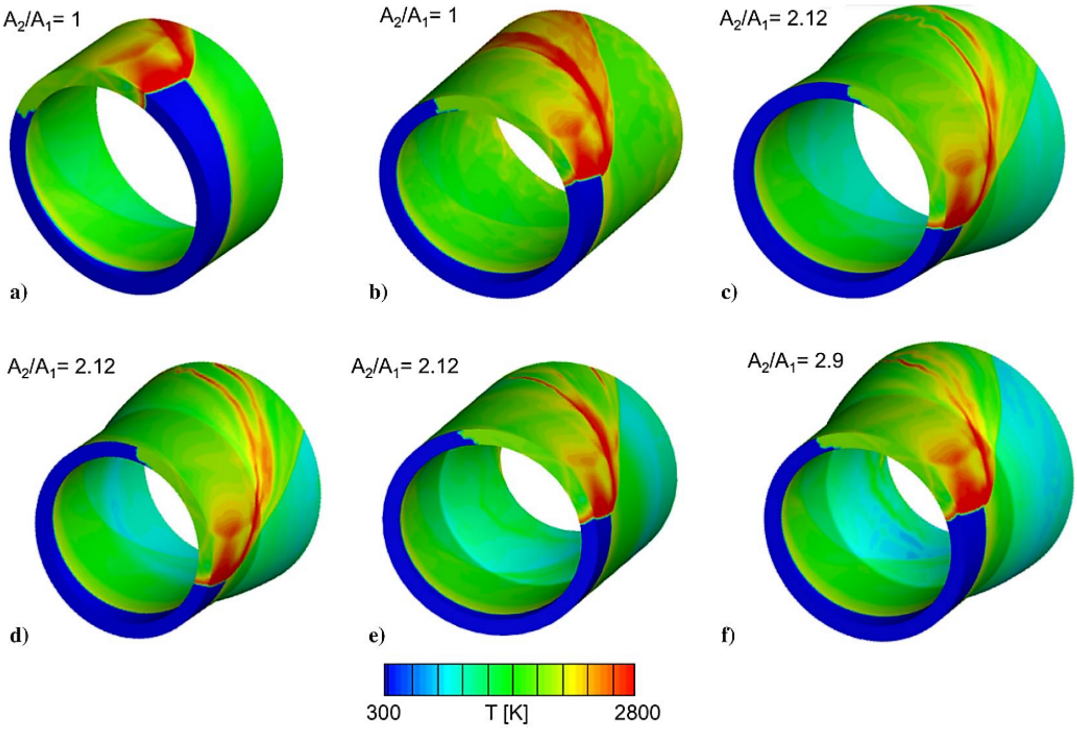


Figure 1.15. Isocontours of static temperature along the inner and outer walls for the baseline and five different nozzle geometries: a) baseline (no nozzle), b) straight duct, c) conical nozzle, d) Bezier outer wall, e) Bezier inner wall, and f) two Bezier surfaces. [7].

Whereas coupling of detonative combustor with downstream axial turbine is assured by the Bezier outer wall divergent nozzle, coupling with upstream compressor con be performed by an isolator (Fig. 1.16). In literature this component has not received enough specific attention for Gas Turbine arrangements of RDE [27] [36], but scope and shape have already been identified. It is a slightly divergent annulus, which links the high-pressure compressor stage to the combustor inlet where the micro-orifice for fuel feeding are placed. Isolator is envisioned to keep the inlet flow properties constant since the detonation wave continually blocks a portion of the annulus entrance from filling (Fig. 1.17). For a practical engine, a minimum length will be required to dampen any waves propagating upstream from the annulus [8]. Moreover, numerical simulations of RDE generally consider direct injection of premixed mixture, from holes or from the full head-end annulus section, but the premixing of fuel and air in the upstream isolator will be prohibited in practical RDE due to concerns with flashback [27].

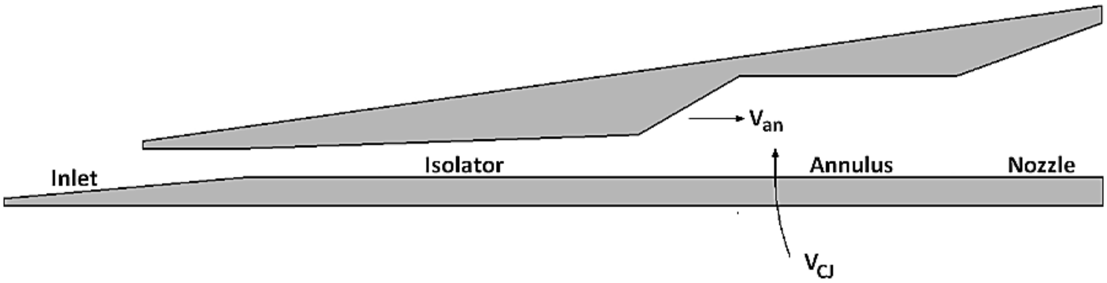


Figure 1.16. Basic engine geometry and stage designations (not to scale). [8].

1.4.4 Future Challenges

Previous paragraphs describe the RDE combustor section, on which the biggest part of the literature focuses. As previously mentioned, scientific community attention has grown during the last decades and understanding the behaviour of this innovative combustion method is the key to assess the feasibility and to control the RDE.
Several progresses have been made and experiments successfully performed, although just for brief time interval, but concerns and future challenges, nowadays, are related to several aspect, listed below:

1. Turbine/PGC aerodynamic design and integration, object of this thesis work
2. Combustor heat flux evaluation and cooling system assessment
3. Detonation stability, scaling and control: turbulence-detonation interaction, hot-spots presence
4. Turbine cooling system and mechanical design
5. Channel curvature and radial thickness effects for combustion and outlet flow properties distribution

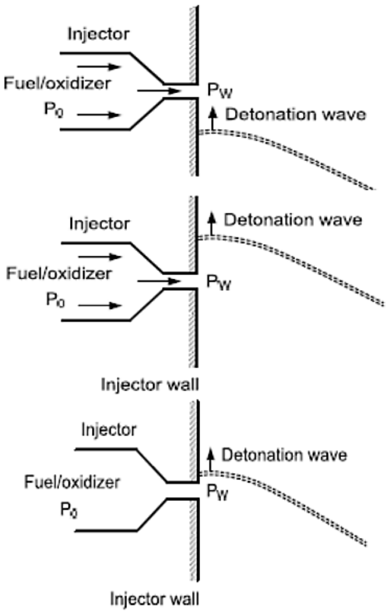


Figure 1.17. Possible injection conditions for fuel and oxidizer: - top, sonic injection ($p_w < p_{cr},$), premixed region expands almost linearly; - middle subsonic injection, ($p_o > p_w > p_{cr}$); - bottom, no injection ($p_w > p_{cr},$) (reverse flow and flashback risk). [3].

6. Prevent backflow into combustor inlet and/or seal

7. Accurate fuel-injection system design and investigation [36]:

    - uniform mixing requirement of oxidizer and fuel in a short time
    - detonation wave initiation in a single direction at start-up
    - flashback risk attenuation: propagation of shock wave back into the injector manifolds may cause destructive flashbacks

8. Instrumentation and Measurement: high frequency, large amplitude range, harsh environmental thermal capabilities required [37]

All of these aspects are relevant, but for the Gas Turbine architecture it must be noted that point 4) has massive effects. It has been already mentioned that TIT is the key for classical gas turbine efficiency improvement, therefore its decrease is not desirable. In order to maintain the usual values of TIT, fundamental is the adoption of film cooling techniques, which, for RDE implementation, impose a new architecture. Indeed, the air bled from the last compressor stage is at lower pressure, both static and total, than the one delivered from the combustor and so it is not possible to eject that air through the film cooling holes. There are two ways to deal with this problem:

- to use an auxiliary, smaller, compressor for the bled air
- to limit the TIT, exploiting only TBC and internal cooling, renouncing to some of the efficiency gain obtained with RDE implementation

1.5 Supersonic Turbine

In order to substitute classical combustors, used in modern Gas Turbines, with RDE and achieve efficiency improvement, one of the main task to be accomplished regards the design of turbine stages able to deal with supersonic flows. Innovative approaches need to be implemented, since classical turbine stages have been studied and tested for flows extremely different from the one coming from a proper designed divergent nozzle [35], downstream of a RDC (Fig. 1.17). Also modern turbomachinery design practice is not adequate for this application, although transonic flows and shock predictions are common in gas turbine, nowadays.

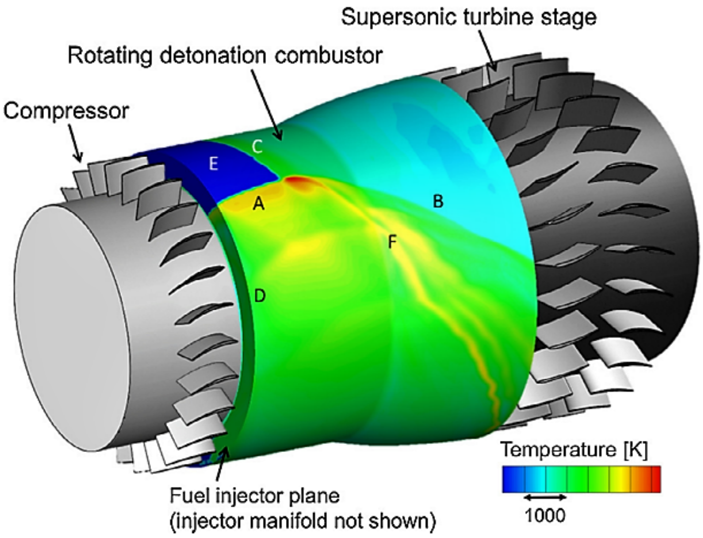


Figure 1.18. Iso-Temperature contour on a three dimensional RDC. A – Detonation front, B – Oblique shock wave, C – contact surface, D – Blocked injection due to the higher post-detonation pressure, E – Refilling zone, F – slip-line. [9]

As previously explained, the flow-field is supersonic and highly unsteady. The only aerodynamic alternative to avoid a supersonic flow consists in adopting a convergent diffuser, in contrast to all the technical solution proposed in literature, downstream of the RDC, in order to force the inhomogeneous combustor outlet flow to become subsonic. In this way it is possible to adopt usual turbine stages, but this option has been shown not to be promising: in [38] calculated turbine efficiencies is below 74% when subsonic axial turbines are exposed to inlet fluctuations of 75% at frequencies up to 10 kHz. The best solution proposed in literature up to now remains the use of a divergent nozzle coupled to novel supersonic turbine [4, 10, 39]. For what concern the flow-field unsteadiness, there are no methodologies to accomplish them during the preliminary design phase. Anyway, recent studies shown that supersonic turbine perform better than subsonic one under fluctuating inlet condition and reduced order models have been proposed to predict the damping capacity of the single row [40–42]. The key aspects of supersonic turbine design are the following:

- assuring the turbine starting, controlling flow angles and blade height evolution

- driving the supersonic flow through the passage by exploiting isentropic expansion and compression characteristic lines
- rounding the blade profile in a way that minimize shock formation
- modelling detached shock and its reflection through the passage, evaluating the induced total pressure loss
- evaluate skin friction losses (wet surface)
- optimize the solidity and any other geometrical parameter to find a trade-off among minimum shock losses, risk of boundary layer detachment and minimum friction losses

All these aspects have been treated in both classical and pioneering studies present in literature and are object of this thesis work. In this innovative branch, A special mention goes to previously mentioned researches [4, 10, 39–42]. Approaches followed in these studies have inspired this thesis work and deserve to be exposed. For what concern the turbine starting, classical turbomachinery literature have already explained the main aspects involved in this phenomenon, which will be deeply treated in the next chapter (Cap. 2). Only three kind of configuration for the turbine stage are possible, displayed in Fig. 1.19:

(a) impulse design, low reaction, with a rotor diffusing channel;

(b) reaction concept with a rotor nozzle design

(c) stator-less configuration with the rotor diffusing the flow

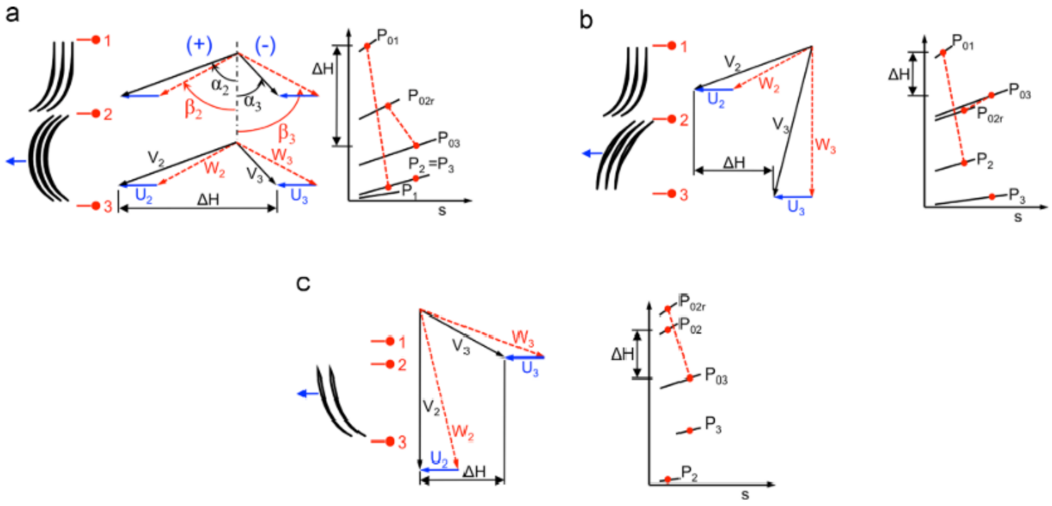


Figure 1.19. Supersonic turbine stage concepts [4]

On the other hand, old NASA technical notes have already faced the problem of designing supersonic blades [15, 43–47], whose resulting shape is similar to the one used for Curtis stage in steam turbines, but is based on an implicit version of the

MOC. Wall segments describe a passage based on the “vortex flow method”, where the inlet parallel flow is converted into a vortex flow field with the proper design of inlet transition arcs. Then the flow is turned through circular arcs and finally the outlet transition arcs provide a uniform parallel flow. This design favours high turning and loading. Obtained profiles need to be thickened and rounded and then tested by means of a numerical simulation (Fig. 1.20).
Whereas concept (a) is unfeasible, results showed the ability of the other design to ingest normal shock waves, allowing the passage to operate in the supersonic regime. Moreover, density gradient contours and pressure losses analysis revealed that the prime source of loss is the leading edge detached shock with its reflections across the turbine passage [39]. This implicit version of the MOC allows producing blade profile twice more efficient than the current state-of-the-art subsonic turbine designs, if exposed to supersonic fluctuating flow [4].

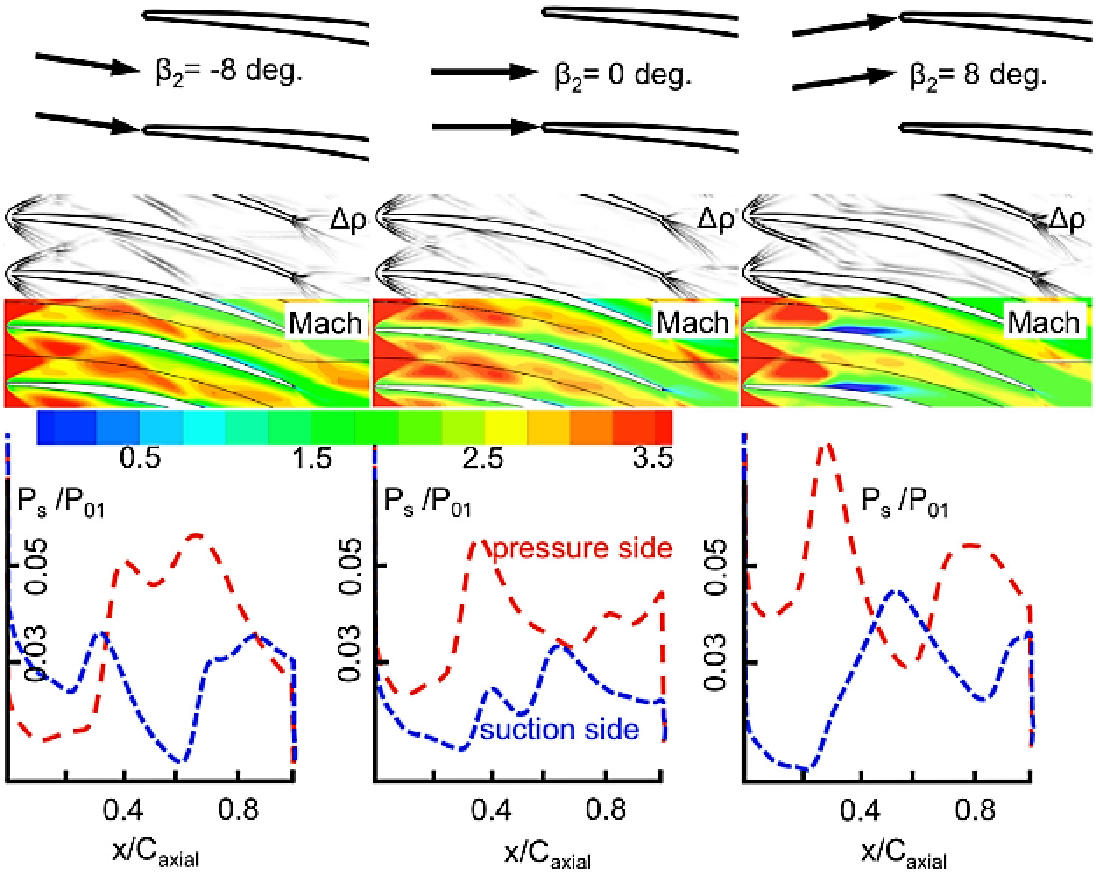


Figure 1.20. Schlieren, iso-Mach number contours and static pressure distribution on the rotor passage at three different inlet flow angles (configuration (c)) [4]

In [10] the MOC has been extensively developed for these application. Instead of exploiting the analytical and implicit MOC-based formulation of the "vortex method", proposed in [43], the authors approaches numerically the system of equations.
In steady, supersonic, inviscid and irrotational flows the governing equations are hyperbolic, being the flow properties at each point independent of the downstream

conditions and the solution procedure is of marching-type. The MOC is a meshless approach with a limited computational demand and so, particularly suitable for optimization algorithms requiring relevant amount of objective function evaluation. The key step involves the determination of the direction along which the cross derivative of the potential velocity is undetermined. These directions represent the lines where velocity derivative is discontinuous, which are known as characteristic lines. The lattice of characteristics begins at the sonic line downstream of the subsonic section caused by the detached shock wave at the leading edge and identified employing the procedure exposed in [12]. This method has been coupled with mixing and friction losses model, related to the BL growth, and produces highly accurate prediction of flow-field inside the turbine passages (Fig. 1.21).

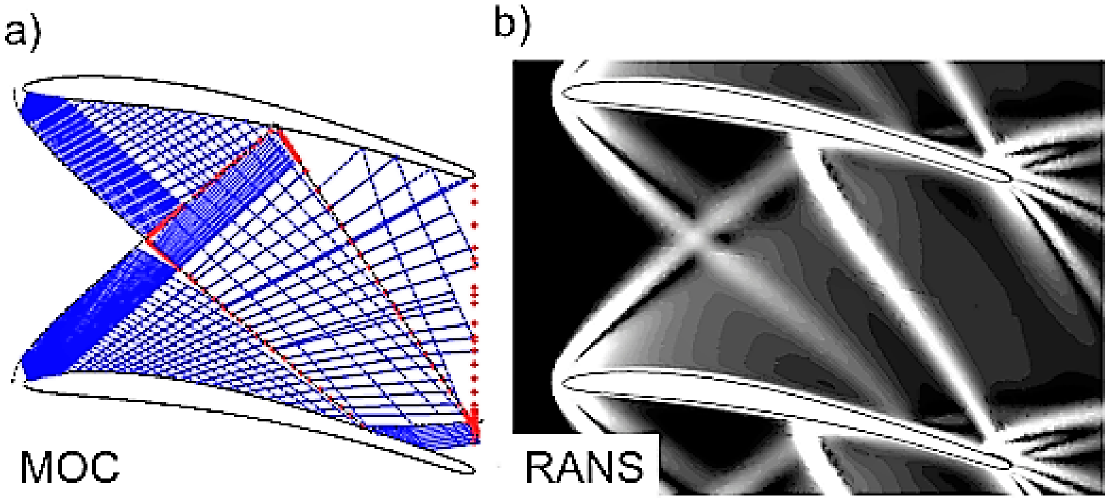


Figure 1.21. (a) Illustration of the characteristic net and the empirically predicted shock waves. (b) Numerical Schlieren visualization obtained with 3D RANS simulations. [10]

Chapter 2

Mean-line design

This chapter describes the completely original mean-line code developed to achieve an advanced preliminary design of a supersonic turbine and it is the core of the entire thesis, because every CFD simulation and every optimization process will start from the results generated by this code.

The first part of the chapter is dedicated to build all of the theoretical elements that are put together to compose the mean-line code. First of all a special care is given to the description and matching of the Kantrowitz limit required for a started blade row (section 2.2). The second element is an accurate evaluation and prediction of the aerodynamic losses, where the mean-line code takes account of:

- the detached bow shock wave at the leading edge estimated using Moeckel theory [12] (subsection 2.3.1);
- the first oblique shock wave reflection (subsection 2.3.2);
- the boundary layer and mixing losses, where first the boundary layer quantities have been calculated using Stratford and Beavers procedure [48] and then Stewart's method has been employed to calculate the losses [14] (subsection 2.3.3).

The final element and one of the most important pieces of the mean-line code is the Goldman version of the method of characteristics [14], applied to generate the profile of both the stator and the rotor of the supersonic turbine (section 2.4).

Finally, section 2.5 explains how all the elements are placed inside the while cycle and shows the logical order in which all the operations have been executed. Two alternative variants of the mean-line code have been developed: in variant A (subsection 2.5.1) the stator blade height is maintained constant, while in variant B (subsection 2.5.2) the channel height is variable both in the stator and in the rotor.

The last section of the chapter is dedicated to the validation process of the mean-line code, accomplished by comparing the code outputs with the results extracted from CFD simulations performed on the same case (section 2.6).

2.1 List of symbols

Nomenclature

$\boldsymbol{b}$	blade height
$\boldsymbol{bl}$	boundary layer
$\boldsymbol{c}$	chord
$\boldsymbol{C_{ft}}$	Zweifel's tangential force coefficient
$\boldsymbol{Cp}$	heat capacity at constant pressure
$\boldsymbol{D}$	diameter
$\boldsymbol{g}$	blade pitch
$\boldsymbol{L_{eul}}$	Euler work
$\boldsymbol{M}$	Mach number
$\boldsymbol{m}$	slope of a straight line
$\boldsymbol{\dot{m}}$	mass flow rate
$\boldsymbol{M_W}$	relative Mach number
$\boldsymbol{N_{bl}}$	number of blades
$\boldsymbol{p}$	pressure
$\boldsymbol{P_{eul}}$	power output
$\boldsymbol{r}$	radius
$\boldsymbol{R}$	air specific gas constant or radius
$\boldsymbol{R_X}$	Reynolds number based on the local free stream conditions
$\boldsymbol{s}$	specific entropy
$\boldsymbol{T}$	temperature
$\boldsymbol{th}$	leading and trailing edge thickness
$\boldsymbol{U}$	peripheral velocity
$\boldsymbol{V}$	absolute velocity
$\boldsymbol{v}$	velocity
$\boldsymbol{W}$	relative velocity
$\boldsymbol{X}$	equivalent flat plate length
$\boldsymbol{Y}$	total pressure losses
$\boldsymbol{\alpha}$	absolute flow angle
$\boldsymbol{\beta}$	relative flow angle
$\boldsymbol{\gamma}$	heat capacity ratio
$\boldsymbol{\delta}$	fully idealized boundary layer thickness
$\boldsymbol{\delta^*}$	displacement thickness
$\boldsymbol{\eta_{t,t}}$	total to total efficiency
$\boldsymbol{\theta}$	momentum thickness
$\boldsymbol{\lambda}$	angle of streamline relative to x-axis
$\boldsymbol{\mu}$	Mach angle
$\boldsymbol{\nu}$	Prandtl-Meyer angle
$\boldsymbol{\rho}$	density
$\boldsymbol{\sigma}$	isentropic contraction ratio from free stream to sonic velocity
$\boldsymbol{\phi}$	local inclination of detached shock relative to x-axis or velocity direction for MOC
$\boldsymbol{\chi}$	stage reaction
$\boldsymbol{\omega}$	rotor angular velocity

Subscripts

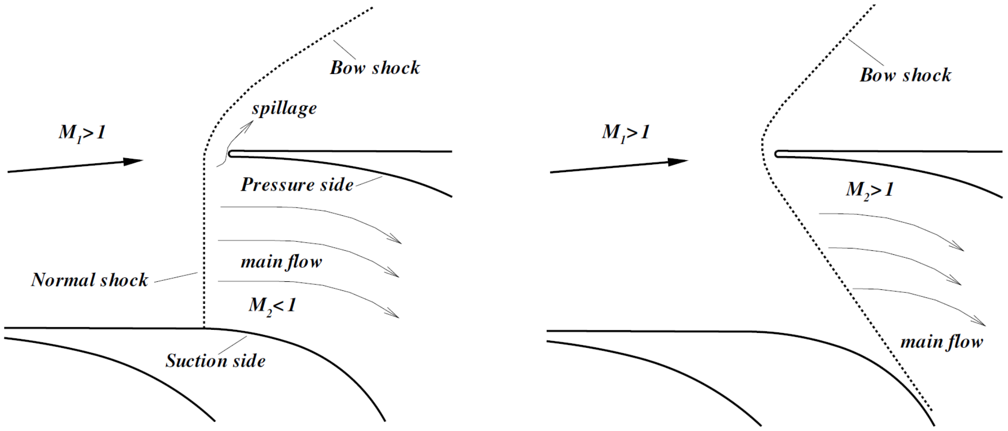
$\mathbf{0}$	free-stream condition (Moeckel's theory)
$\mathbf{1}$	stator inlet
$\mathbf{2}$	stator outlet
$\mathbf{3}$	rotor inlet
$\mathbf{4}$	rotor outlet
$\boldsymbol{ax}$	axial component
$\boldsymbol{bl}$	boundary layer
$\boldsymbol{c}$	centroid of stream tube passing sonic line
$\boldsymbol{cr}$	conditions at Mach number of 1
$\boldsymbol{e}$	external
$\boldsymbol{fs}$	free stream
$\boldsymbol{i}$	inlet or internal
$\boldsymbol{k}$	index for iterative procedures
$\boldsymbol{l}$	lower
$\boldsymbol{m}$	mean-line
$\boldsymbol{n}$	normal component
$\boldsymbol{max}$	maximum
$\boldsymbol{min}$	minimum
$\boldsymbol{o}$	outlet
$\boldsymbol{out}$	outlet
$\boldsymbol{S}$	sonic point of detached shock
$\boldsymbol{SB}$	sonic point of body
$\boldsymbol{t}$	total quantity
$\boldsymbol{te}$	trailing edge
$\boldsymbol{tg}$	tangential component
$\boldsymbol{th}$	throat
$\boldsymbol{tr}$	total relative
$\boldsymbol{u}$	upper
$\boldsymbol{x}$	projection in x direction

Superscripts

$*$	value of a parameter at sonic flow conditions or dimensionless quantities
$'$	total quantity

2.2 Started or unstarted blade row

A convergent passage when subjected to a supersonic flow behaves like a diffuser, reducing the velocity and increasing the pressure of the stream. In these particular conditions, it is critical to assess if the blade row is started or unstarted.
In a Laval nozzle a subsonic flow accelerates in the convergent part, reaches sonic condition in the throat and becomes supersonic in the divergent section [49]. Now suppose hypothetically that this flow could be reversed in order to obtain a supersonic inlet diffuser with sonic condition in the throat section; in practice a flow of this type is unattainable for stability reasons: the amount of mass entering the supersonic portion of the diffuser is independent from what happens downstream, but at the same time disturbances can propagate in the subsonic part leading to a temporary reduction of the mass flow rate; the unbalance in mass flow rate between the two sides causes a continuous accumulation of air at the throat section which finally is resolved through the formation of a normal shock wave in front of the diffuser (unstarted diffuser).
Kantrowitz and Donaldson showed that the strength and the position of the shock depends on the geometry of the diffuser and the Mach number upstream; in particular if the contraction ratio $\frac{A_{th}}{A_1}$ is increased, the normal shock wave first jumps from ahead the diffuser to a stable position after the throat and then if the contraction ratio is increased further but still lower than 1, the flow remains supersonic throughout the whole diffuser without normal shocks (started diffuser).



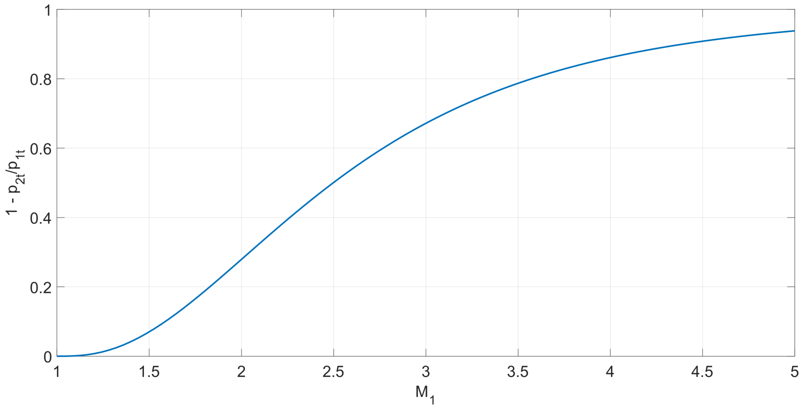
(a) *Unstarted blade row* (b) *Started blade row*

Figure 2.1. Comparison between started and unstarted condition for a supersonic blade row (Figure from [11]).

In the unstarted case the normal shock wave in front of the blade row causes huge total pressure losses which are strongly dependent on the inlet Mach number; these losses can be easily evaluated by applying the normal shock equations and their associated value at varying inlet Mach number is reported in fig. 2.2. For this reason, it is really important to design the stator and the rotor to work in started condition in which the oblique shock waves originated at the leading edge generate significantly lower total pressure losses [4].

M_1	$\frac{p_{2t}}{p_{1t}}$
1.5	0.9297
2	0.7208
2.5	0.4990
3	0.3283
3.5	0.2129
4	0.1387
4.5	0.0916
5	0.0617

(a)



(b)

Figure 2.2. Total pressure losses through a normal shock wave.

A conventional one-dimensional flow theory, which is a quite good approximation in this case, is applied to find the analytical limit for the starting of the blade row [4,49]. First of all, the contraction ratio required to reach a sonic condition in the throat is determined by applying the conservation of mass.

$$\begin{aligned}\frac{A^*}{A_1} &= \frac{\rho_1 v_1}{\rho^* v^*}\\ &= \frac{\rho_1}{\rho_t}\frac{\rho_t}{\rho^*}\frac{M_1\sqrt{\gamma R T_1}}{\sqrt{\gamma R T^*}}\\ &= \frac{(1+\frac{\gamma-1}{2})^{\frac{1}{\gamma-1}}}{(1+\frac{\gamma-1}{2}M_1^2)^{\frac{1}{\gamma-1}}}M_1\frac{(1+\frac{\gamma-1}{2})^{\frac{1}{2}}}{(1+\frac{\gamma-1}{2}M_1^2)^{\frac{1}{2}}}\end{aligned}$$

$$\frac{A^*}{A_1} = \left(\frac{\frac{\gamma+1}{2}}{1+\frac{\gamma-1}{2}M_1^2}\right)^{\frac{\gamma+1}{2(\gamma-1)}} \tag{2.1}$$

Eq. 2.1 actually gives the limit under which the blade row will always be unstarted with a front normal shock. Kantrowitz and Donaldson showed then that the maximum permissible contraction ratio to have a started condition is equal to the contraction ratio that would be required for an isentropic compression to the Mach number of 1 multiplied by the total head ratio across the normal shock.

$$\begin{cases} \left.\frac{A_{th}}{A_1}\right|_{self-started} > \left(\frac{\frac{\gamma+1}{2}}{1+\frac{\gamma-1}{2}M_{normal}^2}\right)^{\frac{\gamma+1}{2(\gamma-1)}} \\ M_{normal}^2 = \frac{(\gamma-1)M_1^2+2}{2\gamma M_1^2-(\gamma-1)} \qquad \text{Mach number behind normal shock} \end{cases} \tag{2.2}$$

Finally in order to have a self-started blade row the contraction ratio should be higher than the limit value calculated using eq. 2.2. As we can see in fig. 2.3 between the self-started and the unstarted region, there is a third intermediate one: in this region it is possible to achieve a started operating mode, but the diffuser must start from a self-started condition and then or the inlet Mach number should be reduced or a variable diffuser geometry should be employed; these two options will not be investigated further because they are not really practical for our case.

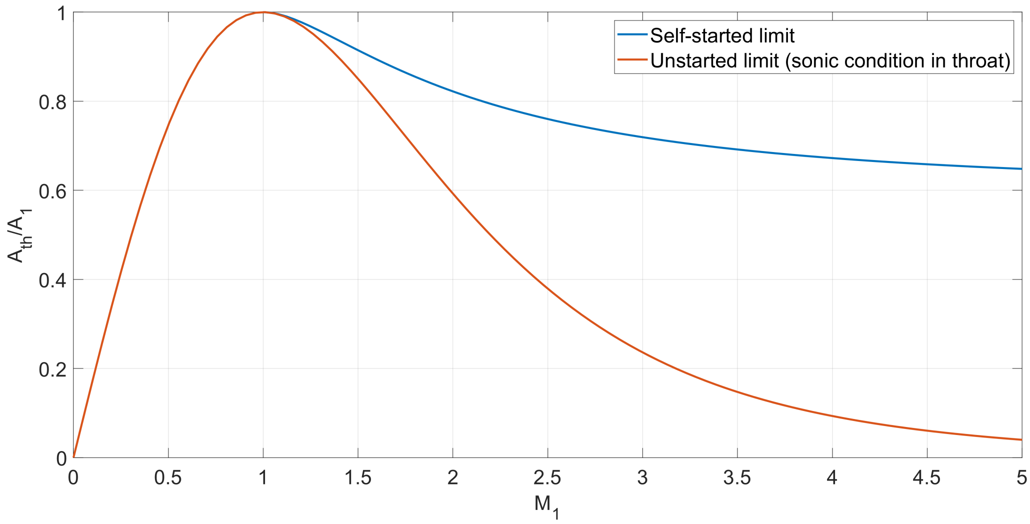


Figure 2.3. Contraction ratio for a started or unstarted blade row with varying inlet Mach number.

The self-starting limit affects also the maximum allowable turning on which the amount of work extracted in a turbine depends; the relation between the two quantities is easily shown by calculating explicitly the area.

$$\frac{A_{th}}{A_1} = \frac{A_{out}}{A_1} = \frac{g_2 b_2 cos(\alpha_2)}{g_1 b_1 cos(\alpha_1)} > \left(\frac{\frac{\gamma+1}{2}}{1 + \frac{\gamma-1}{2} M_{normal}^2} \right)^{\frac{\gamma+1}{2(\gamma-1)}} \tag{2.3}$$

As shown in eq. 2.3, the outlet flow angle is bounded by the self-starting limit and this will affect significantly the design process of the turbine; a solution to ease this limitation on the flow angles and at the same time to increase the amount of work extracted by the turbine is to work with a varying channel height along the blade length and this will be implemented in the mean-line code.

2.3 Evaluation and prediction of aerodynamic losses

Typically a profile working in a supersonic flow environment is less efficient than its subsonic counterpart due to the complex shock pattern generation in the channel. One of the key feature that was developed in the mean-line code is the capability to estimate effectively the losses.
In a supersonic turbine the main source of losses are the following:

- the generation of strong bow shock waves at the leading edge reflected as oblique shock waves on the suction and pressure side of the blades;
- boundary layer growth in a diffusing channel and its interaction with oblique shock waves;

- mixing and secondary flow losses.

Not all of these losses were considered in the mean-line code, but a really good prediction was obtained by taking account the strong bow shock wave at the leading edge (responsible alone of two third of the total losses), its first oblique shock wave reflection, boundary layer and mixing losses; how all of these were estimated and developed in a MATLAB code will be described extensively in the following paragraphs.

2.3.1 Detached bow shock wave

The detached shock generated from the leading edge is the main source of entropy production inside the turbine and it is responsible of two thirds of the total losses; for this reason calculating this component in an accurate way is fundamental to achieve a good loss prediction. The procedure used is divided in two main steps:

1. the shape is predicted by applying the empirical method developed and validated by Moeckel [12];
2. the entropy jump is locally evaluated through the oblique shock wave equations and then averaged along pitch.

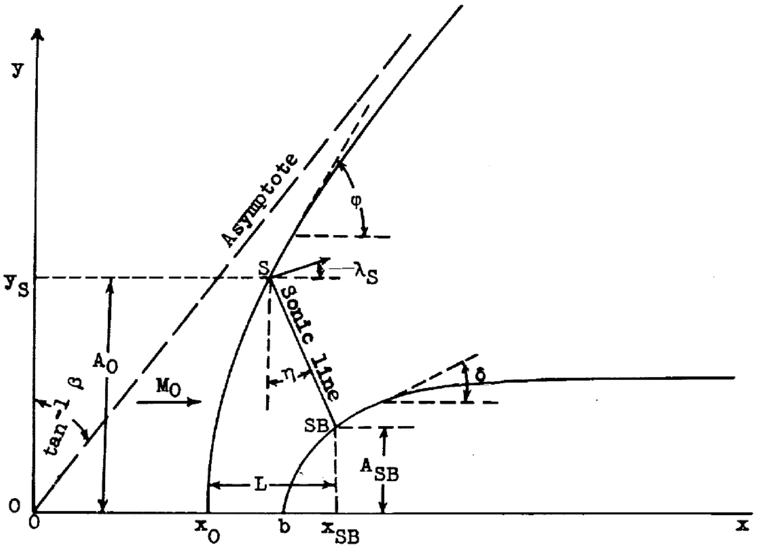


Figure 2.4. Detached shock wave and notation used in analysis (Figure from [12]).

Moeckel developed an approximate method to predict the location of detached shock waves ahead of two-dimensional and axially symmetric bodies by applying the continuity relation on the sonic line. To approximate the problem to an equivalent one-dimensional form, the following two main assumptions are made:

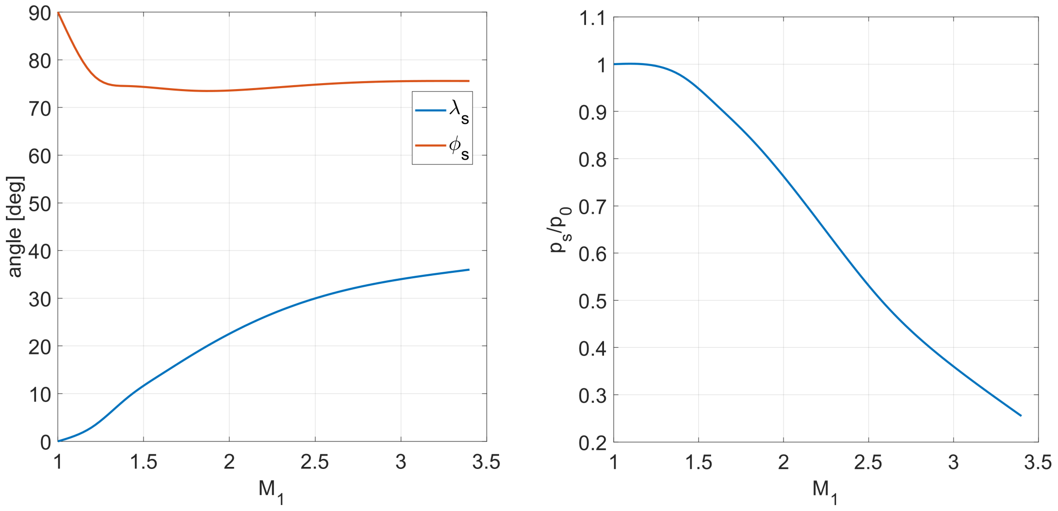
- the form of the detached wave from the axis to its sonic point is adequately represented by an hyperbola asymptotic to the free stream Mach lines;
- the sonic line between the shock and the body is straight and inclined at an angle that depends only on the free stream Mach number.

The second assumption is fundamental because the leading shape geometry ahead the sonic point is no more affecting the shape and location of the bow shock wave, allowing a much easier application of the continuity equation; this model has also some limitations because it oversimplifies the flow when applied to extremely blunt bodies and does not work properly in the proximity of the sonic velocity, but this is not a problem because our application is far from these limit conditions.
Moeckel model uses an hyperbola curve because, as the detached shock waves, these curves are normal to the flow at their foremost point and they are asymptotic to the free-stream Mach lines at large distances.

$$y = \frac{1}{\beta}\sqrt{x^2 - x_0^2} \tag{2.4}$$

where $\beta = \sqrt{M_1^2 - 1}$ is the cotangent of the Mach angle and x_0 is the distance from the vertex of the wave to the intersection of its asymptotes. While β is easily calculated because the Mach number at inlet is known, determining x_0 is more complicated.
First of all it is required to locate the sonic point on the body, which in this case is coincident with the shoulder of the body because the sonic line is straight; the shoulder is the point where the body is inclined to an angle corresponding to the shock detachment, whose value λ_s is experimentally calculated at various inlet Mach number and reported in Fig. 2.5. The angle ϕ_s is known from shock theory because it is the angle at which the velocity behind the shock is sonic.

$$\frac{dy}{dx} = tan\phi = \frac{\sqrt{x_0^2 + \beta^2 y^2}}{\beta^2 y} \tag{2.5}$$



(a) (b)

Figure 2.5. Charts for the determination of supersonic air flow against inclined planes and axially symmetric cones [12].

Rewriting eq. 2.5 in a different form and substituting the sonic point condition calculated before, equation 2.6 is obtained.

$$\frac{x_0}{y_{SB}} = \beta\frac{y_S}{y_{SB}}\sqrt{\beta^2 tan\phi_S^2 - 1} \tag{2.6}$$

In formula 2.6 all the terms are known with the exception of the ratio $\frac{y_S}{y_{SB}}$, which will be now determined by applying the continuity equation on the sonic line; without going in too much detail, which can be found in the original article, the ratio $\frac{y_S}{y_{SB}}$ can be calculated through eq. 2.7, where σ is the contraction ratio required to decelerate the free stream to sonic velocity isentropically.

$$\frac{y_S}{y_{SB}} = \left(1 - \left(\frac{p_0}{p_s}\right)_c \sigma cos\lambda_S\right)^{-1} \tag{2.7}$$

Once all the parameters have been calculated, the detached bow shock wave can be finally drawn in front of the leading edge; in order to assess the reliability and the accuracy of the Moeckel model, the analytical shape of the bow shock wave is compared with a CFD simulation done on a supersonic profile (sec. 2.4) in the same flow condition. In fig. 2.6 in the background there is a post-processed image obtained with a CFD simulation: it is shown the gradient of density to capture in the best possible way the complex shock pattern; the shape of the detached shock wave obtained with Moeckel theory is shown in yellow in the foreground. As proven by the image, the predicted curves are really close to the ones obtained with CFD which confirms that Moeckel model gives a really good approximation with a simplified theory.

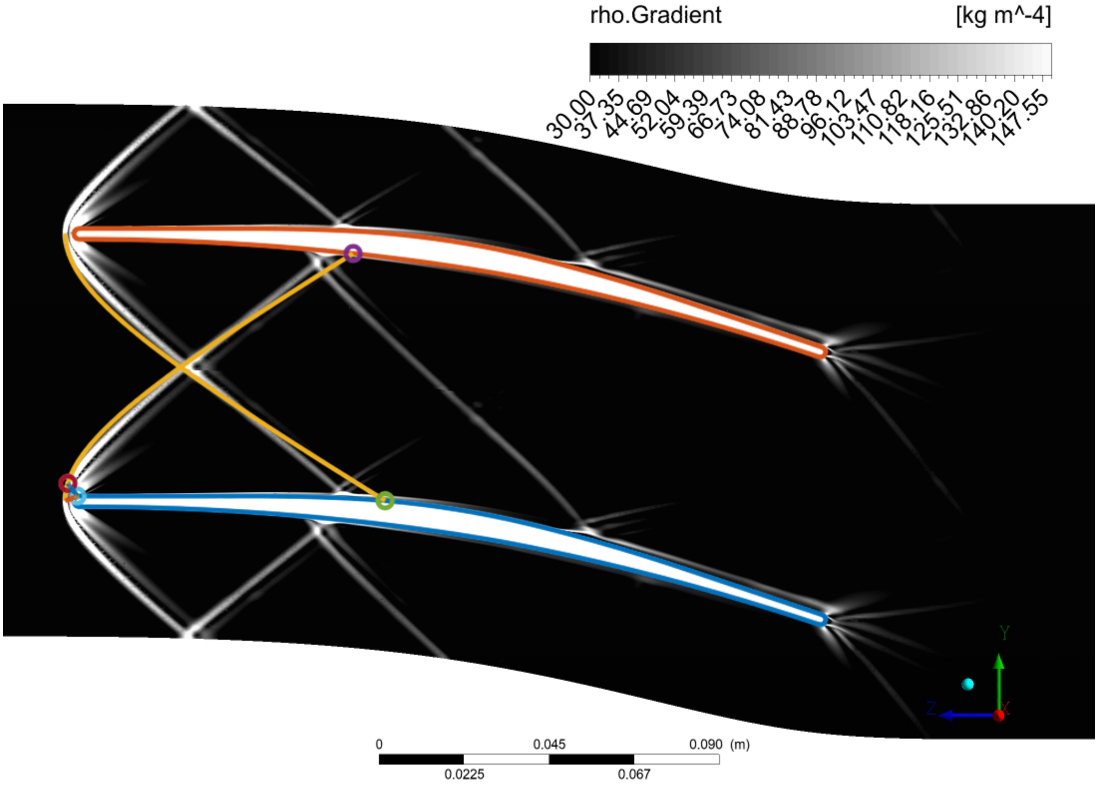


Figure 2.6. Comparison between the detached shock wave shape obtained with Moeckel theory and CFD.

Once the detached shock has been drawn, all the informations required to calculate locally the entropy jump are available by applying the oblique shock relations.

$$\begin{cases} M_{n1} = M_1 sin(\phi) \\ \dfrac{p_2}{p_1} = 1 + \dfrac{2\gamma}{\gamma+1}(M_{n1}^2 - 1) \\ \dfrac{\rho_2}{\rho_1} = \dfrac{(\gamma+1)M_{n1}^2}{(\gamma-1)M_{n1}^2 + 2} \\ \dfrac{T_2}{T_1} = \dfrac{p_2}{p_1}\dfrac{\rho_1}{\rho_2} \\ \Delta s = C_p \log\left(\dfrac{T_2}{T_1}\right) - R\log\left(\dfrac{p_2}{p_1}\right) \end{cases} \tag{2.8}$$

The entropy jump is calculated for each point of the detached shock wave and then the results are averaged for the whole shock wave till the blade surface (Equation 2.9), where the intersection between the shock waves is neglected as a simplifying assumption. In conclusion the total entropy increases in each channel due to the detached shock wave being the sum of the average entropy produced by the upper and lower shock generated at the leading edge (Fig. 2.6).

$$\Delta s_{total-bow-shock} = \frac{1}{y_{2u} - y_{1u}} \int_{y_{1u}}^{y_{2u}} \Delta s_u(y)\, dy + \frac{1}{y_{2l} - y_{1l}} \int_{y_{1l}}^{y_{2l}} \Delta s_l(y)\, dy \tag{2.9}$$

Now by applying Moeckel theory and calculating the entropy jump as shown before, it is possible to do a parametric study to find the best operating condition to minimize the losses.

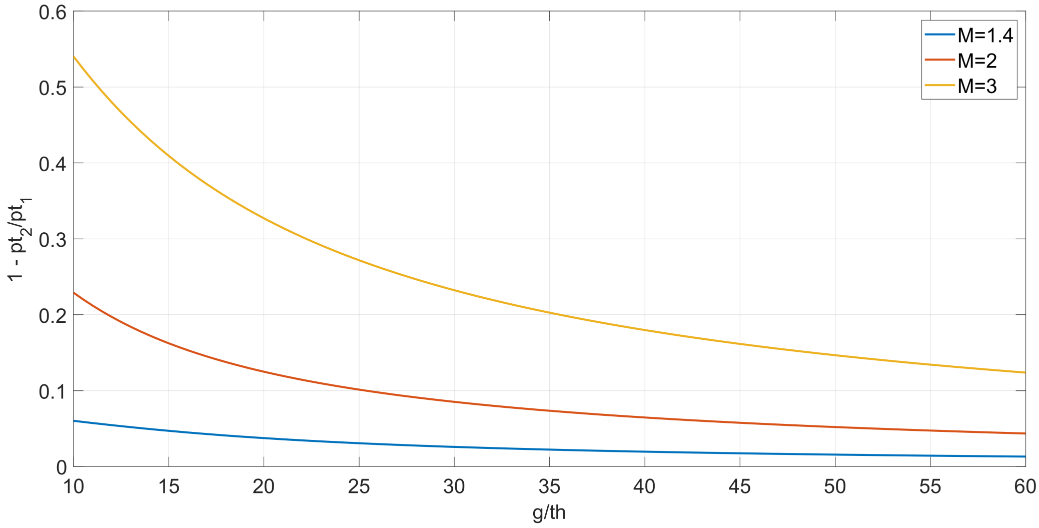


Figure 2.7. Parametric analysis on the total pressure losses caused by bow shock wave.

The results of this analysis are shown in figure 2.7 and the following observations can be made:

- the losses increase with Mach number because the detached shock wave becomes stronger;

- the losses are strongly dependent on the adimensional ratio pitch to thickness $\frac{g}{th}$. This fact has physical meaning because a smaller thickness compared to the blade passage makes the leading edge sharper and reduces the subsonic region behind the bow shock, which is the main responsible of the losses. The reduction in losses decreases faster in the first part and then it slows down till a plateau is reached; finally it is important to remember that a too high ratio cannot be used because the leading edge would be too small to be physically realizable.

2.3.2 First oblique shock wave reflection

When the detached shock wave reaches the blade surface it is reflected as an oblique shock wave; to improve the accuracy in the calculation of the losses the first oblique shock wave will be drawn and the associated losses calculated. Only the first reflection is determined because first of all this is the strongest one and secondly significant approximations are done to estimate it; so if the same assumptions are also applied to the subsequent reflections, the reliability of the results will be seriously affected and the increase in losses would not be that significant.

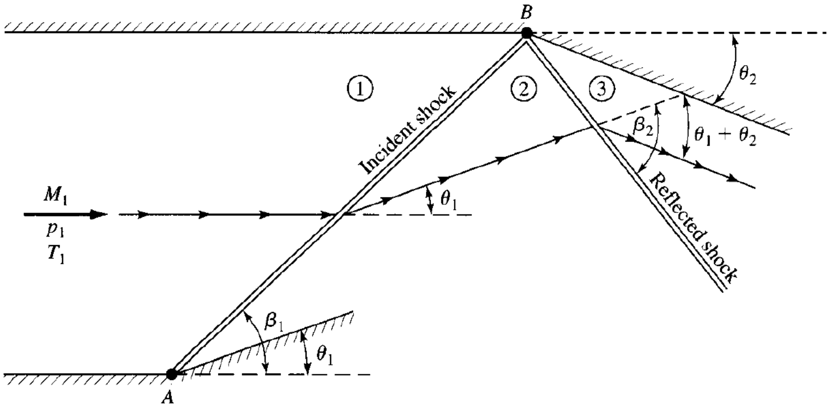


Figure 2.8. Regular reflection from a solid boundary (Figure from [13]).

The basic concept to calculate the reflection is that the flow immediately adjacent to the wall must be parallel to it. As Anderson explains [13] in figure 2.8, in region 2 the streamlines are inclined at an angle θ to the upper wall and originating in B is a reflected shock wave with sufficient strength to turn the flow parallel to the wall and with un upstream Mach number M_2; remember that the strength and the losses of the reflected shock are lower because M_2 is lower than M_1 and for the same reason the shock wave is not specularly reflected.
To calculate the flow angle behind the shock the formula 2.10 is used first to calculate

the angle behind the bow shock wave and then it is used in an implicit equation which is solved numerically to find the slope of the reflected shock.

$$tan\theta = 2\cot\beta\left[\frac{M_1^2 sin^2\beta - 1}{M_1^2(\gamma + cos2\beta) + 2}\right] \tag{2.10}$$

It is important to observe that in this case the first shock is not oblique but a detached bow shock wave, which means that the flow behind the shock is not uniform; for this reason only the last portion of the incident shock is used to calculate the reflection because only the flow closest to the wall must be parallel. Finally the entropy jump is calculated by applying again eq. 2.8 and the comparison of the predicted reflected shock with CFD is shown in figure 2.9.

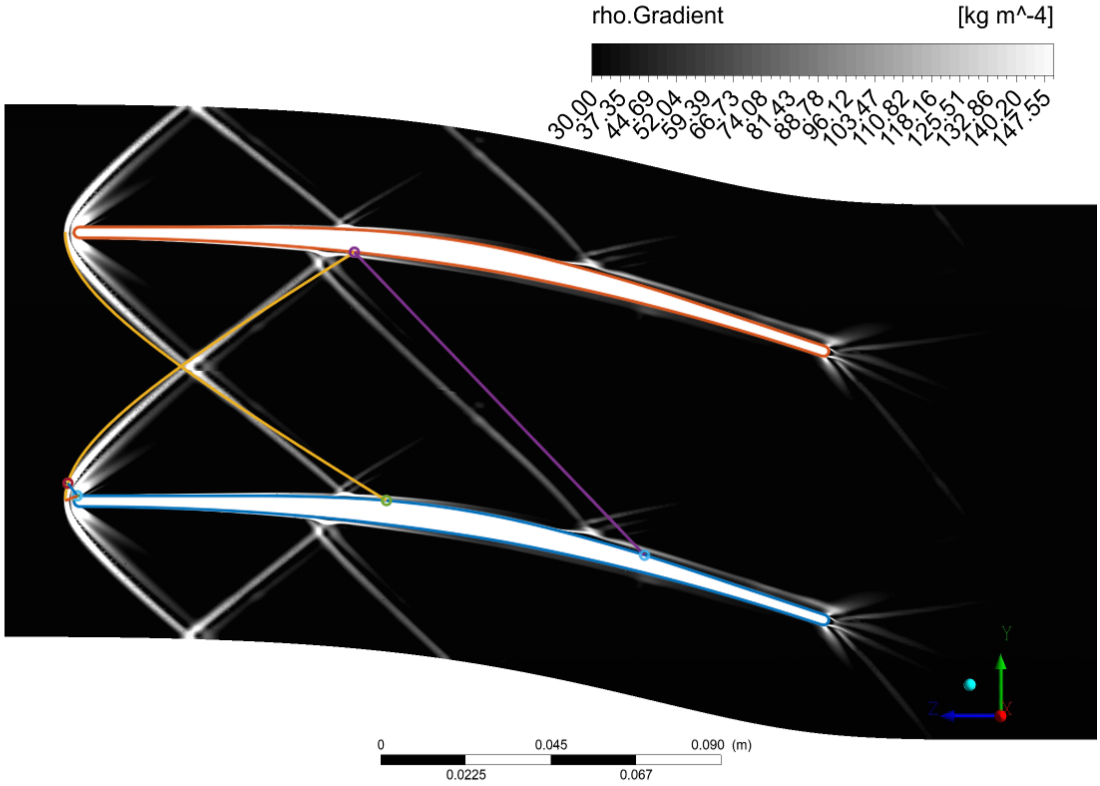


Figure 2.9. Comparison between the reflected oblique shock wave determined analytically (in purple) and CFD.

2.3.3 Boundary layer and mixing losses

The second most important source of losses in a supersonic blade row are boundary layer and mixing losses. To calculate the reduction in efficiency caused by these sources the first step is to estimate the boundary layer quantities.

There are available several method for the calculation of a compressible turbulent boundary layer and for our code the procedure used is the one derived by Stratford and Beavers [48]. Actually the authors of this article started from the extensive work done previously on the topic by many researchers and found a correlation that can resume all the previous investigations reported in eq. 2.11

$$\begin{cases} \theta = f(M)XR_X^{-b} \\ \\ X = P^{-1}\int_0^x P\,dx \\ \\ R_X = \dfrac{a_0}{\nu_0}XM\left(1+\dfrac{M^2}{5}\right)^{-(3-\omega)} \\ \\ P = \left(\dfrac{M}{1+M^2/5}\right)^4 \end{cases} \tag{2.11}$$

where θ is the momentum thickness, X is the equivalent flat plate length, R_X is the Reynolds number based on the local free stream conditions and P is a Mach number function.

In these studies the boundary layer is determined by applying the momentum equation for a compressible turbulent boundary layer, while neglecting in most cases the pressure gradient dependency; also the correlations derived are valid for an insulated wall, thus restricting their range of application. The momentum thickness θ, the displacement thickness δ^* and the fully idealized boundary layer thickness δ are related through the ratios δ^*/θ and θ/δ; in limited pressure gradients, these ratios can be approximated with the results obtained for a flat plate and, as shown by Cope [50], the velocity profile is independent from compressibility effects; after all these considerations, the ratios can be calculated by applying the $\frac{1}{7}$th power law (Figure 2.10). In conclusion first the boundary layer thickness δ is calculated as independent of the Mach number and then the momentum and the displacement thickness are obtained through the $\frac{1}{7}$th power law ratios. In this way the final set of formulae used to calculate the boundary layer quantities is obtained (eq. 2.12 and 2.13).

$$\text{for } R_X \approx 10^6 \begin{cases} \delta = 0.37XR_X^{-\frac{1}{5}} \\ \theta = 0.036(1+\frac{M^2}{10})^{-0.70}XR_X^{-\frac{1}{5}} \\ \delta^* = 0.046(1+0.8M^2)^{0.44}XR_X^{-\frac{1}{5}} \end{cases} \tag{2.12}$$

$$\text{for } R_X \approx 10^7 \begin{cases} \delta = 0.23XR_X^{-\frac{1}{6}} \\ \theta = 0.022(1+\frac{M^2}{10})^{-0.70}XR_X^{-\frac{1}{6}} \\ \delta^* = 0.028(1+0.8M^2)^{0.44}XR_X^{-\frac{1}{6}} \end{cases} \tag{2.13}$$

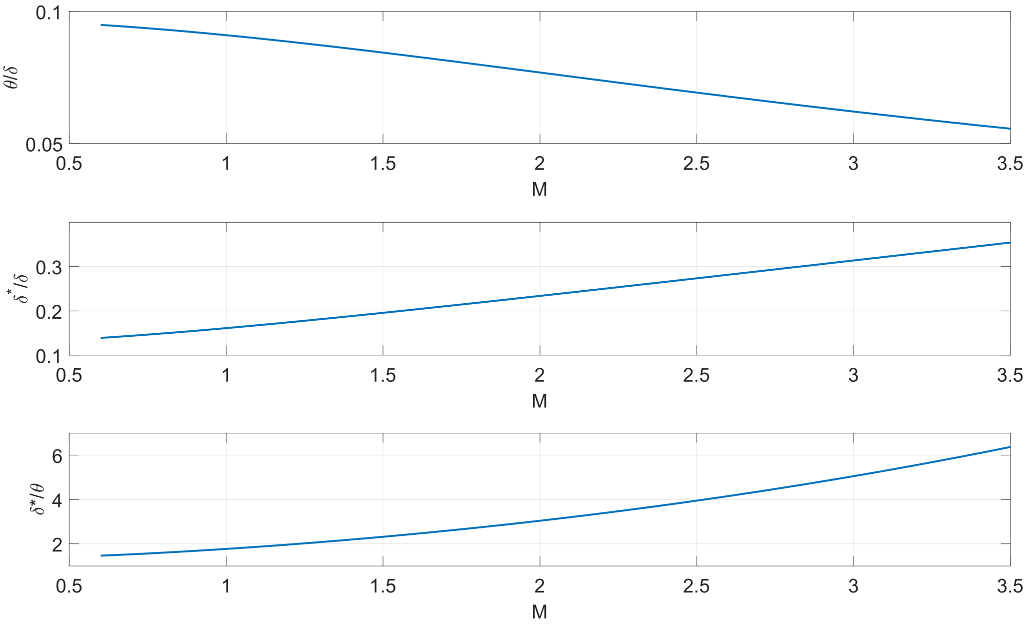


Figure 2.10. Trend of the boundary layer quantities ratios with respect to Mach number.

Once the boundary layer quantities have been calculated the first part of the procedure is completed; now these new data will be used to determine the overall loss coefficients associated with boundary layer and mixing losses through the method demonstrated by Stewart [14].
In a turbomachine blade row the two-dimensional friction loss produced by the air flowing between the blades is usually expressed as a mass averaged total pressure or kinetic energy defect at the blade exit; after the flow leaves the blade, mixing takes place until uniform conditions are reached. The total losses generated in the row are obtained by comparing the completely mixed flow with the inlet conditions and these losses are significantly higher then considering only the mass average loss at the blade exit. In the study done by Stewart the following main observations can be drawn on the topic:

- the loss coefficients based on kinetic energy both before and after mixing are approximately independent of compressibility and for this reason are more desirable in describing the compressible-flow blade loss characteristics;
- an accurate evaluation of the momentum thickness is fundamental to achieve results that are reliable;
- the loss coefficients after mixing are significantly greater than those at the trailing edge and this difference increase with higher velocity flows, especially the supersonic ones;
- the overall loss coefficients are strongly dependent on the trailing edge thickness.

Now before going into the detail of the formulas required to calculate these losses, it is necessary to give a nomenclature to the different stations (notice that this notation is restricted only to this paragraph); as shown in figure 2.11, station 0 represents the inlet of the blade row where the flow condition is completely uniform, station 1 is positioned just downstream the blades with a clear effect of the boundary layer and trailing edge wake and finally station 2 is located sufficiently far from the row to reach a completely mixed flow condition.

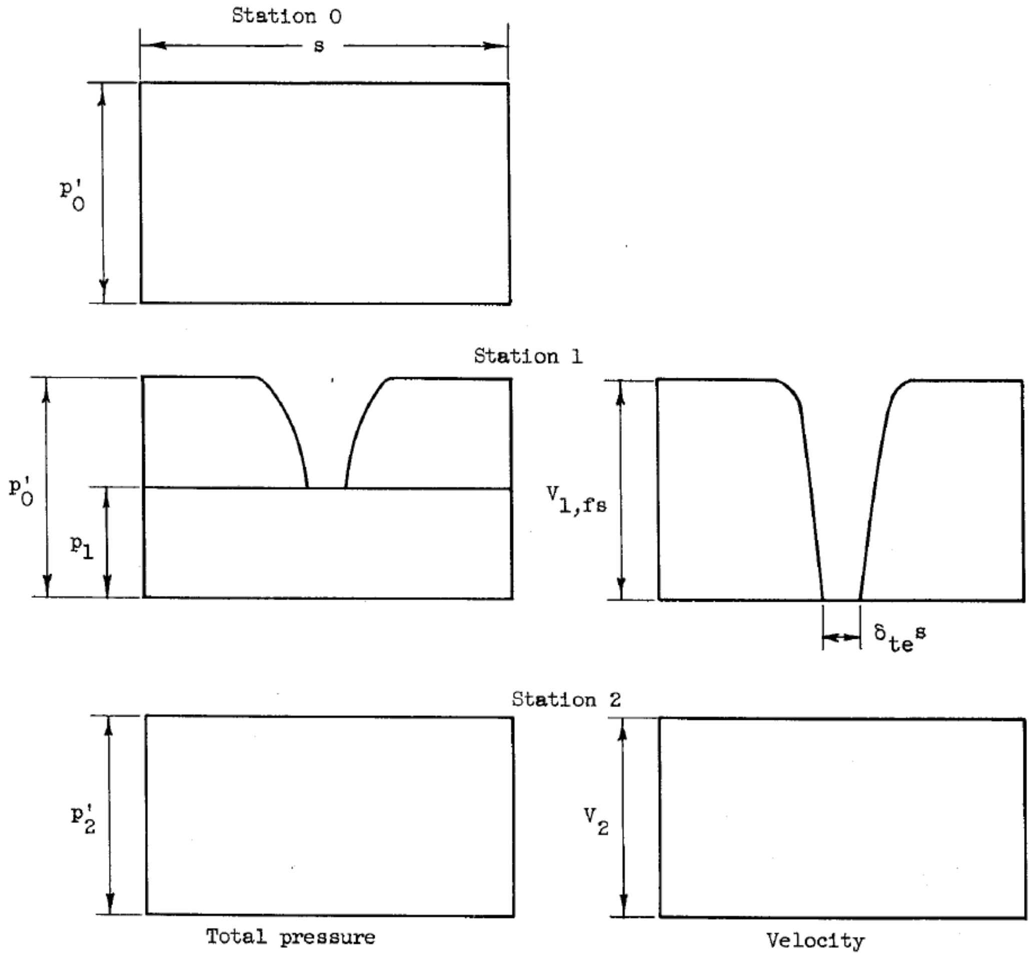


Figure 2.11. Stations nomenclature for boundary and mixing losses calculation (Figure from [14]).

In Stewart's method the boundary layer quantities are averaged between suction side and pressure side; therefore to link this theory with the correlations derived by Stratford and Beavers the following equations are applied:

$$\begin{cases} \delta^{**} = \dfrac{\delta_s^* + \delta_p^*}{gcos(\alpha_1)} = \dfrac{2\delta_{S-B}^*}{gcos(\alpha_1)} \\[2ex] \theta^{**} = \dfrac{\theta_s + \theta_p}{gcos(\alpha_1)} = \dfrac{2\theta_{S-B}^*}{gcos(\alpha_1)} \\[2ex] \delta_{te}^{**} = \dfrac{t}{gcos(\alpha_1)} \end{cases} \tag{2.14}$$

where the subscript S-B stands for Stratford-Beavers, g is the blade pitch and t is the blade thickness at trailing edge.

In the next paragraph the main steps to derive the overall loss coefficients are reported, starting from the conservation of mass, momentum in tangential and axial direction for a compressible flow in dimensionless form (the subscript t is used to indicate total state quantities).

$$\begin{cases} \textbf{Continuity} \\ cos(\alpha_1)(1-\delta^{**}-\delta^{**}_{te})\left(\dfrac{\rho v}{\rho_t v_{cr}}\right)_{fs,1} = \left(\dfrac{\rho v_x}{\rho_t v_{cr}}\right)_2 \dfrac{p_{2t}}{p_{0t}} \\ \\ \textbf{Momentum in tangential direction} \\ sin(\alpha_1)cos(\alpha_1)(1-\delta^{**}-\delta^{**}_{te}-\theta^{**})\left(\dfrac{\rho v^2}{\rho_t v^2_{cr}}\right)_{fs,1} = \left(\dfrac{\rho v_x v_u}{\rho_t v^2_{cr}}\right)_2 \dfrac{p_{2t}}{p_{0t}} \\ \\ \textbf{Momentum in axial direction} \\ \dfrac{p_1}{p_{0t}}\dfrac{\gamma+1}{2\gamma} + cos^2\alpha_1(1-\delta^{**}-\delta^{**}_{te}-\theta^{**})\left(\dfrac{\rho v^2}{\rho_t v^2_{cr}}\right)_{fs,1} = \dfrac{p_2}{p_{2t}}\dfrac{p_{2t}}{p_{0t}}\dfrac{\gamma+1}{2\gamma} + \left(\dfrac{\rho v_x^2}{\rho_t v^2_{cr}}\right)_2 \dfrac{p_{2t}}{p_{0t}} \end{cases} \tag{2.15}$$

Substituting the first equation into the third one with $\frac{p_1}{p_0'} = (\frac{\rho}{\rho'})_{fs,1}(1-A_{fs,1})$, equation 2.16 is obtained.

$$\begin{aligned} &(1-A_{fs,1})\frac{\gamma+1}{2\gamma} + cos^2\alpha_1(1-\delta^{**}-\delta^{**}_{te}-\theta^{**})\left(\frac{v^2}{v^2_{cr}}\right)_{fs,1} = \\ &= \left[\frac{p_2}{p_{2t}}\frac{\gamma+1}{2\gamma} + \left(\frac{\rho v_x^2}{\rho_t v^2_{cr}}\right)_2\right]\left[\frac{cos(\alpha_1)(1-\delta^{**}-\delta^{**}_{te})\left(\frac{v}{v_{cr}}\right)_{fs,1}}{\left(\frac{\rho v_x^2}{\rho_t v^2_{cr}}\right)_2}\right] \end{aligned} \tag{2.16}$$

Then by using the equation of state and energy and rearranging the terms through C and D, a second order equation in function of $(\frac{v_x}{v_{cr}})_2$ is derived.

$$\begin{cases} \left(\dfrac{v_x}{v_{cr}}\right)^2_2 - \dfrac{2\gamma C}{\gamma+1}\left(\dfrac{v_x}{v_{cr}}\right)_2 + \left(1-\dfrac{\gamma-1}{\gamma+1}D^2\right) = 0 \\ \\ C = \dfrac{(1-A_{fs,1})\dfrac{\gamma+1}{2\gamma} + cos^2\alpha_1(1-\delta^{**}-\delta^{**}_{te}-\theta^{**})\left(\dfrac{v^2}{v^2_{cr}}\right)_{fs,1}}{cos(\alpha_1)(1-\delta^{**}-\delta^{**}_{te})\left(\dfrac{v}{v_{cr}}\right)_{fs,1}} \\ \\ D = \left(\dfrac{v_u}{v_{cr}}\right)_2 = \left(\dfrac{v}{v_{cr}}\right)_{fs,1} sin\alpha_1\left(\dfrac{1-\delta^{**}-\delta^{**}_{te}-\theta^{**}}{1-\delta^{**}-\delta^{**}_{te}}\right) \end{cases} \tag{2.17}$$

The solution of the second order equation allows two results and in the original article by Stewart [14] it was always used the minus sign one to find the correct conditions after mixing. This was consistent physically, because with a subsonic flow the increase in available area from the trailing edge to the mixed region reduces the velocity, but in our case the flow is supersonic with an axial Mach number higher than 1, which means that the channel will not behave like a diffuser but as a nozzle; hence to take account of this factor, differently from Stewart the solution of the second order equation selected is the one with the positive sign (eq. 2.18).

$$\left(\frac{v_x}{v_{cr}}\right)_2 = \frac{\gamma C}{\gamma+1} + \sqrt{\left(\frac{\gamma C}{\gamma+1}\right)^2 - 1 + \frac{\gamma-1}{\gamma+1}D^2} \tag{2.18}$$

With the new results, the critical Mach number at station 2 can be calculated and used in equation 2.19 to calculate the static to total ratio for the density at station 2; then the total pressure loss between station 0 and station 2 is determined through equation 2.20.

$$\left(\frac{\rho}{\rho_t}\right)_2 = \left\{1 - \frac{\gamma-1}{\gamma+1}\left[D^2 + \left(\frac{v_x}{v_{cr}}\right)_2^2\right]\right\}^{\dfrac{1}{\gamma-1}} \tag{2.19}$$

$$\frac{p_{2t}}{p_{0t}} = \frac{\left(\frac{\rho v}{\rho_t v_{cr}}\right)_{fs,1} cos\alpha_1(1-\delta^{**}-\delta^{**}_{te})}{\left(\frac{\rho v_x}{\rho_t v_{cr}}\right)_2} \tag{2.20}$$

Finally the losses associated to the boundary layer and mixing are quantified in two different ways: the first one converts the total pressure drop in enthalpy increase, while the second is based on the kinetic energy, which as stated before is independent from compressibility considerations.

$$\begin{cases} \Delta s = -R\log\left(\frac{p_{2t}}{p_{0t}}\right) \\ \bar{e_2} = \frac{\left(\frac{p_{0t}}{p_{2t}}\right)^{\frac{\gamma-1}{\gamma}} - 1}{\left(\frac{p_{0t}}{p_2}\right)^{\frac{\gamma-1}{\gamma}} - 1} \end{cases} \tag{2.21}$$

During the previous steps two data were used that are not directly known in the mean-line code: the critical Mach number and the flow angle at station 1. To solve this issue the following strategy is applied:

- the function to calculate the boundary and mixing losses is inserted inside the main mean-line code, in which the Mach number at station 2 has been already determined; so the critical Mach number at the trailing edge is varied through an iterative procedure in order to obtain the same Mach number after mixing given by the external code (Fig. 2.12);

- the flow angle at trailing edge in first approximation is assumed equal to the flow angle after mixing, known from the external mean-line code.

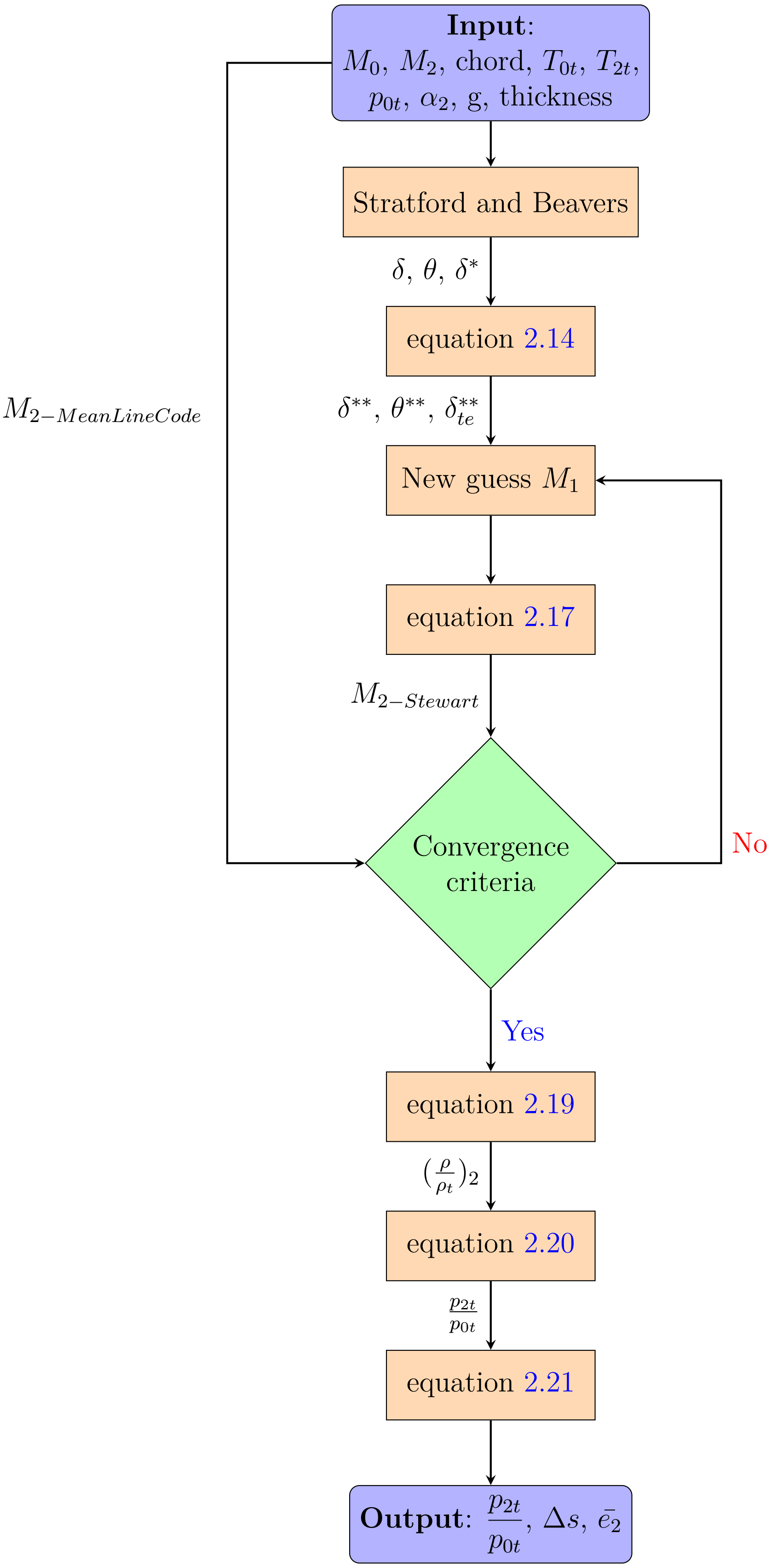

Input:
M_0, M_2, chord, T_{0t}, T_{2t},
p_{0t}, α_2, g, thickness
Stratford and Beavers
δ, θ, δ^*
equation 2.14
δ^{**}, θ^{**}, δ^{**}_{te}
New guess M_1
equation 2.17
$M_{2-Stewart}$
$M_{2-MeanLineCode}$
Convergence criteria
No
Yes
equation 2.19
$(\frac{\rho}{\rho_t})_2$
equation 2.20
$\frac{p_{2t}}{p_{0t}}$
equation 2.21
Output: $\frac{p_{2t}}{p_{0t}}$, Δs, $\bar{e_2}$

Figure 2.12. Flowchart to illustrate the iterative process used to calculate the boundary and mixing losses.

2.4 Blade profile design through the method of characteristics (MOC)

In a supersonic turbine the relative flow velocity inside the rotor, as the name suggests, is supersonic. The main advantage of this particular machine is the capacity to produce a large specific work output, enabling the possibility to have a configuration with a reduced amount of driving fluid and fewer stages; on the other side, the supersonic turbine is also characterized by a lower efficiency due to the complex pattern of shock waves and a higher kinetic energy losses at exit. In order to achieve the highest efficiency possible, a specific method of characteristics developed by Louis J. Goldman [15] has been used to define the shape of the stator and the rotor of the supersonic turbine. Before describing Goldman version of the MOC, let's introduce some basic knowledge about the method of characteristics.

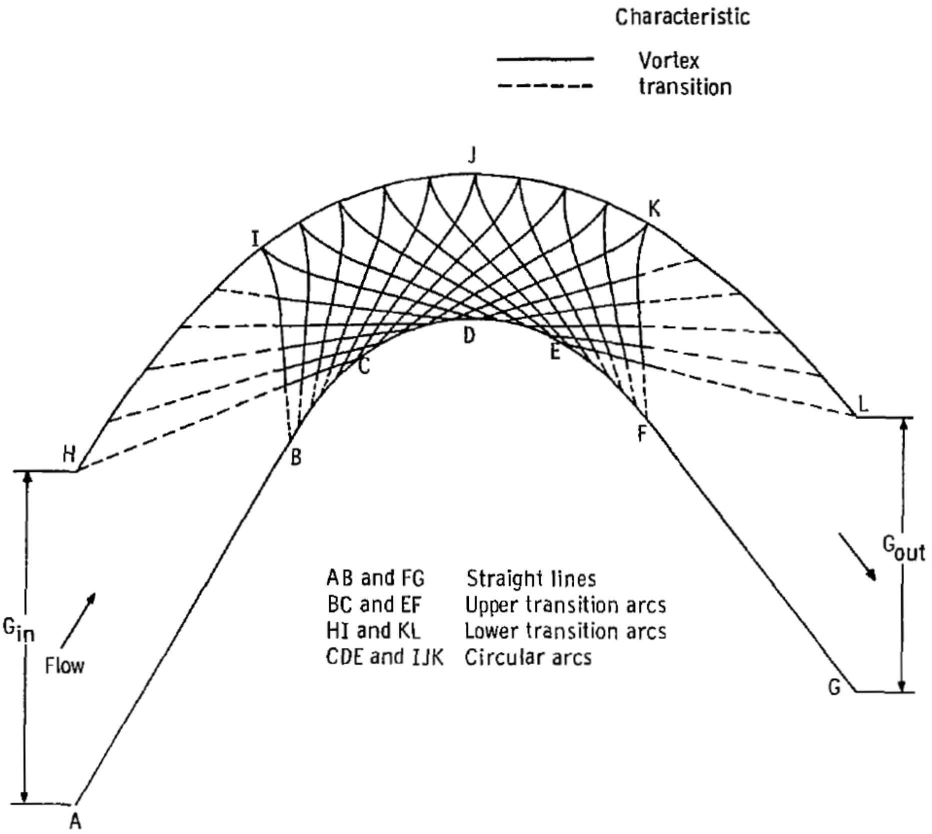

Characteristic
Vortex
transition
J
I
K
D
C
E
L
H
B
F
G
A
G_{in}
Flow
G_{out}
AB and FG Straight lines
BC and EF Upper transition arcs
HI and KL Lower transition arcs
CDE and IJK Circular arcs

Figure 2.13. Design of ideal supersonic passage by method of characteristic (Figure from [15]).

The method of characteristics is a general method for solving hyperbolic partial differential equations, which is exactly the form in which Navier-Stokes equations present themselves when specific hypothesis are included: the flow considered is a two dimensional, steady, inviscid supersonic flow. Actually some of these hypothesis can be relaxed and MOC can still be applied, but considering this simplified form does not modify the main concepts behind this method and also Goldman version used in the code works precisely with the same simplifying hypothesis. To determine the flow properties in the entire domain, three main steps must be performed [13]:

1. locate the characteristic lines: the coordinates of the points of these lines can be determined because the flow variables (p, ρ, T, u, ν) are continuous, while the derivatives ($\partial p/\partial x$, $\partial u/\partial y$, etc.) are indeterminate and across them sometimes the derivatives are even discontinuous;

2. convert the partial differential equations into ordinary differential equations that hold only along the characteristic lines: these new equations are called "compatibility equations" and they relate the velocity magnitude and direction along the characteristic lines;

3. starting from the initial conditions given at some point or region in the flow, solve the compatibility equations step by step along the characteristic lines determining in this way the flow properties in the whole domain. In the most general case the characteristic lines depend on the flow, while the compatibility equations are in function of the location along the characteristic line; because they are influenced by each other, the characteristic and the compatibility equations must be solved simultaneously. The only exception is a two dimensional irrotational flow, for which the compatibility equations are algebraic equations independent of the location on the characteristic line.

It is important to underline that the solution of a hyperbolic partial differential equation requires a downstream-marching method, which means that the properties of the flow must be known on a line from which the method starts and solve the entire flow domain; the flow properties on the line can be assigned through boundary conditions or can be determined through other equations depending on the type of the problem.
In a steady supersonic flow the disturbances cannot propagate upstream, while in a subsonic flow the disturbances are felt everywhere in the domain. This particular behaviour of a supersonic flow is clearly shown by the methods of characteristics: in figure 2.14 the right and left running characteristic lines through point A divide the domain in two separate regions; the upstream region is called domain of dependence and is composed of all those points that may perturb the flow properties at point A, while the downstream one is the region of influence and contains the points affected by the flow in point A.

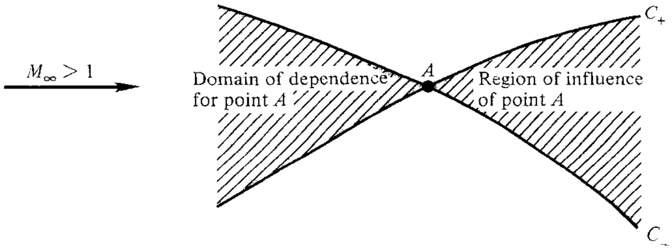

$M_\infty > 1$
Domain of dependence for point A
A
Region of influence of point A
C_+
C_-

Figure 2.14. Domain of dependence and region of influence (Figure from [13]).

Actually there are two method based on the method of characteristics to design the profile of a supersonic blade row: corner-flow method by Shapiro [51] and vortex-flow method by Goldman [15]. Before describing in detail the procedure for the vortex-flow method, it is significant to describe briefly the logic behind the corner flow method and why it was not selected for our purpose.

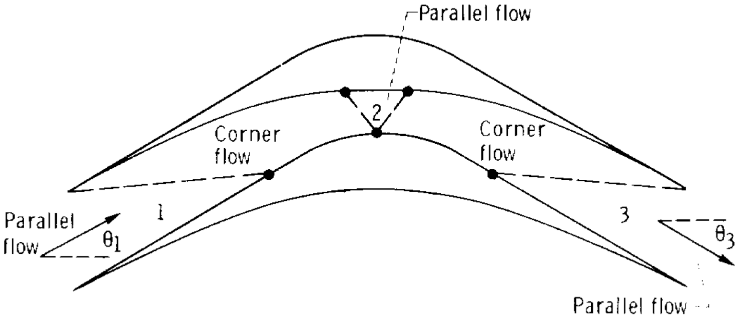
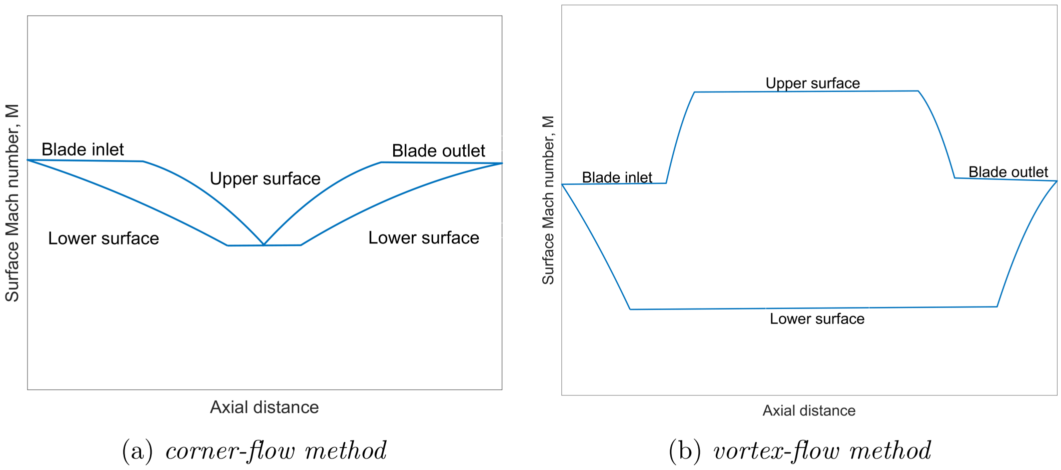


Figure 2.15. Supersonic blade row design by the corner-flow method (Figure from [15]).

The entering flow is uniform and parallel and it is subjected to a corner-type compression in region 1 along the pressure side of the blade; in region 2 the flow is horizontal and finally in region 3 the flow undergoes a corner-type expansion until a uniform and parallel flow of the desired Mach number is reached at the blade exit. This type of blade is really easy to design because in each region there are only one type of waves (compression or expansion), but at the same type it has also two main drawbacks:

- the velocity distribution is not ideal because the loading becomes zero in the centre of the blade;
- for an assigned inlet Mach number, the amount of flow turning is limited by sonic or higher velocity in region 2.



(a) *corner-flow method*

(b) *vortex-flow method*

Figure 2.16. Velocity distribution comparison between corner-flow method and vortex-flow method.

Both these limitations are overcome by the vortex-flow method: the velocity distribution is significantly better with the most loaded region at the centre and the bound on the flow turning is restricted not by the vortex-flow method (Figure 2.16), but by the starting condition of the supersonic blade row; for all these reasons finally the vortex-flow method by Goldman was selected to be implemented in the mean-line code.

The vortex-flow method designs a blade profile that converts a uniform parallel flow at the blade inlet into a vortex flow field, turns the vortex field and finally reconverts it to a uniform parallel flow at the blade exit. The vortex flow method requires several inputs to define the blade profile: inlet Mach number, outlet Mach number, lower surface Mach number, upper surface Mach number, inlet flow angle and the specific heat ratio of the working fluid.

To apply the method of characteristics the domain is divided into small regions and within each of them the flow properties are assumed to be constant (Figure 2.13). The boundaries between the regions are or characteristic lines (which are also Mach lines as demonstrated many times in the theory of MOC) or physical boundaries that the flow is forced to follow.

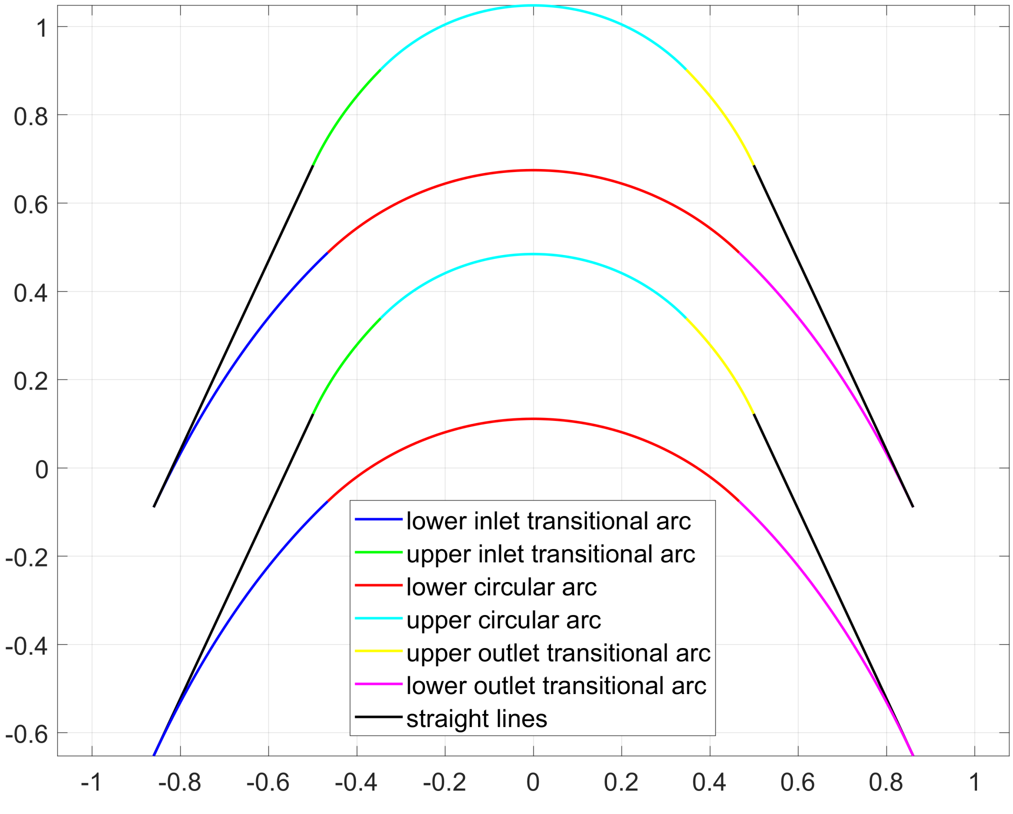


Figure 2.17. Blade profile division in vortex-flow method.

The blade profile is composed of the following fundamental parts and each of them has a specific function (Figure 2.17):

- **inlet transition arcs** transform the assumed uniform flow at the inlet into a vortex flow. On the lower transition arc the Mach number is reduced from the inlet value M_i to the specified lower surface Mach number M_l, while on the upper transition arc the Mach number is increased to reach the upper surface Mach number M_u (Figure 2.16);

- **concentric circular arcs** preserve the vortex-flow structure while turning the flow with a constant surface Mach number;
- **outlet transition arcs** reconvert the vortex flow into a uniform parallel flow. On the lower transition arc the Mach number is increased from M_l, while on the upper transition arc the Mach number is decreased from M_u; both of them reach at the exit section the input specified outlet Mach number M_o (Figure 2.16);
- **straight lines** are added to the upper surface of the blade in order to obtain a closed geometry.

The total amount of flow turning produced by the upper surface or the lower surface is the sum of the turning produced by the transition arcs and the circular arcs. To quantify the total flow deviation it is more convenient to work with the Prandt-Meyer angles ν, defined as the angle through which a flow turns isentropically from sonic velocity to a designated Mach number. The formula to calculate the Prandtl-Meyer angle is shown in equation 2.22, which requires first to determine the critical velocity ratio M^* through equation 2.23.

$$\nu = \frac{\pi}{4}\left(\sqrt{\frac{\gamma+1}{\gamma-1}}-1\right)+\frac{1}{2}\left\{\sqrt{\frac{\gamma+1}{\gamma-1}}arcsin[(\gamma-1)M^{*^2}-\gamma]+arcsin\left(\frac{\gamma+1}{M^{*^2}}-\gamma\right)\right\} \tag{2.22}$$

$$M^* = \left(\frac{\frac{\gamma+1}{2}M^2}{1+\frac{\gamma-1}{2}M^2}\right)^{\frac{1}{2}} \tag{2.23}$$

The flow turning produced by the transition arcs can be now comfortably calculated as the difference between the Prandtl-Meyer angles and it is $\nu_i - \nu_l$ for the inlet lower transition arc, $\nu_o - \nu_l$ for the outlet lower transition arc, $\nu_u - \nu_i$ for the inlet upper transition arc and $\nu_u - \nu_o$ for the outlet upper transition arc. It is important to observe that the maximum flow turning produced by the inlet or outlet transition arcs cannot be grater than the inlet or outlet flow angles, β_i or β_o respectively.

2.4.1 Circular arcs

All the equations used in the profile construction are presented in the their dimensionless form to employ a generalised approach and only at the end the shape will be rescaled to obtain the dimensional blade geometry (dimensionless quantities are distinguished using * as superscript). In the circular part of the channel, as already explained before, there is a vortex flow structure that respects equation 2.24.

$$M^* R^* = 1 \tag{2.24}$$

where $M^* = v/v_{cr}$ is the dimensionless velocity, $R^* = R/r^*$ is the dimensionless radius, v_{cr} is the critical velocity and r^* is the radius of the sonic velocity streamline in the vortex field. Because the upper surface and lower surface Mach number are inputs, the upper and lower radius R^*_u and R^*_l are calculated by equation 2.24.

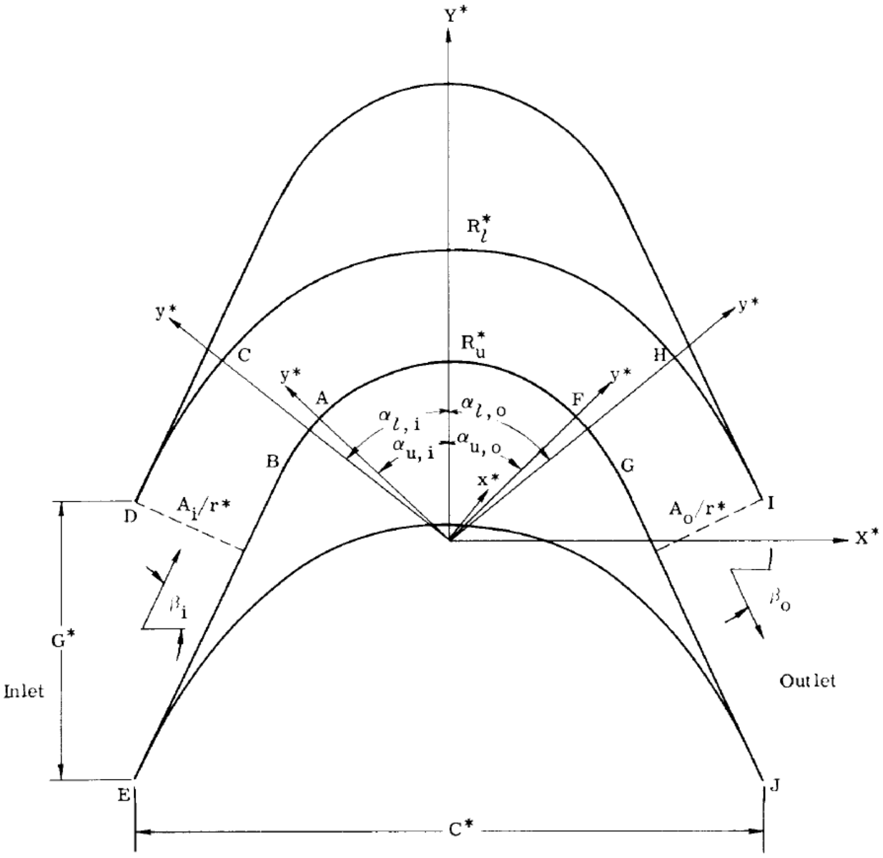


Figure 2.18. Dimensionless supersonic blade profile geometry for vortex-flow method (Figure from [15]).

Once the radius of the two concentric arcs are determined, to draw the arcs it is necessary to calculate the starting and the ending angles. Conventionally it is assumed that the counter-clockwise direction is positive and then all four angles are calculated by taking account of the flow turning produced by the upper and lower transition arcs (Equation 2.25).

$$\begin{cases} \alpha_{l,i} = \beta_i - (\nu_i - \nu_l) \\ \alpha_{l,o} = \beta_o + (\nu_o - \nu_l) \\ \alpha_{u,i} = \beta_i - (\nu_u - \nu_i) \\ \alpha_{u,o} = \beta_o + (\nu_u - \nu_o) \end{cases} \tag{2.25}$$

In equation 2.25 all the quantities are already known or can be calculated using equation 2.22; the only exception is the outlet flow angle β_o, which was not specified in the inputs of the method because it is now derived by applying the continuity equation on the two dimensional blade passage (Equation 2.26).

$$\beta_o = -arccos\left\{\left[\frac{M_i}{M_o}\left(\frac{1+\frac{\gamma-1}{2}M_o^2}{1+\frac{\gamma-1}{2}M_i^2}\right)^{\frac{\gamma+1}{2(\gamma-1)}}\right]cos(\beta_i)\right\} \tag{2.26}$$

2.4.2 Lower inlet transition arc

Figure 2.19 shows the nomenclature used in the equations necessary to build the lower inlet transition arc and notice that also in this case a dimensionless approach can be used.

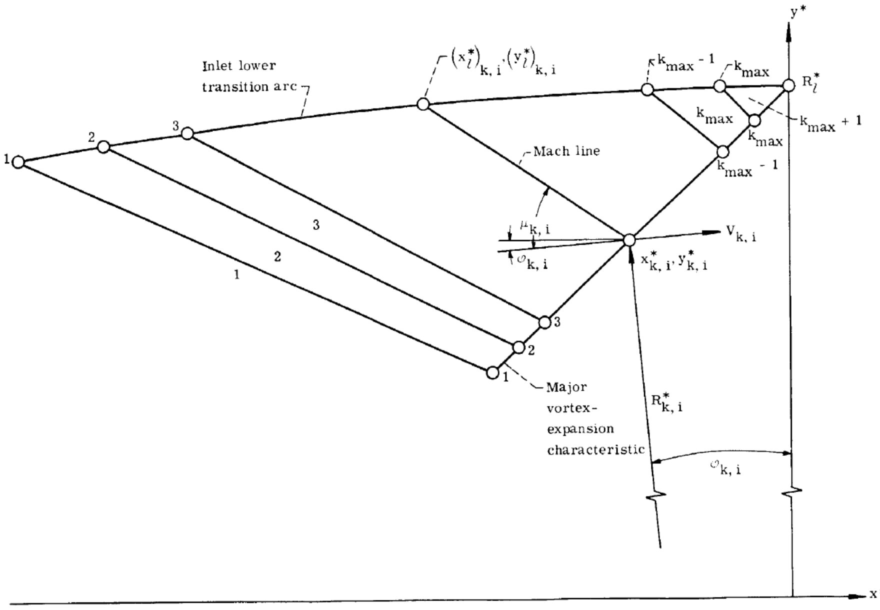


Figure 2.19. Lower inlet transition arc construction (Figure from [15]).

The transition arc starts from ($x^* = 0$, $y^* = R_l^*$); then the following points are obtained sequentially through the intersection of the straight-line wall segments and straight Mach lines associated to a small variation in flow turning. The Mach lines are determined from the major vortex-expansion characteristic, while the wall segments are parallel to the flow direction. Once the lower transition arc is constructed it is rotated of an angle $\alpha_{l,i}$ in order to place it in the correct position in the profile geometry. Now let's explicit all this procedure in its mathematical form.
On the characteristic line, the velocity direction ϕ and the dimensionless radius R^* are linked together by equations 2.27 and 2.28.

$$\phi = \pm\frac{1}{2}f(R^*) + constant \tag{2.27}$$

$$f(R^*) = \sqrt{\frac{\gamma+1}{\gamma-1}}arcsin\left(\frac{\gamma-1}{R^{*2}} - \gamma\right) + arcsin[(\gamma+1)R^{*^2} - \gamma] \tag{2.28}$$

In equation 2.27 a sign must be selected considering physically the problem: a positive sign corresponds to an expansion characteristic line, while a negative one represents a compression line. Then to completely define the major vortex expansion characteristic line, it is necessary to impose as boundary condition that the velocity direction is horizontal $\phi = 0$ at $R^* = R_l^*$.

$$\phi = \frac{1}{2}[f(R^*) - f(R_l^*)] \tag{2.29}$$

Equation 2.29 expresses the flow direction on each point of the major vortex expansion characteristic line. If the domain is divided into small regions, it can be assumed that in the regions adjacent to the characteristic line the flow direction is exactly the same; for k transition arc segments, each of them producing $\Delta\nu$ turning, the flow direction in each segment is:

$$\phi_{k,i} = \nu_i - \nu_l - (k-1)\Delta\nu \tag{2.30}$$

where k is an integer that goes from $k = 1 : [(\nu_i - \nu_l)/\Delta\nu + 1]$. Now let's put together equations 2.22, 2.24, 2.28, 2.29 and 2.30.

$$\phi_{k,i} = \frac{1}{2}[f(R^*_{k,i}) - f(R^*_l)]$$

$$\nu_i - \nu_l - (k-1)\Delta\nu = \frac{1}{2}[f(R^*_{k,i}) - f(R^*_l)]$$

$$\nu_i - \frac{\pi}{4}\left(\sqrt{\frac{\gamma+1}{\gamma-1}} - 1\right) + \frac{1}{2}\left\{\sqrt{\frac{\gamma+1}{\gamma-1}} arcsin[(\gamma-1)M_l^{*^2} - \gamma] + arcsin\left(\frac{\gamma+1}{M_l^{*^2}} - \gamma\right)\right\} - (k-1)\Delta\nu =$$
$$= \frac{1}{2}f(R^*_{k,i}) + \frac{1}{2}\sqrt{\frac{\gamma+1}{\gamma-1}} arcsin\left(\frac{\gamma-1}{R_l^{*^2}} - \gamma\right) + \frac{1}{2}arcsin[(\gamma+1)R_l^{*^2} - \gamma]$$

$$f(R^*_{k,i}) = 2\nu_i - \frac{\pi}{2}\left(\sqrt{\frac{\gamma+1}{\gamma-1}} - 1\right) - 2(k-1)\Delta\nu \tag{2.31}$$

Then for an assigned $\Delta\nu$ and for each value of k, equations 2.28 and 2.31 are solved together and $R^*_{k,i}$ is determined; finally the coordinates of the major expansion characteristic line is obtained by projecting the radius on the axes (Equation 2.32).

$$\begin{cases} x^*_{k,i} = -R^*_{k,i} sin(\phi_{k,i}) \\ y^*_{k,i} = R^*_{k,i} cos(\phi_{k,i}) \end{cases} \tag{2.32}$$

The points of the major expansion characteristic line belong also to the straight Mach lines (Figure 2.19), which are completely defined once the slope is calculated; the slope of the Mach line is determined as an average between the two adjacent regions (Equation 2.33), where the Mach line is inclined at an angle μ respect the velocity direction.

$$\begin{cases} m_{k,i} = tan\left(\frac{\phi_{k,i} + \phi_{k+1,i}}{2} + \frac{\mu_{k,i} + \mu_{k+1,i}}{2}\right) \\ \mu_{k,i} = -arcsin\left(\frac{1}{M_{k,i}}\right) = -arcsin\left[\sqrt{\left(\frac{\gamma+1}{2}\right)R_{k,i}^{*^2} - \left(\frac{\gamma-1}{2}\right)}\right] \\ y^* = m_{k,i}(x^* - x^*_{k,i}) + y^*_{k,i} \qquad \text{Mach line equation} \end{cases} \tag{2.33}$$

The transition arc segments are comfortably determined because they are straight lines parallel to the flow direction ϕ (Equation 2.34).

$$\begin{cases} \bar{m}_{k,i} = tan(\phi_{k+1,i}) \\ y^* = \bar{m}_{k,i}[x^* - (x_l^*)_{k+1,i}] + (y_l^*)_{k+1,i} \qquad \text{Mach line equation} \end{cases} \tag{2.34}$$

Notice that in equation 2.34, $(x_l^*)_{k+1,i}$ and $(y_l^*)_{k+1,i}$ are the coordinates of the lower transition arc, which are unknown except at the starting point $k = [(\nu_i - \nu_l)/\Delta\nu + 1]$ where $x_l^* = 0$ and $y_l^* = R_l^*$. Hence all the remaining points are obtained in a sequential procedure from $k = k_{max}$ to $k = 1$ through the intersection of the straight Mach lines with the straight wall segments (Equation 2.35).

$$\begin{cases} (x_l^*)_{k,i} = \dfrac{\Big[(y_l^*)_{k+1,i} - \bar{m}_{k,i}(x_l^*)_{k+1,i}\Big] - (y_{k,i} - m_{k,i}x^*_{k,i})}{m_{k,i} - \bar{m}_{k,i}} \\ (y_l^*)_{k,i} = \dfrac{m_{k,i}\Big[(y_l^*)_{k+1,i} - \bar{m}_{k,i}(x_l^*)_{k+1,i}\Big] - \bar{m}_{k,i}(y_{k,i} - m_{k,i}x^*_{k,i})}{m_{k,i} - \bar{m}_{k,i}} \end{cases} \tag{2.35}$$

The final step of the procedure is to rotate the lower transition arc of an angle $\alpha_{l,i}$ (Figure 2.18)to place it in the correct reference frame X^* and Y^* of the blade profile geometry (Equation 2.36).

$$\begin{cases} (X_l^*)_{k,i} = (x_l^*)_{k,i}cos(\alpha_{l,i}) - (y_l^*)_{k,i}sin(\alpha_{l,i}) \\ (Y_l^*)_{k,i} = (x_l^*)_{k,i}sin(\alpha_{l,i}) + (y_l^*)_{k,i}cos(\alpha_{l,i}) \end{cases} \tag{2.36}$$

2.4.3 Upper inlet transition arc, lower outlet transition arc and upper outlet transition arc

For the construction of all the other transition arcs the procedure conceptually is identical to the lower transition arc; for the sake of brevity the complete proof with all the steps will not be repeated for each arc, but at the same time the main formulas required to build the profile are here reported to give a complete description.

Upper inlet transition arc

$$\begin{cases} \phi = -\dfrac{1}{2}f(R^*) + constant \qquad \text{Major vortex-compression characteristic line} \\ \phi = \dfrac{1}{2}[-f(R^*) + f(R_u^*)] \\ f(R^*_{k,i}) = 2\nu_i - \dfrac{\pi}{2}\left(\sqrt{\dfrac{\gamma+1}{\gamma-1}} - 1\right) + 2(k-1)\Delta\nu \\ \mu_{k,i} = +arcsin\left(\dfrac{1}{M_{k,i}}\right) = +arcsin\left[\sqrt{\left(\dfrac{\gamma+1}{2}\right)R^{*2}_{k,i} - \left(\dfrac{\gamma-1}{2}\right)}\right] \\ k = 1 : [(\nu_u - \nu_i)/\Delta\nu + 1] \end{cases} \tag{2.37}$$

Lower outlet transition arc

$$\begin{cases} \phi = -\dfrac{1}{2} f(R^*) + constant & \text{Major vortex-compression characteristic line} \\ \phi = \dfrac{1}{2}[-f(R^*) + f(R_l^*)] \\ f(R_{k,i}^*) = 2\nu_o - \dfrac{\pi}{2}\left(\sqrt{\dfrac{\gamma+1}{\gamma-1}} - 1\right) - 2(k-1)\Delta\nu \\ \mu_{k,i} = +arcsin\left(\dfrac{1}{M_{k,i}}\right) = +arcsin\left[\sqrt{\left(\dfrac{\gamma+1}{2}\right) R_{k,i}^{*2} - \left(\dfrac{\gamma-1}{2}\right)}\right] \\ k = 1 : [(\nu_o - \nu_l)/\Delta\nu + 1] \end{cases} \tag{2.38}$$

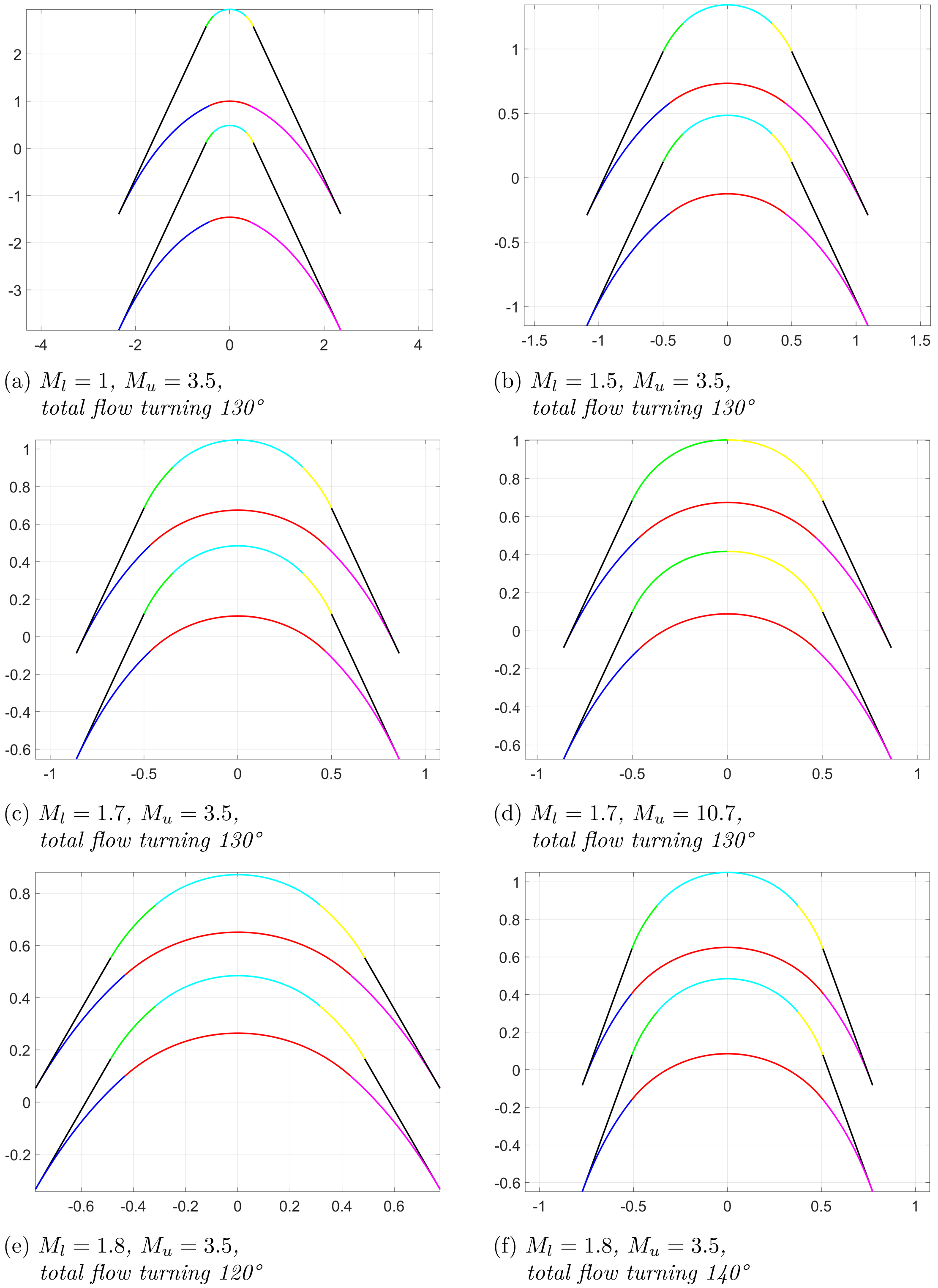
Upper outlet transition arc

$$\begin{cases} \phi = +\dfrac{1}{2} f(R^*) + constant & \text{Major vortex-expansion characteristic line} \\ \phi = \dfrac{1}{2}[f(R^*) - f(R_u^*)] \\ f(R_{k,i}^*) = 2\nu_o - \dfrac{\pi}{2}\left(\sqrt{\dfrac{\gamma+1}{\gamma-1}} - 1\right) + 2(k-1)\Delta\nu \\ \mu_{k,i} = -arcsin\left(\dfrac{1}{M_{k,i}}\right) = -arcsin\left[\sqrt{\left(\dfrac{\gamma+1}{2}\right) R_{k,i}^{*2} - \left(\dfrac{\gamma-1}{2}\right)}\right] \\ k = 1 : [(\nu_u - \nu_o)/\Delta\nu + 1] \end{cases} \tag{2.39}$$

2.4.4 Some examples and blade design limitations

The complete procedure shown before is coded in a MATLAB function, which generates the dimensionless profile given all the inputs specified in the previous section. This function is used here to understand how the different input affect the profile geometry. The common flow condition between the various examples is that the inlet and outlet Mach number are equal to $M_i = M_o = 2.5$ and the specific heat ratio of the fluid used is $\gamma = 1.4$; the main conclusions that can be drawn by observing the results (Figure 2.20) are the following:

- the lower surface Mach number (figures (a) and (b)) and the inlet flow angle (figures (e) and (f)) have a substantial effect on the profile shape, while the upper surface Mach number (figures (c) and (d)) has a really small effect;
- increasing M_u and decreasing β_i generates thinner blades;
- all the parameters affect the solidity: increasing the lower surface Mach number M_l or the specific heat ratio γ raises the blade solidity, while an increase in the upper surface Mach number M_u or the inlet flow angle β_i has the opposite effect.



(a) $M_l = 1,\ M_u = 3.5,$ *total flow turning 130°*

(b) $M_l = 1.5,\ M_u = 3.5,$ *total flow turning 130°*

(c) $M_l = 1.7,\ M_u = 3.5,$ *total flow turning 130°*

(d) $M_l = 1.7,\ M_u = 10.7,$ *total flow turning 130°*

(e) $M_l = 1.8,\ M_u = 3.5,$ *total flow turning 120°*

(f) $M_l = 1.8,\ M_u = 3.5,$ *total flow turning 140°*

Figure 2.20. Supersonic blade profiles produced by the vortex-flow method at varying upper and lower surface Mach number and total flow turning. Common inputs are inlet and outlet Mach number $M_i = M_o = 2.5$ and specific heat ratio $\gamma = 1.4$.

It is important to remember that not all designs are feasible, because on the supersonic turbine there are significant limitations given by the necessity of supersonic starting of the blade row and to avoid issues related to flow separation: the first restriction has been extensively discussed in section 2.2, so let's discuss how the second one has been treated in the code.
Employing a high M_u is desirable for starting conditions, but at the same time this produces a larger adverse pressure gradient on the upper surface that can lead to a premature separation of the flow. Experimental investigations have been carried out on simple shapes with incompressible flows at high pressure gradients and have established that flow separation may be avoided if the pressure recovery coefficient, defined as the ratio of the pressure rise to the dynamic pressure at initial point, is less than 0.5 [52]. Without repeating the whole proof (which can be found on [15]), it is possible to determine the maximum upper surface Mach number $(M_u^*)_{max}$ and the minimum lower surface Mach number $(M_l^*)_{min}$ to obtain a pressure recovery coefficient of 0.5 and prevent a premature flow separation.

$$(M_l^*)_{min} = \sqrt{\frac{\gamma+1}{\gamma-1}} \left\{ 1 - \left(1 - \frac{\gamma-1}{\gamma+1} M_i^{*^2}\right) \left[1 + \frac{1}{2} \left(\frac{\frac{\gamma}{\gamma+1} M_i^{*^2}}{1 - \frac{\gamma-1}{\gamma+1} M_i^{*^2}} \right) \right]^{\frac{\gamma-1}{\gamma}} \right\}^{\frac{1}{2}} \tag{2.40}$$

$$M_o^* = \sqrt{\frac{\gamma+1}{\gamma-1}} \left\{ 1 - \left[1 - \frac{\gamma-1}{\gamma+1} (M_u^*)^2_{max}\right] \left[1 + \frac{1}{2} \left(\frac{\frac{\gamma}{\gamma+1} (M_u^*)^2_{max}}{1 - \frac{\gamma-1}{\gamma+1} (M_u^*)^2_{max}} \right) \right]^{\frac{\gamma-1}{\gamma}} \right\}^{\frac{1}{2}} \tag{2.41}$$

In the MOC function called in the main mean-line code, a further 12% safety factor was added to these limit values ($M_u = 0.88(M_u^*)_{max}$ and $M_l = 1.12(M_l^*)_{min}$) for two essential reasons:

- it is not safe to select as working condition exactly the limit value between a separated and a non separated condition because it can lead to instability in the flow behaviour at the trailing edge;
- the experimental investigations on which the 0.5 value was extracted for the pressure recovery coefficient worked with incompressible flow at high pressure gradients; most probably the value of the coefficient for a supersonic flow is even lower than 0.5 [15].

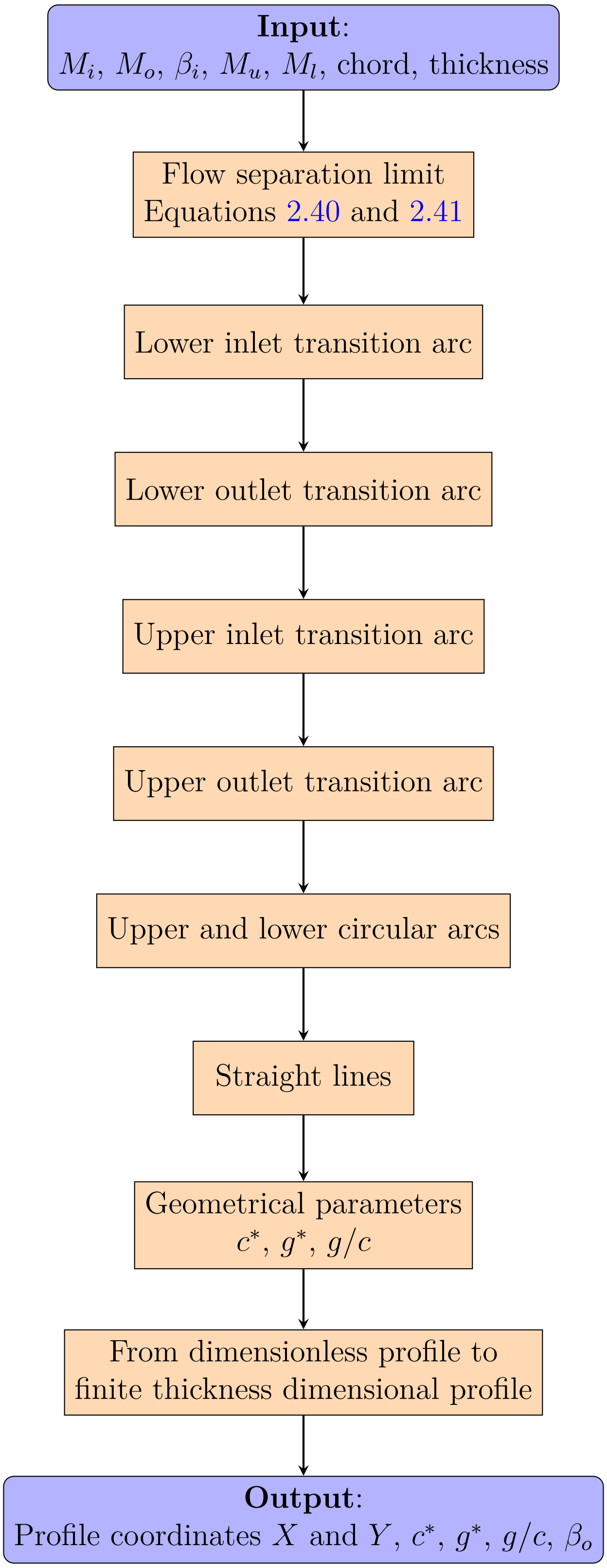

Input:
M_i, M_o, β_i, M_u, M_l, chord, thickness
Flow separation limit
Equations 2.40 and 2.41
Lower inlet transition arc
Lower outlet transition arc
Upper inlet transition arc
Upper outlet transition arc
Upper and lower circular arcs
Straight lines
Geometrical parameters
c^*, g^*, g/c
From dimensionless profile to
finite thickness dimensional profile
Output:
Profile coordinates X and Y, c^*, g^*, g/c, β_o

Figure 2.21. Flowchart to illustrate the procedure used to calculate the supersonic blade profile using the method of characteristics.

2.4.5 From dimensionless to a finite thickness dimensional blade profile

Once the dimensionless profile has been generated, two key geometrical parameters can be calculated: the dimensionless chord c^* and pitch g^* (Figure 2.18). In the mean-line code the only dimensional quantity that will be assigned regarding the profile geometry is the chord, while all the other geometrical parameters (thickness, pitch, etc.) will be obtained through dimensionless ratios carefully selected; starting from this argument, the obvious choice to assign a dimension to the profile is the chord.

$$c^* = \frac{c}{r^*} \tag{2.42}$$

From equation 2.42 the adimensionalizing quantity r^* is determined; then the dimensional infinitesimal thickness blade profile is obtained multiplying the dimensionless coordinates X^* and Y^* by r^*.

The profile generated by the method of characteristics is optimal from an aerodynamic point of view, because the oblique shock waves departing from the sharp leading edge produce lower amount of losses respect to a detached bow shock wave. On the other side a profile with zero thickness at leading and trailing edge is impossible to manufacture and has poor mechanical and thermodynamic performance working with a high velocity and high temperature flow: due to the concentration of stresses or heat the leading edge would either break down or melt.

Hence the first and easier step is to shift the lower and upper surface to a distance equal to the thickness selected for the leading and trailing edge of the blade; the choice of the leading edge thickness is critical because, as seen in section 2.3, most of the losses in a supersonic blade row comes from the detached bow-shock wave at the front. Once an appropriate value of the thickness is selected and the blade surfaces have been moved, it is required to realise a closed blade surface for the new blade.

In the developed MOC function, the circle has been selected to close the profile because it is a really simple curve, but a possible future development could be a study by CFD on different leading edge shapes (like ellipses or even fancier shapes) to evaluate a possible reduction in losses. The constraints that the circle must fulfil are the following:

- passage through the first point on the upper surface;
- passage through the first point on the lower surface;
- tangent in the first point to the upper surface;
- tangent in the first point to the lower surface.

The circle is completely defined by three parameters (Equation 2.43), so it can respect only three of the four constraints reported above: the passage through both points and the tangency to the upper surface in the first point. Even if the fourth conditions is not respected, the profile is closed adequately and does not have significant issues because the slope of the upper and lower surface at leading and trailing edge is exactly the same (Figure 2.22). Luckily this problem is even minor for our working conditions, because at the inlet the flow is completely axial: the circumference will be tangent to both surfaces automatically, because the leading edge slope is zero.

$$x^2 + y^2 + ax + by + c = 0 \tag{2.43}$$

The passage through the points is easily forced by simply substituting in equation 2.43 the coordinates of the two points. On the other side the tangency constraint is more difficult to express, because the circumference equation is implicit; Dini's theorem is then applied to calculate the derivative of an implicit function.

$$\left.\frac{dy}{dx}\right|_{(x_1,y_1)} = -\frac{\left.\dfrac{\partial C(x,y)}{\partial x}\right|_{(x_1,y_1)}}{\left.\dfrac{\partial C(x,y)}{\partial y}\right|_{(x_1,y_1)}} = -\frac{2x_1 + a}{2y_1 + b} \tag{2.44}$$

Finally the three parameters defining the circumference (a, b and c) are found by solving the system of equations 2.45, in which all the imposed conditions are resumed and where m_1 is the slope of the upper surface in the first point.

$$\begin{cases} x_1^2 + y_1^2 + ax_1 + by_1 + c = 0 \\ x_2^2 + y_2^2 + ax_2 + by_2 + c = 0 \\ -\dfrac{2x_1 + a}{2y_1 + b} = m_1 \end{cases} \tag{2.45}$$

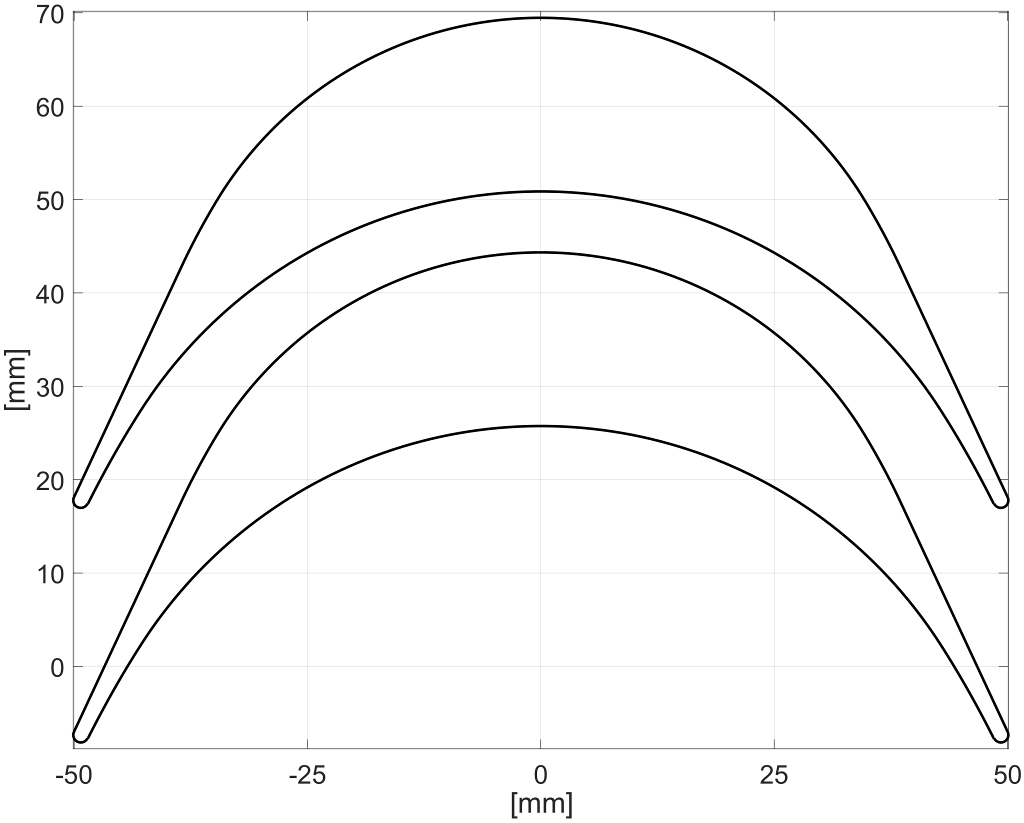


Figure 2.22. Finite thickness dimensional blade profile.

2.5 Mean-line code: description and logic explanation

The mean-line code was not built directly with all its features, but following a software development approach the code started from the basic theory of turbomachinery and then, one by one, all the pieces of theory explained before (Kantrowitz limit for a started blade row, evaluation and prediction of losses, MOC to generate the supersonic profile shape, etc.) have been added and debugged to obtain the final version of the code. Actually there is not one, but two variant of the final version of the code:

- **variant A**: in this case a constant blade height from inlet to outlet was forced in the stator. This constraint was helpful both during the validation phase of the code and also for the CFD-based optimization: in this way the profile behaviour could be more realistically tested with a blade-to-blade approach on the mean-line profile, while a complete 3D-CFD simulation would require a much heavier computational cost. The main drawback of variant A is that the Kantrowitz limit for a started supersonic blade row cannot be extended through a varying channel height, which affects the flow deviation reducing the amount of work extracted.
- **variant B**: another loop is added in the iterative procedure to design the stator; this loop varies the channel height in order to obtain a constant axial velocity along the stator. Because the stator inlet flow angle is 0°, the final blade height at outlet is higher than at inlet. This condition is particularly favourable because it extends the started blade limit and allows a greater amount of work extracted. On the other side the only way to properly test a blade of this type is through a complete 3D-CFD simulation.

The two variants of the mean-line code have in common almost all the inputs, so it feels appropriate to report here the values selected to design the supersonic turbine stage and how these numbers were chosen.

$\mathbf{M_1 = 2}$ Obviously the inlet Mach number is supersonic and the value chosen here is a typical value found in literature regarding this subject ([16], [41]); it is important to mention here that the supersonic turbine stage is not directly positioned after the rotating detonation combustor, but between the two components there is a diverging nozzle which is vital to obtain a better operation of the supersonic turbine. The flow leaving the combustion chamber is strongly non-uniform in terms of pressure, temperature and direction; in case such an unsteady flow feeds directly the turbine, the latter would have a significant reduction in the efficiency. The diverging nozzle will reduce the static temperature, pressure and increase the Mach number; all these variations are beneficial because they alleviate the thermo-mechanical loading, while favouring a started condition on the turbine. On the other side, this device will also increase the total pressure losses.

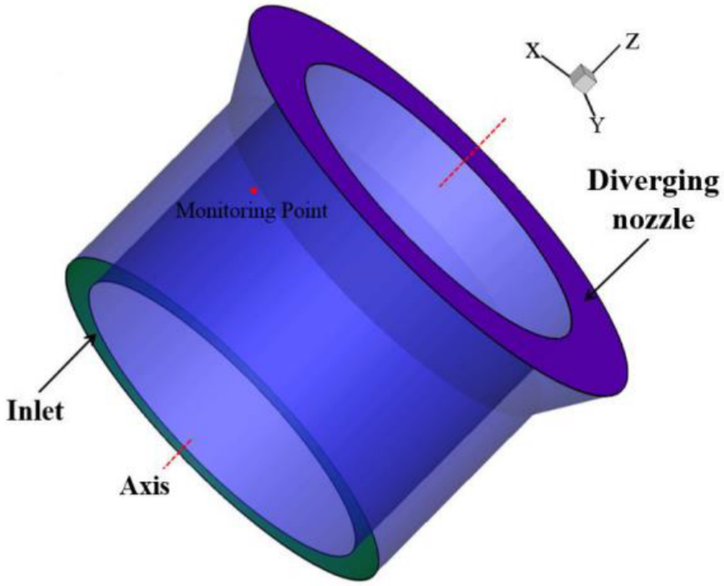


Figure 2.23. Assembly of a rotating detonation combustor with a diverging nozzle downstream (Figure from [16]).

$\boldsymbol{\alpha_1 = 0°}$**,** $\boldsymbol{\alpha_4 = 0°}$**,** $\boldsymbol{\chi = 0.5}$ The flow exiting the nozzle positioned downstream the rotating detonating combustor has an improved uniformity with oscillations considerably reduced; the inlet flow angle was assumed to be equal to zero, because this is exactly the value that resulted from a mass-flow average. Then $\alpha_4 = 0°$ and $\chi = 0.5$ were chosen because the aim was to design a repeated stage turbine with a good efficiency: a completely axial flow at outlet reduces the losses in the subsequent channel, while a reaction stage usually guarantees a higher efficiency and lower deviation angles facilitating the starting of the blade row. Obviously all these inputs cannot be secured with a direct approach, but a multi-cycle iterative approach will be employed, which will be illustrated in detail in the following subsections.

$\boldsymbol{\left.\frac{\mathbf{g}}{\mathbf{th}}\right|_{\mathbf{stator}} = 25}$**,** $\boldsymbol{\left.\frac{\mathbf{g}}{\mathbf{th}}\right|_{\mathbf{rotor}} = 25}$**,** $\boldsymbol{\mathbf{c}_{\mathbf{stator}} = 200\ \mathbf{mm}}$**,** $\boldsymbol{\mathbf{c}_{\mathbf{rotor}} = 200\ \mathbf{mm}}$ Most of the losses in a supersonic blade row are coming from the detached bow shock wave, which is strongly dependent on the dimensionless ratio $\frac{g}{th}$ (Figure 2.7). As discussed before, the higher the ratio, the lower will be the associated losses, but this value should not be pushed too much because the function is non-linear and the benefits reduce at larger $\frac{g}{th}$. A compromise between these two considerations is reached at $\frac{g}{th} = 25$, which is obtained reducing the thickness or increasing the pitch; it is vital to notice that the thickness cannot be reduced below certain limit values for structural and thermodynamic reasons, while increasing the pitch is not a viable option because it is calculated from the solidity obtained through an optimization process. Then the value of the chord is assigned taking account of two aspects:

- the final thickness should be large enough to satisfy the mechanical requirements and to allow enough space to embed advanced system of cooling essential in this harsh environment;

- a longer chord is favourable with a variable height design approach to reduce the blade slope from inlet to outlet.

$\mathbf{T_1 = 1300\ K}$ The static temperature at stator inlet was found in literature ([16], [41]) and the value chosen is a typical temperature of the flow leaving the nozzle integrated with the rotating detonating combustor. Considering the supersonic inlet Mach number $M_1 = 2$, the total temperature at inlet is quite high $T_{1t} = 2340\ K$ and this would be the temperature of the surface of the blade at the leading edge (stagnation point) if no cooling device is employed. To sustain these temperatures it is necessary to apply the most advanced technology available for both the manufacturing of the blade and the design of the cooling system: a possible solution is a single crystal blade made in Nickel alloy, which maintains optimal mechanical performances at high temperatures, cooled with multiple passage internal cooling and extensive film cooling. Another really interesting solution could be the use of ceramic material, which has by far the best mechanical performance at high temperature but suffers from fragility; research is right now ongoing to overcome exactly this issue due to obvious reliability requirements.

$\mathbf{C_{ft_{stator}} = 0.228,\ C_{ft_{rotor}} = 0.245}$ C_{ft} is Zweifel's tangential force coefficient and it is very important because it gives an immediate information of the loading on each blade. In literature there are several studies available on optimal values of C_{ft} for subsonic axial turbines, but almost nothing regarding supersonic turbines. When the blade profile was designed using the method of characteristics, one of the output of the MOC was also the solidity which is directly linked with Zweifel's coefficient; the C_{ft} given by the MOC is quite low and this happens because the main objective of the MOC is to obtain a completely uniform flow at exit using a high number of blades; on the other side, the main objective in our design is a blade row that satisfies uniformity requirements on the flow, while minimizing the losses. The original MOC designed a profile with infinitesimal thickness at leading and trailing edge, so when a finite thickness blade was realized the blade channel and behaviour were modified and hence a new optimal value of the tangential force coefficient must be found. To do this first of all a range of C_{ft} is selected in which the flow is sufficiently uniform and then in this interval the optimal value that minimizes the losses is found through an original completely automated optimization process, which make use of Python, MATLAB and various packages of Ansys. Here a very brief description has been given of the optimization process from which these values come from, but a whole chapter will be dedicated in which all the procedure will be described in detail.

$\mathbf{\dot{m} = 100\ \frac{kg}{s},\ p_{1t} = 15\ bar,\ r_{1i} = 0.345\ m}$ The mass flow rate, the total pressure at inlet and the internal radius at inlet are the last three inputs required by the mean-line code to define completely the turbine stage and the numbers chosen are typical values that are used in gas turbines.

2.5.1 Variant A

Stator

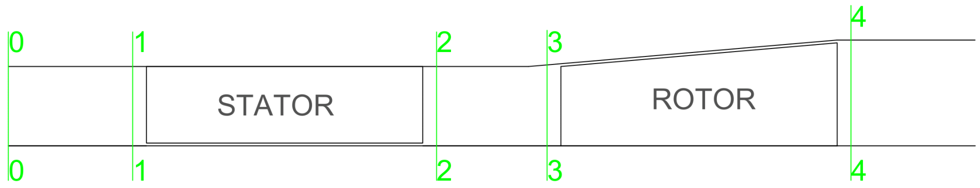


Figure 2.24. Variant A turbine stage layout.

The first step is to completely define the stator inlet by calculating the main physical quantities and the remaining geometrical dimensions.

$$T_{1t} = T_1\left(1 + \frac{\gamma - 1}{2}M_1^2\right) = 2340\ K \tag{2.46}$$

$$p_1 = p_{1t}\left(\frac{T_1}{T_{1t}}\right)^{\frac{\gamma}{\gamma-1}} = 1.917\ bar \tag{2.47}$$

$$V_1 = M_1\sqrt{\gamma R T_1} = 1445\ m/s \tag{2.48}$$

To calculate the channel area the conservation of mass is applied.

$$\begin{cases} \dot{m} = \rho_1 A_1 V_1 \\ r_{1e} = \sqrt{\dfrac{A}{\pi} + r_{1i}^2} = 402.3\ mm \end{cases} \tag{2.49}$$

In variant A the blade height is equal at inlet and outlet and it is calculated quite easily as the difference between inlet and outlet radius.

$$b_1 = b_2 = r_{1e} - r_{1i} = 57.3\ mm \tag{2.50}$$

Now an iterative approach is necessary in order to calculate the physical quantities at stator outlet, which starts with an initial guess of the outlet flow angle α_2. All the next passages are done within a while cycle, which ends when the error calculated through equation 2.51 is lower than the tolerance $toll = 10^{-9}$. Notice also that all the results that are presented in the following equations are the final values obtained when the while cycle has reached convergence.

$$error = \frac{|\alpha_2(k+1) - \alpha_2(k)|}{\alpha_2(k)} \tag{2.51}$$

Once an initial guess of the outlet angle is available, it is possible to determine the outlet area through equation 2.52.

$$\frac{A_2}{A_1} = \frac{g_2 cos(\alpha_2) b_2}{g_1 cos(\alpha_1) b_1} = \frac{\frac{\pi D_{2m}}{N_{bl}} cos(\alpha_2) b_2}{\frac{\pi D_{1m}}{N_{bl}} cos(\alpha_1) b_1}$$

where the D_{1m} and D_{2m} are respectively the inlet and outlet mean diameter and N_{bl} is the number of blades in the stator.

$$A_2 = A_1 \frac{cos(\alpha_2)}{cos(\alpha_1)} \frac{b_2}{b_1} \frac{D_{2m}}{D_{1m}} \tag{2.52}$$

Then the solidity σ is calculated by reversing Zweifel's formula for the tangential force coefficient.

$$\left.\frac{g}{c}\right|_{stator} = \frac{C_{ft_{stator}}}{2cos(\alpha_2)^2 \left[-\left(\frac{v_{1_{ax}}}{v_{2_{ax}}}\right) tan(\alpha_1) + tan(\alpha_2) \right]} = 0.357 \tag{2.53}$$

In formula 2.53 the axial velocity at outlet $V_{2_{ax}}$ is unknown at the first step, so as initial guess it is assumed that $V_{2_{ax}} = V_{1_{ax}}$; at the end of each iteration this velocity will be updated and when the while cycle reaches convergence, the $V_{2_{ax}}$ will also be correct.

$$\begin{cases} g_{stator} = \left.\frac{g}{c}\right|_{stator} c_{stator} \\ \\ N_{bl_{stator}} = \dfrac{2\pi r_{2m}}{g_{stator}} \approx 33 \end{cases} \tag{2.54}$$

The number of stator blades has been rounded to the closest higher integer number for obvious reasons, so the pitch g_{stator} must be recalculated taking account of the actual value of the solidity with an integer number of blades.

$$g_{stator} = \frac{2\pi r_{2m}}{N_{bl_{stator}}} = 71.1\ mm \tag{2.55}$$

Now using the dimensionless ratio $\left.\frac{g}{th}\right|_{stator}$ given as input, it is possible to determine the thickness at leading and trailing edge of the stator.

$$th_{stator} = \frac{g_{stator}}{\left.\frac{g}{th}\right|_{stator}} = 2.8\ mm \tag{2.56}$$

The next step in the while cycle is the estimation of the losses, but in order to predict the reflected shock on the blade surface the profile shape is necessary; the profile shape is generated applying the method of characteristics and the whole procedure is performed in a dedicated MATLAB function (the process is described in detail in section 2.4). Figure 2.25 reports all the inputs required by the MOC function and all of them are available with the exception of M_2; during the while cycle M_2 will be calculated, but that value takes account of both a variable blade height (in case of the rotor) and the aerodynamic losses in the supersonic blade row. The Goldman version of the method of characteristics is two-dimensional and isentropic, which means that the M_2 must be obtained in the same conditions. Calculating this M_2 is not so trivial and a separate MATLAB function has been developed, which does exactly the same operations of the while cycle, but with no losses and with a constant blade height. In conclusion this function for the isentropic constant blade height stator is run before the while cycle and gives the final input for the MOC function.

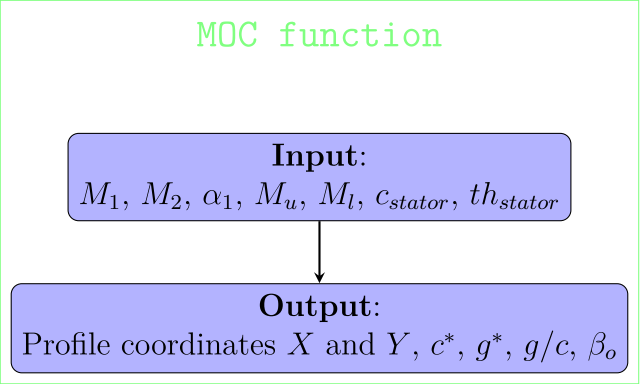

MOC function
Input:
M_1, M_2, α_1, M_u, M_l, c_{stator}, th_{stator}
Output:
Profile coordinates X and Y, c^*, g^*, g/c, β_o

Figure 2.25. Flowchart to illustrate the input and output of the MATLAB function that generates the stator profile using the method of characteristics.

Once the stator profile has been generated, all the data required to estimate the losses produced by the supersonic blade row are available; the entropy increase induced by the detached bow-shock wave and its first reflection are determined through the dedicated MATLAB function, whose inputs are shown in figure 2.26 (for more details see section 2.3.1).

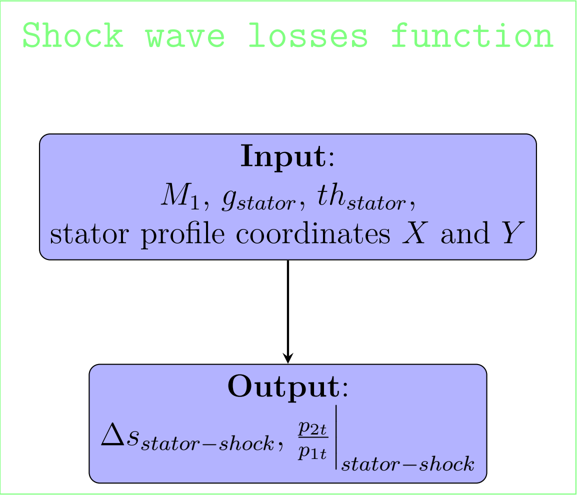

Shock wave losses function
Input:
M_1, g_{stator}, th_{stator},
stator profile coordinates X and Y
Output:
$\Delta s_{stator-shock}$, $\left.\frac{p_{2t}}{p_{1t}}\right|_{stator-shock}$

Figure 2.26. Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the detached bow-shock wave and its first reflections in the stator.

The stator profile generated by the MOC function is shown in figure 2.27 and then the detached bow-shock wave and its first reflection as an oblique shock wave are plotted by the shock wave losses function.

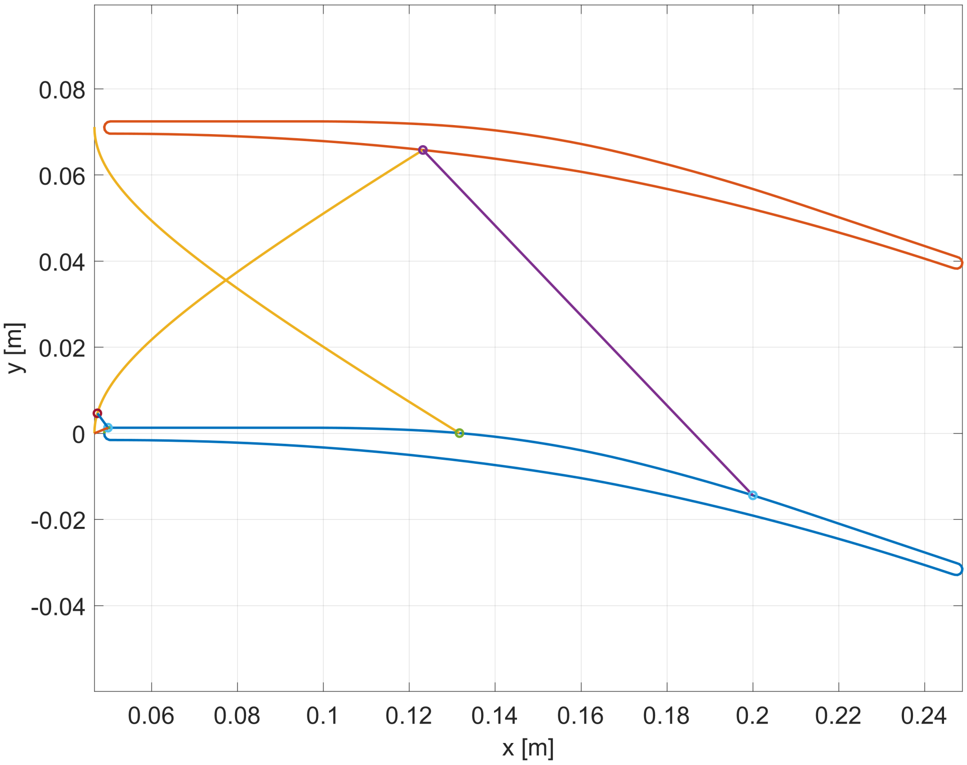


Figure 2.27. Detached bow-shock wave and its first reflection for the stator in variant A .

$$\Delta s_{shock-stator} = 34.05 \ \frac{J}{kg \ K}$$

Then the next step is to estimate the boundary and mixing losses using again a dedicated MATLAB function, whose inputs are shown in figure 2.28 (for more details see section 2.3.3); $T_{1t} = T_{2t}$ because of the conservation of the total temperature in a stator, while as a guess for the first iteration M_2 is assumed equal to M_1.

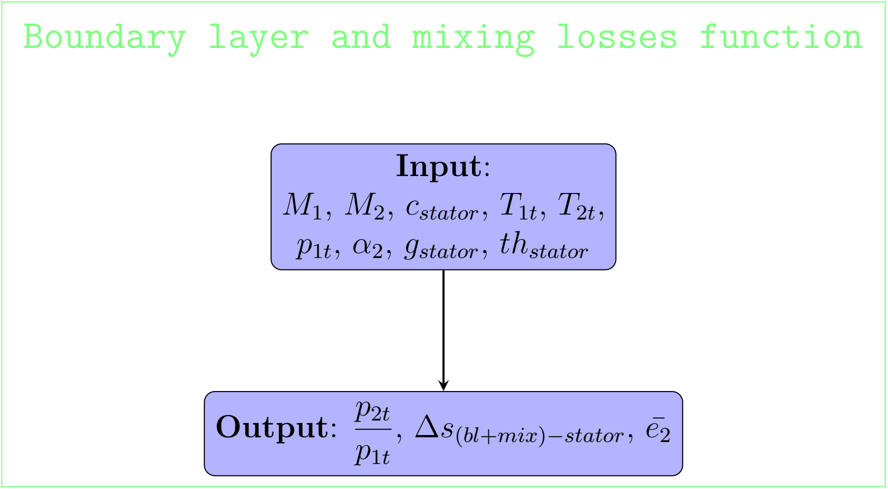

Boundary layer and mixing losses function
Input:
M_1, M_2, c_{stator}, T_{1t}, T_{2t},
p_{1t}, α_2, g_{stator}, th_{stator}
Output: $\frac{p_{2t}}{p_{1t}}$, $\Delta s_{(bl+mix)-stator}$, $\bar{e_2}$

Figure 2.28. Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the boundary layer and mixing in the stator.

$$\Delta s_{(bl+mix)-stator} = 10.12 \ \frac{J}{kg \ K}$$

$$\Delta s_{total-stator} = \Delta s_{shock-stator} + \Delta s_{(bl+mix)-stator} = 44.17 \ \frac{J}{kg \ K} \tag{2.57}$$

From the results obtained in equation 2.57, it is again confirmed that the main source of losses in a supersonic blade row are the shock waves, which in the case of the stator represents 77.1 % of the total losses; this is the prime reason why a significant effort was put to estimate accurately these losses and then parametric studies and optimization process were performed to reduce them as much as possible.
The total entropy jump in the stator is also used to determine the total pressure at stator outlet p_{2t} (Equation 2.58).

$$\Delta s = C_p ln\left(\frac{T_{2t}}{T_{1t}}\right) - Rln\left(\frac{p_{2t}}{p_{1t}}\right) = -Rln\left(\frac{p_{2t}}{p_{1t}}\right)$$

$$p_{2t} = p_{1t} e^{-\dfrac{\Delta s_{total-stator}}{R}} = 12.86 \ bar \tag{2.58}$$

where C_p is the specific heat at constant pressure and R is the universal gas constant. The losses produced in the stator have been expressed till now in terms of entropy, but in order to have an easier value to interpret let's calculate also the total pressure loss coefficient Y.

$$Y = \frac{p_{1t} - p_{2t}}{p_{2t} - p_2} = 20.68 \ \% \tag{2.59}$$

Then at least another quantity is necessary to completely define the outlet of the stator: this data will be the outlet Mach number M_2 obtained through the conservation of mass.

$$\begin{aligned}
\dot{m} &= \rho_2 A_2 V_2 \\
&= \frac{p_2}{RT_2} A_2 V_2 \\
&= \frac{p_{2t}}{\left(1 + \frac{\gamma-1}{2} M_2^2\right)^{\frac{\gamma}{\gamma-1}}} \ \frac{1}{R} \ \frac{1 + \frac{\gamma-1}{2} M_2^2}{T_{2t}} \ A_2 \ \sqrt{\gamma R T_2} M_2 \\
&= \frac{p_{2t}}{\left(1 + \frac{\gamma-1}{2} M_2^2\right)^{\frac{\gamma}{\gamma-1}}} \ \frac{1}{R} \ \frac{1 + \frac{\gamma-1}{2} M_2^2}{T_{2t}} \ A_2 \ \sqrt{\gamma R \frac{T_{2t}}{1 + \frac{\gamma-1}{2} M_2^2}} M_2 \\
&= \frac{p_{2t}}{\sqrt{T_{2t}}} \ \frac{1}{\sqrt{R}} \ A_2 \ \sqrt{\gamma} \left(1 + \frac{\gamma - 1}{2} M_2^2\right)^{-\frac{\gamma+1}{2(\gamma-1)}} M_2
\end{aligned}$$

$$
\begin{aligned}
\dot{m} &= \frac{p_{2t}}{\sqrt{T_{2t}}} \frac{1}{\sqrt{\frac{\gamma R}{\gamma-1}}} A_2 \frac{\gamma}{\sqrt{\gamma-1}} \left(1+\frac{\gamma-1}{2}M_2^2\right)^{-\frac{\gamma+1}{2(\gamma-1)}} M_2 \\
&= \frac{p_{2t}}{\sqrt{Cp\ T_{2t}}} A_2 \frac{\gamma}{\sqrt{\gamma-1}} \left(1+\frac{\gamma-1}{2}M_2^2\right)^{-\frac{\gamma+1}{2(\gamma-1)}} M_2
\end{aligned}
$$

$$
\frac{\dot{m}\sqrt{Cp\ T_{2t}}}{A_2 p_{2t}} = \frac{\gamma}{\sqrt{\gamma-1}} \left(1+\frac{\gamma-1}{2}M_2^2\right)^{-\frac{\gamma+1}{2(\gamma-1)}} M_2 \tag{2.60}
$$

Solving equation 2.60 explicitly is not trivial, so a numerical approach is employed: all the terms are brought on one side of the equation and then the stator outlet Mach number that satisfies the conservation of mass is found by using MATLAB in-built function "fzero".

$$
\Rightarrow M_2 = 1.72
$$

$$
T_2 = \frac{T_{2t}}{1+\frac{\gamma-1}{2}M_2^2} = 1468\ K \tag{2.61}
$$

$$
\begin{cases}
V_2 = \sqrt{\gamma R T_2} M_2 = 1324\ m/s \\
V_{2-ax} = V_2 cos(\alpha_2) = 1245\ m/s \\
V_{2-tg} = V_2 sin(\alpha_2) = 450\ m/s
\end{cases} \tag{2.62}
$$

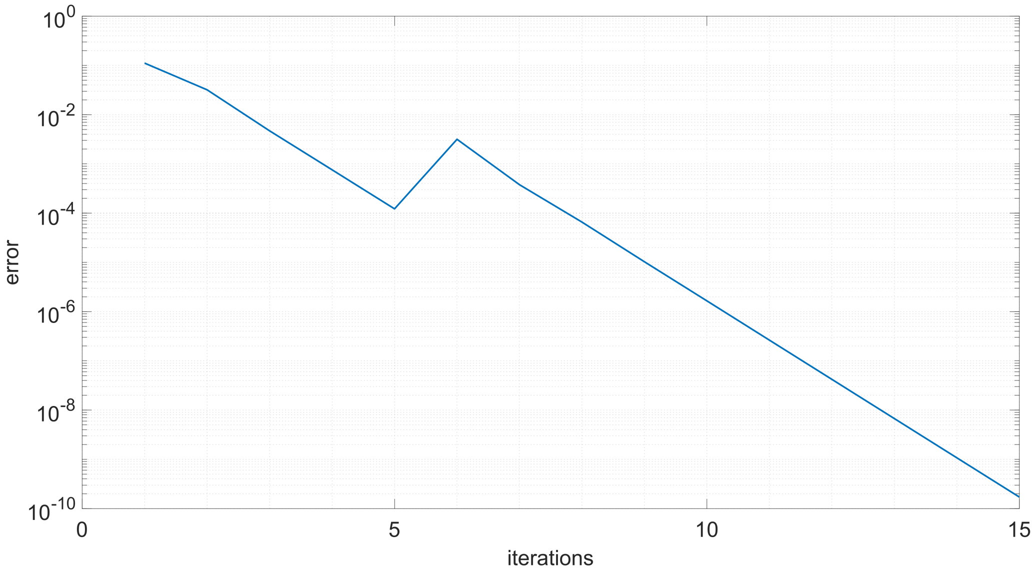


Figure 2.29. While cycle convergence for stator outlet properties.

The final step of the while cycle is to recalculate the stator outlet flow angle for the next iteration, which is determined taking account of the reaction degree χ selected in the input (Equation 2.63). Then the error is calculated using equation 2.51 and the while cycle iterates till the results converge and the tolerance requirement is satisfied (Figure 2.29).

$$\alpha_2(k+1) = atan\left(\frac{2U_2(1-\chi)}{V_{2-ax}}\right) = 19.87° \tag{2.63}$$

where $U_2 = 450\ m/s$ is the peripheral velocity chosen for variant A. Once the stator outlet quantities have been determined, the mean-line code performs two very important verifications; first of all it verifies if the Kantrowitz limit for a started blade row is satisfied or not (Equation 2.64), because, as explained in section 2.2, it is vital that the starting condition is met to avoid designing a very poor performing machine.

$$\begin{cases} \dfrac{A_2}{A_1} > \left(\dfrac{\frac{\gamma+1}{2}}{1+\frac{\gamma-1}{2}M_{normal}^2}\right)^{\frac{\gamma+1}{2(\gamma-1)}} & \checkmark \\ M_{normal}^2 = \dfrac{(\gamma-1)M_1^2+2}{2\gamma M_1^2-(\gamma-1)} & \text{Mach number behind normal shock} \end{cases} \tag{2.64}$$

After that the code verifies that the mass flow rate at stator inlet and outlet is exactly the same.

$$\dot{m}_1 = \dot{m}_2 = 100\ kg/s \qquad \checkmark \tag{2.65}$$

All the calculations that have been performed in the previous pages refer to the mean-line, therefore to complete the design of the stator it is necessary to determine how the various quantities vary along the blade height. In the classic theory of turbomachinery several strategies have been developed and used (free vortex, constant-swirl vortex, exponential vortex, constant nozzle-angle, etc.); for our purposes the most suitable solution is the **constant nozzle-angle**, because it yields a constant flow angle from hub to shroud at the nozzle exit, which is a favourable condition for the design of the rotor. The choice of the constant nozzle-angle is a right fit for this stator because the radial extension of the blade compared to the mean radius is limited ($\frac{b_1}{r_{1m}} = 15.3\ \%$) and at the tip there are no issues related to flow reversal. Finally the system of equations 2.66 derived from the constant nozzle angle strategy gives the axial and tangential velocity distribution at outlet along the blade height.

$$\begin{cases} V_{2_{ax}}(r) = V_{2m_{ax}}\left(\dfrac{r_{2m}}{r}\right)^{sin^2(\alpha_2)} \\ V_{2_{tg}}(r) = V_{2m_{tg}}\left(\dfrac{r_{2m}}{r}\right)^{sin^2(\alpha_2)} \end{cases} \tag{2.66}$$

where the subscript "m" is used for all the quantities on the mean-line.

After determining the absolute velocity V along the blade height, the mean-line code calculates also the relative velocity W(r) (Equation 2.67).

$$\begin{cases} U(r) = \omega r \\ W_{2_{ax}}(r) = V_{2_{ax}}(r) \\ W_{2_{tg}}(r) = V_{2_{tg}}(r) - U(r) \end{cases} \tag{2.67}$$

where $\omega = \dfrac{U_{2m}}{r_{2m}} = 11500\ RPM$ is the angular velocity calculated from the known peripheral velocity on the mean-line.

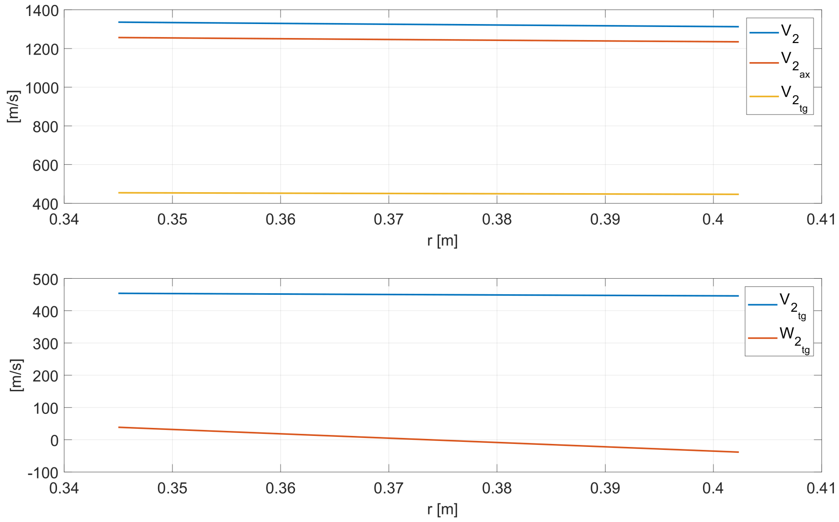


Figure 2.30. Absolute and relative velocity distribution at stator outlet obtained using constant nozzle-angle solution (variant A).

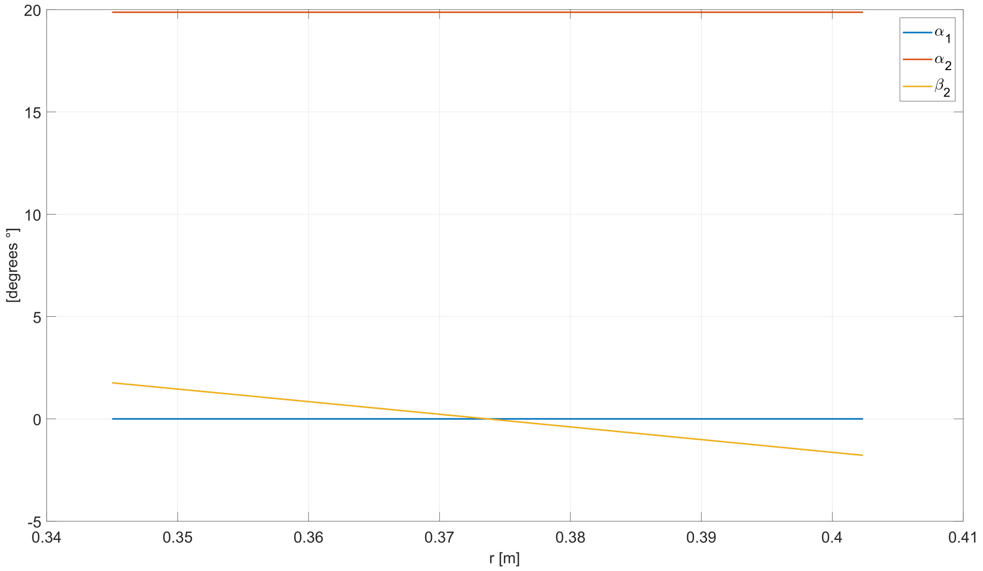


Figure 2.31. Absolute and relative angle distribution at stator outlet obtained using constant nozzle-angle solution (variant A).

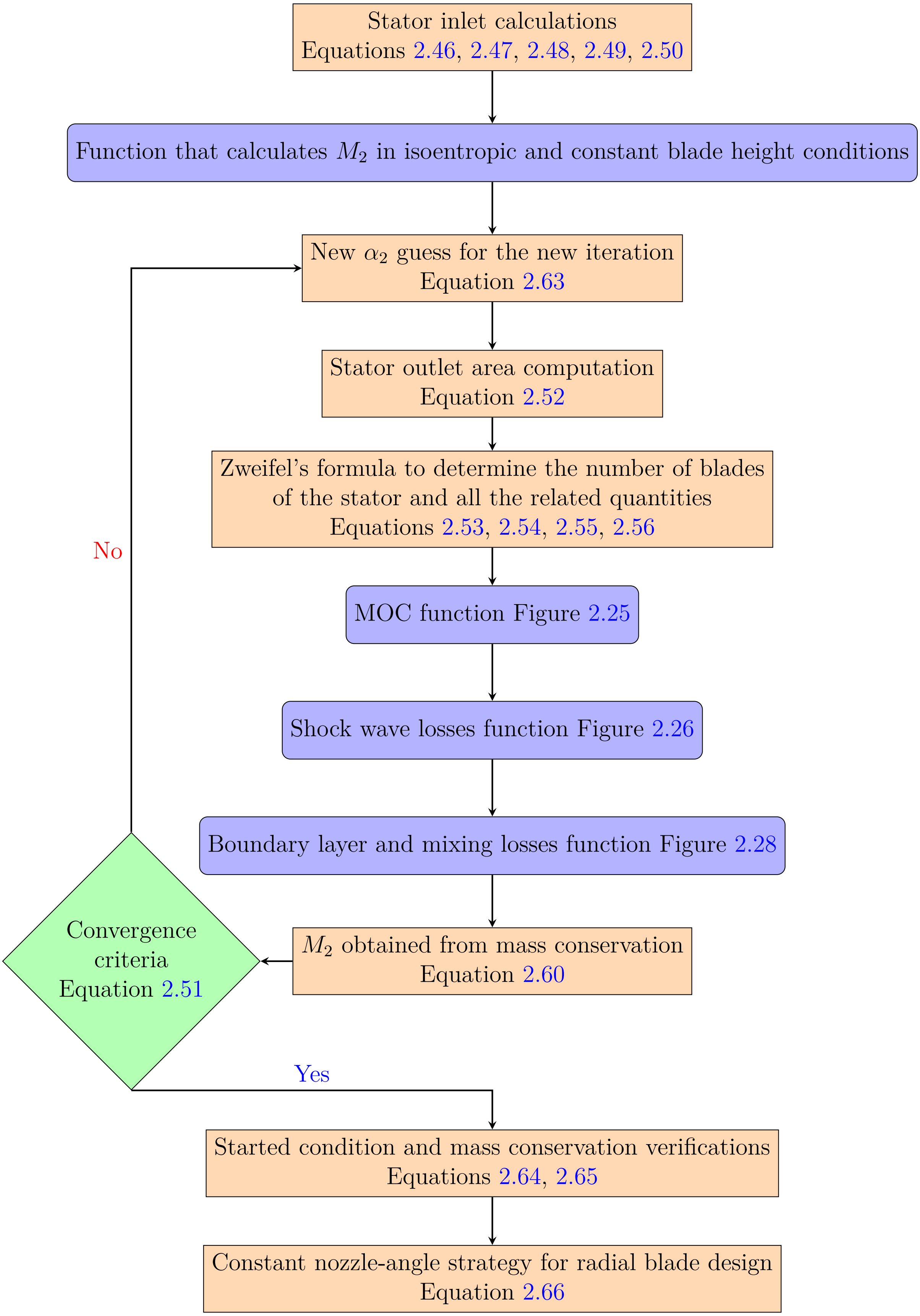

Stator inlet calculations
Equations 2.46, 2.47, 2.48, 2.49, 2.50
Function that calculates M_2 in isoentropic and constant blade height conditions
New α_2 guess for the new iteration
Equation 2.63
Stator outlet area computation
Equation 2.52
Zweifel's formula to determine the number of blades of the stator and all the related quantities
Equations 2.53, 2.54, 2.55, 2.56
MOC function Figure 2.25
Shock wave losses function Figure 2.26
Boundary layer and mixing losses function Figure 2.28
M_2 obtained from mass conservation
Equation 2.60
Convergence criteria
Equation 2.51
No
Yes
Started condition and mass conservation verifications
Equations 2.64, 2.65
Constant nozzle-angle strategy for radial blade design
Equation 2.66

Figure 2.32. Flowchart to illustrate the iterative process used to design the stator in variant A of the mean-line code.

Rotor

Between the rotor and the stator there is obviously a gap: for this version of the mean-line code it is assumed that the stator outlet is equal to the rotor inlet, but the code is structurally written in a way that allows future development in this direction to improve the accuracy in the preliminary design phase.
The procedure developed to design the rotor is quite similar to the one employed for the stator, but there are three significant differences:

- while the stator was designed with a constant blade height in variant A, the rotor design in both versions is always with a variable height solution; this is necessary first of all to facilitate the starting of the rotor blade row, feature that is more difficult to achieve with a constant height solution because of the reduction in Mach number in the stator. The variable height solution is vital in order to obtain together a constant axial velocity along the rotor, a $\chi = 0.5$ reaction turbine and a completely axial flow at outlet;

- in the stator the convergence of the while cycle worked with the stator outlet angle α_2, but in case of the rotor the absolute outlet angle is $\alpha_4 = 0$ for a design choice; for this reason in the rotor the iterative cycle is based on the variation of the blade height and the convergence is reached when the axial component of the velocity at inlet and outlet is exactly the same;

- for the rotor the equations will be written using the relative quantities, which maintains structurally the form of the equations similar to the ones for the stator due to conservation of the relative quantities.

The first step in the procedure is to calculate the outlet blade height, where as a design choice it was decided that the internal radius of the blade remains constant, while only the external radius increases as shown in figure 2.24 (Equation 2.68). As already done for the stator, all the reported results are the final values obtained when the while cycle has reached convergence.

$$\begin{cases} b_4 = \left(\dfrac{b_4}{b_3}(k)\right) b_3 = 74.4\ mm \\ r_{4i} = r_{3i} = 345.0\ mm \\ r_{4e} = r_{4i} + b_4 = 419.4\ mm \end{cases} \tag{2.68}$$

As a guess for the first iteration it is assumed that $\frac{b_4}{b_3} = 1$; then the ratio is updated each iteration considering the normalized distance between the desired axial velocity and the one obtained in that iteration (Equation 2.69). Finally the convergence of the while cycle is reached when the error defined in equation 2.70 is lower than the established tolerance, in this case $toll = 10^{-9}$.

$$\left.\frac{b_4}{b_3}\right|_{k+1} = \left.\frac{b_4}{b_3}\right|_{k} \left(1 - \frac{V_{4_{ax}} - V_{3_{ax}}}{V_{3_{ax}}}\right) = 1.2975 \tag{2.69}$$

$$error = \frac{|V_{4_{ax}} - V_{3_{ax}}|}{V_{3_{ax}}} \tag{2.70}$$

When the blade height changes from inlet to outlet, it is very important to remember that also the mean-line radius and all the associated quantities change.

$$\begin{cases} r_{4m} = \dfrac{r_{4i} + r_{4e}}{2} = 382.2 \ mm \\ U_4 = \omega \ r_{4m} = 460.27 \ m/s \end{cases} \tag{2.71}$$

As already mentioned before, the flow direction at rotor outlet is completely axial to reduce the losses and then by applying the velocity triangles relations it is possible to calculate the relative flow angle at rotor outlet (Equation 2.72).

$$\begin{cases} \alpha_4 = 0° \\ \beta_4 = atan\left(-\dfrac{U_4}{W_{4_{ax}}}\right) = -20.29° \end{cases} \tag{2.72}$$

where for the first iteration it is assumed that $W_{4_{ax}} = W_{3_{ax}}$. After that the mean-line code calculates the rotor solidity σ using Zweifel's formula for the tangential force coefficient adapted for the rotor.

$$\left.\frac{g}{c}\right|_{rotor} = \frac{C_{ft_{rotor}}}{2cos(\beta_4)^2 \left[\left(\dfrac{W_{3_{ax}}}{W_{4_{ax}}}\right) tan(\beta_3) - tan(\beta_4)\right]} = 0.377 \tag{2.73}$$

Once the solidity is known, first the rotor number of blades is approximated with the closest higher integer number for obvious reasons and then the actual pitch of the rotor is calculated.

$$\begin{cases} g_{rotor} = \left.\dfrac{g}{c}\right|_{rotor} c_{rotor} \\ N_{bl_{rotor}} = \dfrac{2\pi r_{3m}}{g_{rotor}} \approx 32 \end{cases} \tag{2.74}$$

Fortunately the number of blades of the stator and rotor are not exactly the same, because otherwise the turbine stage would have suffered from significant stator-rotor coupling issues.

$$g_{rotor} = \frac{2\pi r_{3m}}{N_{bl_{rotor}}} = 73.4 \ mm \tag{2.75}$$

Now using the the dimensionless ratio $\left.\frac{g}{th}\right|_{rotor}$ given as input, it is possible to determine the thickness at leading and trailing edge of the rotor.

$$th_{rotor} = \frac{g_{rotor}}{\left.\dfrac{g}{th}\right|_{rotor}} = 2.9 \ mm \tag{2.76}$$

The next step in the while cycle is the estimation of the losses, but the profile shape is necessary to predict the reflected shock on the blade surface; the profile shape is generated applying the method of characteristics and the whole procedure is performed in a dedicated MATLAB function (the process is described in detail in section 2.4).

All the inputs required by the MOC function are reported in figure 2.33 and all of them are available with the exception of the relative Mach number M_{3W}. As already explained for the stator, the M_{3W} that is given as input to the MOC function is the one obtained solving the rotor in isentropic and constant blade height conditions, which are exactly the basic hypothesis of the Goldman's version on the method of characteristics; calculating this M_{3W} is not so trivial and a separate MATLAB function is developed, which does exactly the same operations of the while cycle, but with no losses and with a constant blade height. Then this function is executed before the while cycle and prepares the last input required by the MOC function for the rotor.

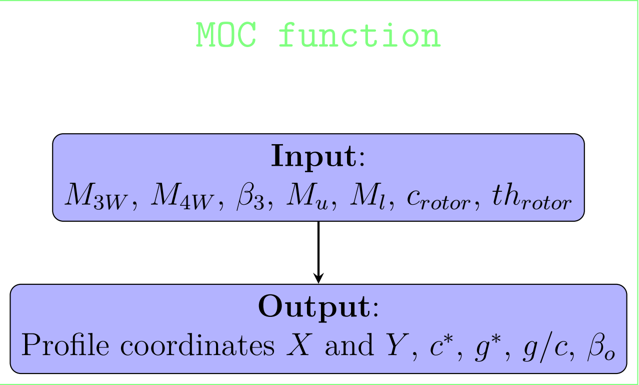

MOC function
Input:
M_{3W}, M_{4W}, β_3, M_u, M_l, c_{rotor}, th_{rotor}
Output:
Profile coordinates X and Y, c^*, g^*, g/c, β_o

Figure 2.33. Flowchart to illustrate the input and output of the MATLAB function that generates the rotor profile using the method of characteristics.

Now that the rotor profile has been generated, all the data required to estimate the losses produced by the supersonic blade row are available; the entropy increase induced by the detached bow-shock wave and its first reflection are determined through the dedicated MATLAB function, whose inputs are shown in figure 2.34, where the subscript "tr" means total-relative (for more details see section 2.3.1).

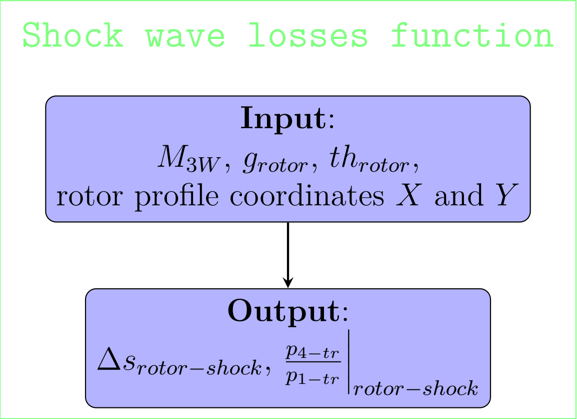

Shock wave losses function
Input:
M_{3W}, g_{rotor}, th_{rotor},
rotor profile coordinates X and Y
Output:
$\Delta s_{rotor-shock}$, $\left.\frac{p_{4-tr}}{p_{1-tr}}\right|_{rotor-shock}$

Figure 2.34. Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the detached bow-shock wave and its first reflections in the rotor.

In figure 2.35 is shown the rotor profile generated by the MOC function and then the detached bow-shock wave and its first reflection as an oblique shock wave are plotted by the shock wave losses function.

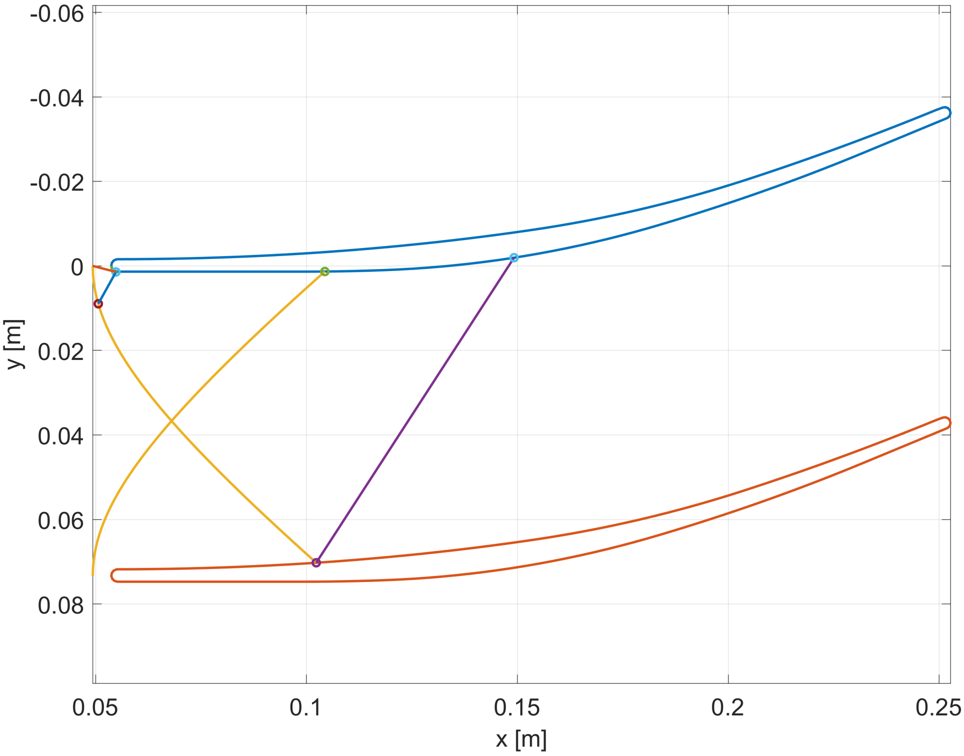


Figure 2.35. Detached bow-shock wave and its first reflection for the rotor in variant A .

$$\Delta s_{shock-rotor} = 17.68\ \frac{J}{kg\ K}$$

Please note a significant reduction in shock losses between stator and rotor, because this type of losses are strongly dependent on the inlet Mach number: for the rotor $M_{3W} = 1.62$, which is lower than the value for the stator $M_1 = 2$.
Then the next step is to estimate the boundary and mixing losses using again a dedicated MATLAB function, whose inputs are shown in figure 2.36 (for more details see section 2.3.3). To calculate T_{4-tr} the conservation of the rothalpy between rotor inlet and outlet is imposed (Equation 2.78), which is fundamental to consider the variation of the mean radius.

$$T_{3-tr} = T_3 + \frac{W_3^2}{2Cp} = 2239\ K \tag{2.77}$$

$$T_{4-tr} = T_{3-tr} + \frac{U_4^2 - U_3^2}{2Cp} = 2244\ K \tag{2.78}$$

$$p_{3-tr} = p_3 \left(1 + \frac{\gamma - 1}{2} M_{3W}^2\right)^{\frac{\gamma}{\gamma-1}} = 11.02\ bar \tag{2.79}$$

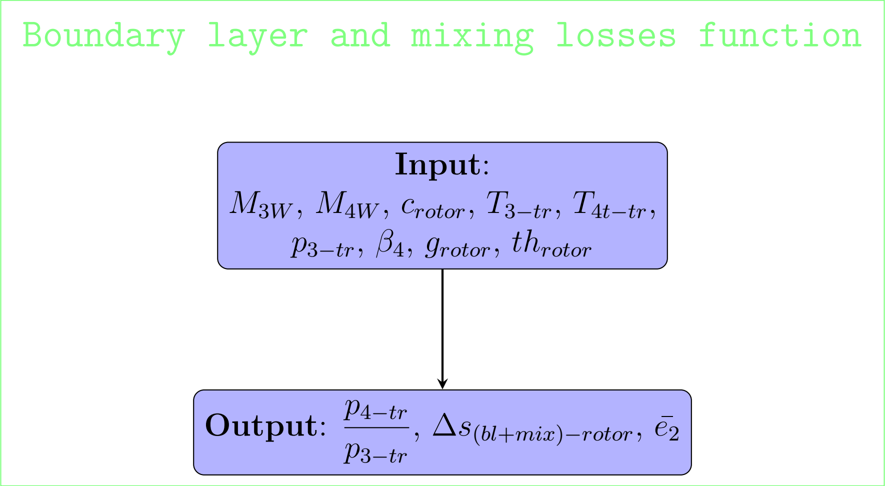

Boundary layer and mixing losses function
Input:
M_{3W}, M_{4W}, c_{rotor}, T_{3-tr}, T_{4t-tr},
p_{3-tr}, β_4, g_{rotor}, th_{rotor}
Output: $\frac{p_{4-tr}}{p_{3-tr}}$, $\Delta s_{(bl+mix)-rotor}$, $\bar{e_2}$

Figure 2.36. Flowchart to illustrate the input and output of the MATLAB function that determines the losses associated to the boundary layer and mixing in the rotor.

$$\Delta s_{(bl+mix)-rotor} = 10.41\ \frac{J}{kg\ K}$$

$$\Delta s_{total-rotor} = \Delta s_{shock-rotor} + \Delta s_{(bl+mix)-rotor} = 28.09\ \frac{J}{kg\ K} \tag{2.80}$$

From the results obtained in equation 2.80, we can notice that there is a reduction in losses in the rotor due to a weaker detached-bow shock wave, while the values of the losses associated to boundary layer and mixing are really similar. On the other side, the shock wave losses are still the main source of entropy production representing 62.9 % of the total losses in the rotor.

The total entropy jump in the rotor is used also to calculate the total relative pressure at rotor outlet p_{4-tr} (Equation 2.82). The approximation done in equation 2.81 is possible because the two relative temperature are really close and the term $ln(\frac{T_{4-tr}}{T_{3-tr}}) = 0.0021$ is negligible.

$$\Delta s = C_p ln\left(\frac{T_{4-tr}}{T_{3-tr}}\right) - Rln\left(\frac{p_{4-tr}}{p_{3-tr}}\right) \approx -Rln\left(\frac{p_{4-tr}}{p_{3-tr}}\right) \tag{2.81}$$

$$p_{4-tr} = p_{3-tr} e^{-\frac{\Delta s_{total-rotor}}{R}} = 10.00\ bar \tag{2.82}$$

Then also for the rotor the total pressure loss coefficient is calculated to obtain a value that gives a direct interpretation of the performance of the supersonic rotor blade row.

$$Y = \frac{p_{3-tr} - p_{4-tr}}{p_{4-tr} - p_4} = 12.5\ \% \tag{2.83}$$

Before applying the conservation of mass, it is necessary to calculate the outlet area considering the relative flow angles and the blade height variation.

$$\begin{cases} \dfrac{A_4}{A_3} = \dfrac{g_4 cos(\beta_4) b_4}{g_3 cos(\beta_3) b_3} = \dfrac{cos(\beta_4)}{cos(\beta_3)} \dfrac{b_4}{b_3} \dfrac{D_{4m}}{D_{3m}} = 1.24 \\ \\ A_3 = (2\pi r_{3m}) b_3 cos(\beta_3) \end{cases} \tag{2.84}$$

Now finally the rotor outlet Mach number M_{4W} is calculated by forcing the conservation of mass along the rotor. In this case the whole demonstration will not be repeated, because the procedure is similar but with relative quantities. The final formula for the mass conservation in the rotor is equation 2.85.

$$\frac{\dot{m}\sqrt{Cp\, T_{4-tr}}}{A_4 p_{4-tr}} = \frac{\gamma}{\sqrt{\gamma - 1}} \left(1 + \frac{\gamma - 1}{2} M_{4W}^2\right)^{-\frac{\gamma+1}{2(\gamma-1)}} M_{4W} \tag{2.85}$$

Equation 2.85 is solved using a numerical approach, where all the terms are brought on one side and then the rotor outlet Mach number is found by using MATLAB in-built function "fzero".

$$\Rightarrow M_{4W} = 1.79$$

After that the code calculates all the remaining quantities on the rotor mean outlet radius.

$$T_4 = \frac{T_{4-tr}}{\left(1 + \dfrac{\gamma - 1}{2} M_{4W}^2\right)} = 1367\ K \tag{2.86}$$

$$\begin{cases} W_4 = \sqrt{\gamma R T_4} M_{4W} = 1327.42\ m/s \\ W_{4_{tg}} = W_4 sin(\beta_4) = -460.27\ m/s \\ W_{4_{ax}} = W_4 cos(\beta_4) = 1245.06\ m/s \end{cases} \tag{2.87}$$

$$\begin{cases} V_{4_{tg}} = W_{4_{tg}} + U_4 = 0\ m/s \\ V_{4_{ax}} = W_{4_{ax}} = 1245.06\ m/s \\ M_4 = \dfrac{V_4}{\sqrt{\gamma R T_4}} = 1.68 \end{cases} \tag{2.88}$$

The final step of the while cycle is to calculate the error as the normalized distance between the desired axial velocity and the one obtained in that iteration (Equation 2.70). The iterative procedure ends if the error of the while cycle is lower than the established tolerance ($toll = 10^{-9}$), otherwise a new iteration starts with the new guess given by equation 2.69. The while cycle presents a really good monotonous convergence as shown in figure 2.37.

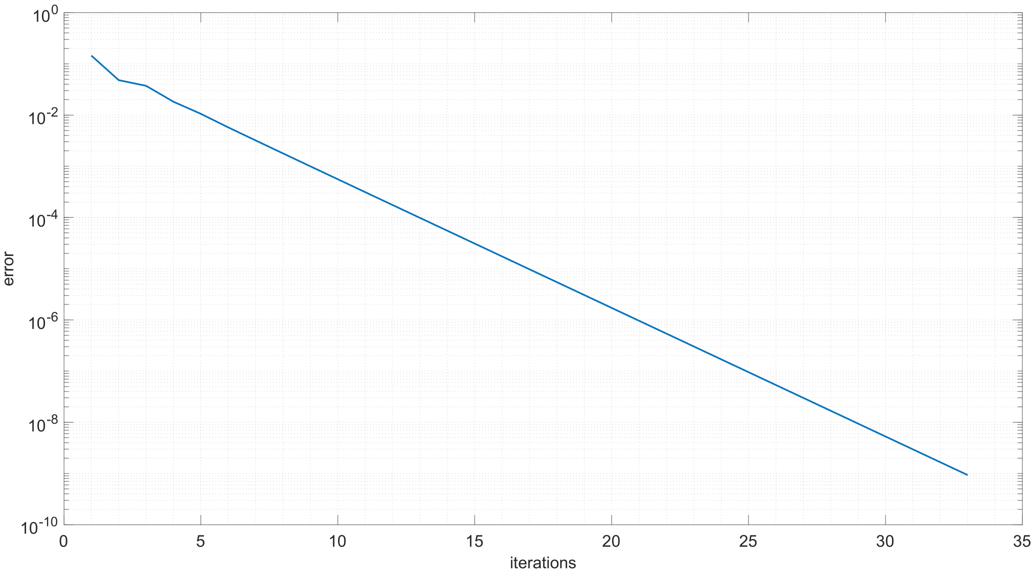


Figure 2.37. While cycle convergence for rotor outlet properties.

Subsequently the mean-line code starts the verification process of the results obtained for the rotor; first the code checks if the rotor blade row is working in a started condition or not using the Kantrowitz theory (Equation 2.89): a supersonic blade row must always work in started condition, otherwise the losses would be too high.

$$\begin{cases} \dfrac{A_4}{A_3} > \left(\dfrac{\frac{\gamma+1}{2}}{1+\frac{\gamma-1}{2}M_{normal}^2} \right)^{\frac{\gamma+1}{2(\gamma-1)}} & \checkmark \\ M_{normal}^2 = \dfrac{(\gamma-1)M_{3W}^2+2}{2\gamma M_{3W}^2-(\gamma-1)} & \text{Mach number behind normal shock} \end{cases} \tag{2.89}$$

After that the code verifies that all the mass flow rate coming from the stator goes through the rotor.

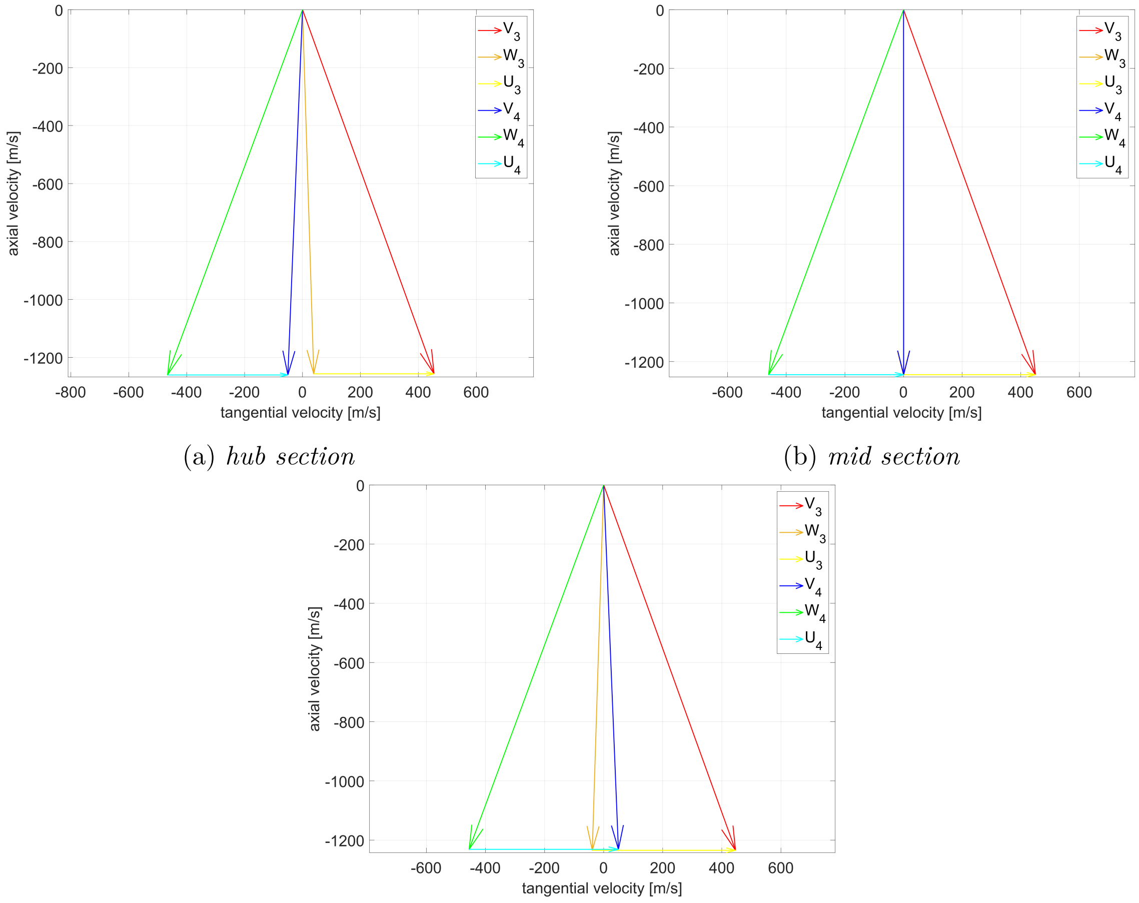
$$\dot{m}_3 = \dot{m}_4 = 100\ kg/s \qquad \checkmark \tag{2.90}$$

Till now all the calculations that were performed for the rotor regarded only the mean radius, hence to complete the design of the blade row it is necessary to establish a strategy to determine how the various quantities vary in the radial direction. The strategy employed for the rotor is the **constant angle**, because the radial extension of the blade compared to the mean radius is limited ($\frac{b_4}{r_{4m}} = 19.5\ \%$) and at the tip there are no issues related to flow reversal. Finally the system of equations 2.91 derived from the constant nozzle angle strategy gives the axial and tangential velocity distribution at outlet along the blade height, where in this case the equations are written using relative quantities. The blade clearance was not considered in this version of the code, but future development can expand on this topic, where a really interesting study could be the estimate of the clearance losses for this type of machine.

$$\begin{cases} W_{4_{ax}}(r_4) = W_{4m_{ax}}\left(\dfrac{r_{4m}}{r_4}\right)^{sin^2(\beta_4)} \\ \\ W_{4_{tg}}(r_4) = W_{4m_{tg}}\left(\dfrac{r_{4m}}{r_4}\right)^{sin^2(\beta_4)} \end{cases} \tag{2.91}$$

After determining the relative velocity $W(r_4)$ along the blade height, the mean-line code calculates also the relative velocity $V(r_4)$ (Equation 2.92).

$$\begin{cases} U_4(r_4) = \omega r_4 \\ V_{4_{ax}}(r_4) = W_{4_{ax}}(r_4) \\ V_{4_{tg}}(r_4) = W_{4_{tg}}(r_4) + U(r_4) \end{cases} \tag{2.92}$$



(a) *hub section*

(b) *mid section*

(c) *tip section*

Figure 2.38. Velocity triangles at hub, mid and tip section of the rotor (variant A).

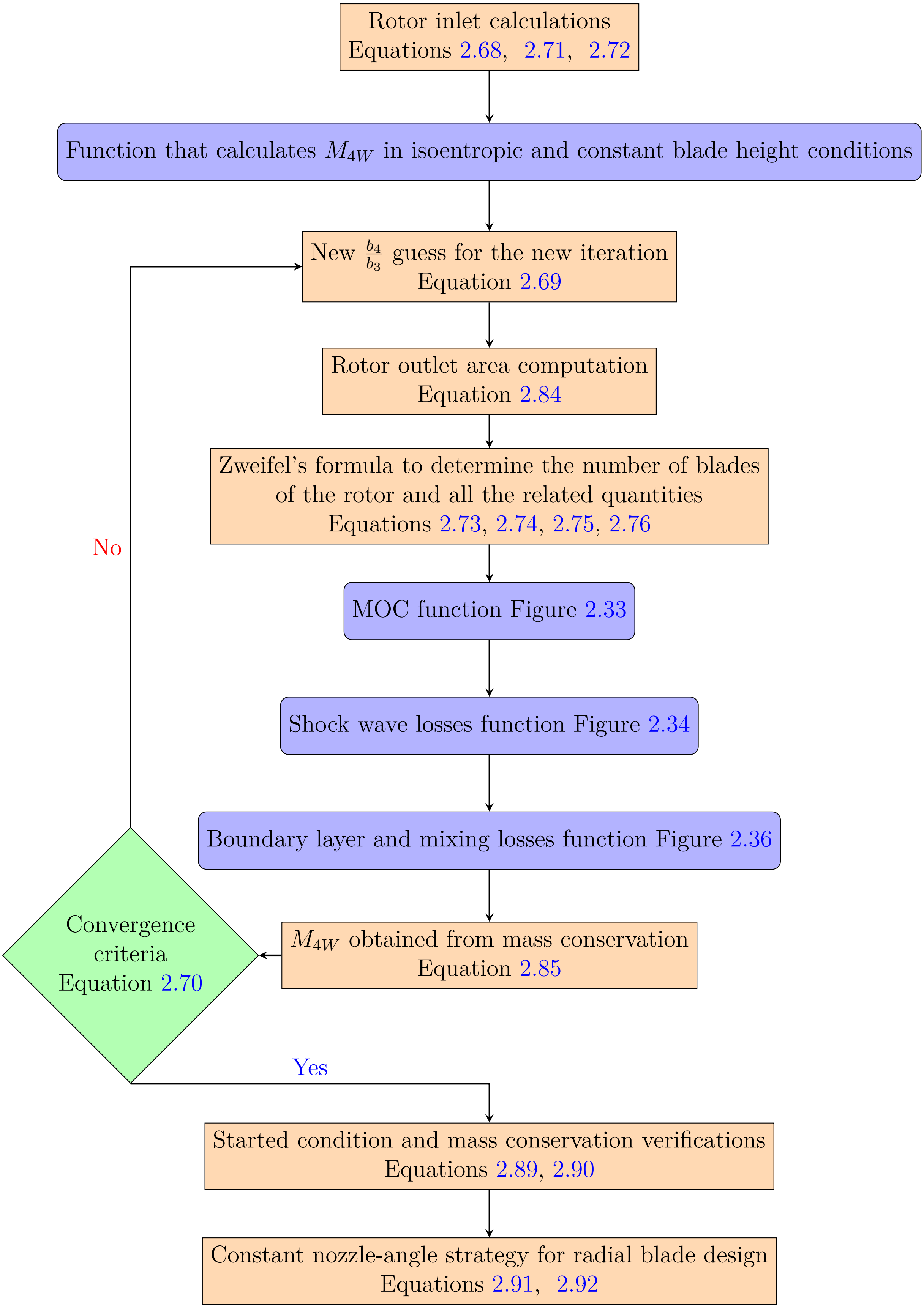

Rotor inlet calculations
Equations 2.68, 2.71, 2.72
Function that calculates M_{4W} in isoentropic and constant blade height conditions
New $\frac{b_4}{b_3}$ guess for the new iteration
Equation 2.69
Rotor outlet area computation
Equation 2.84
Zweifel's formula to determine the number of blades
of the rotor and all the related quantities
Equations 2.73, 2.74, 2.75, 2.76
MOC function Figure 2.33
Shock wave losses function Figure 2.34
Boundary layer and mixing losses function Figure 2.36
M_{4W} obtained from mass conservation
Equation 2.85
Convergence
criteria
Equation 2.70
No
Yes
Started condition and mass conservation verifications
Equations 2.89, 2.90
Constant nozzle-angle strategy for radial blade design
Equations 2.91, 2.92

Figure 2.39. Flowchart to illustrate the iterative process used to design the rotor in variant A of the mean-line code.

Finally with these last few passages the mean-line code completes the advanced procedure developed to design a supersonic turbine stage with its stator and rotor. Now that the machine has been defined, in its final step the code evaluates the Euler work (Equation 2.93), the power output (Equation 2.94) and the turbine total to total efficiency (Equation 2.95).

$$L_{eul} = U_3 V_{3_{tg}} - U_4 V_{4_{tg}} = 202.50 \ \frac{kJ}{kg} \tag{2.93}$$

$$P_{eul} = \dot{m} L_{eul} = 20.25 \ MW \tag{2.94}$$

The total to total efficiency, or isentropic efficiency, is defined for a turbine as the ratio between the actual work extracted by the machine and the ideal work that would be obtained in isentropic conditions.

$$\eta_{t,t} = \frac{\Delta h_t}{\Delta h_{t,is}} = \frac{L_{eul}}{Cp(T_{1t} - T_{4t,is})} = \frac{L_{eul}}{CpT_{1t}\left[1 - \left(\frac{p_{4t}}{p_{1t}}\right)^{\frac{\gamma-1}{\gamma}}\right]} = 56.9 \ \% \tag{2.95}$$

The supersonic turbine generates a large amount of work and power even if the flow angles are limited to satisfy the Kantrowitz limit for a started blade row: this is possible thanks to a high velocity flow that produces a large tangential component even if the flow angles are small. Considering only this aspect, using higher Mach numbers allows an important increase in work extraction because the Kantrowitz limit is less restricting and the velocities are bigger.

As a first impression, by observing the value obtained for the total to total efficiency, it seems that the designed turbine has a bad performance. It is very important to understand that a supersonic turbine will always be less efficient than a subsonic one because the complex shock waves generate huge amount of losses in the turbine passage: this point has been thoroughly discussed both from a theoretical point of view and then proved with the mean-line code results (such results are considered very reliable as shown in section 2.6). To do a fair comparison, the efficiency obtained above should be compared with machines that are working in the same supersonic environment: Paniagua et al. [4] revealed up to 80 % of total pressure loss when conventional subsonic turbine operate continuously at an inlet Mach number of 3.5; compared to those values, the supersonic turbine designed by the mean-line code is a major improvement with 20.7 % of total pressure losses in the stator and 12.5 % in the rotor.

At this point is rather reasonable to ask why it is so important to put so much effort in designing a supersonic turbine in a really complicated environment, if it is already known that it is less efficient than a classic solution with a subsonic flow? The supersonic turbine is designed to work downstream a rotating detonating combustor, in which the combustion process propagates with a velocity of km/s and produces a significant pressure increase; this particular thermodynamic cycle (Fickett-Jacobs) can theoretically increase the efficiency by 15 % and develop a much higher impulse on small engines. Hence even if the supersonic turbine is less efficient than a conventional one, the overall engine has a higher performance and efficiency,

lower emission of pollutants and CO_2, more compactness and lower mass (For more details see chapter 1).
On the other side it is vital to increase the efficiency of the turbine as much as possible and this is why one of the main focus of the thesis was to find or develop optimization processes dedicated to these type of machines:

- the optimal value of Zweifel's tangential force coefficient that minimizes the losses is found through an original completely automated optimization process, which make use of Python, MATLAB and various packages of Ansys;
- the profile shape obtained with the method of characteristics will be optimized through an evolutionary shape-optimization code developed by the laboratory of fluid machines at Politecnico di Milano, based on a geometrical parametrization technique based on B-Splines, a high-fidelity and experimentally validated CFD solver and a surrogate-based evolutionary algorithm [53].

2.5.2 Variant B

The variant B of the mean-line code wants to design a repeated stage turbine, which means that the absolute velocity at rotor outlet is equal to the absolute velocity at stator inlet. This type of turbine is used in the intermediate stages of a multi-stage machine, because the same optimized turbine is used in series to extract work with flow velocities that remain constant at inlet and outlet.
The variant B of the code starts with what has already been done with the other version: in variant A the flow velocity is completely axial at inlet and outlet of the stage, but the values are different. This happens because while in the rotor the axial velocity is constant, in the stator the axial velocity at outlet is lower than the inlet due to a constant stator blade height solution. Hence the stator blade height must be varied in order to achieve a constant axial velocity, that in the end will lead to a repeated stage configuration.
From a code development point of view, the stator design process is carried out with a double while loop:

- in the inner while loop all the operations that are performed are exactly the same of variant A;
- in the outer while loop the stator blade height is varied according to the normalized distance between the desired axial velocity and the one that is obtained in that iteration (Equation 2.96). Finally the convergence of the while cycle is reached when the error defined in equation 2.97 is lower than the established tolerance, in this case $toll = 10^{-9}$.

$$\left.\frac{b_2}{b_1}\right|_{k+1} = \left.\frac{b_2}{b_1}\right|_{k}\left(1 - \frac{V_{2_{ax}} - V_{1_{ax}}}{V_{1_{ax}}}\right) \tag{2.96}$$

$$error = \frac{|V_{2_{ax}} - V_{1_{ax}}|}{V_{1_{ax}}} \tag{2.97}$$

Most of the procedure used in variant B is exactly the same of variant A, so the description is shortened. What can be more interesting to the reader are the results obtained with this alternative version and therefore in the next pages the main quantities and figures describing the turbine will be reported.

	stator inlet	stator outlet	rotor inlet	rotor outlet
r_{1i} [mm]	345.0	345.0	345.0	345.0
r_{1e} [mm]	402.3	433.8	433.8	495.2
r_{1m} [mm]	373.7	389.4	389.4	420.1
b [mm]	57.3	88.8	88.8	150.2
V [m/s]	1445	1543	1543	1445
V_{tg} [m/s]	0	540	540	0
V_{ax} [m/s]	1445	1445	1445	1445
M	2	2.26	2.26	2.27
α [°]	0	20.48	20.48	0
W [m/s]			1445	1558
W_{tg} [m/s]			0	-583
W_{ax} [m/s]			1445	1445
M_W			2.12	2.45
β [°]			0	-21.95
T [K]	1300	1155	1155	1010
T_t [K]	2340	2340	2340	2050
T_{tr} [K]			2195	2219
p [bar]	1.92	1.06	2.06	0.51
p_t [bar]	15	12.49	12.49	6.03
p_{tr} [bar]			9.99	7.95

Table 2.1. Inlet and outlet quantities for the stator and rotor designed with the variant B of the mean-line code.

	stator	rotor	turbine
N_{bl}	36	35	
b_{out}/b_{in}	1.55	1.69	
g [mm]	68.0	69.9	
th [mm]	2.72	2.80	
Δs_{shock} [J/(kg K)]	34.07	42.17	
Δs_{bl+mix} [J/(kg K)]	18.38	23.21	
Y	21.90	27.32	
L_{eul} [kJ/kg]			291.60
P_{eul} [MW]			29.16
$\eta_{t,t}$			54.09 %

Table 2.2. Stator, rotor and turbine stage main parameters (variant B).

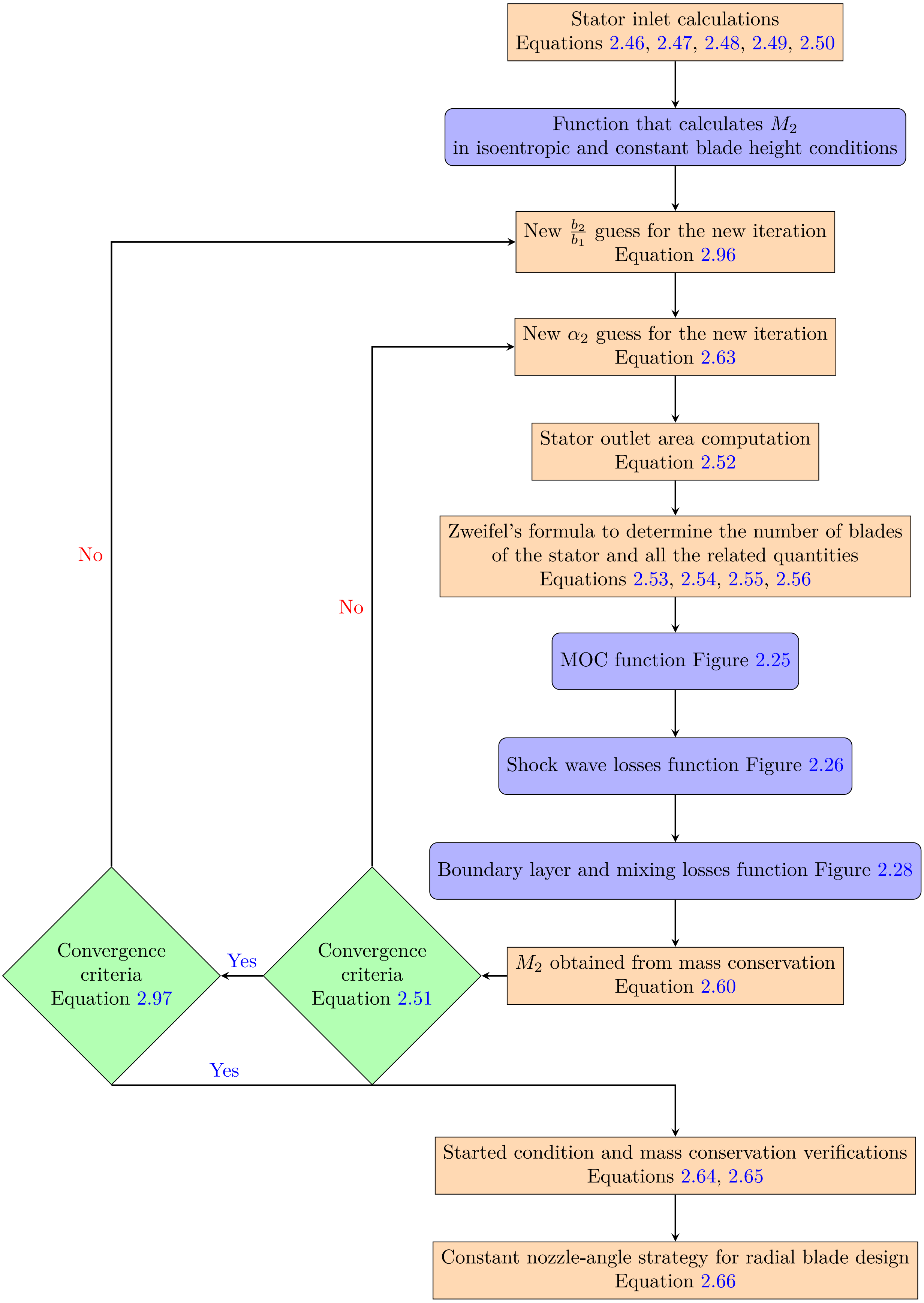

Stator inlet calculations
Equations 2.46, 2.47, 2.48, 2.49, 2.50
Function that calculates M_2
in isoentropic and constant blade height conditions
New $\frac{b_2}{b_1}$ guess for the new iteration
Equation 2.96
New α_2 guess for the new iteration
Equation 2.63
Stator outlet area computation
Equation 2.52
Zweifel's formula to determine the number of blades
of the stator and all the related quantities
Equations 2.53, 2.54, 2.55, 2.56
MOC function Figure 2.25
Shock wave losses function Figure 2.26
Boundary layer and mixing losses function Figure 2.28
M_2 obtained from mass conservation
Equation 2.60
Convergence
criteria
Equation 2.51
Yes
Convergence
criteria
Equation 2.97
No
No
Yes
Started condition and mass conservation verifications
Equations 2.64, 2.65
Constant nozzle-angle strategy for radial blade design
Equation 2.66

Figure 2.40. Flowchart to illustrate the iterative process used to design the stator in variant B of the mean-line code.

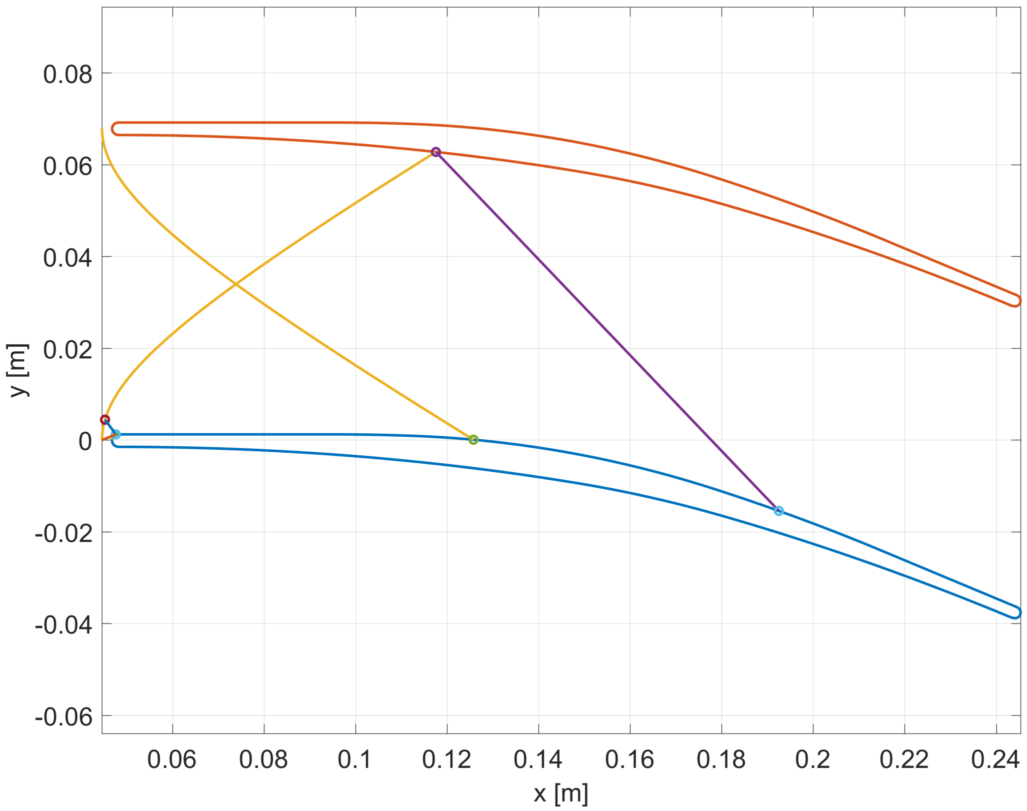


Figure 2.41. Detached bow-shock wave and its first reflection for the stator in variant B.

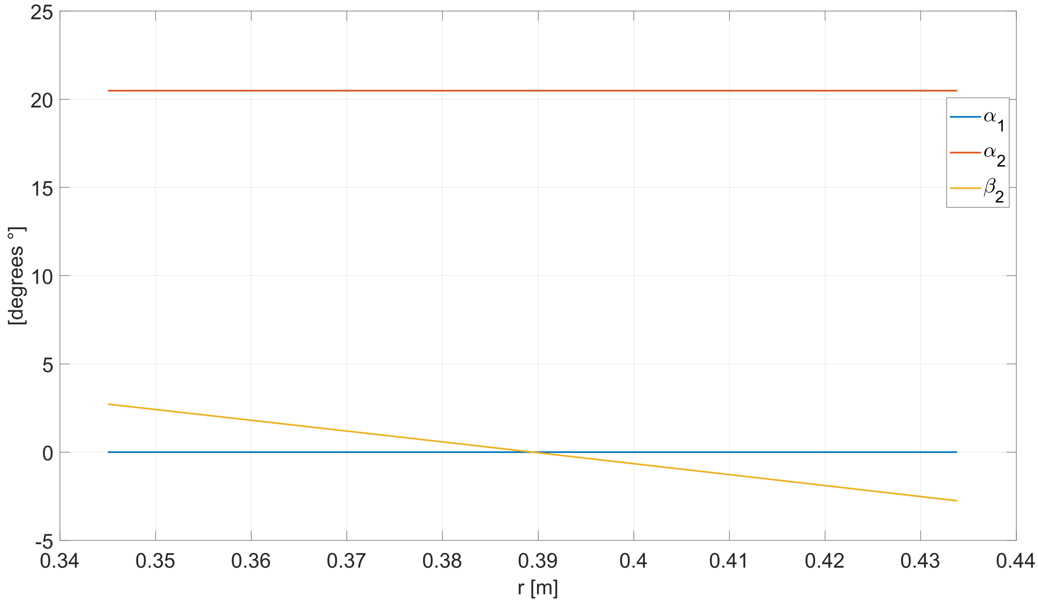


Figure 2.42. Absolute and relative angle distribution at stator outlet obtained using constant nozzle-angle solution (variant B).

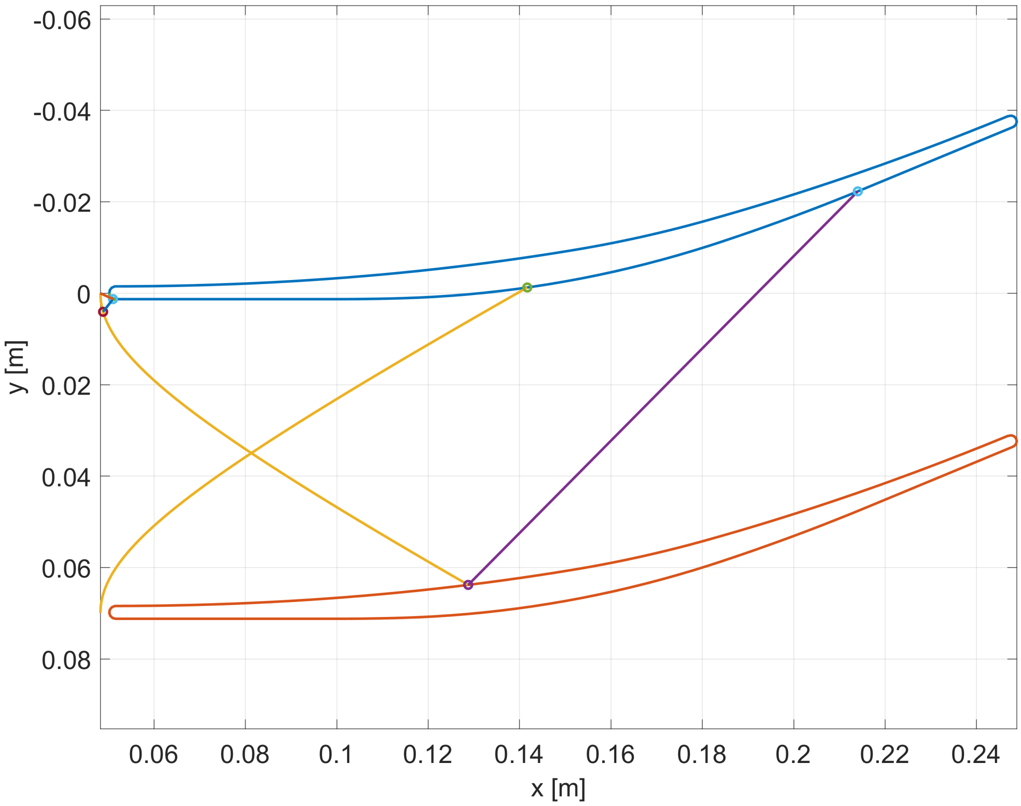


Figure 2.43. Detached bow-shock wave and its first reflection for the rotor in variant B .

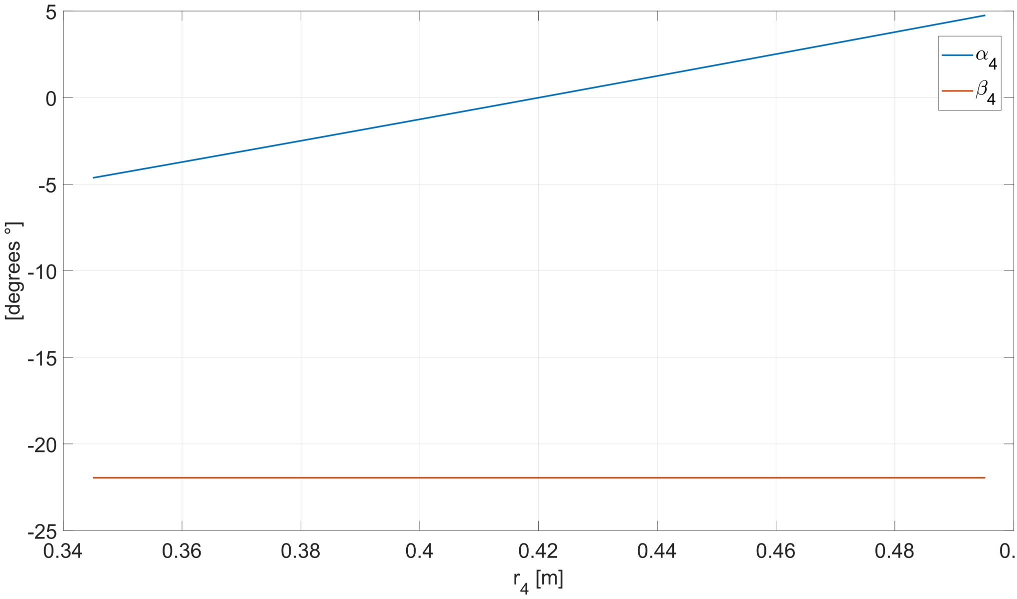


Figure 2.44. Absolute and relative angle distribution at rotor outlet obtained using constant nozzle-angle solution (variant B).

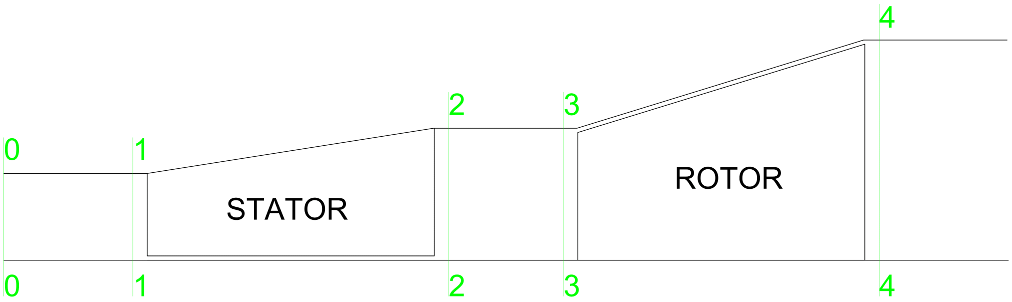
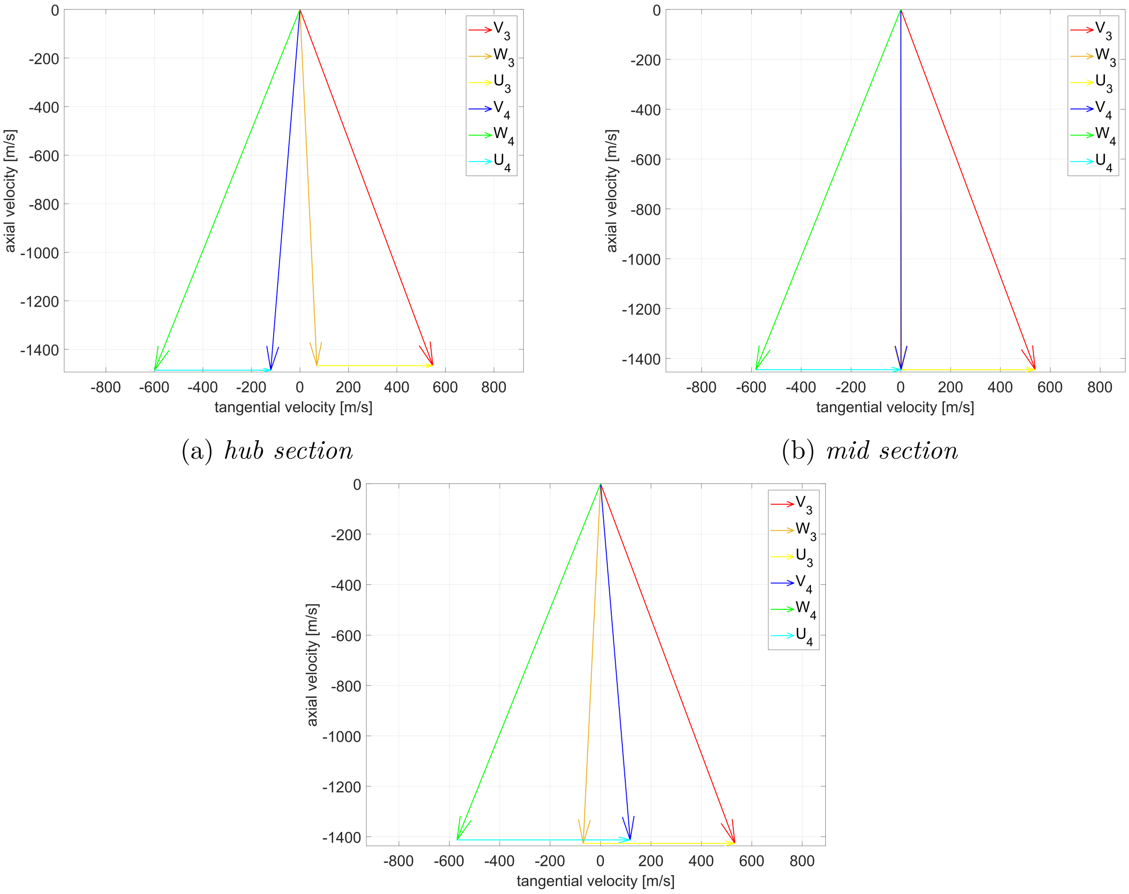


Figure 2.45. Variant B turbine stage layout.



(a) *hub section*

(b) *mid section*

(c) *tip section*

Figure 2.46. Velocity triangles at hub, mid and tip section of the rotor (variant B).

2.6 Mean-line code results validation through CFD simulations

This chapter describes in detail the mean-line code that was developed to do an advanced preliminary design of a supersonic turbine, but up to this section no discussion was dedicated to the reliability of the results generated by the code. The validation procedure is accomplished by comparing the code outputs with the results that are extracted from CFD simulations performed on the stator blade.
The stator blade of variant A was selected as a test for the validation procedure because in this way the profile behaviour could be more realistically tested with a blade-to-blade approach on the mean-line profile and without a complete 3D-CFD simulation, which requires a much heavier computational cost. The simulations were performed using Ansys CFX and the details regarding the solver, the setting of the simulation and the post-processing are thoroughly discussed in the next chapters. Obviously, this verification process does not represent a complete validation of the code, which would have required, considering the many inputs of the code, a huge number of complete 3D-CFD simulations; the time and computational cost for the whole procedure is enormous, so after the really promising results of the first validation procedure around the selected working conditions, it was decided to dedicate our efforts to the vital optimization process required to improve the efficiency of the supersonic turbine. In particular, the first parameter to be optimized is the Zweifel's tangential force coefficient, directly related to blade solidity.
Indeed, as briefly introduced while describing the inputs of the mean-line code (section 2.5), the optimal value of Zweifel's tangential force coefficient was determined through an original completely automated optimization process (for more details see chapter 5). Many simulations were performed during this optimization process and the associated results will be used here to accomplish the verification process of the mean-line code. In the following figures, the error between the mean-line code and the CFD simulation result for a generic quantity X is defined in equation 2.98.

$$error\ \% = 100\ \frac{|X_{CFD} - X_{meanLineCode}|}{X_{CFD}} \tag{2.98}$$

In figure 2.47 the trend of the entropy with a varying Zweifel's tangential force coefficient is reported: the mean-line code losses prediction is really good with a maximum error lower than 8 % and lower than 1 % in the selected working condition of the turbine. Also the trend in Fig. 2.48 for the total pressure loss coefficient is in good accordance with CFD simulations. With a lower C_{ft} (higher number of blades and smaller pitch g), the error increases because a higher number of oblique shock wave reflections are generated in the passage, while the mean-line code takes account of only the first one; at higher C_{ft} than the design point (lower number of blades), the increase in losses is explained by the small bubble of separation formed behind the reflection point on a more loaded blade surface. Moreover, more loaded blades imply the existence of undesired over-speed regions, where the BL stability becomes critical.

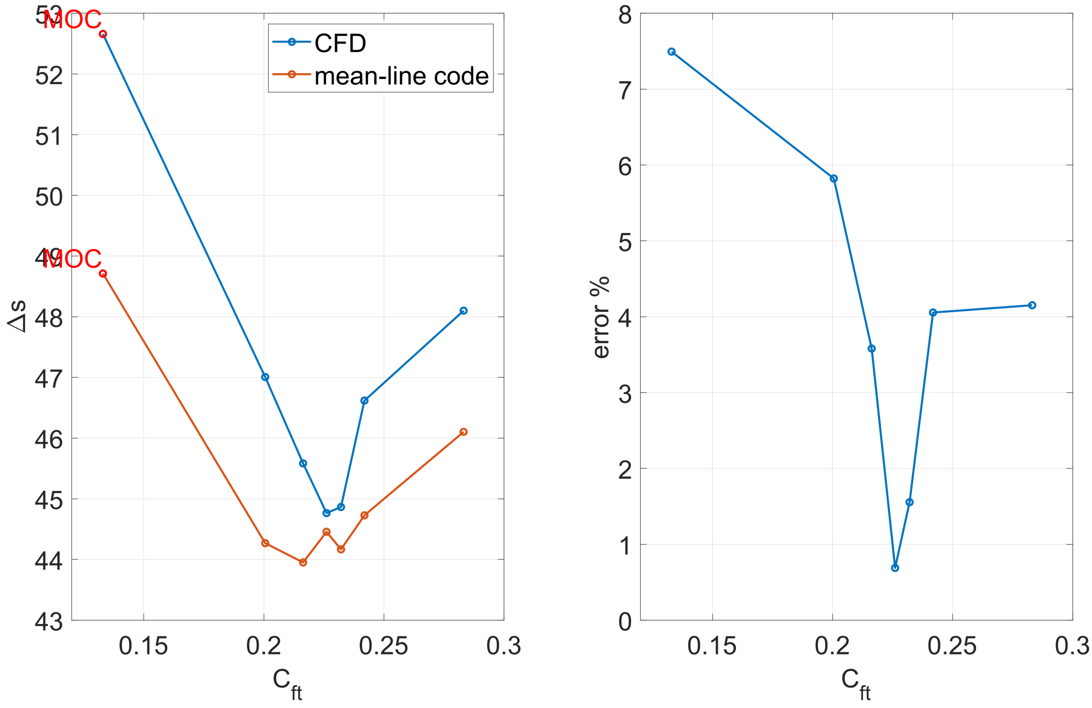


Figure 2.47. Entropy production comparison between the mean-line code and the CFD simulations.

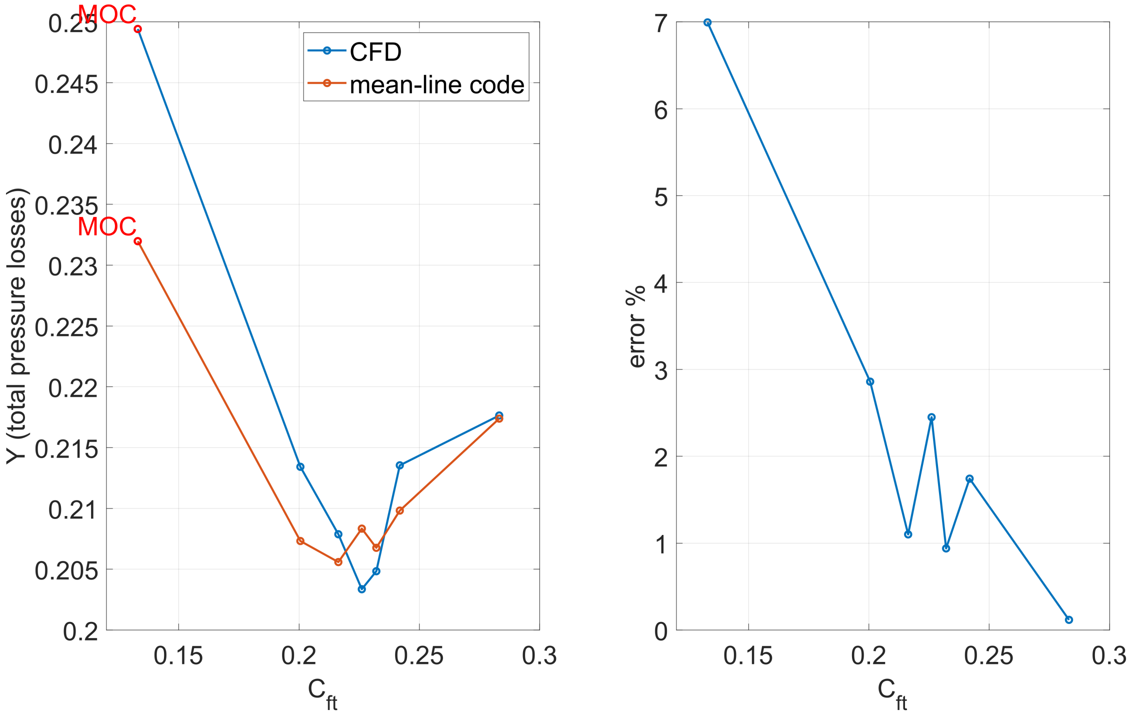


Figure 2.48. Total pressure losses comparison between the mean-line code and the CFD simulations for the stator.

The smallest value of C_{ft} is associated to the original value of the solidity given by the method of characteristics: as shown in the plot, the entropy generated in that condition is almost 18 % higher than the optimal value; this clearly justifies the original optimization process developed to find the optimal value of Zweifel's tangential force coefficient.

The optimization process developed is CFD-based because, even though the mean-line code is able to predict quite accurately the losses, it is not able to catch properly the minimum of the curve.

In figure 2.49 it is possible to observe that the estimate of the mean-line code is very satisfying in terms of total pressure with an error lower than 2%: this is really helpful because the iterative procedure employed to design the rotor starts with accurate and reliable quantities.

Finally figure 2.50 shows the standard deviation of the stator outlet flow angle measured at a chord distance from the trailing edge; when the tangential force coefficient increases (lower number of blades), the flow obviously is less uniform, while the maximum uniformity condition is achieved at the the original value of the solidity given by the MOC. The optimum value of the C_{ft} obtained searching for the minimum of losses must satisfy at the same time also the minimum requirement of flow uniformity, established as a necessary condition.

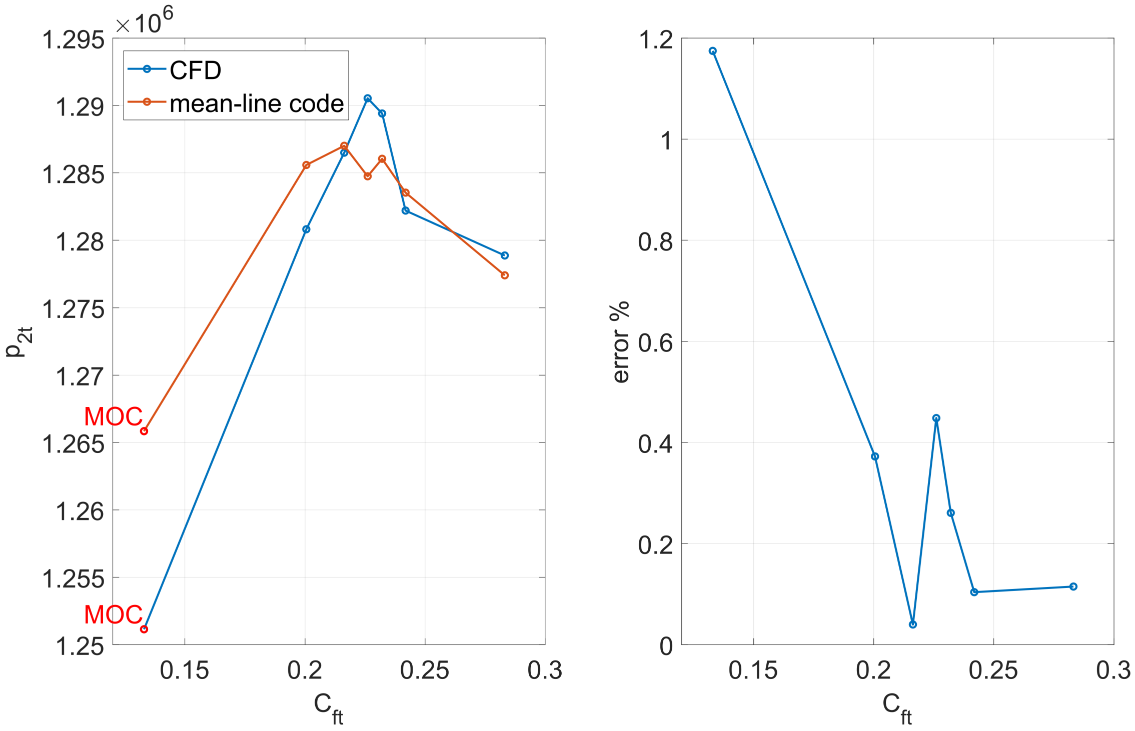


Figure 2.49. Stator outlet total pressure comparison between the mean-line code and the CFD simulations.

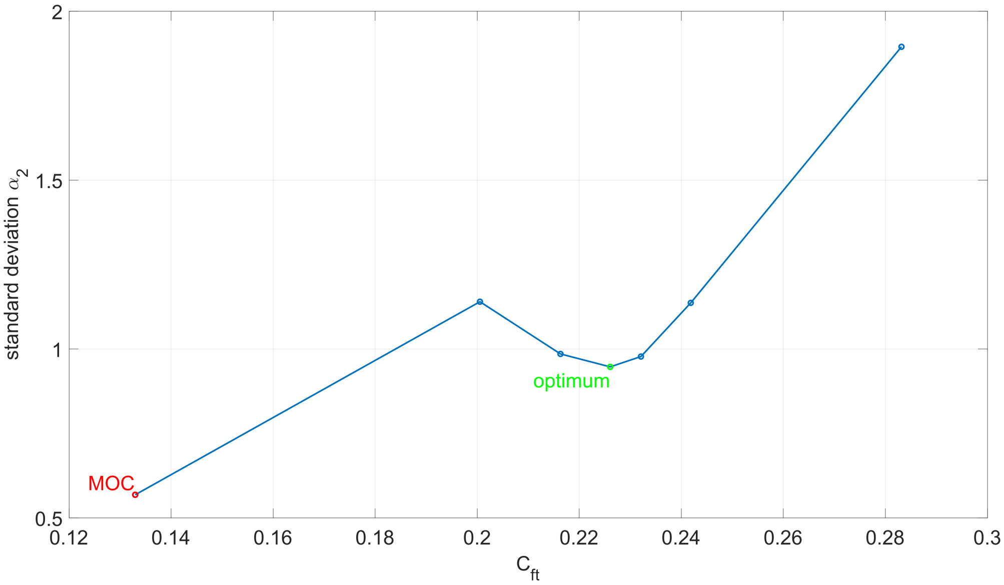


Figure 2.50. Standard deviation of the stator outlet flow angle obtained from CFD simulations and calculated at a chord distance from the trailing edge.

Chapter 3

Computation Fluid Dynamic

3.1 Navier-Stokes equations

Modern calculators allow to deal successfully with a lot of problem related to turbomachinery (see Chapter 4), although several aspects require approximate physical model and parameter calibration through experiments or statistical laws. In particular, it is possible to simulate the flow-field evolution in time scale of industrial interest in an acceptable amount of time, although with frequency still too low with respect to the turbulence one. Despite the relevant progress in CFD, it is always necessary to reduce the model complexity acting on geometrical features and on thermo-fluid dynamic assumptions. This is particularly true since recursive calculations have to be implemented for design optimization of the machine. Assuming known basic fluid dynamic and turbomachinery concepts, some considerations need to be exposed before reporting the governing equations solved by means the CFD techniques:

- Navier-Stokes equations for compressible fluids include mass conservation (3.1b), momentum conservation (3.1c) and energy conservation (3.1a) equations along with two equations of state, inherent the fluid nature itself. So, five non-linear scalar partial differential equations and two algebraic equations (sometimes the equation of state are substitute with tabular relation to deal with real gas models).

- Differential equations includes diffusive terms, linked to disturbances propagation without mass transportation (second derivatives in the equations), and convective term, associated to system transport properties (first derivatives in equations).

- Governing equations include stress terms, that are expressed in function of the strain rate.

- The equations are generically time-variant, but the turbulent effects, essentially statistical and characterized by frequencies much larger than those of industrial interest, are often introduced through turbulent viscosity and appropriate models.

Derivations of the governing differential equations are available in every text concerning classical fluid-dynamic, with various level of mathematical rigour. Depending on the hypotheses and simplifications, it is possible to face different form of these equations.

The mathematical model is based on the formulation of conservation laws, i.e. the variation of the total amount of a quantity inside a given domain is equal to the balance between the amount of that quantity entering and leaving the considered domain, plus the contribution from eventual sources generating that quantity. It is possible to adopt different modelling strategies for the fluid flow. Below, for example, are reported the *tout-court* Navier-Stokes partial differential system of equations (3.1) for Calorically Perfect (3.1e) Ideal Gas (3.1f), in differential and conservative form. Moreover, the gas is modelled as a Newtonian fluid, which means that the stress tensor (not explicitly visible here) is isotropic, linear in strain rates and it is divergence zero for fluid at rest. This assumption is common for lots of fluid in nature, but with some relevant exception such as blood, paint, shampoo and so on (none of them of interest for this thesis purpose).

$$\frac{\partial}{\partial t}(\rho\, e_{tot}) + \nabla\cdot\Big(\rho\,\mathbf{u}\; h_{tot} + \mathbf{q} + \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{u} - \mu(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}})\cdot\mathbf{u}\Big) = \rho\mathbf{g}\cdot\mathbf{u} \tag{3.1a}$$

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\,\mathbf{u}) = 0 \tag{3.1b}$$

$$\frac{\partial}{\partial t}(\rho\,\mathbf{u}) + \nabla\cdot(\rho\,\mathbf{u}\otimes\mathbf{u}) = -\nabla p + \nabla\cdot\Big\{\mu\Big(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}} - \tfrac{2}{3}(\nabla\cdot\mathbf{u})\mathbf{I}\Big)\Big\} + \rho\mathbf{g} \tag{3.1c}$$

$$\mathbf{q} = -\lambda\nabla T \tag{3.1d}$$

$$h_{tot} = c_p T + \frac{\|\mathbf{u}\|^2}{2} = e_{tot} + \frac{p}{\rho} \tag{3.1e}$$

$$p = \rho R_0 T/w \tag{3.1f}$$

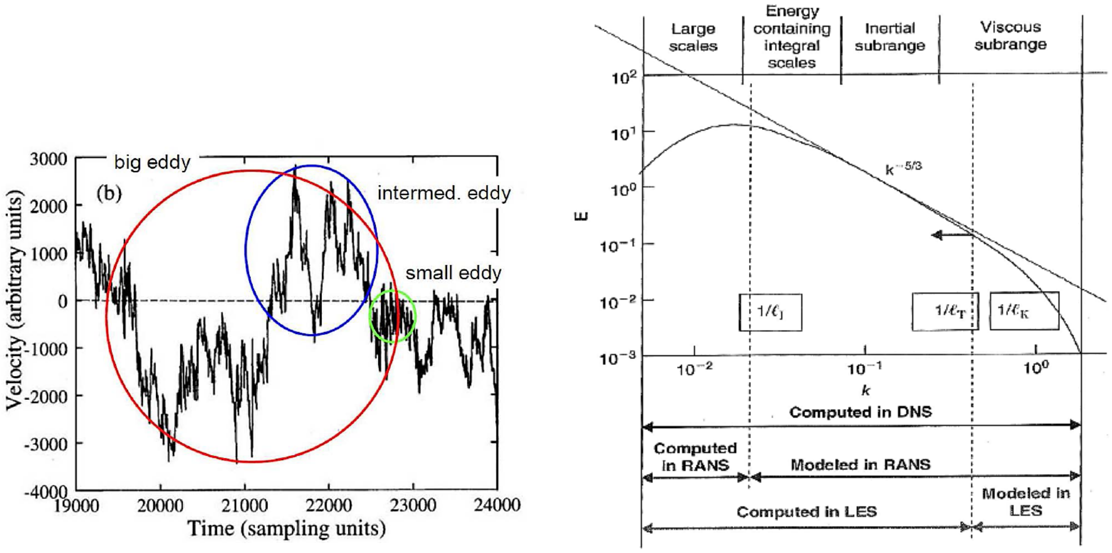
with p [Pa] static pressure; ρ [kg m^{-3}] density; $\mathbf{u}$ [m s^{-1}] velocity vector; $\mathbf{g}$ [m s^{-2}] gravitational acceleration vector; T [K] static temperature; h_{tot} [J kg^{-1}] total enthalpy; e_{tot} [J kg^{-1}] total internal energy; $\mathbf{q}$ [J s^{-}1 m^{-}2] heat flux density, c_p [J kg^{-1} K^{-1}] constant pressure specific heat capacity; w [kg mol^{-1}] molar mass of gas mixture; R_0 [J K^{-1} mol^{-1}] universal gas constant; μ [Pa s] dynamic viscosity; λ [J s^{-1} m^{-1} K^{-1}] fluid thermal conductivity.

For the sake of completeness it must be noted that μ, λ are function of temperature and pressure (or other two thermodynamic intensive properties) and their expression or tabular data need to be implemented in the code to well-pose the mathematical problem, along with adequate boundary and initial conditions.

3.2 Turbulence modelling

Navier-Stokes equations are rather complex and strongly non-linear. This is due to the flow compressibility, but, most of all, to the convection term, which is responsible of turbulence. This is a great problem when Navier-Stokes are attacked analytically, since this strong non-linearity implies non-uniqueness of the solution. In this frame, it is important to distinguish between laminar flows and turbulent ones. Laminar flows are characterised by a stable, parallel and well ordered movement with no strong interaction between the different layers of the flow and low momentum convection. Turbulent flows are a much more complicated phenomenon. Turbulence consists of fluctuations in the flow field in time and space. It is a complex process, mainly because it is three-dimensional, unsteady and consists of many scales. According to [54], turbulent flow is characterised by the following features:

- randomness: the turbulent flows seem chaotic and unpredictable, in contrast with the laminar flow;
- non-linearity: it means that the flow is unstable. This causes two important features in a turbulent flow:
  - small perturbations can grow up to finite amplitude disturbances. If these disturbances exceed the stability criteria, the flow becomes more unstable, reaching a chaotic state;
  - it causes vortex stretching, a process by which three-dimensional turbulent flows keep their vorticity as constant.
- diffusivity: given the macroscopic mixing processes inside the turbulent flows, they are characterised by an higher rate of diffusion of momentum and heat;
- vorticity: turbulence is characterised by high levels of fluctuations of all the physical quantities, included vorticity. The swirling structures which can be identified in a turbulent flow are called eddies. Turbulent flow is characterised by a wide range of eddy sizes, from the width of the region occupied by the turbulent flow up to very small eddies (Kolmogorov scale). The largest eddies contain most of the energy and it is handed down from the largest to the smallest scale by means of non-linear interactions (Fig. 3.1).
- dissipation: the vortex stretching mechanism transfers energy to smaller and smaller scales of the vortices, up to a scale where the vortex itself is smeared out by the fact that gradients are too strong. This is a dissipative phenomenon dominated by the molecular viscosity.



(a) (b)

Figure 3.1. Turbulent eddies scale

Turbulence occurs when the inertia forces in the fluid become significant compared to viscous forces, and it is characterized by a high Reynolds Number. In principle, the Navier-Stokes equations describe both laminar and turbulent flows without the need for additional information. However, turbulent flows at realistic Reynolds numbers span a large range of turbulent length and time scales, and would generally involve length scales much smaller than the smallest finite volume mesh, which can be practically used in a numerical analysis. The Direct Numerical Simulation (DNS) of these flows would require computing power which is many orders of magnitude higher than available in the near future. To enable the effects of turbulence to be predicted, a large amount of CFD research has concentrated on methods that make use of turbulence models. Turbulence models have been specifically developed to account for the effects of turbulence without recourse to a prohibitively fine mesh and direct numerical simulation. Most turbulence models are statistical turbulence model, as described below. The two exceptions described in literature are the Large Eddy Simulation (LES) model and the Detached Eddy Simulation (DES) model.
When looking at time scales much larger than the time scales of turbulent fluctuations, turbulent flow could be said to exhibit average characteristics, with an additional time-varying, fluctuating component. For example, a velocity component may be divided into an average component, and a time varying component. In general, turbulence models seek to modify the original unsteady Navier-Stokes equations by the introduction of averaged and fluctuating quantities to produce the URANS equations. These equations represent the mean flow quantities only, while modelling turbulence effects without a need for the resolution of the turbulent fluctuations. All scales of the turbulence field are being modelled. Turbulence models based on the RANS equations are known as Statistical Turbulence Models due to the statistical averaging procedure employed to obtain the equations. Simulation of the RANS equations greatly reduces the computational effort compared to a Direct Numerical Simulation and is generally adopted for practical engineering calculations. However, the averaging procedure introduces additional unknown terms containing products of the fluctuating quantities, which act like additional stresses in the fluid. These terms, called 'turbulent' or 'Reynolds' stresses, are difficult to determine directly and so become further unknowns. The Reynolds (turbulent) stresses need to be modelled by additional equations of known quantities in order to achieve "closure." Closure implies that there is a sufficient number of equations for all the unknowns, including the Reynolds-Stress tensor resulting from the averaging procedure. The equations used to close the system define the type of turbulence model [19].

3.2.1 Reynolds Averaged Navier-Stokes equations

As described previously, turbulence models seek to solve a modified set of transport equations by introducing averaged and fluctuating components. Once variables in Navier-Stokes equations have been replaced with their decomposition quantities, the averaging procedure must be performed. In compressible flows, instantaneous variables are split into a mass-weighted mean and a fluctuating part.

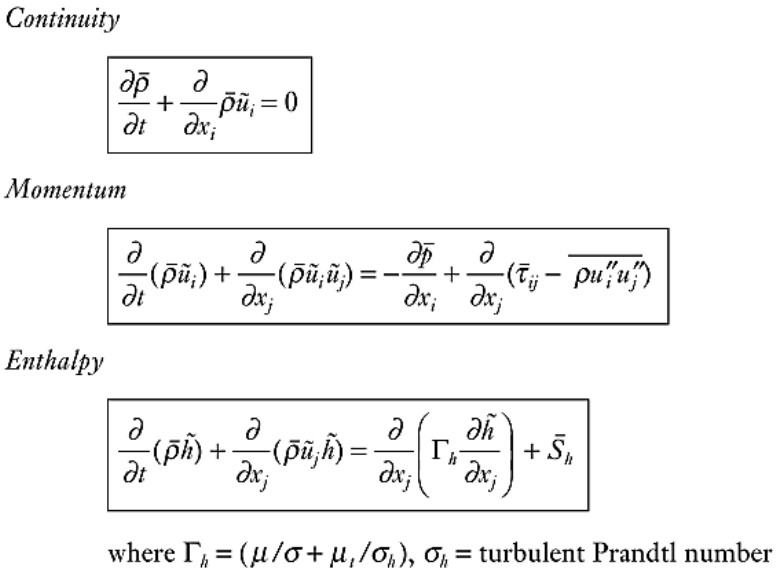
$$\phi = \tilde{\phi} + \phi'' \quad with \quad \tilde{\phi} = \frac{\widetilde{\phi\rho}}{\tilde{\rho}} \qquad \overline{\rho\phi''} = 0$$

This decomposition is called *Favre averaging* and it is an outstanding mathematical (not physical) simplification; it removes density fluctuations from the equations, not from the turbulence. Small disturbance fluctuations in compressible flows can be categorized into acoustic, total temperature (or entropy) and vorticity disturbances. Morkovin's hypothesis states that the effects of density fluctuations on the turbulence structure are unimportant if $\rho' << \bar{\rho}$.
Pressure and density are the only variables decomposed in the Reynolds fashion:

$$\rho = \bar{\rho} + \rho' \quad with \quad \bar{\rho} = \frac{1}{t}\int_t \rho(t)dt$$

$$p = \bar{p} + p' \quad with \quad \bar{p} = \frac{1}{t}\int_t p(t)dt$$

Performing the time averaging procedure, the system displayed in Fig 3.2 is obtained: In the Navier-Stokes equations, each component is decomposed as reported above and



Continuity

$$\boxed{\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial}{\partial x_i}\bar{\rho}\tilde{u}_i = 0}$$

Momentum

$$\boxed{\frac{\partial}{\partial t}(\bar{\rho}\tilde{u}_i) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_i\tilde{u}_j) = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(\bar{\tau}_{ij} - \overline{\rho u''_i u''_j})}$$

Enthalpy

$$\boxed{\frac{\partial}{\partial t}(\bar{\rho}\tilde{h}) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{h}) = \frac{\partial}{\partial x_j}\left(\Gamma_h \frac{\partial \tilde{h}}{\partial x_j}\right) + \bar{S}_h}$$

where $\Gamma_h = (\mu/\sigma + \mu_t/\sigma_h)$, σ_h = turbulent Prandtl number

Figure 3.2. Favre averaged Navier-Stokes equations from [17]

then a time averaging is performed. The linear terms are replaced by the corresponding mean terms, while the non-linear term is given by a mean component and a non-linear combination of fluctuating components under the divergence operator. In the momentum equation fluctuating component acts like a stress added to the viscous one: this is called Reynolds stress and it is a symmetric tensor. Reynolds stresses physically represent the driving force (turbulent transport) that enhances diffusivity and mixing rate in turbulent flows. Similarly, in the energy equation the non-linear convective term causes the presence of a combined fluctuating term, which is called turbulent heat flux. It represents the way in which the turbulent transport enhances the diffusivity with respect to the molecular one. Boussinesq proposed a purely formal analogy with the Newton's stress strain-rate law, which represents the Reynolds stress tensor as summation of two components: an isotropic part, related to the Turbulent Kinetic Energy (TKE) (k in eq. 3.2), and a deviatoric one proportional to the mean strain rate matrix.

$$\overline{r} = -\overline{\rho u_i'' u_j''} = \nabla \cdot (\rho\, \mathbf{u}'' \otimes \mathbf{u}'') = -\overline{\rho}\frac{2}{3}k\mathbf{I} + 2\mu_t\overline{\mathbf{D}} \qquad with \qquad k = \frac{\|\mathbf{u}''\|^2}{2} \tag{3.2}$$

The term μ_t is called turbulent viscosity or eddy viscosity, while λ_t is called turbulent thermal conductivity. In this way, the problem is closed, but a modelling for these two parameters is required. The value of turbulent thermal conductivity comes from assumption on the turbulent Prandtl number, generally posed in a range between 0.85 and 0.95, but other solution are needed to model the turbulent viscosity.
Two approaches are available to face this problem:

- Boussinesq eddy-viscosity approximation, which assumes that the principal axes of the Reynolds stress tensor are coincident with those of the mean strain rate tensor ($\overline{\mathbf{D}}$ in eq. 3.2) at all points in a turbulent flow. This assumption is inaccurate, since it is incapable of capturing the more subtle relationships between turbulent energy production and turbulent stresses caused by anisotropy of the normal ones, sometimes leading to incorrect simulation of the flow.

- Reynolds Stress Models (RSM) deal with this theoretical gap since they are turbulent models which close the Reynolds Averaged Navier–Stokes (RANS) system by solving additional transport differential equations or algebraic equations Explicit Algebraic Reynolds Stress Model (EARSM) for each of the six Reynolds stress component.

In this two main class of turbulence closure methodologies, different models are available and differentiate each one with respect the other primarily for the number and type of additional equations involved to model the Reynolds stresses.
In Ansys CFX are present several fluid model options:

- Laminar flow
  - None (Laminar): Turbulence is not modelled.
- Boussinesq eddy-viscosity approximation, modelled as the product of a turbulent velocity scale and a turbulence length scale. $\mu_t = \rho U_t l_t$
  - Zero Equation (mixing length) model. Very simple eddy viscosity models compute a global value for μ_t from the mean velocity and a geometric length scale using an empirical formula. Because no additional transport equations are solved, these models are termed 'zero equation.' A constant turbulent eddy viscosity is calculated for the entire flow domain.
  - Eddy Viscosity Transport Equation: a one-equation variation of the k-epsilon model. In this transport differential equation there is the presence of a destruction term, which accounts for the structure of turbulence and is based on the von Karman length scale.
  - Two-equation models: widely used, as they offer a good compromise between numerical effort and computational accuracy. Both the velocity and length scale are solved using separate transport equations (hence the

term ‘two-equation’). The turbulence velocity scale is computed from the turbulent kinetic energy k, which is provided from the solution of its transport equation. The turbulent length scale is estimated from two properties of the turbulence field, usually the turbulent kinetic energy and its dissipation rate ϵ (or specific dissipation rate $\omega = \epsilon/k$), provided from the solution of its own transport equation.

- standard **k-**ϵ. Its main disadvantage is how it performs in the near wall region: the ϵ equation is problematic in these zones due to the presence of a term, singular at the wall. For this reason, special treatment is required i.e. wall function must be used. **K-**ϵ models, despite this problem, are some of the most prominent turbulence models, implemented in every general purpose CFD codes and are considered the industry standard model. It has proven to be stable and numerically robust with a well-established regime of predictive capability.
- Standard **k-**ω. One of its advantages is the near wall treatment for low-Reynolds number computations. The model does not involve the complex non-linear damping functions required for the **k-**ϵ model and is therefore more accurate and robust. A low-Reynolds **k-**ϵ model would typically require a near wall resolution of $y^+ < 0.2$, while a low-Reynolds number **k-**ω model would require at least $y^+ < 2$. The main problem with the **k-**ω model is its strong sensitivity to free-stream conditions. Depending on the value specified for ω at the inlet, significant result variations can be obtained.
- renormalisation group (RNG) **k-**ϵ: A variation of the k-epsilon model.
- Baseline (BSL)-zonal **k-**ω based model
- SST-zonal **k-**ω based model: recommended for accurate boundary layer simulations and always adopted in this thesis work. It is the standard for turbomachinery flow simulations (paragraph 3.2.2). Developed to overcome the weaknesses of BSL-zonal **k-**ω based model.

- Reynolds stresses modelling
  - **k-**ϵ EARSM / BSL EARSM: These models are a simplified version of the Reynolds stress models with application to problems with secondary flows as well as flows with streamline curvature and/or system rotation.
  - ω RSM / BSL RSM
  - QI / SSG / LRR RSM: Provides high accuracy for some complex flows.
- LES-based approach, in which some region of the flow are not averaged, but solved directly. It is based on the concept of resolving only the large scales of turbulence and to model the small scales. The classical motivation for LES is that the large scales are problem-dependent and difficult to model, whereas the smaller scales become more and more universal and isotropic and can be modelled more easily. LES is based on filtering the Navier-Stokes equations over a finite spatial region (typically the grid volume) and aimed at only resolving the

portions of turbulence larger than the filter width. All of the following model can be adopted only for unsteady simulation.

- LES Smagorinsky / LES WALE / LES Dynamic Model.
- DES.
- Stress Blended Eddy Simulation (SBES). This is a refinement of DES.
- Scale-Adaptive Simulation (SAS) SST.

3.2.2 k-ω SST

One of the main problems in turbulence modelling is the accurate prediction of flow separation from a smooth surface. Standard two-equation turbulence models often fail to predict the onset and the amount of flow separation under adverse pressure gradient conditions. This is an important phenomenon in many technical applications, particularly for airplane aerodynamics and for turbomachinery flows, because the stall characteristics of a turbomachine (or a plane) are controlled by the flow separation from the blade (or from the wing). In general, turbulence models based on the ϵ-equation predict the onset of separation too late and under-predict the amount of separation later on. This is problematic, as this behaviour gives an excessively optimistic performance characteristic for an airfoil, not conservative at all.
The BSL model combines the advantages of **k-**ω and **k-**ϵ models, but still fails to properly predict the onset and amount of flow separation from smooth surfaces. This deficiency primarily arises because these models, which do not account for the transport of the turbulent shear stress, overpredict the eddy-viscosity.
The **k-**ω based SST model fixes this defect by adopting a limiter to the formulation of the eddy-viscosity (3.3c). Currently, the latter is the most prominent two-equation models in this area. It accounts for the transport of the turbulent shear stress and gives highly accurate predictions of the onset and the amount of flow separation under adverse pressure gradients. Both BSL and SST models implement a blending function (3.3d) to smoothly activate the ϵ or the ω transport equation, based on the distance from the closer wall.

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho u_j k)}{\partial x_j} = P_k - \beta' \rho \omega k + \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{k3}}\right)\frac{\partial k}{\partial x_j}\right] \tag{3.3a}$$

$$\frac{\partial(\rho\omega)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j \omega - \left(\mu + \frac{\mu_t}{\sigma_{\omega 3}}\right)\frac{\partial \omega}{\partial x_j}\right) = \frac{\alpha_3 \omega}{k} P - \beta_3 \rho \omega^2 + \frac{(1-F_1)2\rho}{\omega \sigma_{\omega 2}}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} \tag{3.3b}$$

$$\mu_t = \frac{\rho a_1 k}{\max(a_1 \omega, S_{inv} F_2)} \tag{3.3c}$$

$$F_1 = \tanh\left\{\left[\min\left(\max\left(\frac{\sqrt{k}}{0.09\omega d}, \frac{500\nu}{d^2\omega}\right), \frac{4\rho\sigma_{\omega 2} k}{\mathrm{CD}_{k\omega} d^2}\right)\right]^4\right\} \tag{3.3d}$$

Superior performance of this model has been demonstrated in a large number of validation studies and therefore it is, nowadays, the standard for turbomachinery flow simulation. To benefit from this model, a resolution of the BL of more than 10 points is required, along with a $y^+ < 3$ at wall, to solve directly the viscous sublayer. For free shear flows, the SST model is mathematically identical to the **k-**ϵ model. One of the advantages of the SST formulation is the near wall treatment for low-Reynolds number computations, where it is more accurate and more robust.

3.3 Turbulent Boundary Layer

In not hypersonic flows, which is always the case in turbomachinery the no slip condition holds. It ensures that the fluid at the wall has its same velocity; in the case in which the wall is stationary, the velocity of the fluid in that position is null. After this statement is clear that there must be a region where the velocity passes from the free stream value to zero; the region close to the wall, where there are strong velocity gradients, is called boundary layer, and the velocity profile there it is in general independent from the free stream condition. The fluid there cannot be described with the classical Reynolds number theory. Approximating the fluid in the boundary layer with the wall bounded shear-flow model, which is almost the case when the flow is attached to the wall, we obtain a simplified equation for the velocity:

$$\left(\frac{\mu}{\rho} + l_t^2 \frac{\partial U}{\partial y}\right) \frac{\partial U}{\partial y} = \tau_{wall} \tag{3.4}$$

where:

l_t is the turbulent mixing length;

y is the coordinate normal to the wall;

U is velocity component parallel to the wall;

τ_{wall} is shear stress at the wall.

Equation 3.4 can be made adimensional with the introduction of a friction velocity and a viscous length scale, resulting in 3.5.

$$\left(1 + (l_t^+)^2 \frac{\partial U^+}{\partial y^+}\right) \frac{\partial U^+}{\partial y^+} = 1 \tag{3.5}$$

where:

$U_\tau = \sqrt{\tau_{wall}/\rho}$ is the friction velocity;

$\delta_v = \frac{\mu}{\rho U_\tau}$ is the viscous length scale;

$y^+ = y/\delta_v$ is the adimensional velocity;

$U^+ = U/U_\tau$ is the adimensional distance.

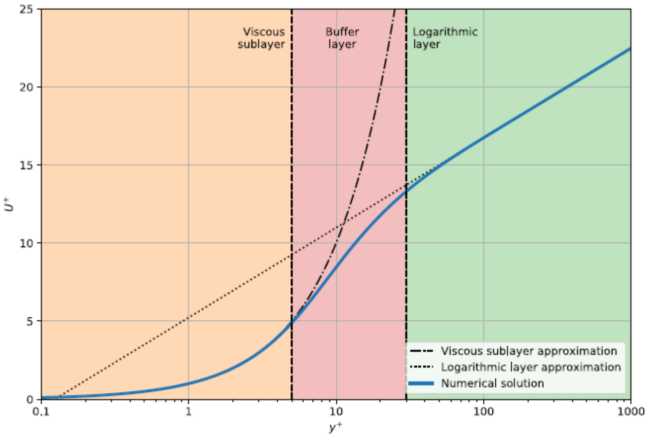


Figure 3.3. Boundary layer - dimensionless velocity profile

Equation (3.5) can be easily solved in two different conditions: really close to the wall, where $y+ \rightarrow 0$, and far from the wall, where $(l_t^+)^2 \frac{\partial U^+}{\partial y^+} \gg 1$. The solutions in this two extreme conditions are:

$$U^+ = \begin{cases} y^+ & \text{if } 0 < y^+ < 5 \rightarrow \text{viscous sub layer} \\ \dfrac{1}{n} \log x + C & \text{if } 30 < y^+ < 1000 \rightarrow \text{logarithmic layer} \end{cases} \tag{3.6}$$

These considerations hold whenever we are considering an incompressible turbulent boundary layer, but with the strong hypotheses implicitly adopted and the approximation introduced. Indeed this wall function, which includes a very important universal logarithmic law, is the most simple among the various wall functions that can be adopted for BL approximation. The parameter y^+, anyway, plays a relevant role for whatever turbulence model we are using, since its value at the wall gives us information about if the near-wall mesh is fine enough for the purpose. In particular, for URANS equations, the $k-\omega$ based models are able to solve directly the viscous sublayer if a mesh resolution fine enough is adopted, as explained in paragraph 3.2.2, so y^+ is required to be less than 1 or at least less than 3. On the other side, the $k-\epsilon$ based models present singularity near the wall, so aWhat as been described high value of y^+, at least 30, is needed, letting that wall functions approximate the BL solution.

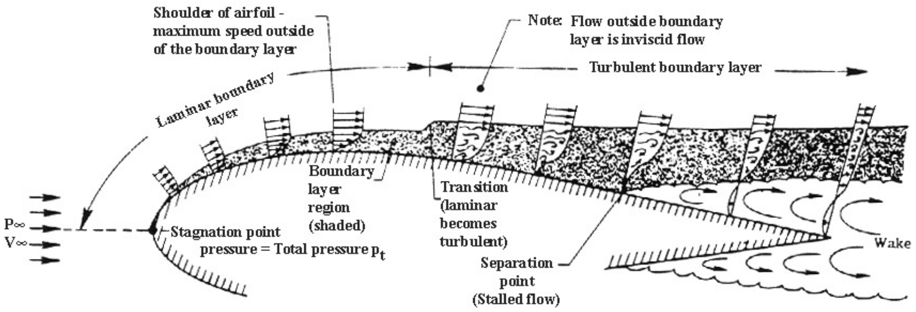

Shoulder of airfoil - maximum speed outside of the boundary layer
Note: Flow outside boundary layer is inviscid flow
Turbulent boundary layer
Laminar boundary layer
Boundary layer region (shaded)
Transition (laminar becomes turbulent)
P∞
V∞
Stagnation point pressure = Total pressure p_t
Separation point (Stalled flow)
Wake

Figure 3.4. Boundary layer behaviour along blade profile

These phenomenons are, in reality and in actual numerics, further complicated by the compressibility role and by the transitional behaviour of the boundary layer. Indeed, the latter always presents an initial laminar behaviour that, along the blade profile, change into a full turbulent behaviour, after a transitional phase characterized by intermittent phenomenology (Fig 3.4). What has been described and observed in every experimental work is much more difficult to simulate. Nowadays, a certain number of models exist able to do that, but still relying on characteristic parameters to be tuned in every different situation. For this reason, intermittency models for transition BL behaviour have not been included in the simulations present in this work.

3.4 Finite Volume Method - Ansys CFX

Assuming known all the mathematical and physical aspects of the governing equations and the turbulence models, it is possible to deal with the numerical approaches, developed during the last century to solve these equation. It is important to remark that Navier-Stokes equations can be solved analytically only in very few cases, not of practical interest and for this reason CFD is still under continuously development.

Computational Fluid Dynamics can be defined as a computer-based tool for simulating the behaviour of systems involving fluid flow, heat transfer, and other related physical processes. It works by solving the equations of fluid flow in integral form over a region of interest, with specified conditions on the boundary of that region.

Historically, computers have been used to solve fluid flow problems for many years, starting with codes for specific class of problems. From the mid-1970s, the complex mathematics required to generalize the algorithms began to be understood and general purpose CFD solvers were developed. Recent advances in computing power, together with powerful graphics and interactive 3D manipulation of models, have made the process of creating a CFD model and analysing results much less labour intensive, reducing time and, hence, cost. As a result of these factors, CFD is now an established industrial design tool, helping to reduce design time scales and improve processes throughout the engineering world, although not at the same level of other computational techniques such as mechanical stress or motion simulation. CFD provides a cost-effective and accurate alternative to scale model testing, with variations on the simulation being performed quickly, offering obvious advantages [19]. The ultimate aim of developments in the CFD field is to provide a capability comparable with other CAE (computer-aided engineering) tools such as stress analysis codes. The main reason why CFD has lagged behind is the tremendous complexity of the underlying behaviour, which precludes a description of fluid flows that is at the same time economical and sufficiently complete [17].

The main part of every CFD code is related to domain and equation discretization and to their solving procedure. The Navier-Stokes equations can also be solved in conjunction with other algebraic or differential equations, describing related processes, such as combustion. To solve these systems of equations, there are a number of different methods which are used. The most common, and the one on which **Ansys CFX** is based, is known as the Finite Volume Method (FVM). In this technique, the region of interest is divided into small sub-regions, called control volumes. The equations are discretized and solved iteratively for each control volume. As a result, an approximation of each variable value at specific points throughout the domain can be obtained and exploited to derive a full flow behaviour picture. The FVM uses the integral form of the conservation equations as its starting point. The solution domain is subdivided into a finite number of contiguous control volumes, and the conservation integral equations are applied to each Control Volume (CV). At the centroid of each CV lies a computational node at which the variable values are to be calculated. Interpolation is used to express variable values at the CV surface in terms of the nodal (CV-center) values. Surface and volume integrals are approximated using suitable quadrature formulae. As a result, one obtains an algebraic equation for each CV, in which a number of neighbour nodal values appear. The Finite Volume (FV) method can accommodate any type of grid, so it is suitable for complex geometries. The grid defines only the control volume boundaries and need not to be related to a coordinate system. The method is conservative by construction, so long as surface integrals (which represent convective and diffusive fluxes) are the same for the CVs sharing the boundary. The FV approach is perhaps the simplest to understand and to program. All terms that need to be approximated have physical meaning which is why it is popular with engineers. The disadvantage of FV methods compared to Finite Difference schemes is that methods of order higher than second are more difficult to develop in 3D. This is due to the fact that the FV approach requires three levels of approximation: interpolation, differentiation, and integration [55].

3.4.1 Steps of a CFD simulation

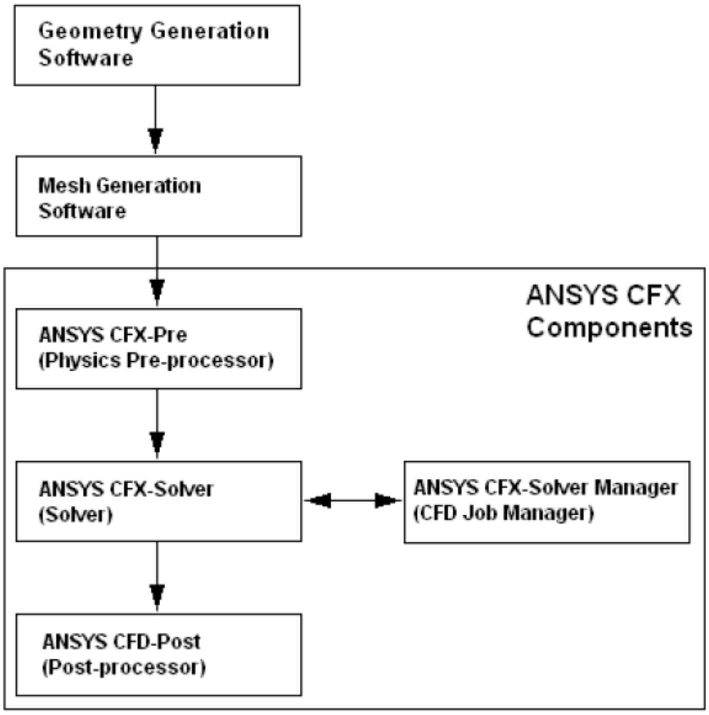

Geometry Generation Software
Mesh Generation Software
ANSYS CFX Components
ANSYS CFX-Pre (Physics Pre-processor)
ANSYS CFX-Solver (Solver)
ANSYS CFX-Solver Manager (CFD Job Manager)
ANSYS CFD-Post (Post-processor)

Figure 3.5. Ansys CFX modules

The process of performing a single CFD simulation is split into four components and is described hereafter making reference to Ansys CFX (Fig. 3.5):

1. Creating the Geometry. In this work coordinate files produced during the 0D design in **Matlab** were provided as the input geometry.

2. Grid generation (i.e. divide the computational domain in small sub-regions):

   - Define solid and fluid regions;
   - Setting properties for the mesh;
   - Create surface boundary names;
   - Proper mesh generation.

   This interactive process is the first pre-processing stage and it is performed with Ansys **TurboGrid**. The objective is to produce a mesh for input to the physics pre-processor, from a closed geometric solid.

3. Defining the Physics of the Model. This interactive process is the second pre-processing stage and is used to create input required by the Solver. The mesh files are loaded into the physics pre-processor, CFX-Pre, and user actions are required:

   - Selection of the physical and chemical phenomena that need to be modelled
   - Definition of fluid properties
   - Specification of appropriate boundary conditions at cells, which coincide with the domain boundary

4. Solving the CFD Problem. CFX-Solver produces the required results in a non-interactive process. Every CFD code solves the problem as follows:

   (a) The partial differential equations are integrated over all the CVs in the region of interest. This is equivalent to applying basic conservation laws to each control volume. When summed the internal surface integrals cancel out, thus global conservation is built into the method.

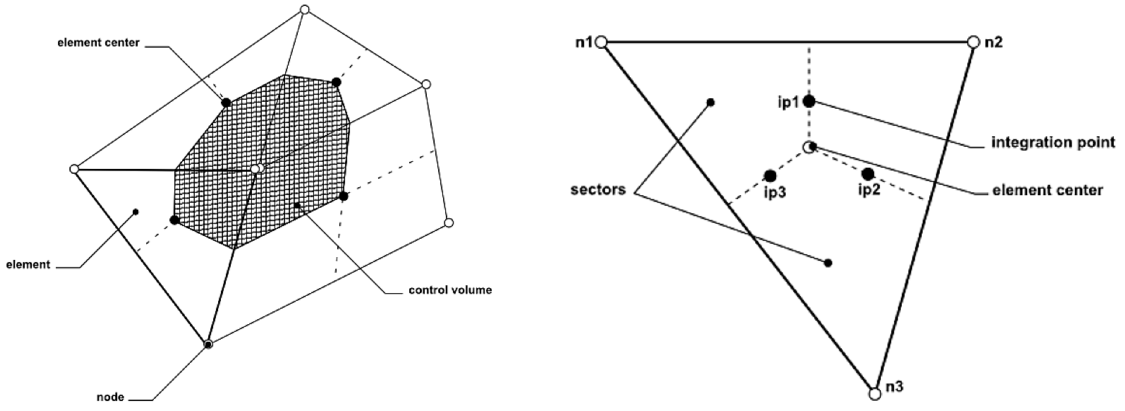
   (b) These integral equations are converted to a system of algebraic equations by generating a set of approximations for the terms in the integral equations.

   (c) The algebraic equations are solved iteratively. An iterative approach is required because of the non-linear nature of the equations, and, as the solution approaches the exact one, it is said to converge. For each iteration, an error, or residual, is reported as a measure of the overall conservation of the flow properties. Accuracy and convergence of the solution depend on a number of factors, including the size and shape of the control volumes and the size of the final residuals. In general, the larger the number of cells, the better the solution accuracy. Both the accuracy of a solution and its cost in terms of necessary computer hardware and calculation time are dependent on the grid fineness. Optimal meshes are often non-uniform: finer in areas where large variations occur from point to point and coarser in regions with relatively little change.

5. Visualizing results in the Post-processor (CFX-Post). The solver produces a results file that is then passed to the post-processor and can be visualized and further elaborate.

3.4.2 Domain discretization

The most important idea behind the computational fluid dynamics is the discretization of the domain and of the balance equations. There are three distinct streams of numerical solution techniques: finite difference, Finite Element Method (FEM) and spectral methods. The finite volume method, a special finite difference formulation, is central to the most well established CFD codes: Ansys CFX, FLUENT, Open-Foam and STAR-CD [17]. **Ansys CFX** uses an *element-based finite volume method*, which first involves discretizing the spatial domain using a mesh. The mesh is used to construct finite volumes, which are used to conserve relevant quantities such as mass, momentum, and energy. ANSYS CFX uses a co-located (non-staggered) grid layout such that the control volumes are identical for all transport equations, i.e. all solution variables and fluid properties are stored at the same nodes (mesh vertices). A control volume (the shaded area) is constructed around each mesh node using the median dual (defined by lines joining the centers of the edges and element centers surrounding the node) [19].



(a)

(b)

Figure 3.6. Control Volume Definition

In order to create mesh for turbomachinery analysis, it is possible to use Ansys TurboGrid. This software is tailored for the specific aim of creating the most appropriate grid for turbomachinery simulation through an automated procedure, starting from CAD geometries or from coordinate files (.crv). The last way is the one used in this work, since the blade shape have been defined during the mean-line 0D design (Chapter 2). This software is based on an automated topology and mesh (ATM) technology, that produces high-quality hexahedral in an automated manner. With minimal user input, the ATM method adapts the block topology for the given blade geometry, yielding a smooth mesh with high orthogonality. For this thesis purpose, two main input parameters have played an important role:

- Target Passage Mesh Size, which allows the user to specify the number of nodes present in the full passage to be simulated;
- The Near-wall Element Size Specification parameter, which specifies the boundary layer mesh refinement through the $y{*}+$, evaluated with the selected Reynold number.

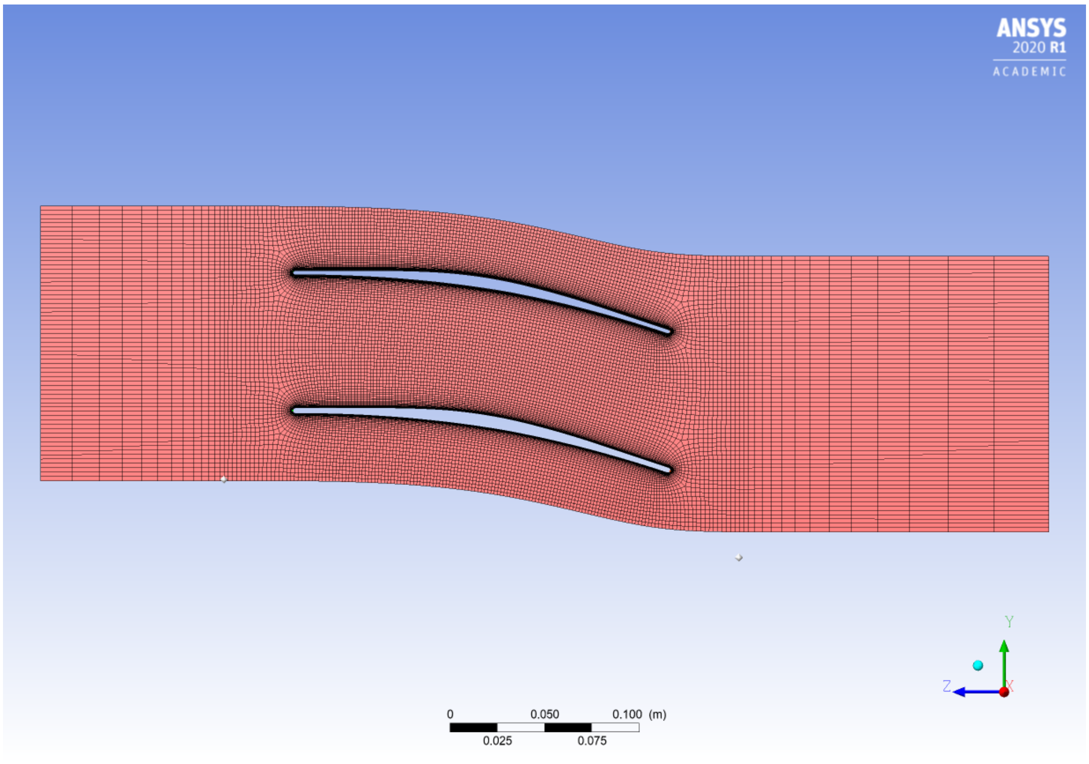


Figure 3.7. TurboGrid blade-to-blade coarse mesh

3.4.3 Integration and Discretization of Governing Equations

The governing equations have to be discretized in time and space to be solved numerically. To illustrate the element-based finite volume methodology, the conservation equation (Eq. 3.7) of a scalar quantity will be considered, expressed in Cartesian coordinates:

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u_j \phi)}{\partial x_j} = S_\phi + \frac{\partial}{\partial x_j}\left(\Gamma_\phi \frac{\partial \phi}{\partial x_j}\right) \tag{3.7}$$

These equations are integrated over each control volume, and Gauss' Divergence Theorem is applied to convert volume integrals involving divergence and gradient operators to surface integrals. If control volumes do not deform in time (i.e not deforming mesh), then the time derivatives can be moved outside of the volume integrals and the integrated equations become:

$$\frac{\partial}{\partial t}\int_V \rho\phi dV + \int_S \rho u_j \phi \cdot dn_j = \int_S \Gamma_\phi \frac{\partial \phi}{\partial x_j} \cdot dn_j + \int_V S_\phi dV \tag{3.8}$$

where:

- V denote volume regions of integration; volume integrals represent source or accumulation terms;
- S denote surface regions of integration; surface integrals represent the summation of fluxes, which can be advective or diffusive ones;
- dn_j are the differential Cartesian components of the outward normal surface vector.

Volume integrals are discretized within each element sector and accumulated to the control volume to which the sector belongs, while surface integrals are discretized at the integration points (ipn) located at the center of each surface segment within an element and then distributed to the adjacent control volumes. Because the surface integrals are equal and opposite for control volumes adjacent to the integration points, the surface integrals are guaranteed to be locally conservative. Again, it must be noted that last statement is one of the more attractive point for the FVM, when dealing with partial differential equations with convective-diffusion terms, such as the Navier-Stokes system. After discretizing the volume and surface integrals, the integral equations become:

$$\underbrace{\frac{V}{\Delta t}\left(\frac{3}{2}(\rho\phi) - 2(\rho\phi)^o + \frac{1}{2}(\rho\phi)^{oo}\right)}_{\text{unsteady term}} + \underbrace{\sum_{ip} \dot{m}_{ip}\phi_{ip}}_{\text{advective term}} = \underbrace{\sum_{ip}\left(\Gamma_\phi \frac{\partial \phi}{\partial x_j}\Delta n_j\right)}_{\text{diffusion term}} + \underbrace{V\overline{S_\phi}}_{\text{source term}} \tag{3.9}$$

where:

- V is the control volume;
- Δt is the time step;
- Δn_j is the discrete outward surface vector;
- ip subscript refers evaluation at an integration point
- ^{o} superscript denotes the old time level.
- $m_{ip} = (\rho u_j \Delta n_j)_{ip}$

For the discretization phase, every integral term in Eq. 3.8 have been substitute with a discrete approximation by means of different strategies. Ansys CFX shows a smaller range of choices for the discretization to the users, with respect to other software, such as OpenFoam or also Ansys Fluent. This is because the algorithm did already implement the best discretization schemes with the ability to automatically switch from one to the other, based on the particular simulation and even iteration.

For the unsteady term, for example, it is possible to chose between a First Order Backward Euler scheme or a Second Order Backward Euler. For transient blade row simulation in this thesis, the latter have been chosen. Both methods are robust, implicit, conservative in time and do not have a time step size limitation, but the first discretization method is bounded and only first-order accurate in time (it will introduce discretization errors that tend to diffuse steep temporal gradients), while the second discretization method is second-order accurate but is not bounded and may create some non-physical solution oscillations. In several schemes for compressible flows, such the ones implemented in Ansys CFX, the solution is achieved by time-marching the calculation from an initial state, also for steady simulation. Due to the time-marching nature of the calculation, care must be placed on the time-step setting. Despite being implicit, the non-linearity can lead to numerical instability at the beginning of the calculation and in presence of very small cells (such as the ones in the BL O-grid to get $y^+ = 1$).

For what concern the space discretization of the equation (i.e diffusion and convective term discretization schemes), in CFD dedicated literature [17, 55, 56] a huge amount of methods and analysis is present, but here just a brief reference to the schemes implemented in Ansys CFX will be exposed.
The advection term (3.7) requires the integration point values of ϕ_{ip} to be approximated in terms of the nodal values ϕ_i. The advection schemes implemented in ANSYS CFX can be cast in the form:

$$\phi_{ip} = \phi_{up} + \beta \nabla\phi\Big|_{up} \cdot \Delta\vec{r} \tag{3.10}$$

where:

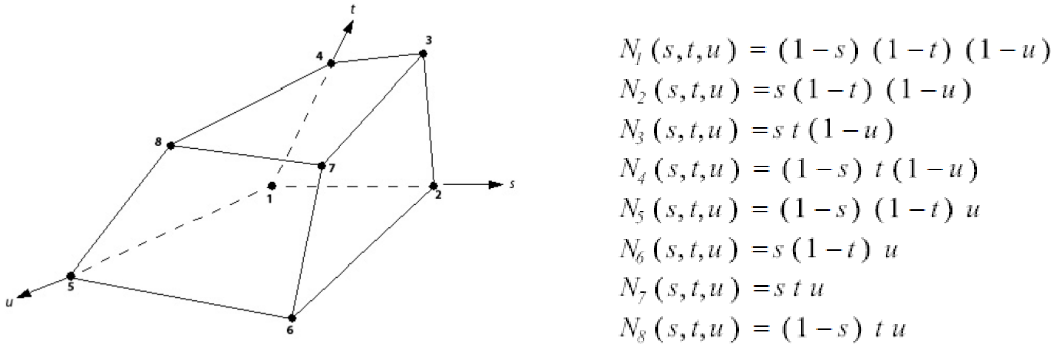
- ϕ_{up} is the **upwind** node value
- β (gradient limiter) and $\nabla\phi$ (3.11) identify the discretization scheme and are evaluated from the upwind node
- $\Delta\vec{r}$ the vector from the upwind node to the *ip*.

The High Resolution Scheme adopt a special non-linear formulation for β at each node, computed to be as close to one as possible without introducing new extrema. The advective flux is then evaluated using the values of β and $\nabla\phi$ from the upwind node. Gradient limiter β, also known as slope limiter, is used to prevent spurious oscillations, which would otherwise appear in the solution flow field near shocks, discontinuities, or near rapid local changes in the flow field. The gradient limiter attempts to invoke and enforce the monotonicity principle by prohibiting the linearly reconstructed field variable on the cell faces to exceed the maximum or minimum values of the neighbouring cells. β formulation is based on boundedness principles and can be demonstrate to be total variation diminishing in its 1D form. This limiter is of a non-differentiable type and uses the Minimum Modulus function (Roe, 1985) to limit and clip the reconstructed solution overshoots and undershoots on the cell faces. This scheme has been always adopted in this thesis work.

Peculiarity of the software formulation is its being element-based. Solution fields and other properties are stored at the mesh nodes, but, for many of the terms, the solution field or solution gradients must be approximated at integration points. ANSYS CFX uses finite-element *linear* shape functions to perform these approximations with $\phi = \sum_{i=1}^{N_{node}} N_i \phi_i$ (Fig. 3.8). This shape functions formulation, coupled with the Green-Gauss theorem, led to calculate gradient values at node location as:

$$\nabla\phi = \frac{1}{V}\sum_{ip}(\phi\Delta\vec{n}_{ip}) \tag{3.11}$$

For the diffusion term, again, CFX appeals to the shape function, following the standard finite-element approach. For a derivative in the x direction (diffusion term in 3.9) at integration point *ip*:



$$
\begin{aligned}
N_1(s,t,u) &= (1-s)\,(1-t)\,(1-u)\\
N_2(s,t,u) &= s\,(1-t)\,(1-u)\\
N_3(s,t,u) &= s\,t\,(1-u)\\
N_4(s,t,u) &= (1-s)\,t\,(1-u)\\
N_5(s,t,u) &= (1-s)\,(1-t)\,u\\
N_6(s,t,u) &= s\,(1-t)\,u\\
N_7(s,t,u) &= s\,t\,u\\
N_8(s,t,u) &= (1-s)\,t\,u
\end{aligned}
$$

(a) (b)

Figure 3.8. Hexahedral shape functions

$$\left.\frac{\partial\phi}{\partial x}\right|_{ip} = \sum_{n}\left.\frac{\partial N_n}{\partial x}\right|_{ip}\phi_n \tag{3.12}$$

The summation is over all the element shape functions, whose Cartesian derivatives can be expressed in terms of their own local derivatives by means of the Jacobian transformation matrix.

3.4.4 Linear Solver

Whatever is the strategy adopted to deal with every governing equation (see section 3.4.5), once the equations have been discretized on every CV, the problem reduces to solve a classical linear system $[A][\phi] = [b]$ (Fig. 3.9). To do that the direct solvers (Matrix inversion, Gauss Elimination, LU decomposition) are not convenient, since they are expensive in storage and CPU time so not suitable for large matrices like the ones involved in FVM for CFD. The only alternative are the iterative solvers, which start from a solution guess and, iteration after iteration, converge towards the exact solution.

ANSYS CFX uses a Algebraic Multigrid (AMG) accelerated Incomplete Lower-Upper (ILU) factorization technique [19, 57] for solving the discrete system of linearised

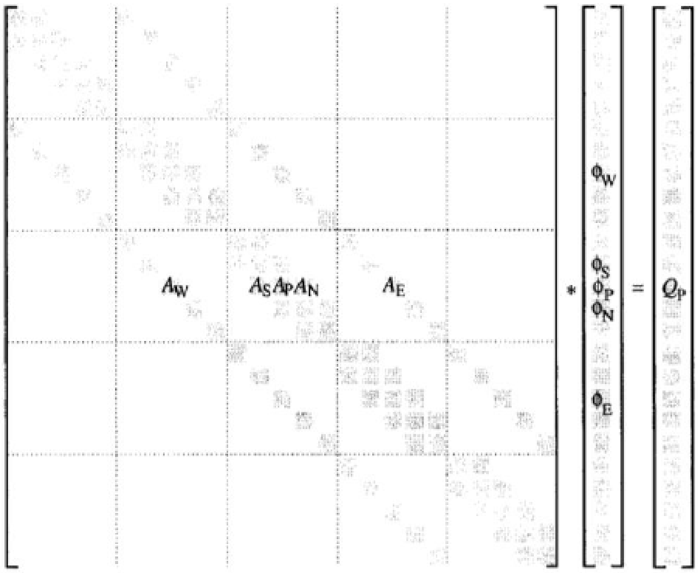
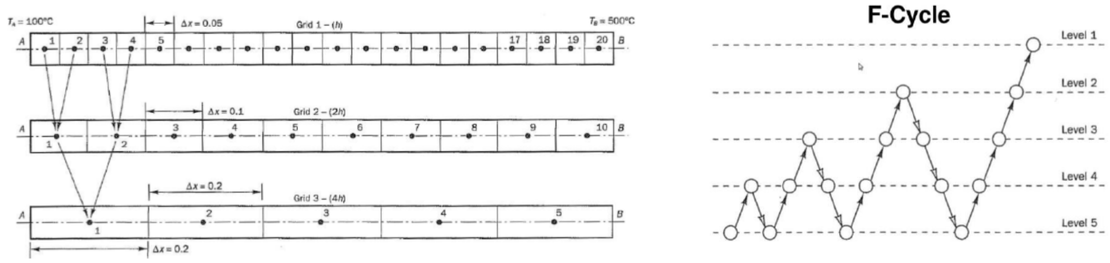


Figure 3.9. Representation of a linear system in matrix form for a scalar variable

equations. Since the matrix solvers are iterative, they will produce a residual as a measure of the error in the solution. A smaller residual gives a more accurate result. The residual is defined as:

$$[r]_k = [b] - [A][\phi]_k \tag{3.13}$$

where k subscript indicates the k-th iteration. A tolerance for each property has to be specified to control the accuracy of the linear solver. Specifying a low tolerance might be time consuming, and it is therefore important to consider an appropriate compromise between accuracy and the number of iteration loops needed to achieve convergence. The AMG technique consists in solving the first iterations on the real mesh (the refined one) and virtually coarsening the mesh for the latest iterations. These last results are then transferred on the fine mesh. From a numerical point of view, any iterative solver is good at reducing the error of the same wavelength of the mesh size and this means that errors with longer wavelength needs a lot of iterations to disappear. The coarsening of the mesh used by the AMG approach increases mesh spacing, without need of re-meshing, thus being able do reduce the errors in few iterations for all the wavelengths. AMG forms a system of discrete equations for a coarse mesh by summing the fine mesh equations. This results in virtual coarsening of the mesh spacing during the course of the iterations, and then re-refining the mesh to obtain an accurate solution. This reduces significantly the number of iterations required to reach the target residuals. Furthermore, the influence of boundaries and far-away points are more easily transmitted to interior of coarse mesh with respect to fine mesh. Convergence of the solution is based on the reduction of equation residual over successive solutions. The residual (3.13) is normalized to make it independent of the scale of the analysed problem. Before solving an equation for a particular field, the initial residual is evaluated on the current values of the field. After each solver iteration, the residual is re-evaluated. The procedure stops if any of the following conditions are reached:



(a)

(b)

Figure 3.10. Algebraic Multigrid procedure

- the residual falls below the solver tolerance;
- the ratio of current to initial residuals falls below the solver relative tolerance;
- the number of iterations exceeds a maximum specified by the user.

3.4.5 Solving Procedures

As briefly described in section 3.4.3, CFD codes contain discretization techniques suitable for the treatment of the key transport phenomena, convection (transport due to fluid flow) and diffusion (transport due to variations of ϕ from point to point) as well as for the source terms (associated with the creation or destruction of ϕ) and the rate of change with respect to time. Discretization process results in a system of linear algebraic equations to be solved iteratively due to the underlying, complex and non-linear physical phenomena described. The complexity and size of the set of equations depends on the dimensionality of the problem, the number of grid nodes and the discretization practice [17]. The solution of the linear equation system provides the values of the physical quantities at the node in a fixed time step.

The iterative solving procedures reported in literature and implemented in CFD codes are various and follow different logic. A simplistic categorization of the solution algorithms can be made on the basis of three main aspects that identify the logic of the codes themselves:

1. Time resolution techniques: implicit or explicit
2. Pressure-based or Density-based solver formulation
3. Segregated or Coupled equations "arrangement"

Although combination of the previous aspects returns eight solver types, only four of them are the most common and realizable and are represented in Fig. 3.11.

Ansys CFX is a **pressure-based implicit coupled solver**, so all the hydrodynamic equations are solved as a single system, then energy equation and, at the end, turbulent equations are solved. The coupled solver is faster than the traditional pressure-base segregated ones and fewer iterations are required to iterations required for convergence to a steady-state, or to calculate the solution for each time step in a time-dependent analysis. For this reason the continuity equation does not need to be manipulated to obtain a Poisson equation for the pressure, as it is done with the pressure-based

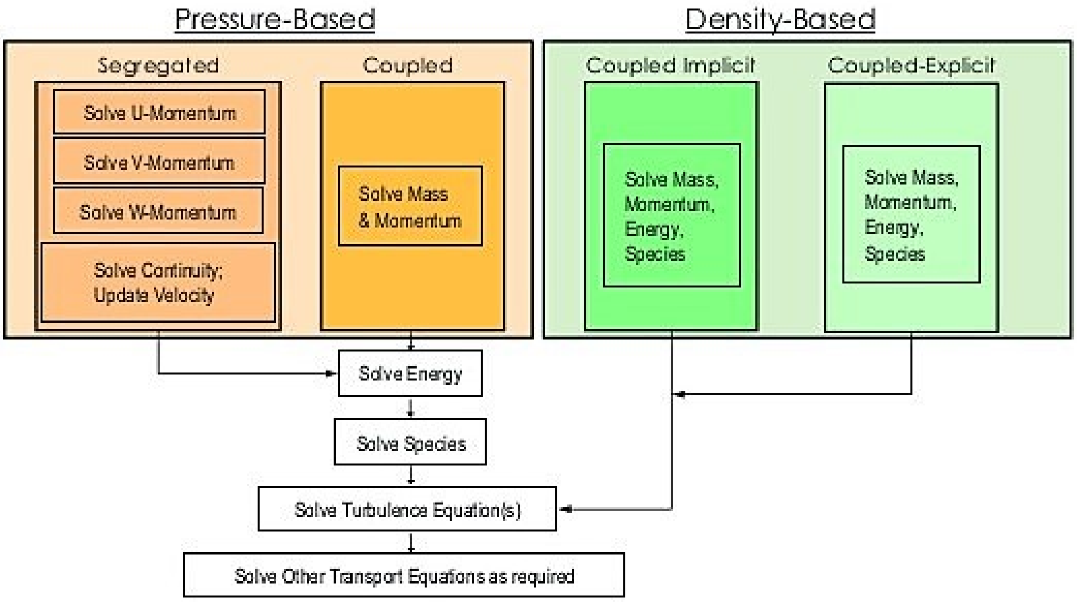

Pressure-Based
Density-Based
Segregated
Coupled
Coupled Implicit
Coupled-Explicit
Solve U-Momentum
Solve V-Momentum
Solve W-Momentum
Solve Continuity; Update Velocity
Solve Mass & Momentum
Solve Mass, Momentum, Energy, Species
Solve Mass, Momentum, Energy, Species
Solve Energy
Solve Species
Solve Turbulence Equation(s)
Solve Other Transport Equations as required

Figure 3.11. Schematization of different solver approaches: 1) pressure-based implicit segregated, 2) pressure-based implicit coupled, 3) density-based implicit coupled, 4) density-based explicit coupled

segregated solver algorithms (such as SIMPLE, SIMPLER, SIMPLEC, PISO and PIMPLE [17, 57]). In particular, the linear set of equations that arise by applying the FVM to all elements can be written in the forms:

$$\sum_j a_{ij}\phi_j = b_i \qquad \text{scalar unknowns on the } j\text{-th node} \tag{3.14}$$

$$\sum_j [A]_{ij}\vec{\phi_j} = \vec{B_i} \qquad \text{coupled scalar unknowns on the } j\text{-th node} \tag{3.15}$$

with ϕ the unknowns, a the equation coefficients, b the right hand side, i identifies a "neighbour" CV (or node), allowing the solver to deal with every kind of mesh. When dealing with transport equation of a scalar quantity, such as energy, TKE, specific energy dissipation rate and so on, the algebraic system of equation takes the form of 3.14, while with the hydrodynamic quantities the system takes the form of 3.15, with

$$\vec{\phi_j} = \begin{bmatrix} u_j \\ v_j \\ w_j \\ p_j \end{bmatrix} \tag{3.16}$$

The advantages of such a coupled treatment over a non-coupled or segregated approach are several: robustness, efficiency, generality, and simplicity. The principal drawback is the high storage needed for all the coefficients, twice the space needed for segregated algorithm. The complete flow diagram of the coupled solver is reported in Appendix B.

Chapter 4

Computational Models for Turbomachinery Flows

Design of a turbomachine is a long and not trivial path, which requires deep knowledges in various field of engineering, applied physics and numerical analysis. For this reason, there are a lot of people involved in this task for every single project since the first steps of the design phase up to the maintenance procedure during the entire machine life. OEMs deploy teams of engineers for the stage-by-stage design of an advanced gas turbine HGP and this procedure is rarely started from scratch, also in completely new products. Complexity of flows in turbomachine is really high and different model are available in the industrial practice and in literature with different level of accuracy. All these models are subjected to evolution during years but the core logic remains almost unchanged. These models are listed below and will be discussed in the next sections:

1. 0-D mean-line model: preliminary calculation method, object of chapter 2
2. Spanwise models:
   - 'classic' through-flow methods (*channel method* and *streamline curvature method*)
   - CFD axisymmetric model (also known as CFD-based through-flow method)
3. Quasi-3D CFD model: blade-to-blade simulation, object of chapter three.
4. Coupled stator-rotor CFD models for multi-row/multi-stage simulations
5. Fully 3D, time-resolved: high-fidelity methods

4.1 Spanwise model

The complete analysis of the flow through a turbomachine is a complex and three-dimensional problem, whose full numerical solution, also at present day, is not suitable for optimization procedure involving relevant amount of flow field simulation. First step in designing a turbomachine is to perform a 0-D mean-line analysis (point 1 in previous list), where the first design choices can be done, assessed and a first parameter optimization can be performed. Then, a more accurate flow field analysis is needed and a step forward in this term of accuracy, maintaining a relatively low computational cost, is to solve the meridional flow, exploiting a spanwise model (point 2 2). The problem of turbomachinery flow-field evaluation is here reduced to two separate and more manageable problems known as the hub-to-tip and the blade-to-blade (3). Aim in the hub-to-tip problem (i.e. spanwise model) is to solve for the flow only on a meridional surface of the turbomachine.

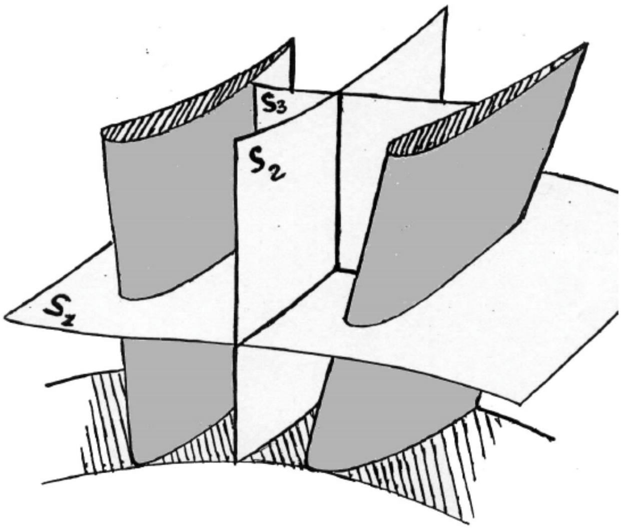


Figure 4.1. Stream flow surfaces: blade-to-blade surface S1; meridional surface S2; secondary flow surface S3

The first of these methodologies, the 'classical' one, is the channel method which directly deal with the *radial equilibrium* problem. Both the naive and the rigorous derivation of generalized radial equilibrium equation (4.1) can be easily found in literature along with its description and implications, under the assumption of axisymmetric flow.

$$V_a \frac{\partial V_a}{\partial r} - V_a \frac{\partial V_r}{\partial z} + \frac{V_t}{r} \frac{\partial r V_t}{\partial r} = \frac{\partial h_T}{\partial r} - T \frac{\partial s}{\partial r} \tag{4.1}$$

with V_a axial velocity component; V_t tangential component; V_r radial component; r, θ, z cylindrical coordinate; s entropy, evaluated from empirical correlation losses; h_T total enthalpy, from velocity triangles and Euler work.

The basic idea is to solve the equation considering an adequate amount of stream tubes (i.e. annular channel in which the annulus is divided), imposing the mass flow rate at the inlet. Stream tubes vary their own dimension to ensure the mass conservation and so it's possible to solve numerically the eq. (4.1). $V_a(r)$ and $\rho(r)$ are obtained and the procedure con be again iterated up to convergence in every stream tube station (Fig. 4.2).

An improved approach to study the meridional flow field is the so-called *Streamline Curvature Through-Flow method*, which calculates the flow on a meridional stream

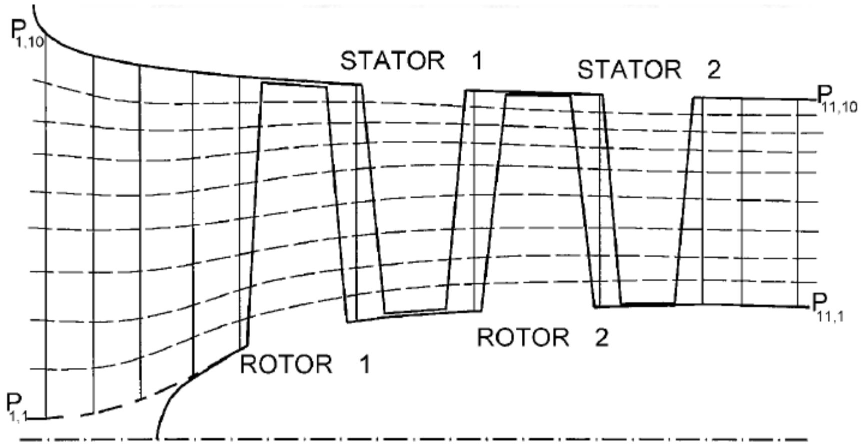


Figure 4.2. Meridional surface and quasi-normal traces

surface right across the annulus in duct and bladed regions (Fig. 4.3). The Streamline Curvature Through-Flow method is capable of obtaining supersonic solutions with the restriction that meridional Mach number should not exceed unity anywhere in the turbomachine and this imply that for supersonic turbine, treated in this thesis work, a further development should be necessary. The difference with the Channel method relies on the much more features involved: equation written in inherent stream-wise and normal coordinates; stream-surface orientation in circumferential and radial directions; viscous losses addressed with a 'drag' volume force.

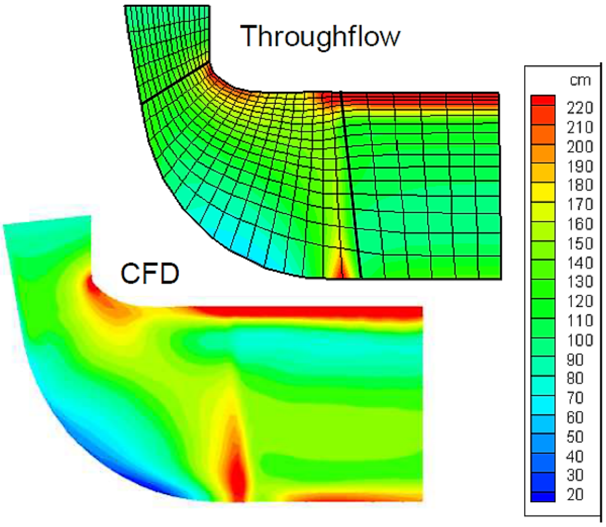


Figure 4.3. Comparison of the meridional flow-field of a mixed flow turbine compared with a 3D CFD calculation (ANSYS CFX) [18]

A modern approach (2) to the axisymmetric flow solution is the CFD through-flow model (Fig. 4.4), in which Euler or even URANS equations are projected on the meridional plane direction and solved by means FV or Finite Element (FE) technique, both suitable for explicit and implicit temporal integration. This method does not exhibit a certain number of the Streamline Curvature method shortcomings. Indeed, with a through-flow model that directly solves the Euler/Navier-Stokes equations, the flow-field inside a compressor or a turbine can be computed, whatever the flow regime is. Shocks can also be captured and choked mass flow predicted naturally. Furthermore, these codes benefit from the well known CFD techniques for solving the Euler/Navier-Stokes equations and, if solved by time-marching techniques, the unsteady capability is naturally included in the method. Also, as it shares the same numerics as 3D calculation codes, the coupling of both tools is made easier. The integration of both tools is indeed a desired feature in the design tool chain of a turbomachine. Finally, if the Navier-Stokes equations are used, the viscous flow developing on the annulus end-walls is captured and 2-D recirculation flows can be predicted.

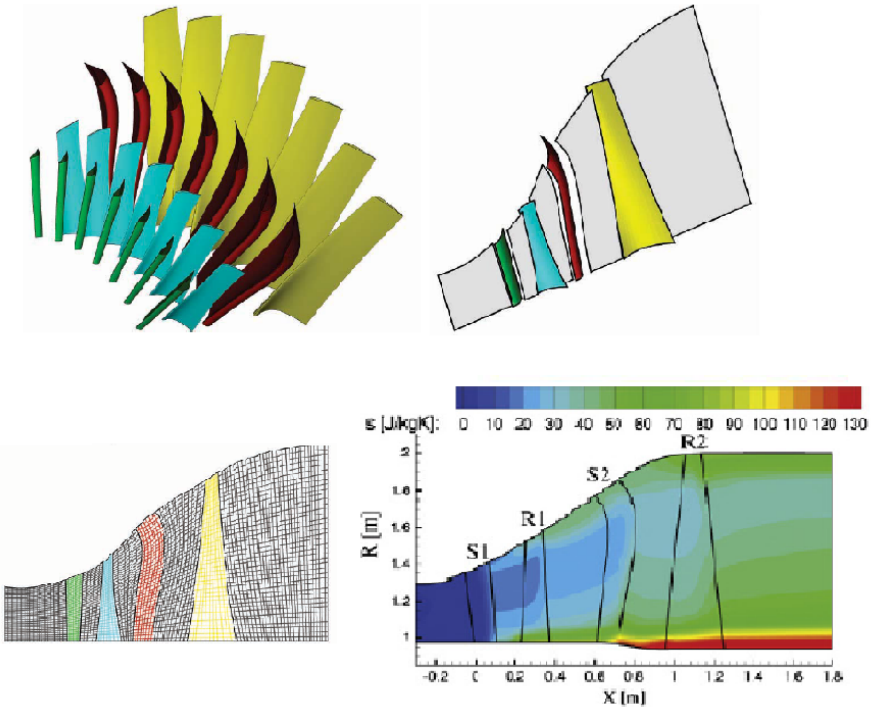


Figure 4.4. From left to right and top to the bottom: CAD turbine geometry reproduction; Blades replaced by mean surfaces; Mean surfaces projected on the meridional plane; Results in term of entropy production

4.2 Blade-to-blade simulation

Next step in turbomachinery flow evaluation is to calculate the flow on a blade-to-blade (S1 in Fig. 4.1) stream surface, given the stream surface shape. The objective is to design the detailed blade profile. The stream surface is best thought of as a stream tube (Fig. 4.5) with an associated stream surface thickness and radius which are obtained from the throughflow calculation. Accurate specification of the radius and thickness variation is essential as they can have a dominant effect on the blade surface pressure distribution. This is the reason why the coupling between spanwise model and blade-to-blade simulation is a powerful tool for the aerodynamic design of a turbomachine. Generally, only the blade-to-blade surface relative to a stream tube located at a mean radius is solved, but, if more accurate evaluations are needed, it's possible to solve more blade-to-blade flow-field. The coupling of this method with a spanwise model can be considered a *quasi-3D* simulation.

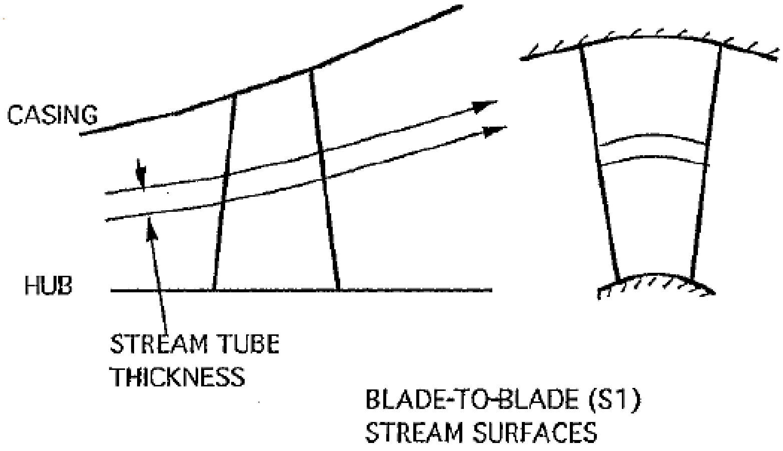


Figure 4.5. Blade-to-blade surface evolution

Alternatively, the blade-to-blade simulations can be performed in order to assess the performance of an annular or a linear cascade, ignoring the through-flow method coupling. This is the kind of analysis performed in this thesis work to study the two dimensional behaviour of the supersonic turbine blade.

4.3 Coupled stator-rotor simulation

With the increasing of computer performances, CFD simulations have been more extensively exploited and physical models involved are more complex. Nowadays, three dimensional unsteady simulations, with complete set of URANS equations, are possible for the full wheels with coupled stages, although still a relevant amount of time is required. To reduce the computational effort it is possible to simulate just an equal annular portion of rotor and stator with a finite number of channel (Fig. 4.6). Unfortunately, this approach, that exploits simple spatial periodicity, is unfeasible because good practices in turbomachinery design impose a mutually prime number of blades in two subsequent rows [19]. For this reason only full coupled wheel simulations are allowed, but, during the design phase, these kind of simulations are not of interest, since faster tool are required to optimize machine performances in an iterative process of aerodynamic flow feature improvement.
Different modern CFD techniques have been developed to deal with these requirements and, most of all, with the problem of unequal pitch at the connecting interface between two stage domains. Some of them are well established and have been used in this thesis work. A brief explanation of these methods is reported in the next sections (4.3.1 and 4.3.2)

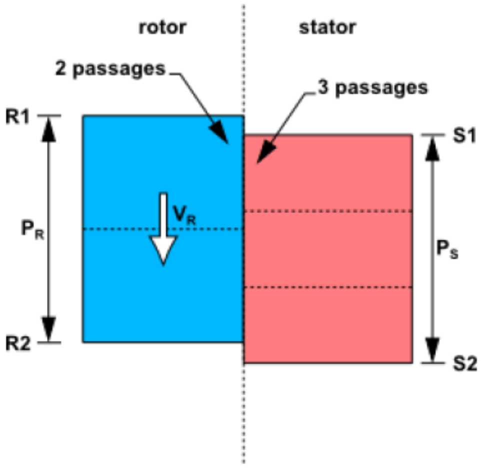


Figure 4.6. Use of 'classical' spatial periodicity to simulate equal sectors of rotor and stator, possible only when blade numbers are not mutual prime. [19]

4.3.1 Mixing-plane

Full wheels unsteady simulation can be avoided, thanks to modern computational techniques, but reducing the accuracy level. Since the unsteady effects (e.g. shock wave reflection between the stages, fluttering, fluid dynamic periodic disturbances) are not of interest during the intermediate design steps, the *mixing-plane* technique is widely adopted. It is based on a tangential average of the fluid dynamic quantities exiting from the first domain; this averaged values are used as inlet boundary condition for the second stage and so on for all the connected domain to be simulated. The *mixing-plane* approach is a powerful tool and has been extensively used in this thesis work, also for thin annular cascade simulation. This averaging is a type of mixing process and it assumes that the mixing out of the non-uniform flow occurs instantaneously at a 'mixing plane' rather than gradually through the downstream blade rows, but there is no guarantee that the loss generated by instantaneous mixing will be the same as that generated by the gradual mixing that occurs in practice [20]. Summarizing, advantages and disadvantages inherent in this method are listed in Table 4.1.

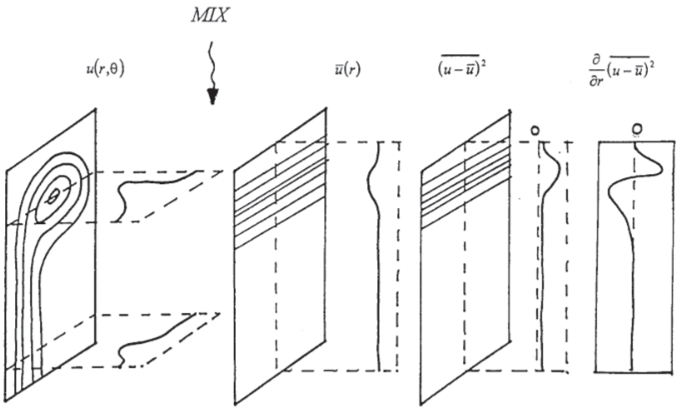

MIX
$u(r,\theta)$
$\overline{u}(r)$
$\overline{(u-\overline{u})^2}$
$\frac{\partial}{\partial r}\overline{(u-\overline{u})^2}$
o
o

Figure 4.7. three-dimensional mixing problems [20].

Table 4.1. Pros and Cons of mixing-plane technique

advantages	disadvantages
Steady formulation for both the stages	Fictional mixing losses introduced
Only periodicity in space is required so different stator and rotor pitches can be treated	Not realistic flow pictures if strong circumferential gradients exist at the inlet/outlet of a cascades
No information have to specified in-between the blade rows	Increase of computational time due to convergence at the interface

4.3.2 Transient Blade Row Models

A further step towards a complete full stage unsteady simulation is done using the *Transient Rotor Stator* (Ansys CFX terminology) frame change model. Transient turbomachinery flow calculations are computationally demanding, but faster calculation can be performed by making use of a reduced model of the turbomachine. This model reduction can be accomplished if pitch-change methods such as *Time Transformation* or *Fourier Transformation* are used. The reduction in size of computational domain reduces memory requirements (by a factor of 1/number of passages) and results in faster solution convergence to a steady-periodic state [19]. In practice it is recommended to simulate al least 5 to 10 periods, monitoring relevant quantities in specific domain location, to confirm that steady-periodic behaviour has been reached. With *Time Transformation* technique, the unequal pitch problem is overcome by applying a time transformation to the flow equations so that simple periodic boundary conditions can be used on pitch-wise boundaries. The idea is to reproduce a "phase shifted periodic boundary conditions". The basic principle is that the pitch-wise boundaries R1/R2 and S2/S2 (Fig. 4.8) are periodic to each other at different instances in time. For example the relative position of R1 and S1 at $t0$ is reproduced between sides R2 and S2 at an earlier time $t0$ - ΔT , where ΔT is defined by (P_R -P_S)/V_R. The *Time Transformation* method can be used to apply periodic conditions that are phase-shifted in time, also for boundary disturbance cases (*Frozen Gust Analysis*).

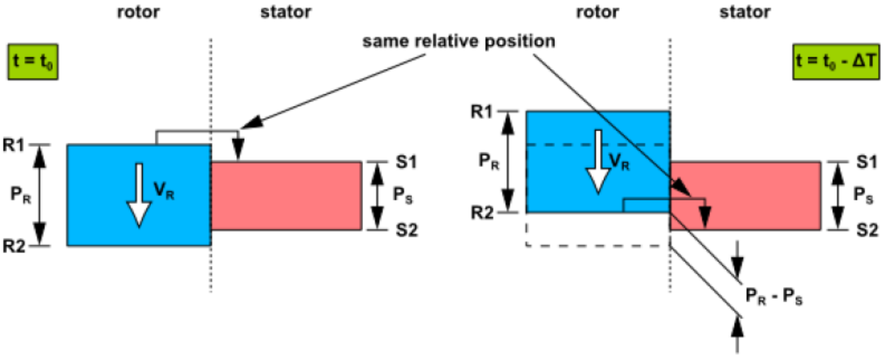


Figure 4.8. Phase Shifted Periodicity. P_R and P_S are the rotor and stator pitches, respectively, V_R is the rotor velocity [19].

Chapter 5

Research of optimal solidity by an automated optimization process

In chapter (4), the most important computational models developed for turbomachinery application have been introduced. In this thesis, not every model have been exploited for the turbine design. This is due mainly because supersonic turbine cascades are a pioneering technology to be designed from scratch. Nevertheless, the research activities in this thesis work make extensive use of simulation tools and their results, which are listed in this chapter and the following one, revealed important properties for the annular cascades. For sure, much more simulations will be required in the foreseeable future in this branch of research. Indeed, although the blade profiles can be well designed by means of the methods described in Chapter 2, the lack of experimental evidence and semi-empirical formulas for losses, incidence angle and so on, imply that an extensive campaign of blade-to-blade simulations is needed. Once useful results have been acquired, other simulation models can be implemented, starting with through-flow codes and 3D blade shape design. For blade-to-blade simulations, the parameter listed in Table 5.1 have been adopted:

Table 5.1. Simulation settings for rotor and stator blade CFD simulation

Settings (supersonic flow-regime option)	STATOR	ROTOR
Inlet Mach number	2	1.6088 (relative frame)
Static Pressure [bar]	1.9171	2.5390
Inlet Axial Speed [m/s]	1445.5	1238.2 (relative frame)
Static Temperature [K]	1300	1476
Inlet Turbulence Intensity	5%	5%
Blade Wall	Adiabatic and No-slip	Adiabatic and No-slip
Hub and Shroud Walls	Adiabatic and Free-slip	Adiabatic and Free-slip
Reference Diameter [cm]	74.73	74.73

5.1 Solidity effect on rounded blade profile

During the mean-line design of the turbine stage, the MOC code produced blade shape with a solidity not exactly related to a loading criteria. Indeed, Goldman approach was simply related to the supersonic, inviscid and irrotational flow analysis in Cartesian coordinates; the velocity distribution (see Fig. 2.16) along the blade wall is indirectly prescribed adopting the "vortex-flow" method, but this methodology does not take account of the performance detriment associated to the particular flow features generated by the rounded blade profiles. A more accurate flow-field prediction have been assured thanks to the application of Moeckel [12] detached bow shock wave prediction. Better results have been achieved extending that methodology with the incorporation of an original scheme, conceived during this thesis work and based on classical gas dynamic theory, that allows to guess position and strength of the first reflected shock wave. Actually, with the solidity, defined as $\sigma = chord/pitch$, given by the implicit-MOC code and the thickness assumed from published loss model for supersonic turbine [39], it was expected the presence of several other shock wave reflection. This is quite intuitive, if basic gas-dynamic notion are considered and, consequently, an analysis about the solidity effect on turbine performances is required. The blade to blade simulation campaign, which characterizes the second half of this thesis, starts exactly from these considerations. CFD analysis confirms the shock pattern (Fig. 5.1) supposed and partially simulated in the mean-line design. Of particular interest is the presence not only of shock reflections, but also of expansion waves and their interaction with BL (Fig. 5.2). Although a deeper analysis is required to fully understand how the supersonic flow-field interacts with transitional behaviour of boundary layer, some "classical" patterns are present. Where shock waves hit the blade profile, the BL thickens and the velocity defect region is visibly more extended, which can transform in a small bubble of separation for a more loaded blade. It can be also noticed how strong is the detached bow shock wave in front of the blade. This is the main source of loss for the turbine and every kind of machine aerodynamic optimization must face this phenomenon. In this simulation the performances are worsen by the numerous reflections along the blade channel and, of course, by the presence of a subsonic wake downstream the trailing edge. The entropy production Δs is equal to 50.04 $\frac{J}{kg\ K}$.

5.2 Loading Criteria

At this point a new value for the solidity is required, but in literature only very few relations are present to determine this value for axial turbines. These relations are the loading criteria, which typically limit the product between mass flow rate per passage times the tangential force on the blade to properly optimize the solidity value. Indeed, cascade profiles are subjected to at least two opposite requirements: minimization of wetted surface and minimization of BL separation risk. Fixing the blade chord value, the first requirement implies to adopt a low number of blades (i.e. low solidity value), whereas the second one points an opposite request for the blade numbers.

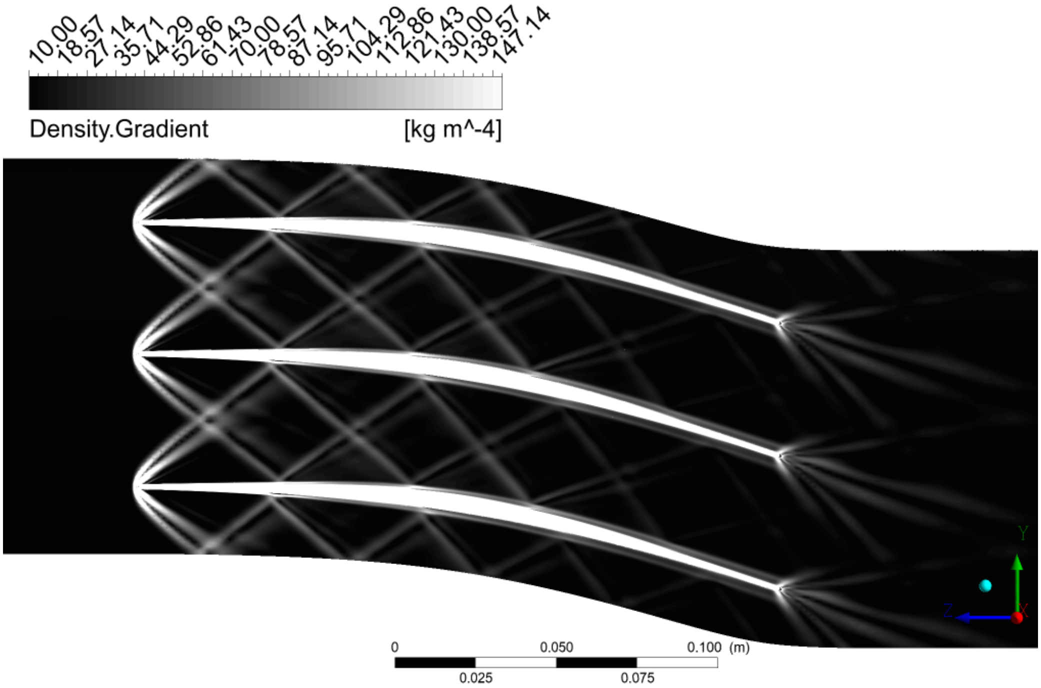


Figure 5.1. Density gradient contours in arbitrary linear scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with solidity imposed by the Method of Characteristics

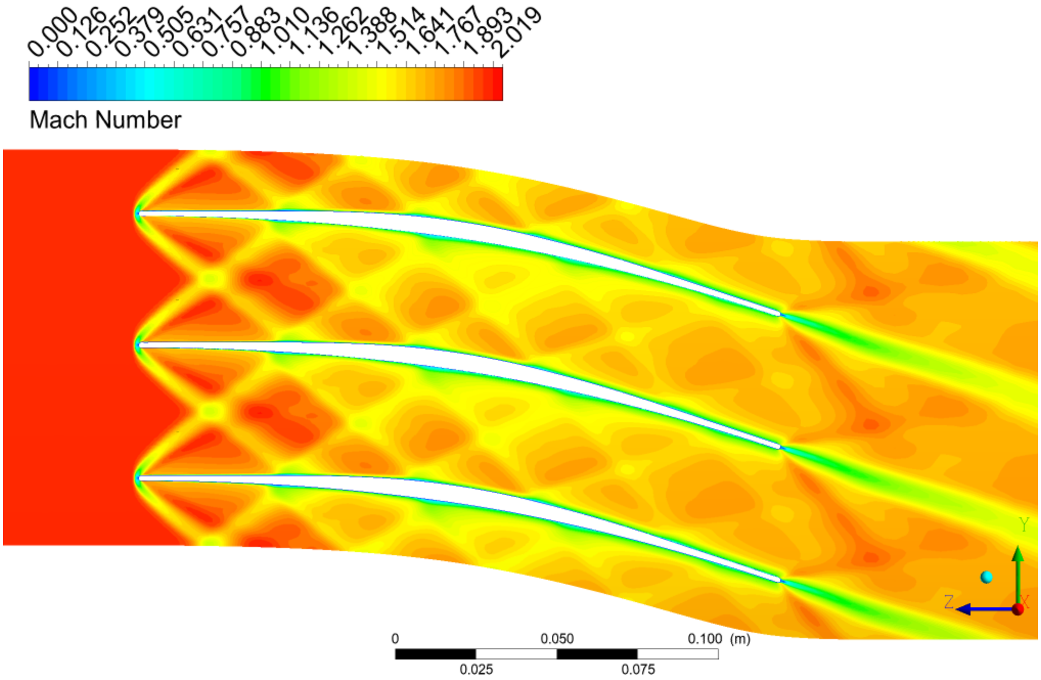


Figure 5.2. Mach contours of supersonic stator blade with solidity imposed by the Method of Characteristics

A trade-off must be find and semi-empirical models have been formulated and experimentally validated for every kind of turbomachine. Nevertheless, axial turbines are not so sensitive to the necessity of increase the number of the blades in order to avoid separation, since in these applications the pressure gradient is not adverse.

5.2.1 Zweifel criterium

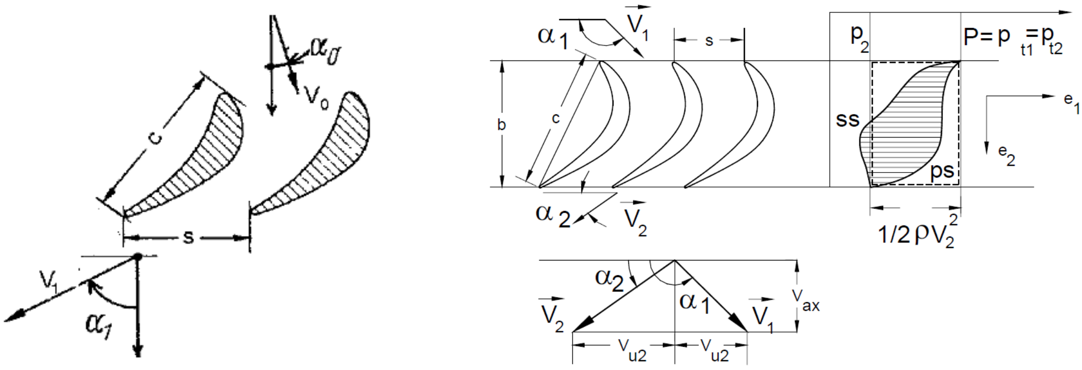
Two relevant criteria are available in literature (detailed explanations are present in [58] or in the original reports):

1. Ainley-Mathieson criterium. It looks for the minimum losses based on experimental measurement.

2. Zweifel criterium. Adopted in the mean-line code. It limits the tangential force on the blade. Zweifel introduced an ideal tangential blade force, which is based on an ideal pressure distribution shown in Fig. 5.3b. This ideal pressure distribution has the maximum total pressure over the entire pressure side. On the suction side, the pressure falls instantaneously to p_2 being the lowest pressure possible without pressure rise. This ideal pressure, which of course cannot be realized, gives a tangential force per unit of length.

$$C_{Ft} = \frac{F_t}{\frac{1}{2} b \rho V_1^2} = \frac{2}{\sigma_{ax}} \cos^2 \alpha_1 \left(\frac{V_{0m}}{V_{1m}} \tan \alpha_0 - \tan \alpha_1 \right) \tag{5.1}$$

where C_{ft} stands for "tangential force coefficient", $\sigma_{ax} = b_x/s$ is the axial solidity (i.e. ratio between the axial chord projection and the pitch), α are the flow angle with Anglo-American convention (Fig. 5.3a) and V_m are the axial speed components.



(a) *Anglo-American convention for flow angles*

(b) *Real and ideal pressure distribution around a turbine blade [59]*

The recommended value for the tangential force coefficient C_{Ft} found in literature is 0.8 or in the range of $0.6 \div 1.2$, if a wider and more modern sample of subsonic axial turbine is considered. The C_{Ft} value obtained with the solidity coming from the MOC is equal to 0.135 (i.e. 56 blades)for the stator. This discrepancy with respect to the usual turbomachines is an important signal, which represent how different are flow-field and forces in case of a supersonic blade passages. The blades, in fact, are less loaded (so less subject to BL separation) because the aim of the implicit-MOC code is only to carefully lead the flow particles into the blade passage with a smooth change of direction.

The choice to use C_{Ft} as driving parameter for the optimization process of row performances is not mandatory, but reveals to be particularly adequate. Its meaning is not simply related only to blade solidity but it takes in account also the flow deflection, through the flow angle in their frame of reference, and with a minor impact to the change in the axial velocity components. These last two quantities are involved in almost every step of the mean-line code, including the imposition of a reaction degree, the Kantrowitz starting limit and the MOC code itself. Moreover, it must be noted that the original MOC profile is sharp, according to the isentropic expansion theory (expansion and compression characteristic lines), while the real one have been rounded according to a suggested parameter from [39] of *pitch/th* ratio equal to 25 (*th* stands for *thickness* and it is referred to profile thickness at leading and trailing edge).

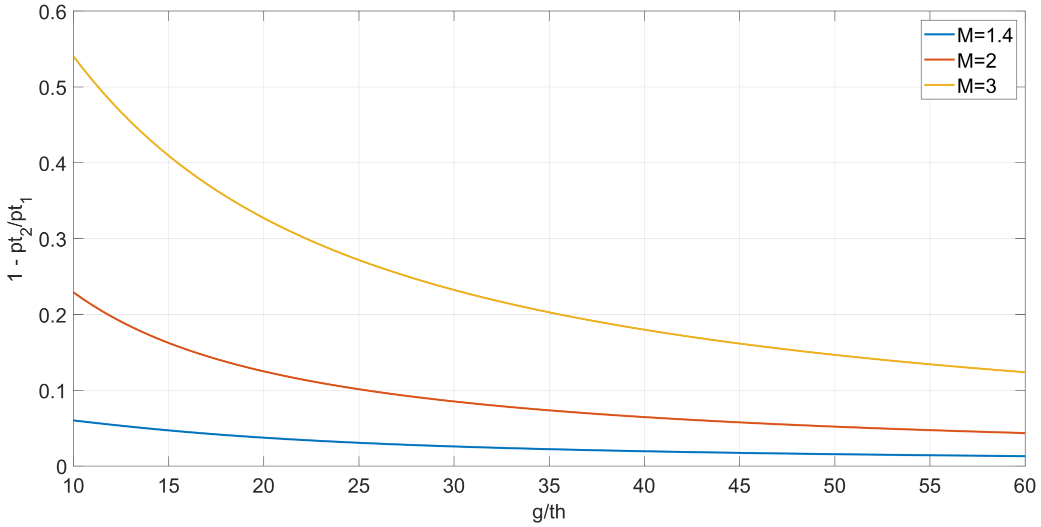


Figure 5.4. Evolution of the pressure losses with pitch to leading edge thickness from mean-line code parametric analysis.

5.2.2 Upper limit for C_{Ft}

Meanwhile the lower limit of C_{Ft} value have been identified as equal to the one indirectly prescribed by the MOC, there is the necessity to select an upper limit, based on fluid dynamic considerations. During the initial design phase and thanks to early blade to blade simulations, shock pattern and outlet flow angle variability have been investigated. It immediately became clear that the turbine blades should not be too far apart, since the flow at domain exit could present low adherence to the prescribed outlet angle. In other words, blades are not able to impose the correct direction to the flow if the solidity is too low (i.e. too high value for C_{Ft}). It is important to remember that the flow leaving the nozzle positioned downstream the rotating detonating combustor is not uniform in terms of pressure, temperature and direction, with around 5 degrees of oscillation around the mean value; the stator has the critical task to deliver the most uniform flow possible at outlet, which is fundamental for the rotor to operate properly and extract work from the fluid. Moreover, the shock pattern in this situation could produce strong disturbances into the intra-row gap by directly impinging the downstream rotor blades or affecting the **subsonic wake**, an already problematic region to deal with. For this reason, the mean-line code portion dedicated to the modelling of bow shock and its reflection is not thought to face shock waves which exit the blade passage without any intersection with the profile curves. The upper limit accepted for the stator C_{Ft} is 0.3 (i.e. 26 blades)(Fig. 5.5 and 5.6). In both cases shock-BL interactions are relevant. Indeed, the shock reflected on the blade pressure side impinges the rear section of the lower blade, where the flow field is already delicate to manage. In Fig. 5.7, and, to a lesser extent, in Fig. 5.5, is present a large flow region of undesired over-speed (red zone in Mach contours plot) and, immediately downstream, there is a flow separation zone caused by shock incidence and consequent flow-field adjustments in terms of expansion waves and secondary shocks.

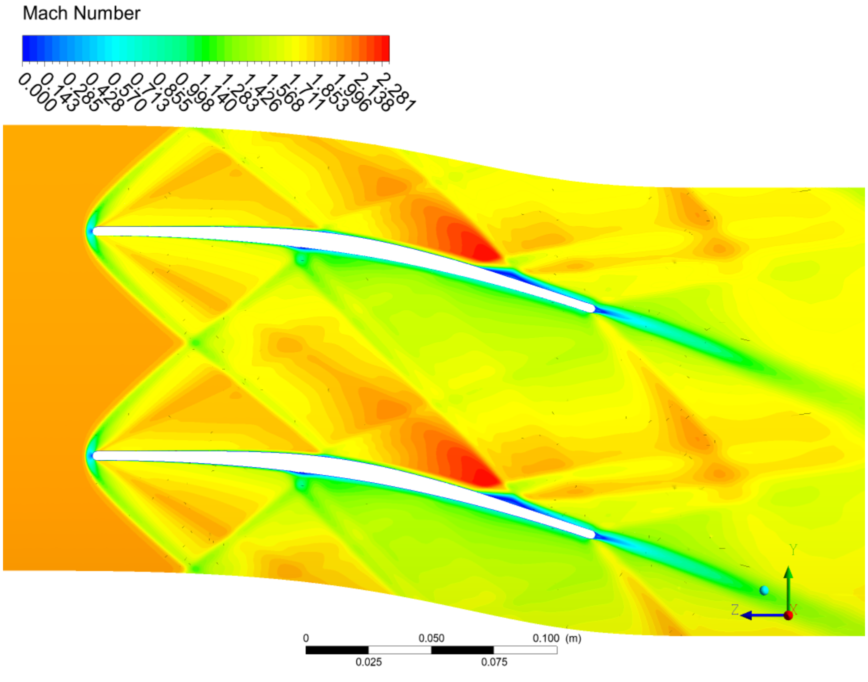


Figure 5.5. Mach contours of supersonic stator blade with solidity imposed by the mean-line code $C_{Ft} = 0.3$; 26 blades

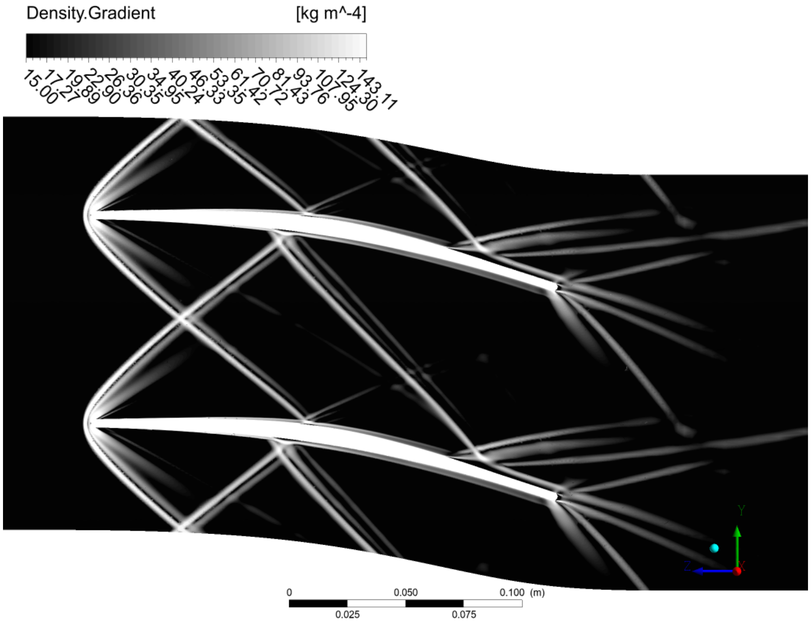


Figure 5.6. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with maximum solidity accepted by the mean-line code $C_{Ft} = 0.3$; 26 blades

For the sake of completeness, results coming from a simulation with $C_{Ft} = 0.44$ (i.e. 17 blades) are also reported (Fig. 5.7 and 5.8).

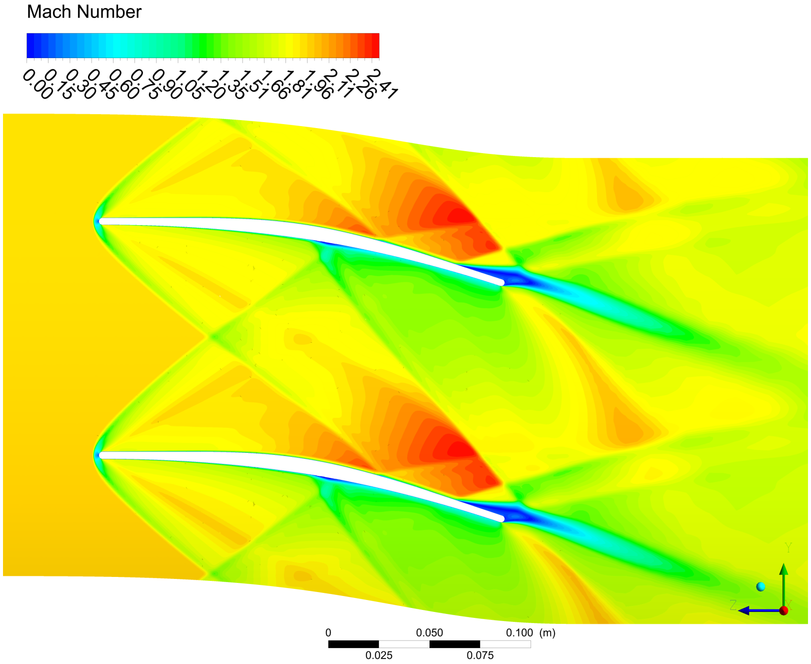


Figure 5.7. Mach contours of supersonic stator blade with solidity imposed by the mean-line code $C_{Ft} = 0.44$; 17 blades

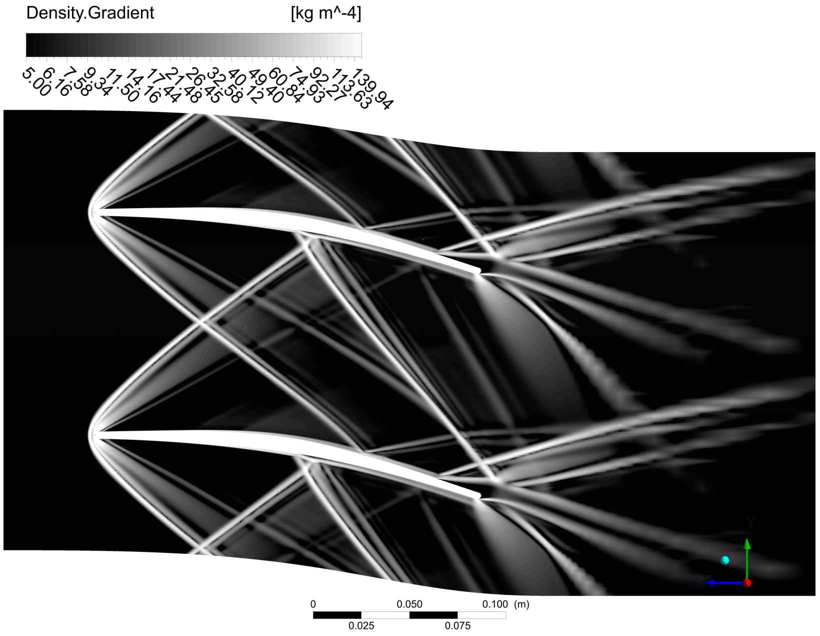


Figure 5.8. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with maximum solidity accepted by the mean-line code $C_{Ft} = 0.44$; 17 blades

5.3 In-depth description of the automated optimization process

For a given blade geometry with given inlet and exit flow angles, a distinct pitch/chord ratio at which the profile losses are minimum exists [59]. Starting from this consideration by Schobeiri, a series of preliminary CFD simulations with a varying Zweifel's tangential force coefficient C_{ft} were performed and it was observed a clear minimum in the search interval, which has been "artificially" restricted due to the fluid dynamic considerations described in the previous section (5.2). Hence, it can be assumed that the blade profile performance, evaluated in term of entropy production across the row or, equivalently, in term of total pressure loss, is a unimodal function of C_{Ft}. A function $f(x)$ is said to be unimodal in a given interval $[x_a; x_b]$ if, given a value x_m, it is monotonically decreasing for $x \leq x_m$ and monotonically increasing for $x \geq x_m$. For this function, the minimum value is $f(x_m)$ and there is no other local minimum (see Fig. 5.9). With unimodal function any extreme found is guaranteed to be the global extreme.

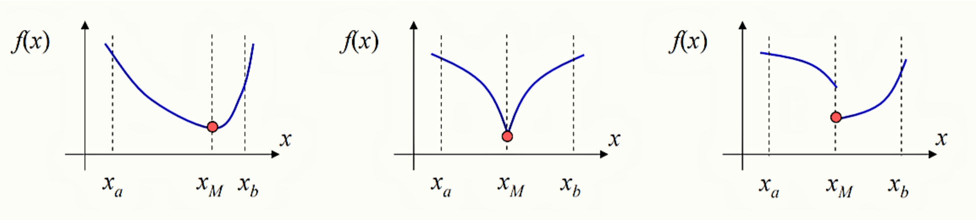


Figure 5.9. Examples of unimodal function

Once the existence of a minimum is confirmed, there are mainly two different ways to find it:

- calculate the entropy production for different values of C_{ft}, interpolate the data with a curve and then find its minimum;

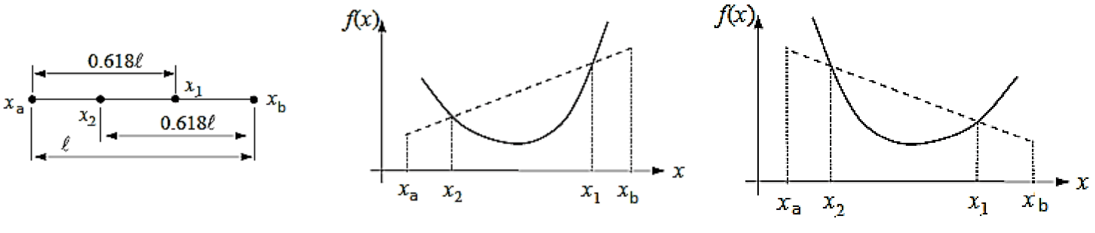
- employ an optimization algorithm, which is able to catch the minimum.

The first option is easier, but the main drawback is that the final result is sensible to the choice of the interpolating nodes and of the interpolating function; therefore the second methodology was selected for our purpose. The optimization algorithm chosen is the golden section search method, which has a simple, robust logic and converges towards a local minimum without the need of an interpolating function or to calculate derivatives. A weak point of this method is that it converges towards a local minimum, but within the selected interval, as already explained before, the trend of the entropy generated has only one minimum; therefore, the local minimum is also the absolute minimum in the interval.
In the golden section search method, the design domain is iteratively divided into sub-domains in proportion to the golden ratio $\phi = 1.618$. As shown in Figure 5.10a, the first point is identified as $x_1 = x_a + 0.618(x_b - x_a)$, while the second point is $x_2 = x_b - 0.618(x_b - x_a)$. The method works as follows:

1. If $f(x_1) > f(x_2)$, then the minimum is between x_a and x_1, and the region to the right of x_1 is excluded from the search, as illustrated in Figure 5.10b. We assign $x_b = x_1$ to continue the search

2. If $f(x_1) < f(x_2)$, then the minimum is between x_2 and x_b, the region to the left of x_2 is excluded from the search, as illustrated in Figure 5.10c. We assign $x_a = x_2$ to continue the search.

3. If $|f(x_2) - f(x_1)| < \epsilon$, then we found the optimal point at $x_m = x_b$ or x_a.



(a) (b) (c)

Figure 5.10. Illustration of the golden section search method [21].

Actually, the golden-section search method takes a lot of function evaluations to find an optimal solution. When the number of design variables increases, the computation time required for performing function evaluations is substantial, making these methods less desirable [21]. However, for our purpose where only one variable is considered

(Zweifel tangential force coeffcient C_{ft}), this method reveals to be easy to implement and acceptably time consuming.

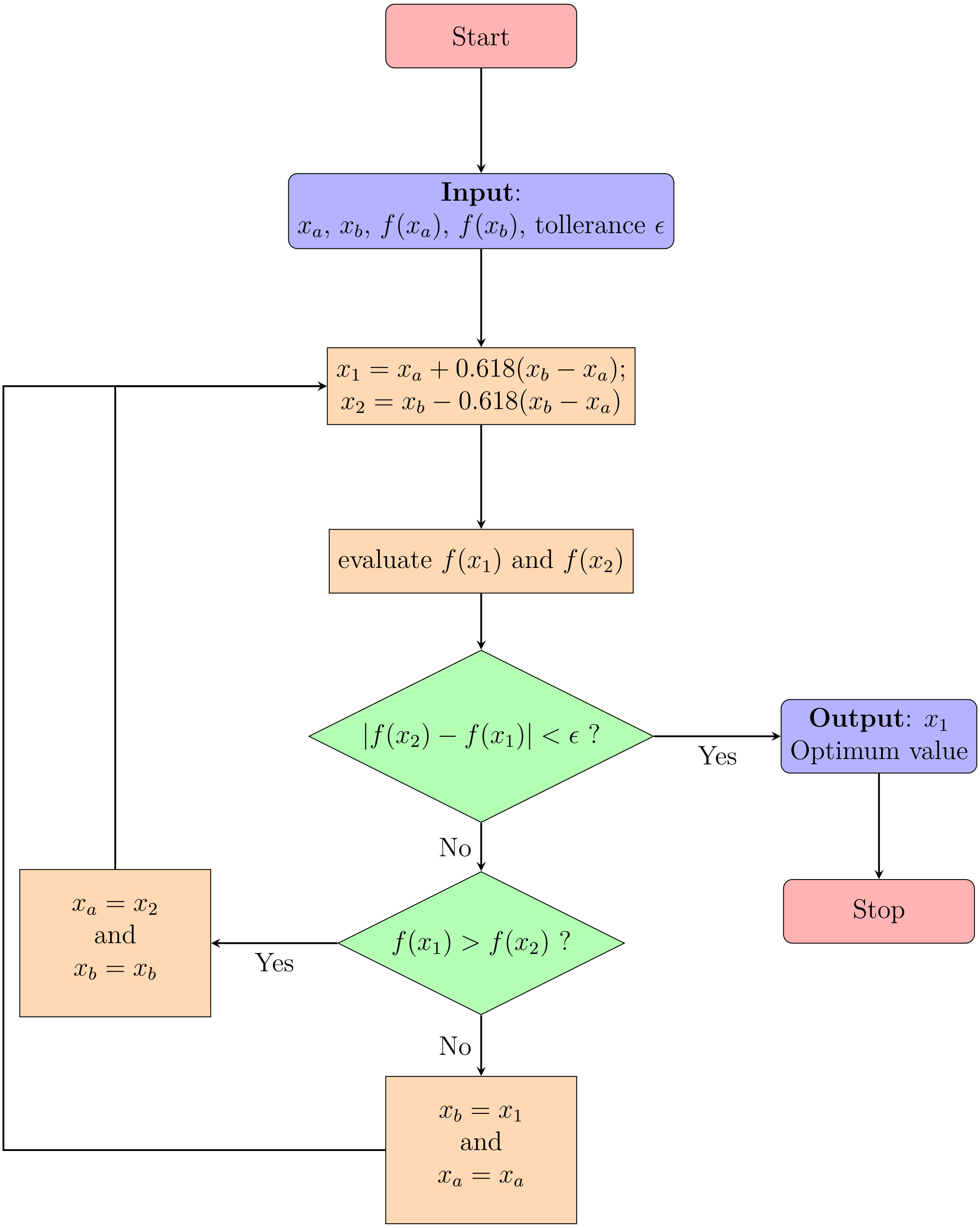

Start
Input:
x_a, x_b, $f(x_a)$, $f(x_b)$, tollerance ϵ
$x_1 = x_a + 0.618(x_b - x_a)$;
$x_2 = x_b - 0.618(x_b - x_a)$
evaluate $f(x_1)$ and $f(x_2)$
$|f(x_2) - f(x_1)| < \epsilon$?
Yes
Output: x_1
Optimum value
Stop
No
$f(x_1) > f(x_2)$?
Yes
$x_a = x_2$
and
$x_b = x_b$
No
$x_b = x_1$
and
$x_a = x_a$

Figure 5.11. Flowchart of the golden-section search method. Note that the function $f(x)$ corresponds to the entropy production Δs evaluated by means of CFD simulations as the difference between mass-flow averaged entropy at domain outlet and the entropy at the inlet.

In the previous lines, the optimization algorithm was explained using the expression $f(x)$: x is the value of Zweifel's tangential force coefficient C_{ft}, while with $f()$ is intended the whole process that starts from the mean-line code to generate the blade profile, sets-up the CFD simulation, extracts the required data for post-processing and finally calculates the entropy generated in the supersonic blade row. Doing all these operations manually is less than ideal, because the process is prone to human error and it is really time consuming from a user point of view; on the other side a fully automatized code programming is definitely more complicated, but has much more potential: the entire procedure requires less time (around 15 hours on a average laptop), but most importantly the design parameters and simulation settings can be changed easily and the procedure re-launched without causing a huge time loss for the user.

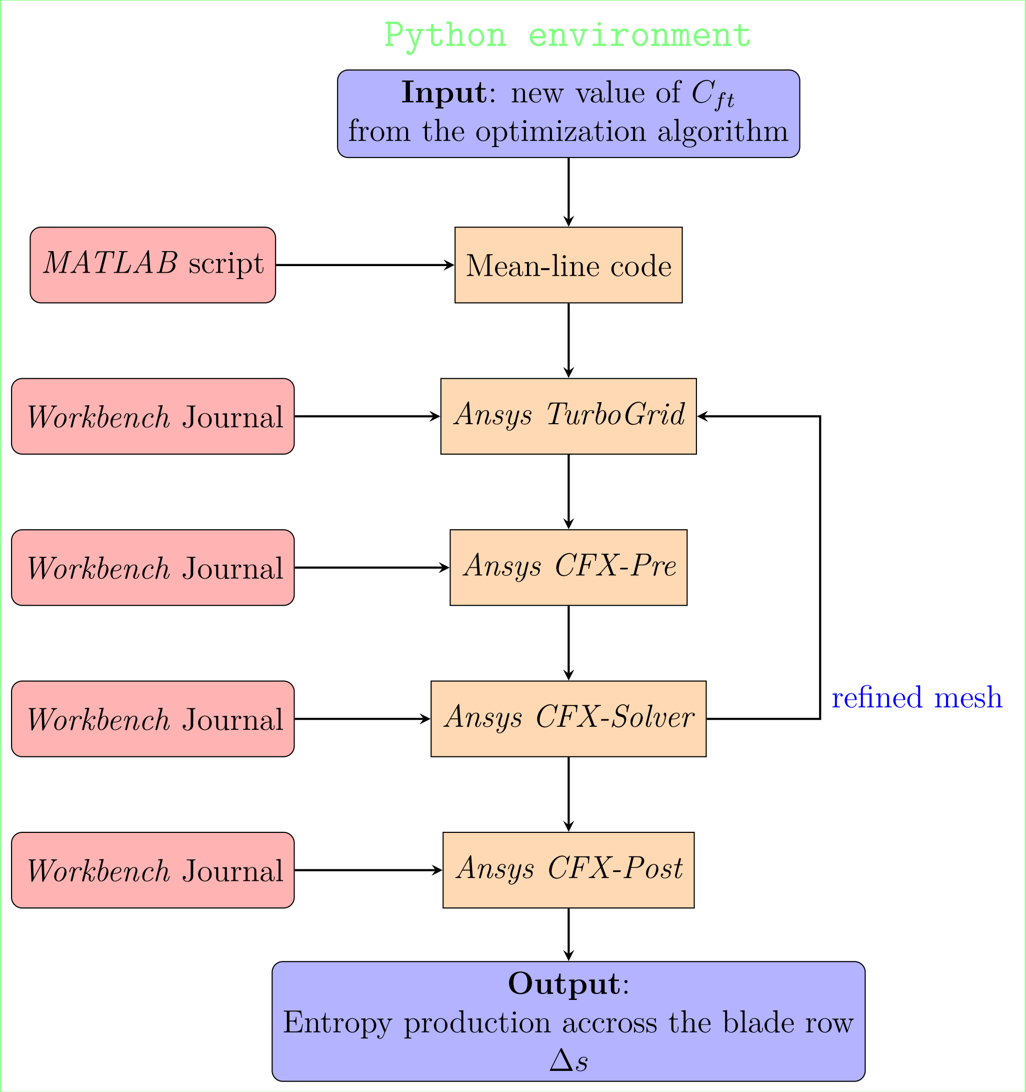


Figure 5.12. Flowchart to illustrate the procedure used to calculate the entropy production for each value of C_{ft} given by the optimization algorithm.

While all the operations required to generate the blade profile are already available

from the mean-line code written in MATLAB, it is necessary to prepare all the instructions to automatically mesh the domain, set the simulation, run the simulation, post-process it and the extract the main required data. Ansys Workbench has this really useful feature for scripting, called record Journal, which allows to register all the operations performed in the program. During the creation of the journal, all the operations must be carefully taught to employ the most general approach, because all the instructions must be valid not only for that specific case, but for each simulation that is required by the optimization procedure. Obviously the instructions cannot be exactly the same for all the simulations, because some quantities are dependent on the C_{ft}, like the number of blades or the profile shape, and they must be changed in the recording for the associated case.
Finally when the automation of all the parts have been completed, a Python code is written to put all the pieces together and this can be considered as the brain of the procedure: Python gives at each iteration the new value of C_{ft} through the optimization algorithm, it launches all the different programs with their associated scripts or journals, it modifies the recording with the values associated to that case and finally stores all the relevant results. In the following list, it is reported the order in which the various programs are called by Python to calculate the entropy production for each value of C_{ft} (Fig. 5.12):

1. *MATLAB*: the first step is to run the mean-line code (for more details see chapter 2) with the new value of C_{ft}; notice that it is fundamental to run the mean-line code each time, because the search of the optimal value of Zweifel's tangential force coefficient is done while maintaining all the other dimensionless numbers exactly the same; for example, just changing the number of blades in a row using the same profile, which is much easier to implement, is an incorrect approach because some of the dimensionless numbers are changing, especially the ratio pitch to leading edge thickness on which the losses are strongly dependent. Output are the coordinate files (.crv) for the profile and the number of blades for the following step. The blade profile curves are provided in Cartesian coordinate, but first it is needed to transform the planar blade passage obtained with the MOC into an annular passage. This topological transformation is described in Appendix D.

2. *Ansys TurboGrid*: it loads the coordinate files and read the number of blades (i.e. number of passage in a single stator/rotor row) and creates the mesh. Actually the meshing of the domain and the next two steps regarding the simulation are performed twice for each case: the first mesh is really coarse and it is used to initialize the entire domain in fast manner and then a significantly more refined mesh (100k cells) is employed to actually simulate the behaviour of the supersonic row and to extract the required data. This strategy has been really successful to reduce the computational cost and this was really important considering that all the simulations were performed on an average laptop. Note that settings for the mesh are fixed in term of BL refinement and number of node for passage.

3. *Ansys CFX-Pre*: it sets the simulation (previously defined thermodynamic parameter and computational models) and load the mesh with relative information

about boundary conditions.

4. *Ansys CFX-Solver*: it runs the simulation adopting a parallel computing approach with the available CPU cores.

5. *Ansys CFX-Post*: it elaborates the simulation results and extracts several information for the post processing. The most important are the mass-flow averaged entropy at domain outlet and the mass-flow averaged entropy at the domain inlet, computing the difference Δs between this two quantities. Δs is the output.

Saving the results has been an essential part in the code development, because it is necessary to find the right compromise to not completely fill the memory; for each of the two cases tested at each iteration, the code creates a dedicated folder in which the main data and results extracted in the post-processing are stored. Finally in the main results folder, text files are created to report a summary of the data associated to the optimization algorithm (convergence error, the extent of the search interval at each iteration, C_{ft} values and associated entropy production, number of iterations, ...).

5.4 Optimal Solution

Outcomes of the optimization procedure are reported hereafter for stator and rotor rows. C_{Ft} values of interest for the stator are included in the interval [0.135; 0.3], and the optimal solution have been found to be $C_{Ft} = 0.226$ with 33 blades (Fig. 5.13). For the rotor row the interval is [0.233; 0.4] and the optimal solution have been found to be $C_{Ft} = 0.245$ with 32 blades (Fig. 5.14). The mesh independence analysis and the y+ calculation is reported in Appendix C.
Luckily, the blade numbers are mutual prime and this is a minimum requirement for subsequent rows in order to avoid undesirable fluid-structure interaction. This expedient allows at least to avoid that the blades could vibrate and oscillate synchronously.

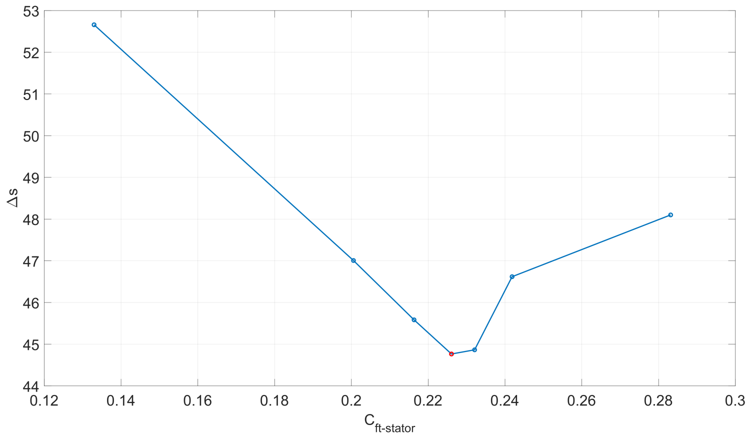


Figure 5.13. Research of optimal solidity to obtain the minimum entropy increase across stator.

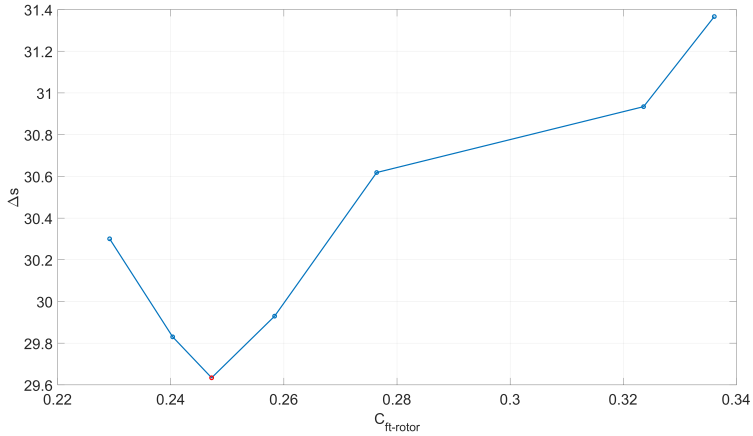


Figure 5.14. Research of optimal solidity to obtain the minimum entropy increase across rotor.

5.4.1 Stator Cascade

With the optimum value of C_{Ft} for the stator, the entropy variation Δs is limited to 44.504 [J kg^{-1} K^{-1}] and the total pressure loss coefficient, if evaluated with respect to the inlet kinetic energy, is $Y = \dfrac{P^0_{inlet} - P^0_{outlet}}{P^0_{inlet} - P_{inlet}} = 15.84\%$. Bow shock wave is reduced in extension and number of reflections (see Fig. 5.15), but still retaining a good capability of impose a deviation to the flow-field, contrasting its inertia. Shocks interact with the BL in position where the flow is not problematic (see Fig. 5.16). Indeed, separation bubble are not present and the usual BL thickening in the interaction zone is modest. Subsonic pockets are present only in the wake region and in front of the leading edge. In the latter zone, the flow passes throughout a section of the bow shock strong enough to be considered normal and, consequently, to concentrate an important amount of total pressure losses.

While deflecting the flow, blades produce an increase of static pressure and static temperature up to, respectively, 2.634 [bar] and 1491.4 [K] at the outlet. The Mach number, instead, decreases to 1.69. All these quantities are mass-flow averaged at the outlet section. Giving this values it is clear that the blades act as a convergent supersonic diffuser, whereas a subsonic classical turbine can be idealized as a convergent subsonic nozzle.

By way of comparison, the blade loading chart is reported in Fig. 5.17, considering the three main simulations analysed in this chapter. Here, it is possible to have a look to almost every phenomenon described up to now from a different point of view. For sure, this blade loading chart is very different with respect to the classical turbine ones, but important dissimilarities are visible also among the three simulations.

In the first part of the profile, all blades are not loaded, as imposed by the 0D code, but the extension of this zone increases with C_{Ft} (i.e. decreasing the number of blades). Where the profile begins to deflect the flow, the loading progressively increases. The difference can be explained considering that cascades with high solidity are characterized by more gradual pressure variation due to the mutual interaction between pressure-side and suction-side pressure field. In the chart, are clearly visible also the positions where shocks interact with BL, by looking at the peaks of pressure curves.

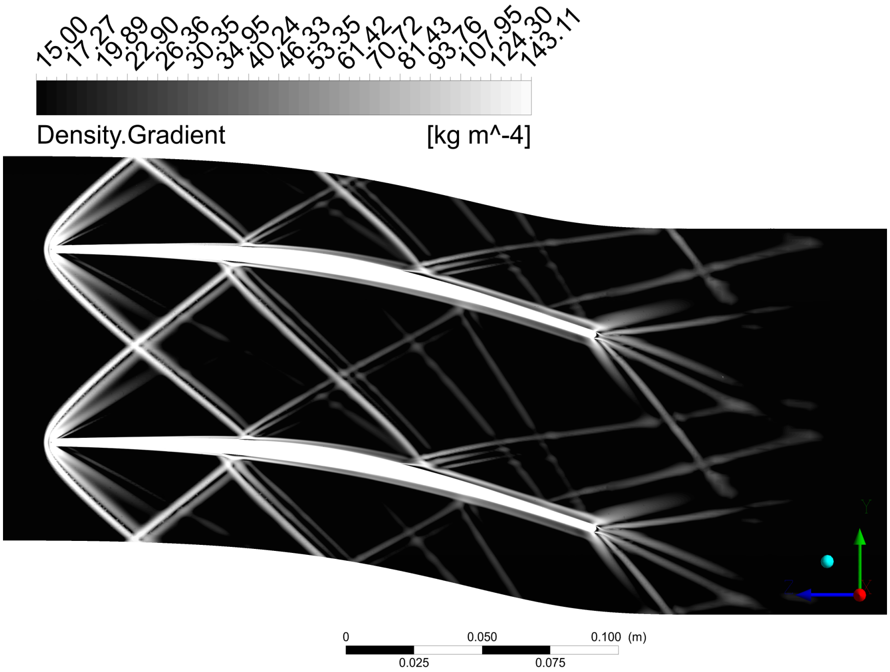


Figure 5.15. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) of supersonic stator blade with optimum solidity $C_{Ft} = 0.226$, 33 blades

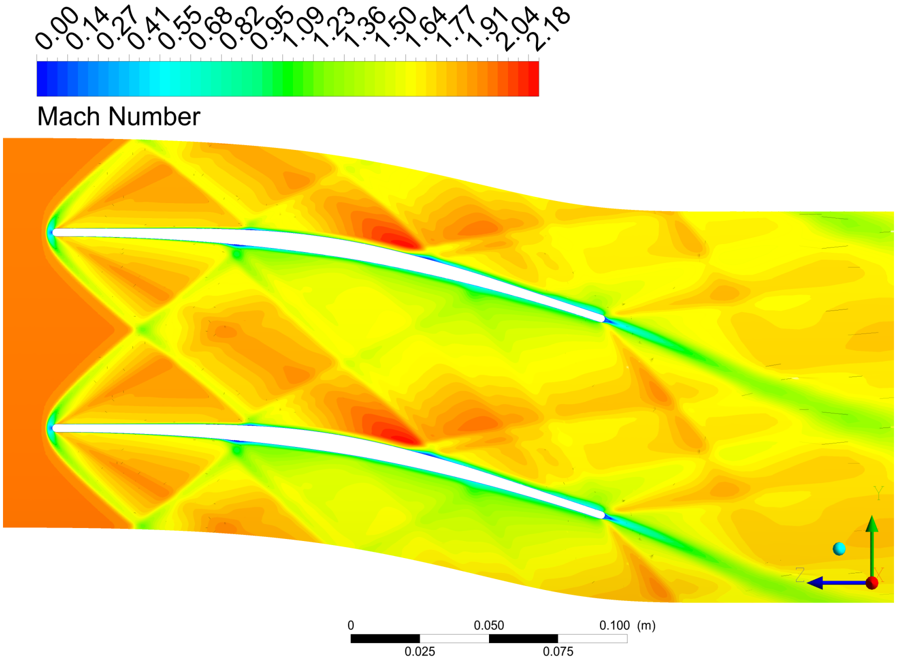


Figure 5.16. Mach number contours of supersonic stator blade with optimum solidity $C_{Ft} = 0.226$, 33 blades

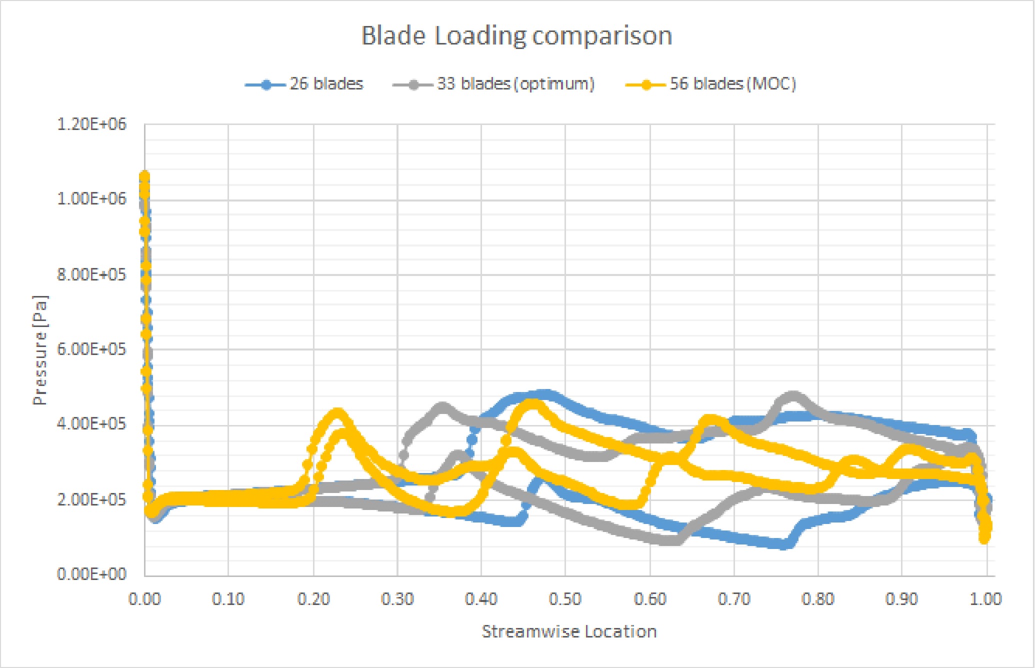


Figure 5.17. Blade loading chart of supersonic stator blade with various C_{Ft} (number of blades)

5.4.2 Rotor Cascade

With the optimum value of C_{Ft} for the rotor, the entropy variation Δs is limited to 29.634 $[\mathrm{J\,kg^{-1}\,K^{-1}}]$. The total pressure loss coefficient is $Y = \dfrac{P^0_{inlet} - P^0_{outlet}}{P^0_{inlet} - P_{inlet}} = 13.35\%$. An aspect to note is that, while the optimization code for the stator led to a reduction of entropy production Δs compared to the starting MOC-based profile of 25%, the advantages for the rotor are of minor importance, leading to only a 5% Δs reduction. The main reason this happens is to be found in the shock wave lattice (see Fig. 5.18). The inlet rotor Mach number is lower than the stator one so, the losses associated with the detached bow shock wave are less relevant if compared to the other loss sources in the rotor passage. Indeed, the other effect of having a lower inlet Mach number is that the bow shock is characterized by a higher slope. Whatever is the number of blades among those evaluated by the Golden-section search method, the bow shock is continuously reflected much more times than in the stator channel. For this reason, despite the low total pressure losses caused by the bow shock, there are no advantages for the overall rotor row performances, because of its reflections, whose strength and positions play an important role in the entropy production.

In Fig. 5.19, it can be seen that the flow-field is particularly regular, there are no regions of flow separation or incipient separation and the subsonic wake zone is confined. Moreover, the outlet mass-flow average Mach number is equal to 1.262 in the stationary frame. This low value allow one more degree of freedom for the design of the hypothetical following stage: force the flow to become subsonic with a convergent-divergent nozzle shape or proceed with one more supersonic stage.

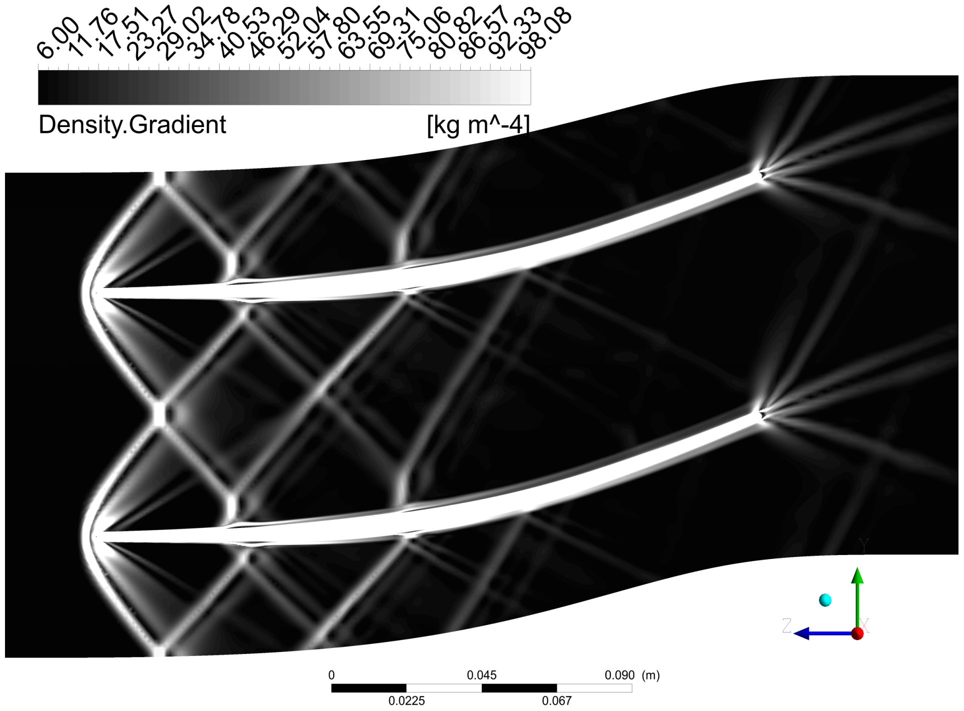


Figure 5.18. Density gradient contours in arbitrary linear scale (i.e. numerical-Schlieren visualization) of supersonic rotor blade with optimum solidity $C_{Ft} = 0.245$, 32 blades. Relative frame of reference

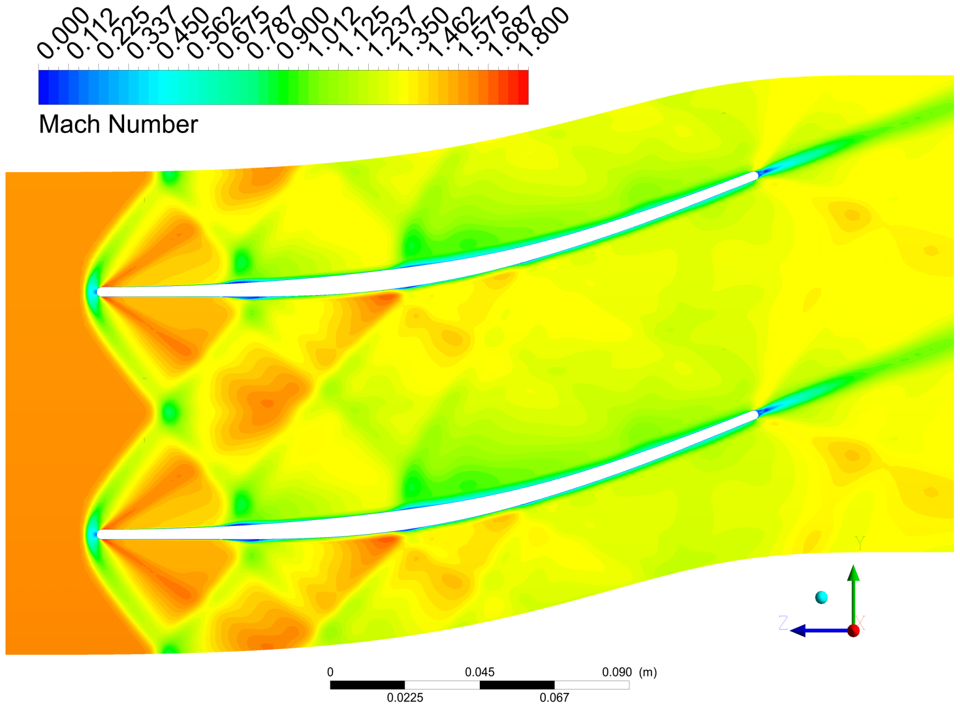


Figure 5.19. Mach number contours of supersonic rotor blade with optimum solidity $C_{Ft} = 0.245$, 32 blades. Relative frame of reference

5.5 Outlet flow angle analysis

Aim of the optimization procedure was to find the blade number, for both stator and rotor, that maximizes row performances. Results of CFD simulations in term of entropy production and total pressure loss have been compared with 0D prediction model (2.6), with particular interest on the shock pattern prediction capability of the mean-line code. Another fluid dynamic quantity of considerable importance that deserves an in-depth analysis is the outlet flow-angle. Whatever is the mean-line code accuracy and despite the effort to improve its prediction capability, it is still a reduced order model with respect to the CFD. For this reason, since the relevant turbomachinery parameters are treated as lumped ones, it is not possible to extract information about the flow-field non-uniformities from the mean-line code. This is the case of stator outlet flow angle. The entire machine architecture and, in particular, the blade profiles are designed to deal with precise flow angle to achieve certain power, efficiency and torque. If, for some reason, discrepancy and variability of flow properties are too relevant, the mean-line code loses accuracy and reliability. In case of outlet flow angle, it can be subjected to excessive variability due to residual shock and expansion waves and can suffer of discrepancy with respect to 0D value if the deviation angle (i.e. difference between flow angle and blade metal angle) is too large (flow not well-oriented and blade too loaded).

In this section a detailed analysis of outlet flow angles at stator domain exit is proposed.

Blades	17	26	33	56
α_2 average	20.75	21.82	21.77	18.36
α_2 standard deviation	1.8805	1.9442	0.9136	0.3438
α_2 peak-to-peak amplitude	6.01	5.92	2.80	0.99
α_2 mass-flow average	20.73	21.70	21.80	18.34
α_2 0D code prediction	19.55	19.97	19.87	19.97

Table 5.2. Comparison of outlet flow angle: CFD simulation vs mean-line code results

In the "56 blades" row ($C_{Ft} = 0.135$), the last column of Table 5.2, the CFD simulation results show the best value for the outlet flow angle. The mean-line code prediction of α_2 is closer to the value simulated in Ansys CFX, both considering the *line-averaged* α_2 and the mass-flow weighted average α_2 values. This should not be surprising since 56 blades are exactly the number of blades obtained with the MOC, so in this situation the flow is better forced through the passage. Unfortunately, as explained in section 5.1, this configuration is not acceptable in term of entropy production, due to the numerous shock reflections along the channel. Also the standard deviation value is coherent with the phenomenology just described; in fact, it is the lowest value achieved.

For what concern the other three simulations here reported for comparison purpose (see Fig. 5.20 and Fig. 5.21), the "17 blades" ($C_{Ft} = 0.44$) one gives results that are the worse in term of standard deviation or, in other words, the flow angle at stator domain outlet presents a too high irregularity with such a low number of blades.

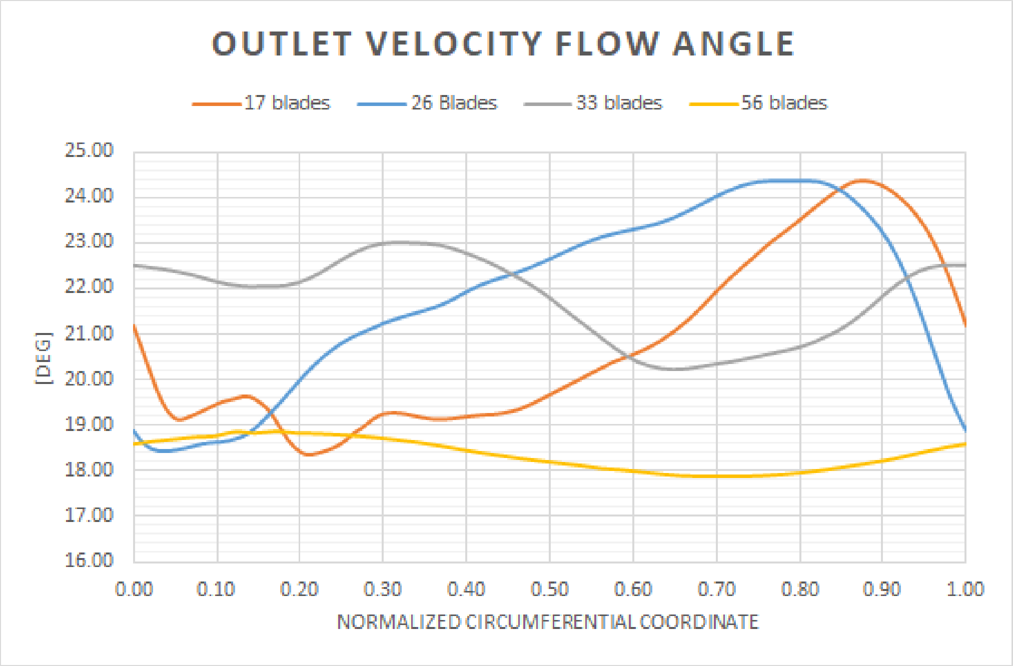


Figure 5.20. Plot of outlet velocity flow angle. To compare the curves extracted from four different CFD simulations, the curvilinear (circumferential) coordinate have been normalized with respect to the pitch dimension for every stator configuration reported here.

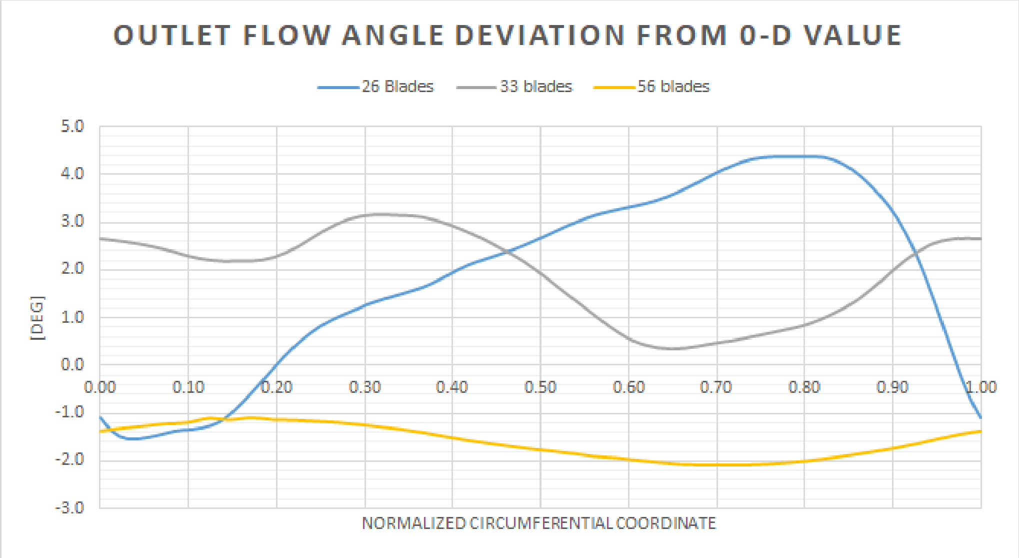


Figure 5.21. Plot of deviation angle. To compare the curves extracted from four different CFD simulations, the curvilinear (circumferential) coordinate have been normalized with respect to the pitch dimension for every stator configuration reported here.

The standard deviation value is also too high for the "26 blades" ($C_{Ft} = 0.3$) configuration and it is more than double the value of the "33 blades" ($C_{Ft} = 0.226$) configuration. The peak-to-peak value reduction with the increasing of the blades number confirms the conclusions reached so far. In both simulations, the deviation angle is roughly equal to 2 degree, but the relevant fact is that, implicitly, a good trade-off between flow outlet angle variability (standard deviation parameter) and efficiency have been found without using a multi-objective optimization algorithm. This is, from a fluid dynamic point of view, directly related to the shock pattern and to the right and left running characteristics position, whose choice implies the amount of losses and the non-uniformity of outlet flow quantities.

Chapter 6

2D turbine stage simulation

The computational methods to analyse the flow in cascades presented in Chapter 4 are useful design tools. Blade-to-blade simulation is the first step to achieve a full understanding of the 3D flow-field behaviour in a turbomachine. In this chapter, the analysis starts considering the design condition of the supersonic turbine. Stator-rotor coupling is approached with the mixing-plane technique and then by adopting a "phase shifted periodic boundary conditions", which allows to capture unsteady flow behaviour in the rotor row. The latter methodology is based on the *Time Transformation* pitch-change method (theoretical background are exposed in section 4.3.2). In the second half of the chapter, results from simulations in off-design condition are analysed. In those sections, the variations of the stator inlet velocity angle and, later, of the inlet Mach number are considered. In both cases simulations are performed first only on the stator annular cascade domain and then on the entire stage, with the mixing-plane technique.

6.1 Full stage simulation in design condition

After the value of C_{Ft} have been chosen with the golden-section search method, single rows have been simulated by imposing as boundary conditions the values coming from the mean-line code (see Tab. 6.1). The following step to assess the supersonic turbine performance was to simulate the coupled stages by means of the mixing-plane technique (theoretical background have been briefly reported in paragraph 4.3.1). The results coming out from this simulation turned out to be consistent with the mean-line code performance prediction, with a certain level of approximation. Indeed, the reported results in Tab. 6.1 and in Tab. 6.2 are promising for the future development of the supersonic turbine and its integration with a RDE, but some approximations have been introduced into the simulation procedure. First of all, the blade-to-blade simulations described in paragraph 5.4.2 and the ones reported hereafter do not take care of the height increase of the rotor blades. To assess the influence also of this parameter a full 3D mixing-plane simulation is required. This is the main reason why some flow-field parameters show deviation with respected to the 0D results. Losses calculated in this simulation are respectively $\Delta s = 45.29$ [J Kg^{-}1 K^{-}1] for the stator row and $\Delta s = 26.97$ [J Kg^{-}1 K^{-}1] for the rotor row. It must be noted that the entropy production value in the stator row, if coupled with the rotor domain, is in line with the value evaluated simulating the stator alone. Some small discrepancies are present in the coupled rotor row, whose quantities of interest are different with respect to the rotor stand-alone simulations. Losses are lower in the mixing-plane approach since the Mach number at the rotor inlet, coming from the numerically mixed stator outlet, is a bit lower with respect to the one predicted by the 0D code, so the shock waves are less strong (contours are reported in Fig. 6.1, 6.2, 6.3 and 6.4). Anyway, lack of accuracy is present in both the mean-line code and in the mixing-plane simulation, since no end-wall effect have been included. To deal with this source of losses is easy in a 3D CFD simulation, but it could not be the case for the mean-line code. A full comparison between the 0D code results and the blade-to-blade CFD simulation outcomes is carried out in section 2.6.

The performance parameters (see Tab. 6.2) of the machine are promising for the development of this pioneering turbine class, since there are several design parameters and optimization chance that can lead to a decrease of the entropy production.

Table 6.1. Relevant quantities, mass averaged, at stator inlet and rotor outlet.

Quantity	Inlet	Outlet	Ratio (Out/In)
Static Temperature	1300 K	1667.95 K	1.283
Total Temperature	2340.1 K	2170.11 K	0.927
Static Pressure	1.91856 [bar]	356720 [bar]	1.859
Total Pressure	15 [bar]	8.97 [bar]	0.598
Static Enthalpy	1006.26 [kJ kg$^-$1]	1375.82 [kJ kg$^-$1]	1.367
Total Enthalpy	2050.94 [kJ kg$^-$1]	1880.2 [kJ kg$^-$1]	0.917

Table 6.2. Results quantities, mass averaged, at stator inlet and rotor outlet. Note that the 0D code provides a value of $\eta_{tt} = 56.4$ and of 172 [kJ kg$^-$1] for the specific (Eulerian) work. Discrepancies with respect to CFD simulation are due to the lack of a deviation angles model.

	Results
Torque (single rotor blade)	124.184 [Nm]
Torque (all blades)	3973.89 [Nm]
Specific work	171.5 [kJ kg$^-$1]
Dimensionless mass flow	0.0284926
Power Coefficient	0.00517448
Total-to-total isentropic. efficiency η_{tt}	54.85%

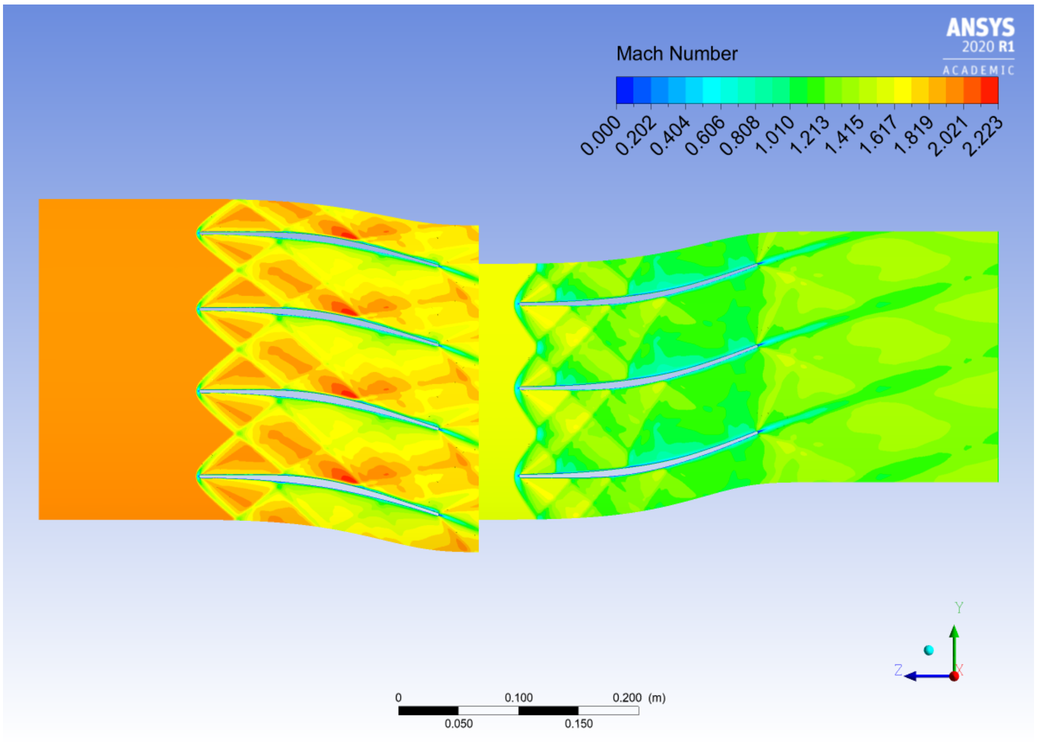


Figure 6.1. Mach number contour in design condition. Relative Mach number reported for the rotor domain

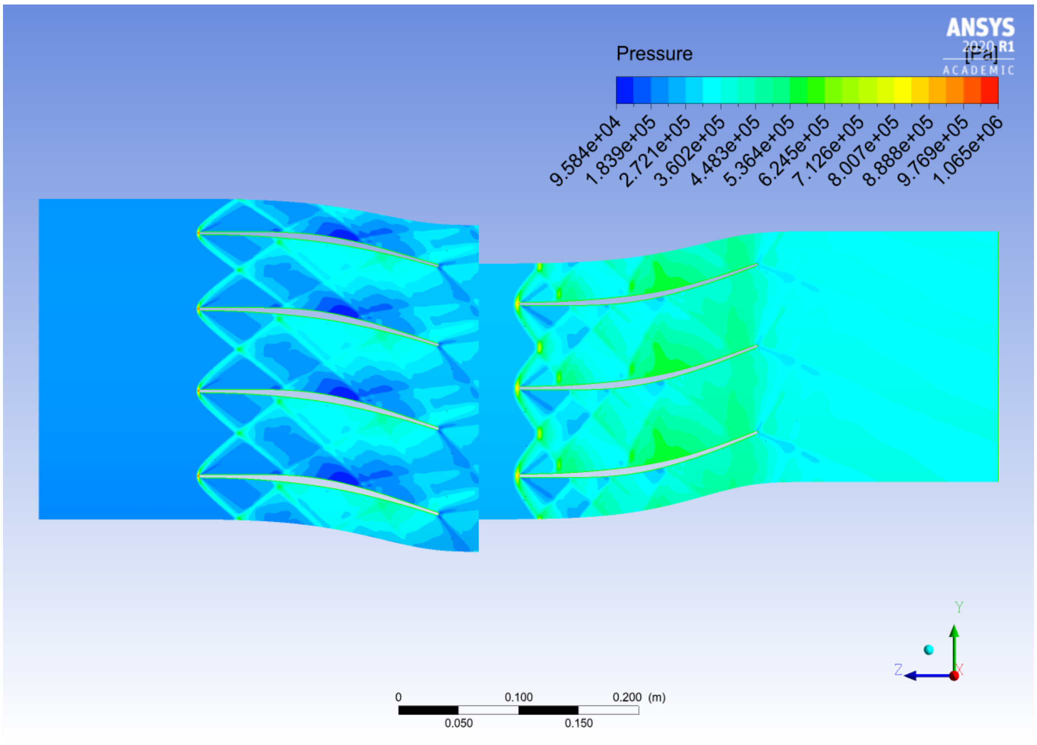


Figure 6.2. Static pressure contour in design condition

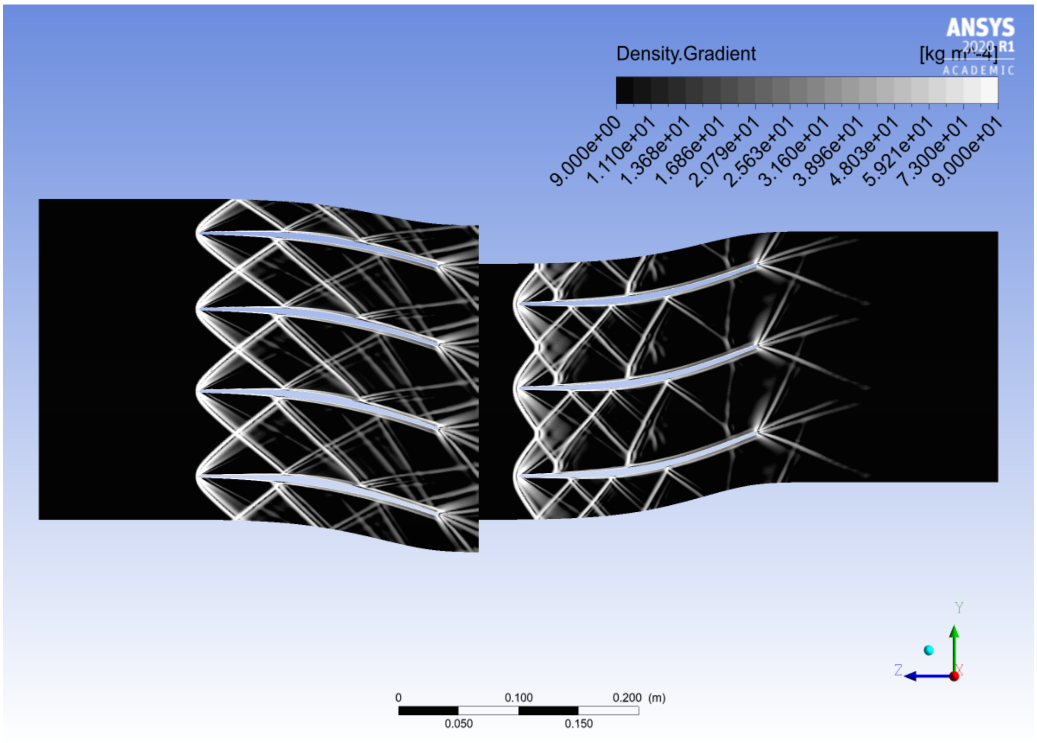


Figure 6.3. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization) in design condition

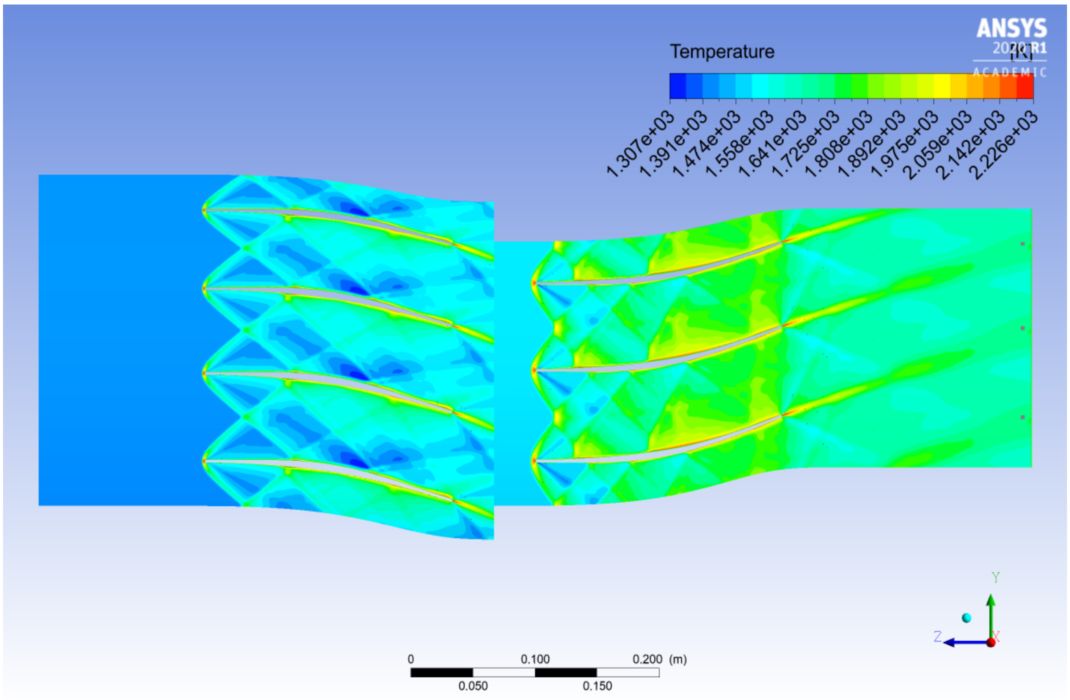


Figure 6.4. Static temperature contour in design condition

6.2 Frozen Gust Analysis

The goal of this approach is to set up a transient blade row calculation to model an inlet disturbance (frozen gust) using the Ansys CFX *Time Transformation* model [19]. The basic idea is to consider fluid dynamic values at the outlet stator domain as input for an unsteady simulation of the rotor, evaluating the fluid-flow at each time step only in rotor domain. The difference with respect a mixing-plane approach is, first of all, in the fact that those outlet values are not circumferential averaged, so the rotor is subjected to non uniform time-dependent inlet boundary conditions. The simplification inside this procedure consists in this domain reduction that allow a saving in term of storage and computational resources. When a *Frozen Gust Analysis* is performed, the upstream row, the stator one in this case, is not influenced by the downstream flow-field variations and for this reason can be said to be *frozen*. From the rigorous mathematical point of view, this setting is based on a wrong assumption. Indeed, the URANS system of equations is always parabolic, whatever is the flow regime, and for this reason small disturbances in the flow-field can travel upstream in space. Nevertheless, the lower computation effort can make the *Frozen Gust* approach feasible for a coarse optimization of the blade, based on the transient phenomena that can be captured. Moreover, the fluid regime in this application is almost everywhere supersonic (the only exception is the wake region) and the capability of internal small disturbances to travel upstream is greatly reduced. It is intuitive that, considering the unsteadiness in a turbomachine simulation, the results in term of efficiency, entropy production and total pressure loss are lower if compared with the ones simulated with steady-state setting. Nevertheless, here the most interesting parameters are those related to the outlet flow angle average value and its standard deviation (Fig. 6.5). Understanding the physics beyond and assessing the stage performance in terms of flow angle quantities is very important for the RDE applications, since the latter produce a flow that is inherently periodic, unsteady and not uniform. Results of the flow angle analysis are reported below in Tab. 6.3 and in Fig. 6.5. Monitor points have been positioned close to the periodic boundary in correspondence of inlet domain section (subscript *in*), leading edge (subscript *LE*), trailing edge (subscript *TE*) and outlet domain section (subscript *out*).

Table 6.3. Relevant quantities, mass averaged, at stator inlet and rotor outlet.

	β_{in} [deg]	β_{LE} [deg]	β_{TE} [deg]	β_{out} [deg]
mean value	-1.80	-1.50	19.20	18.92
standard deviation	0.06483	0.49133	0.02810	0.01359
peak.to-peak amplitude	0.2605	1.8865	0.1352	0.0458

Considering that the mean-line code designed a value of 0 [deg] for the relative speed flow angle at the leading edge and of 20.38 [deg] for the relative speed flow angle at trailing edge, the simulation reports the presence of both a negative incidence angle and a negative deviation angle. To solve this problem of discrepancy, a better prediction model for incidence and deviation should be included into the 0D code, based on experiments or a more extensive simulation campaign of MOC-based supersonic blades. Nevertheless, the most important result achieved with this simulation is the assessment of the row capability to damp inlet oscillations. The standard deviation values and the peak-to-peak values confirm that this happens not only due to the mixing phenomenon downstream of the blade, but already in the blade passage due to the good design of characteristic line positions. This damping capacity can be observed for every quantity of interest evaluated at inlet and outlet surfaces.

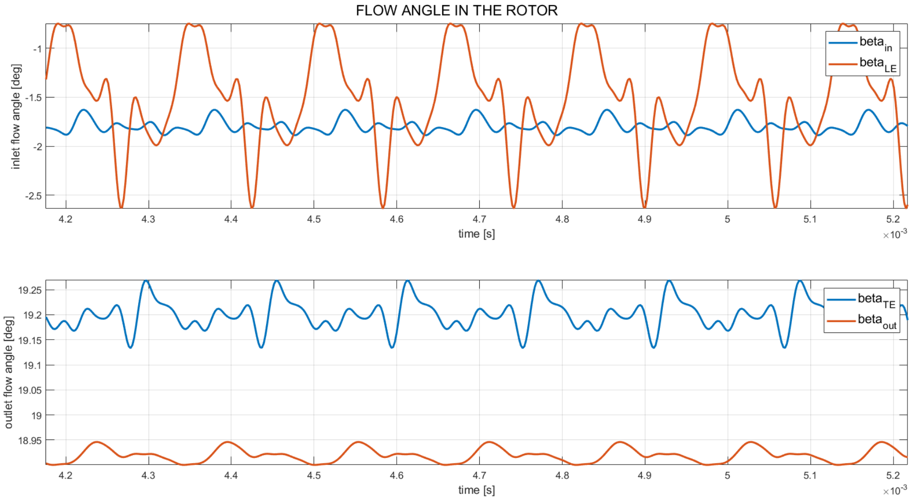


Figure 6.5. Plot of velocity flow angle for rotor inlet and outlet vs time

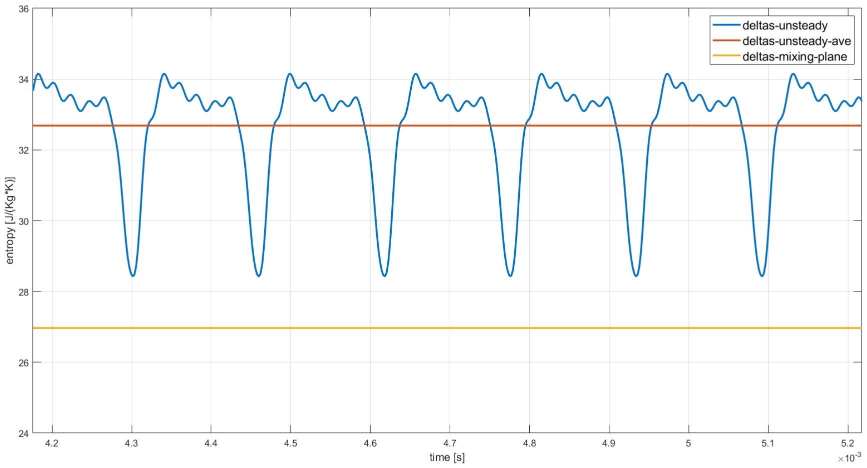


Figure 6.6. Plot of entropy increase throughout the rotor passage vs time

For what concern the entropy production throughout the blade passage, the oscillatory behaviour is evident and a small increase of this value is present, as expected, when compared to the one obtained by means of the mixing-plane technique. The negative peaks in Fig. 6.6 shows the moment in which the rotor cascade is in the best position with respect to the stator row (Fig. 6.7), since the flow-field is much similar to the one for which the blades are designed. In Fig. 6.8, instead, the situation is opposite

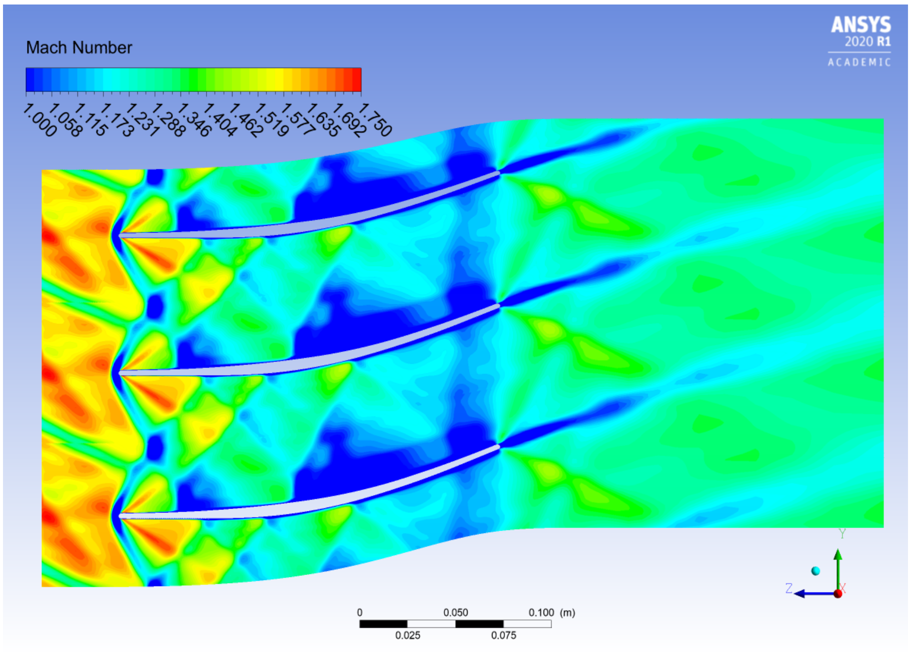


Figure 6.7. Mach number contours at 3.7e-4 [s]. Arbitrary scale: low limit fixed to 1

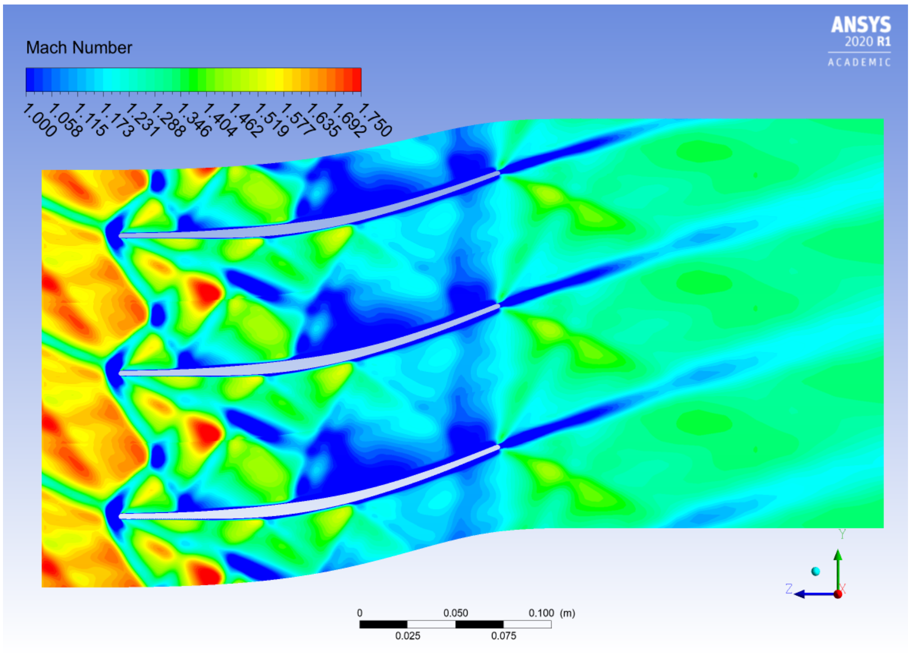


Figure 6.8. Mach number contours at 7.8e-4 [s]. Wake, became again supersonic, hits the rotor leading edge. Arbitrary scale: low limit fixed to 1. Note the presence of larger subsonic zone behind the shock waves.

and it is the worst configuration. When the wake, although being again supersonic, hits the leading edge of the rotor blade, the shock pattern become more problematic. The wake is, by definition, a region of velocity defect and so a low Mach number flow is encountered by the rotor blade. Lower Mach numbers are associated with lower oblique shock angles, but, in this situation, the shock is detached and the effect is an enlargement of the subsonic zone in front of the leading edge. A problem arises because, in this zone, the total pressure losses are the higher possible and only the wake absorption can mitigate these effects. It follows that the inter-row distance is a very important parameter to be optimized, much more than in classical subsonic axial turbines.

6.3 Off-design performances

At this point the off-design performances of the stator row and, than, of the full stage later remains to be investigated. The two fluid dynamic parameters chosen to be varied are the inlet flow angle and the inlet Mach number. This choice is motivated not only by the need to understand how a supersonic turbomachine works in a wide range of different inlet conditions, but also because the RDE, for which the turbine stage have been designed, present high variability of these parameters. Further investigation should be performed also for inlet pressure and temperature variations.

6.3.1 Full stage: inlet flow angle variation

For this set of off-design simulation, every inlet boundary conditions have been fixed but the inlet angle. Since Ansys CFX deals with annular cascade, the information about inlet angle have been provided to the solver by defining the axial and circumferential component of velocity. The angle have been varied between -16 [deg] to +16 [deg] with a spacing of 4 degrees, leading to 8 off-design simulations. The effect of these settings is a reduction of the mass flow rate (Tab. 6.4) and, consequently, the turbine unstarting for extreme value of the inlet angle. Actually, only the simulation with inlet angle equal to -16 [deg] did not converge (i.e. value of residuals too high), because of the stator blade passage unstarting, despite every effort made to deal with the presence of a normal shock too close to the mixing-plane interface.

Table 6.4. *annular* mass flow rate passing through the thin layer which, in Ansys CFX, describes the blade-to-blade passage

α_{in} [deg]	-12	-8	-4	0	4	8	12	16
mass flow rate [kg/s]	0.8273	0.8376	0.8438	0.8458	0.8438	0.8376	0.8273	0.8131

In Fig. 6.9, the values of entropy increase are reported for the full turbine stage and for only the rotor blades. Looking at those values, what can be noticed is that the losses curve is not "symmetric" with respect to the on-design condition. It appears that the supersonic stator blades are able to provide good performances with a positive incidence angle (further discussion in paragraph 6.3.2). As can be expected at this point of the discussion, the rotor blade are not so sensitive to the variation of the

stator inlet flow angle, mainly because the stator itself have been demonstrated to possess good capability to deviate the flow. For what concern the sensitivity of β_2 (i.e rotor outlet relative flow angle) to the stator inlet flow angle variations, good results have been achieved (Fig. 6.10). Standard deviation values are very low and this confirms again the low sensitivity of the machine, at least to the flow angle disturbances. Also the averaged values of β_2 are in a small range of deviation from the reference value at 0 [deg] or from the mean-line code designed value.
By way of example, Mach number contours and numerical-Schlieren visualization are reported below, resulted from the -12 [deg] simulation (Fig. 6.11 and Fig. 6.12) and the +12 [deg] simulation (Fig. 6.13 and Fig. 6.14).

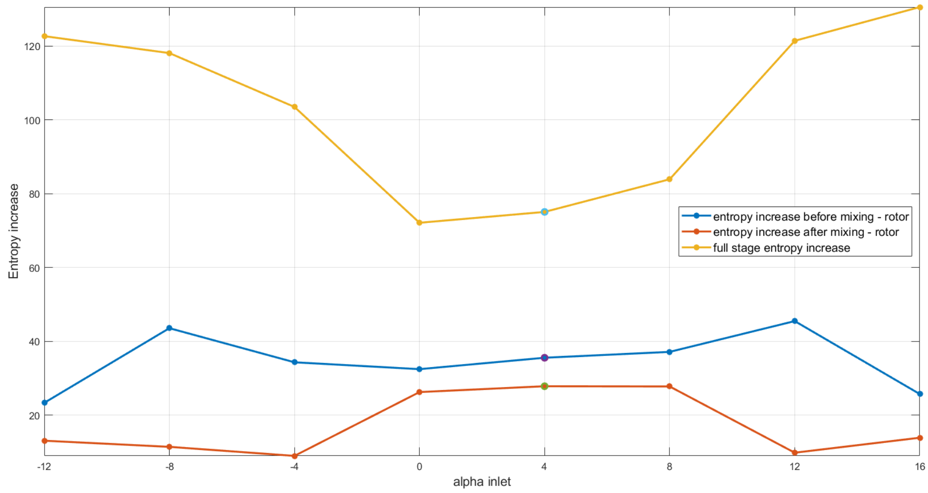


Figure 6.9. Plot of entropy increase vs inlet flow angle

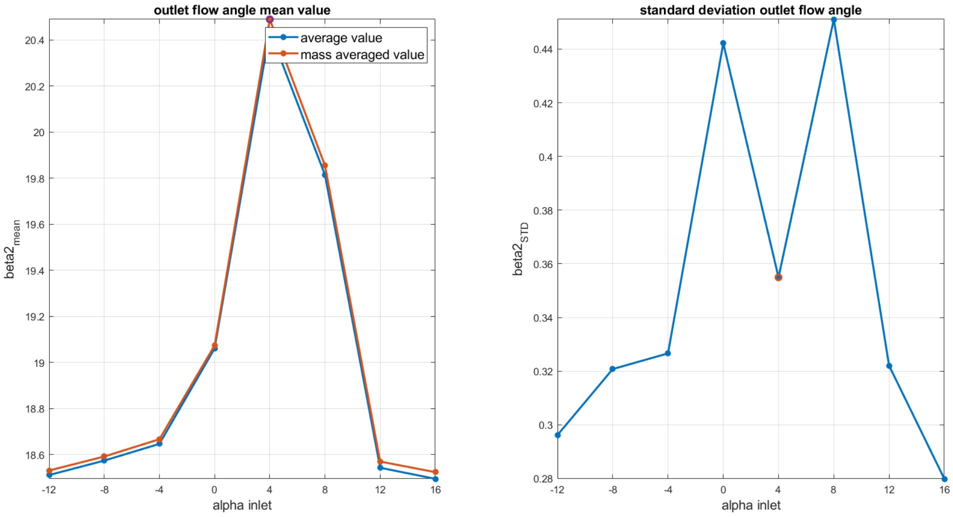


Figure 6.10. Outlet flow angle averaged value vs inlet angle (left); Outlet flow angle standard deviation vs inlet angle (left)

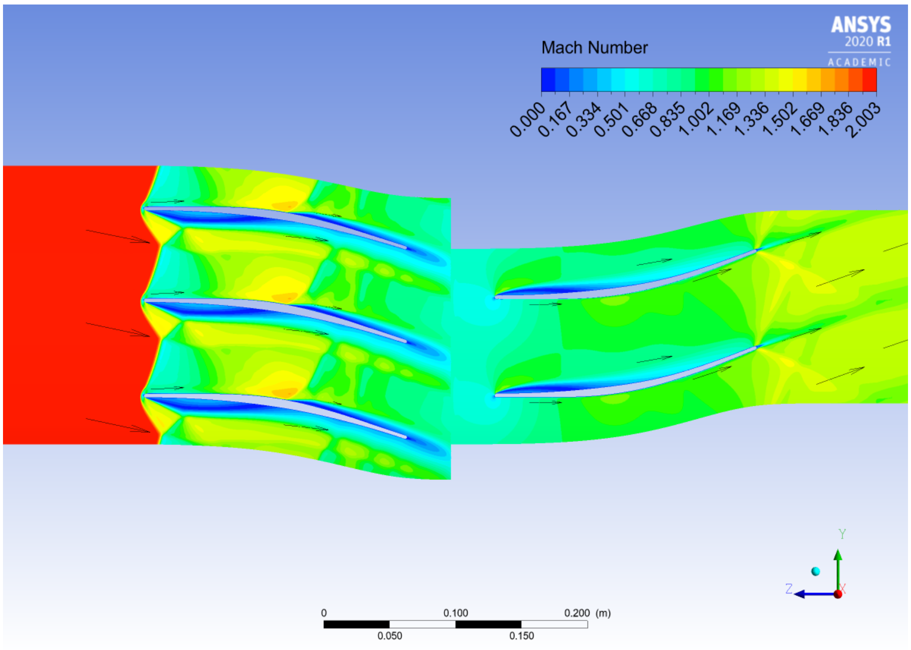


Figure 6.11. Mach number contours. Inlet incidence angle: -12 [deg]. $\eta_{tt} = 18.86\%$. Subsonic pocket downstream the bow shock, close to the BL, in wake region and at the rotor inlet. This configuration is close to the unstarting.

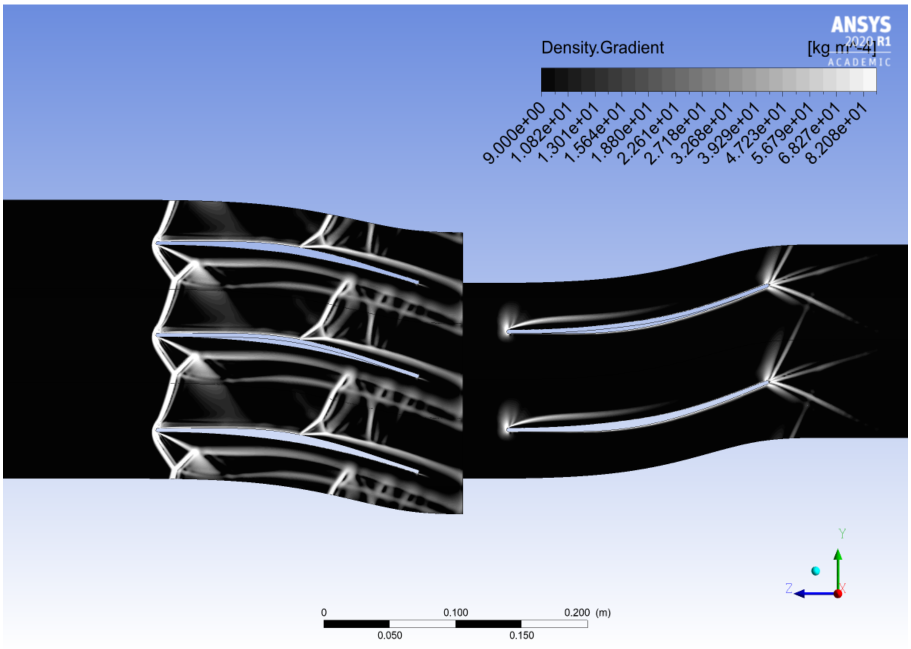


Figure 6.12. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet incidence angle: -12 [deg]. $\eta_{tt} = 18.86\%$. The shock lattice confirms that this configuration is close to the Kantrowitz limit. Note that, because of BL separation, the useful stator passage section is reduced.

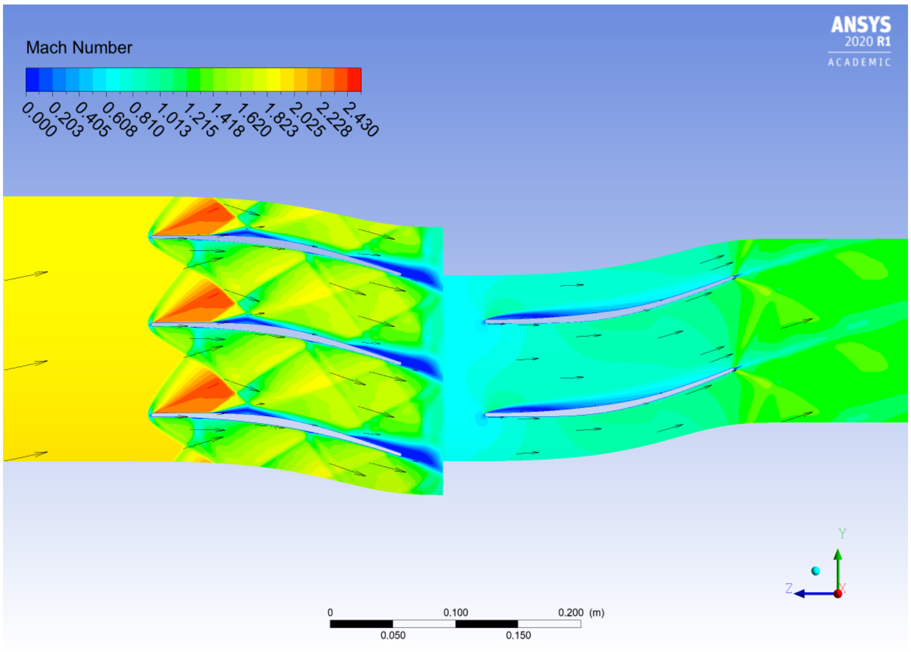


Figure 6.13. Mach number contours. Inlet incidence angle: +12 [deg]. $\eta_{tt} = 21.98\%$

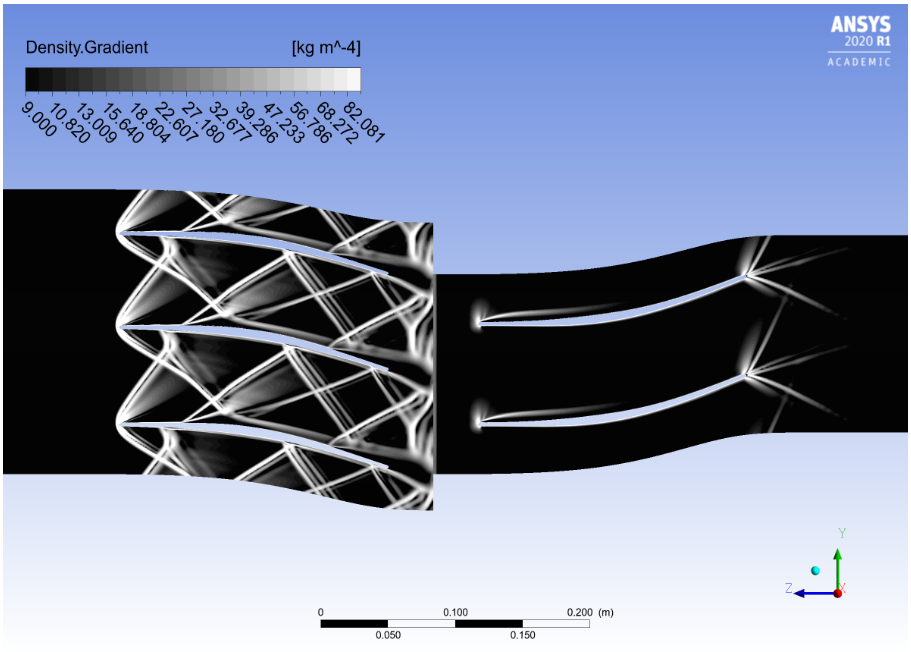


Figure 6.14. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet incidence angle: +12 [deg]. $\eta_{tt} = 21.98\%$

6.3.2 Stator row: inlet flow angle variation

A deeper analysis of the single supersonic stator blade reveals to be necessary. The goal was to better understand what is the stator behaviour in a more extended range of inlet flow angles. Here, the full interval have been simulated, spacing from -16 deg to +16 deg. Every simulation achieved convergence, also in unstarted configuration. In Fig. 6.15, the loss curves confirm the trend already captured in the mixing-plane simulations. It must be noted that losses in stator are crucial for the stage performances, mainly because of the shock waves lattice, and, for high value of incidence angle, also for its interaction with the BL. The rotor plays a crucial role only for the starting of the entire machine, but its contribution in term of entropy production is always relatively small. The simulation results also suggest further considerations: a robust optimization of the machine must deal with the fact that supersonic profiles do not well perform, if subjected to negative incidence flow-field. This fact suggests that, in a future integration between the turbine and the RDE, it should be better to design the blade in order to operate not in its "steady-state" maximum efficiency point, but with a positive incidence angle in order the extend to operation range also to the negative values of the inlet flow angle.

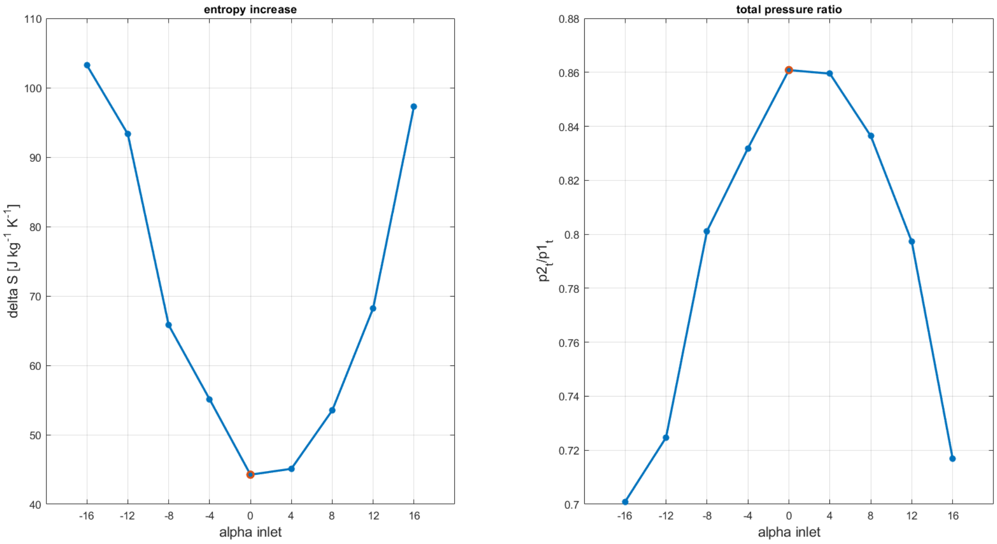


Figure 6.15. Entropy production (left) and Total pressure ratio (right) for the stator blade vs inlet flow angle

In Fig. 6.16 are reported values of outlet flow angle and the respective trend for standard deviation index. The flow-field is characterized by a medium level of variability at the outlet surface. The on-design condition is the best, considering both the deviation from the 0D code angle prediction and the standard deviation index. Meanwhile, in off-design condition the profile progressively loses the capability to impose the flow direction and the discrepancies with respect to the nominal condition are equal to 4 [deg] in the worst scenario.

Moreover, it is interesting to observe the Mach contours in the unstarted configuration with -16[deg] incidence angle (Fig. 6.17). The stator is unstarted: the presence

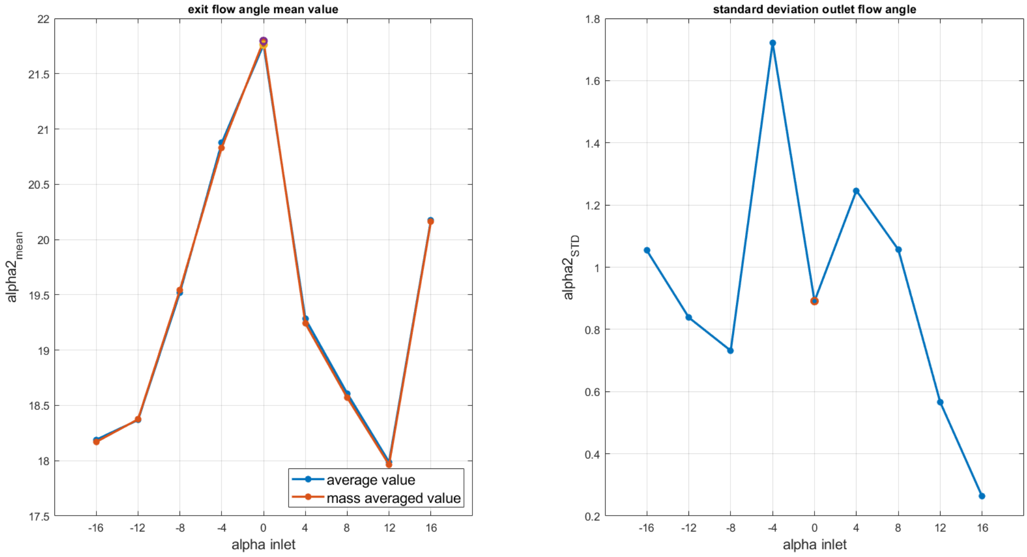


Figure 6.16. Mean value of stator outlet flow angles (left) and standard deviation (right) vs stator inlet flow angle

of a normal shock wave close to the inlet section cause tremendous losses and BL separation. The subsonic region is more extended on the front part of the suction side of the blades, while on the correspondent regions on the pressure side, the flow must accelerate to overcome the leading edge.

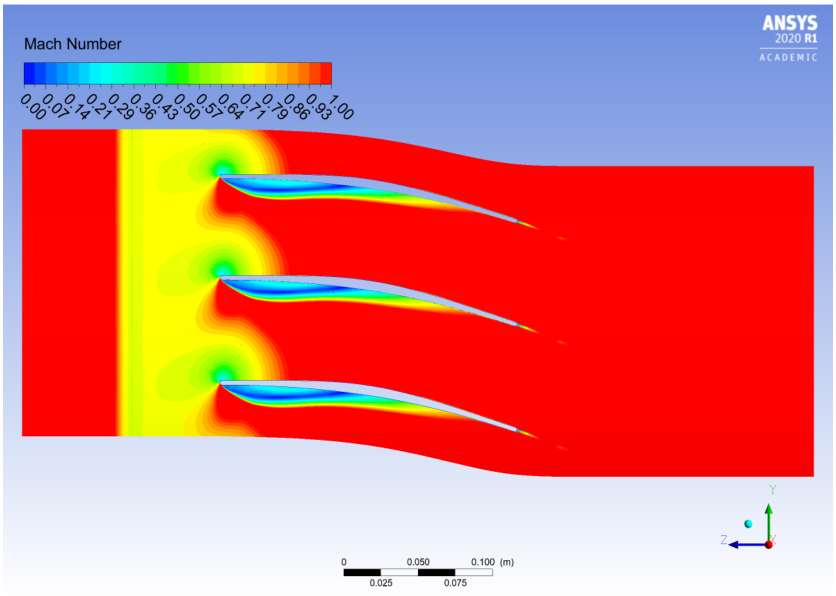


Figure 6.17. Mach number contours of the -16 [deg] inlet flow angle configuration. The scale is limited to a value of 1 for the upper bound to better evidence the supersonic regions (in red).

For what concerns the BL development, in Fig. 6.18 are reported the value of the z-component of the shear stress at the blade wall. For the sake of clarity, only values coming from simulation with +12 [deg], 0[deg] and -12 [deg] are plotted.

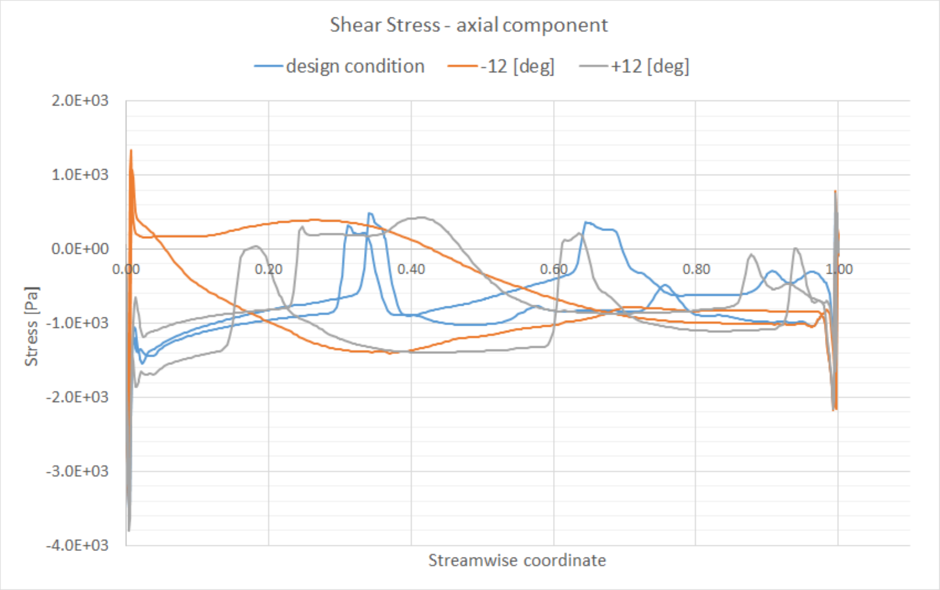


Figure 6.18. Shear Stress axial component vs stream-wise coordinate. Where the value are positive, there is the presence of boundary layer separation. For the design condition, these regions are very limited and present only where shock waves hit the BL; the worst situation is in the -12[deg] configuration, where front part of the blade pressure side experience extensive reverse flow due to BL separation, because of the negative incidence.

6.3.3 Full stage: inlet Mach number variation

In this series of simulations, the turbine stage have been exposed to an inlet flow with different value of the Mach number. Static pressure and static temperature have been kept fixed. This choice was arbitrary. In future simulation campaign, other inlet parameters combination can be studied. The closer alternative is to fix the total quantities and vary the Mach number. The inlet flow angle in all the the following simulation is the nominal one, i.e. 0 [deg]. The first results obtained are related to the unstarting of the turbine stage. It can be noted, in the following diagrams, that the range of inlet Mach number varies from a value of 2 to 2.8. For value lower than 2, the nominal design condition, the simulations did not converge because of the presence of a normal shock upstream the rotor blade and too close to the mixing-plane, causing numerical instabilities. Actually, the mean-line code already faced this problem by gradually increasing the height of the rotor blade, in order to accelerate the flow field and, therefore, satisfy the Kantrowitz limit. Unfortunately, simulating the average stator and rotor profiles, it was not possible to take into account the increase in channel size and therefore, when simulating the full stage at $M < 2$, the rotor does not start as it can be expected from the Kantrowitz condition applied to a constant height rotor. In Fig. 6.19 are reported the losses in terms of mass weighted averaged entropy. The latter is reported at every surface of interest: stator inlet ($S1_{in}$), stator outlet ($S1_{out}$), rotor inlet ($R1_{in}$), rotor outlet ($R1_{out}$). Stator outlet and Rotor inlet surfaces are the interface boundary where the numerical mixing takes place by circumferential averaging the flow-field. It can be noted how the entropy in $R1_{in}$ is always a bit higher than the one in $S1_{out}$, because of the mixing process. Overall, the stage losses increase almost linearly with the inlet Mach number. Same trend is present for the single rows of stator and rotor. At higher Mach, greater growth is observed. This is due not only to the BL and the mixing process, but, most of all, to the reflected oblique shock that impacts the blade profile in proximity of the trailing edge, so favouring a premature flow separation.

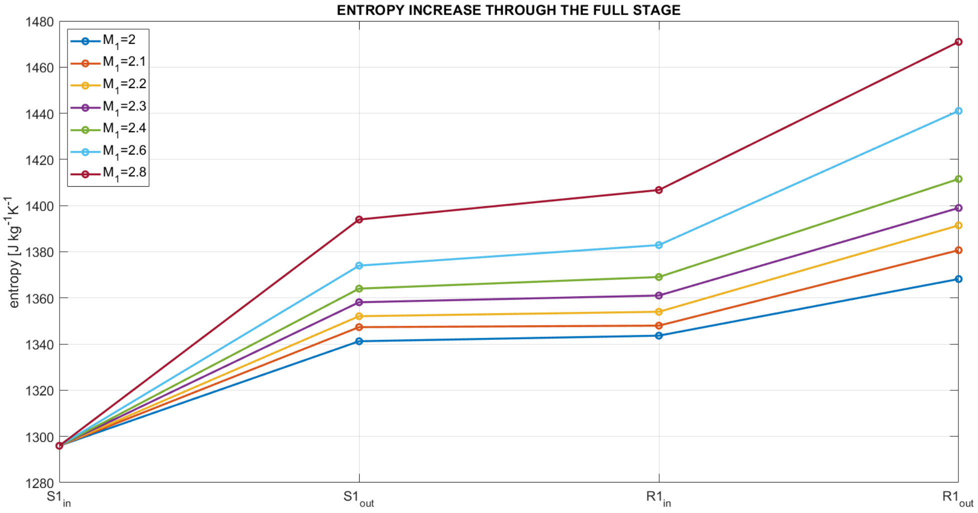


Figure 6.19. Entropy values throughout the full stage domain

In Fig. 6.20, rotor outlet flow angle in relative frame of reference have been analysed. The flow regime is supersonic and, increasing the Mach number, the flow inertia becomes relevant enough to produce important deviations from the blade metal angle. Moreover, in these conditions, the characteristic lines are not any more in the designed position and the physical effect is an increase in the outlet flow angle variability, highlighted by higher values of the flow angle standard deviation. In Fig. 6.21, are

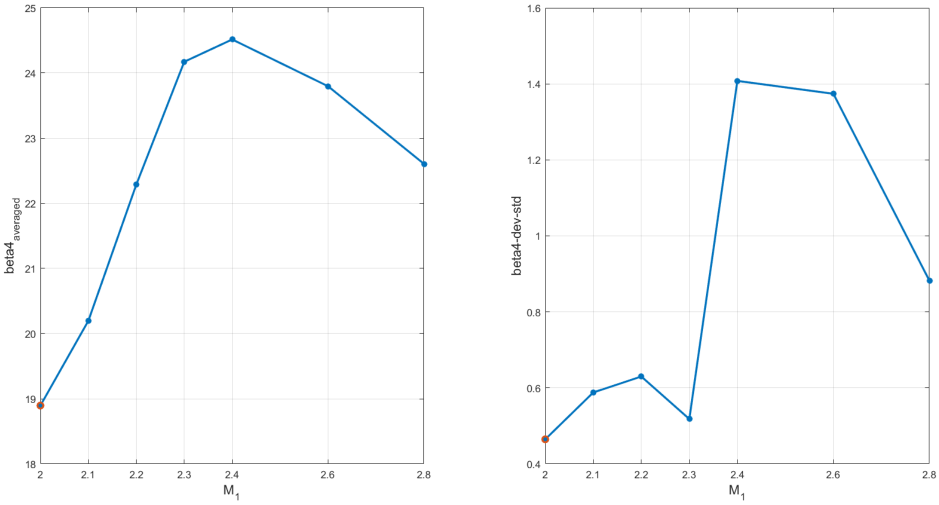


Figure 6.20. Mean value of rotor relative outlet flow angles (left) and standard deviation (right) vs stator inlet Mach number

plotted the Mach number contours coming from the $M = 2.6$ configuration. It is interesting to note how the off-design condition at the stator inlet have direct effects on the rotor inlet not only in terms of Mach number distribution or shock waves position (Fig. 6.22), but in terms of incidence angle. The inlet Mach number increase is correlated to the deviation angle at the stator blade; the resulting effect is that the rotor is subjected to a flow coming with a positive incidence angle.

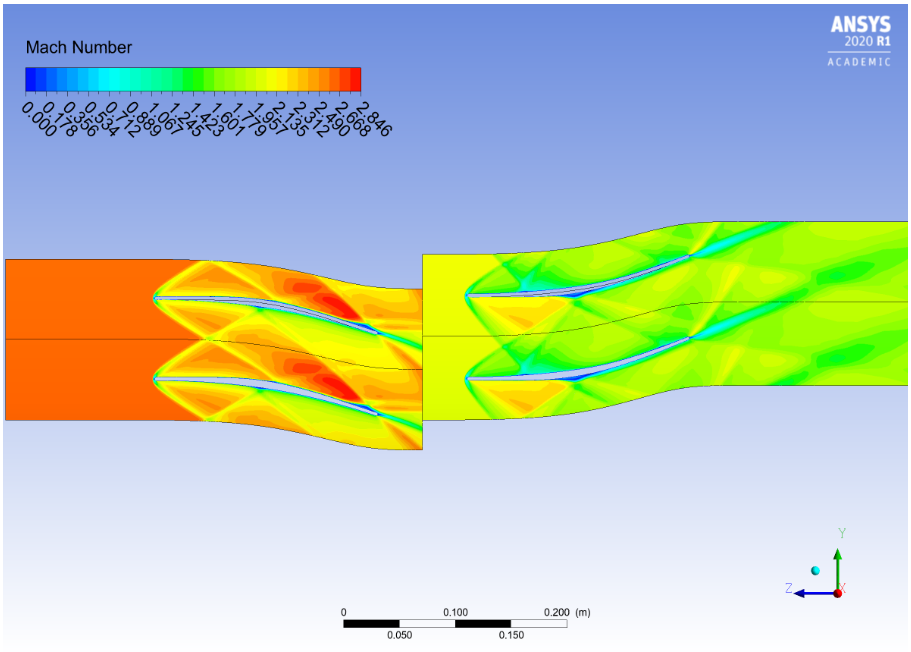


Figure 6.21. Mach number contours. Inlet $M = 2.6$

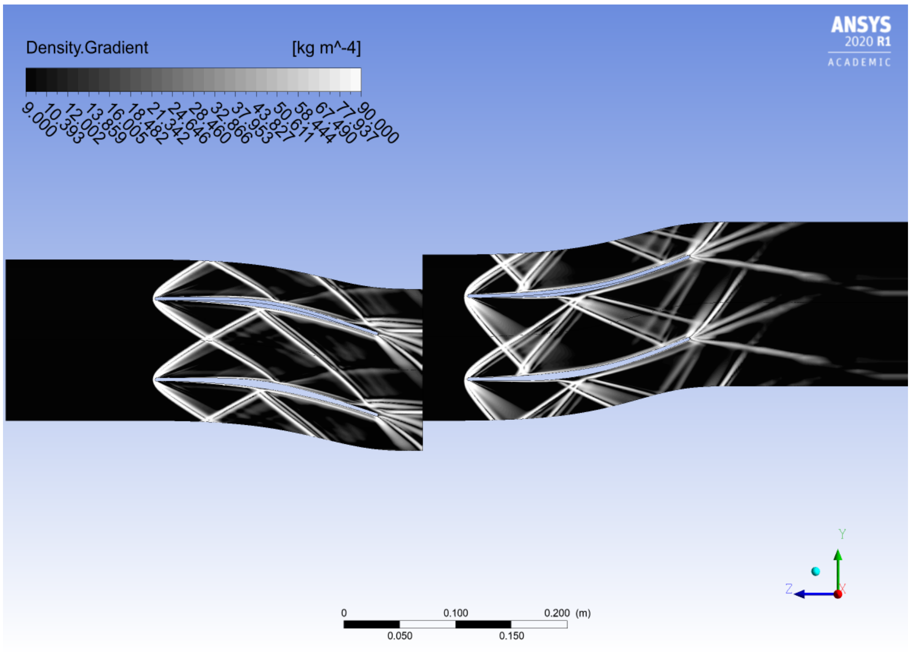


Figure 6.22. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 2.6$

6.3.4 Stator row: inlet Mach number variation

The last part of this set of simulations is focused on the fluid dynamic phenomena occurring in the stator channel without the influence of the downstream blade. A more extensive analysis was deemed appropriate to understand the rangeability of the stator, exploring also the Mach value lower than the designed one. Meanwhile the full turbine was unstarted for $M < 2$ at inlet, this is not the case of the stator blade. The mass flow rate through the passage varies linearly (see Tab. 6.5) with the inlet Mach number, since the inlet static conditions are still fixed. The range explored with these simulation is between 1.6 and 2.8 inlet Mach number. The following features con be noted:

- the outlet flow angle is reduced by a few degrees when the speeds are lower than the design condition, while there is a low over-speed sensitivity (Fig. 6.23);
- the condition of uniformity of the outgoing flow is reduced as the speed increases;
- the losses have a fairly linear trend around the design condition ($M = 2$) (Fig. 6.24); at higher Mach, the trend is more than linear since the reflected oblique shock affects the trailing edge region, favouring a premature flow separation;

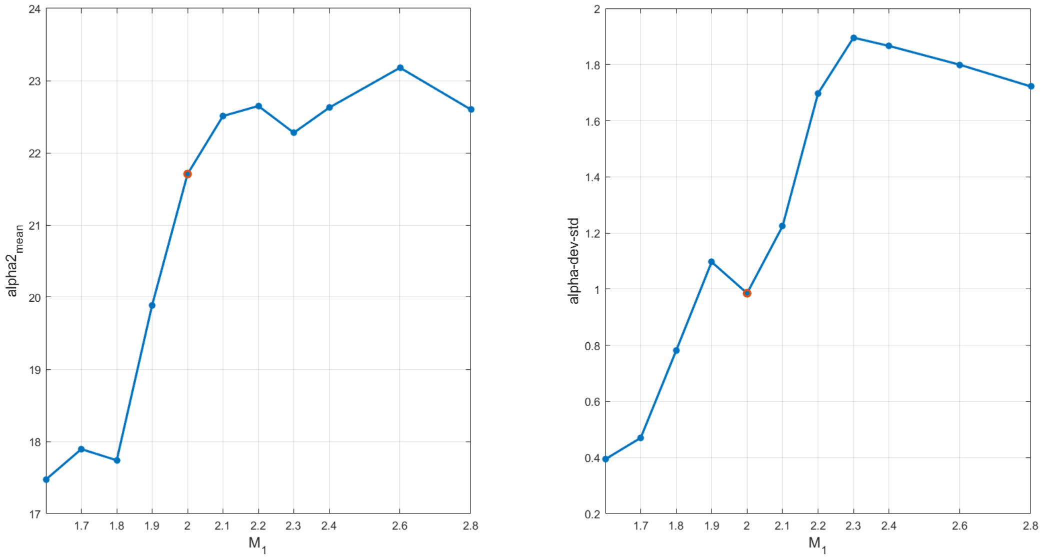


Figure 6.23. Mean value of stator outlet flow angles (left) and standard deviation (right) vs stator inlet Mach number

Table 6.5. *annular* mass flow rate passing through the thin layer which, in Ansys CFX, describes the blade-to-blade passage. Fixing the static conditions, the mass flow rate varies linearly with the inlet Mach number. No presence of chocking.

Inlet Mach number	1.6	1.7	1.8	1.9	2	2.1	2.2	2.3	2.4	2.6	2.8
Mass flow rate [kg/s]	0.34	0.36	0.38	0.4	0.42	0.44	0.47	0.49	0.51	0.55	0.59

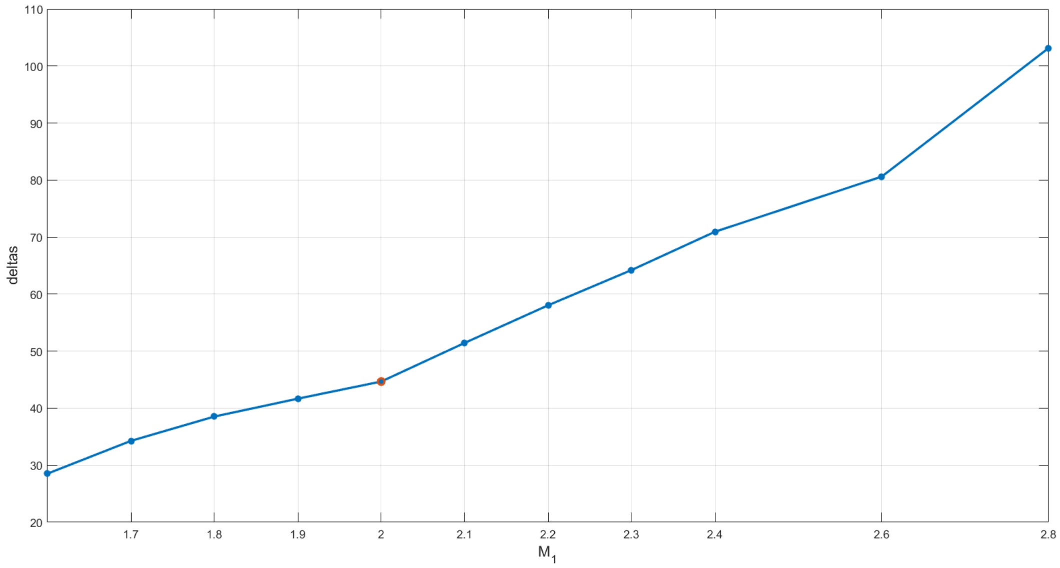


Figure 6.24. Stator entropy increase [J kg^{-}1 K^{-}1] vs inlet Mach number.

By way of example, hereafter are reported the blade loading chart (Fig. 6.25), the resulting Mach contours and numerical-Schlieren visualizations from some of the stator simulations in off-design condition: Fig. 6.26 and Fig. 6.27 for inlet Mach number equal to 1.6, Fig. 6.28 and Fig. 6.29 for inlet Mach number equal to 2.4, Fig. 6.30 and Fig. 6.31 for inlet Mach number equal to 2.8.

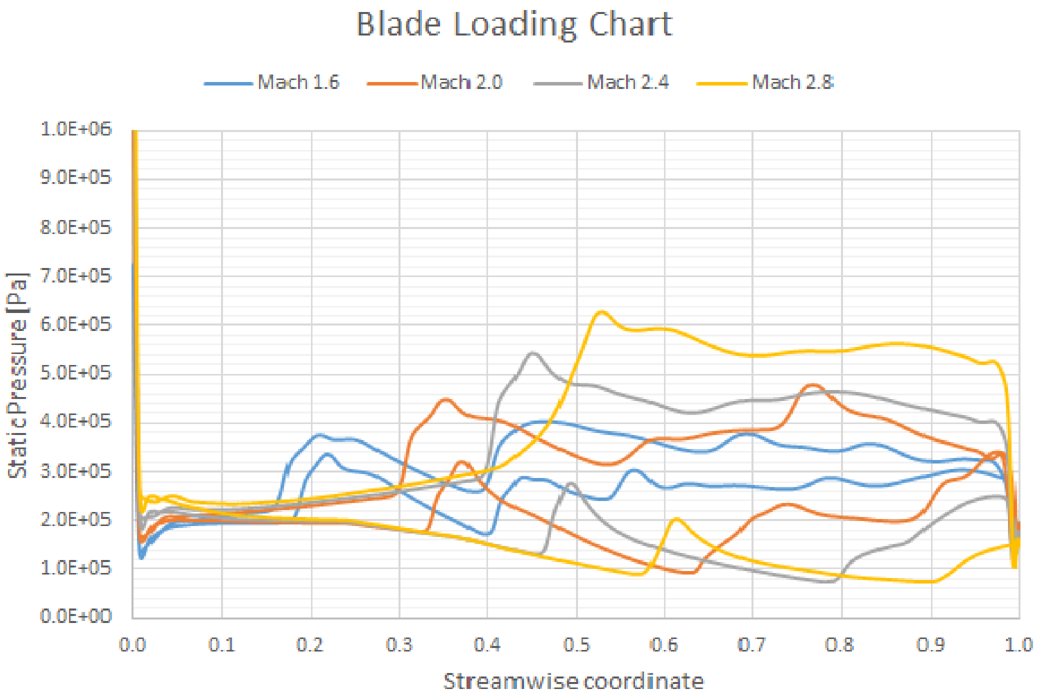


Figure 6.25. Blade loading chart for different inlet mach number conditions.

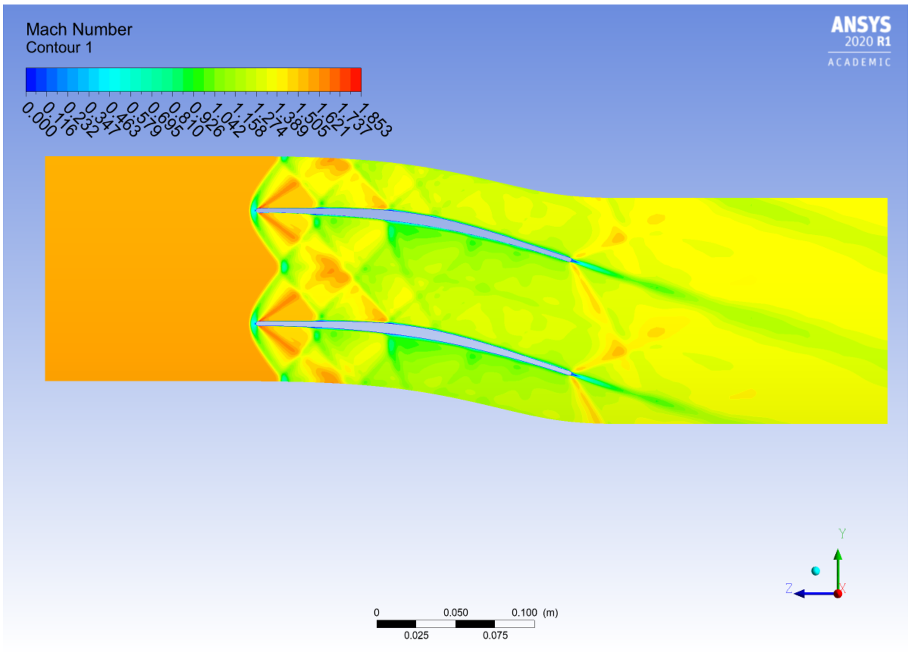


Figure 6.26. Mach number contours. Inlet $M = 1.6$

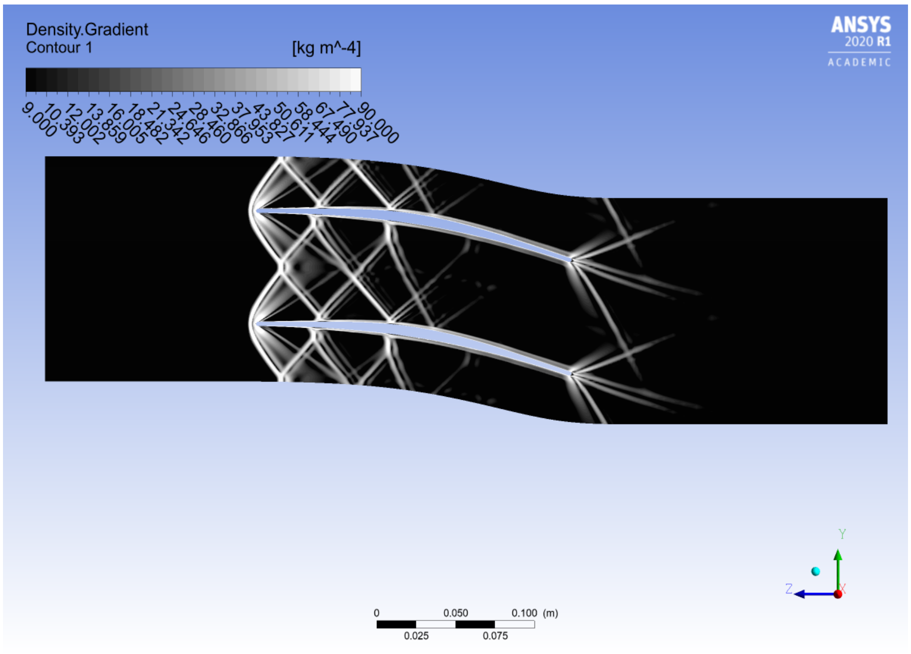


Figure 6.27. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 1.6$. Note that the shock lattice is *tighter* than the one resulting from simulation in nominal condition (Fig. 5.15)

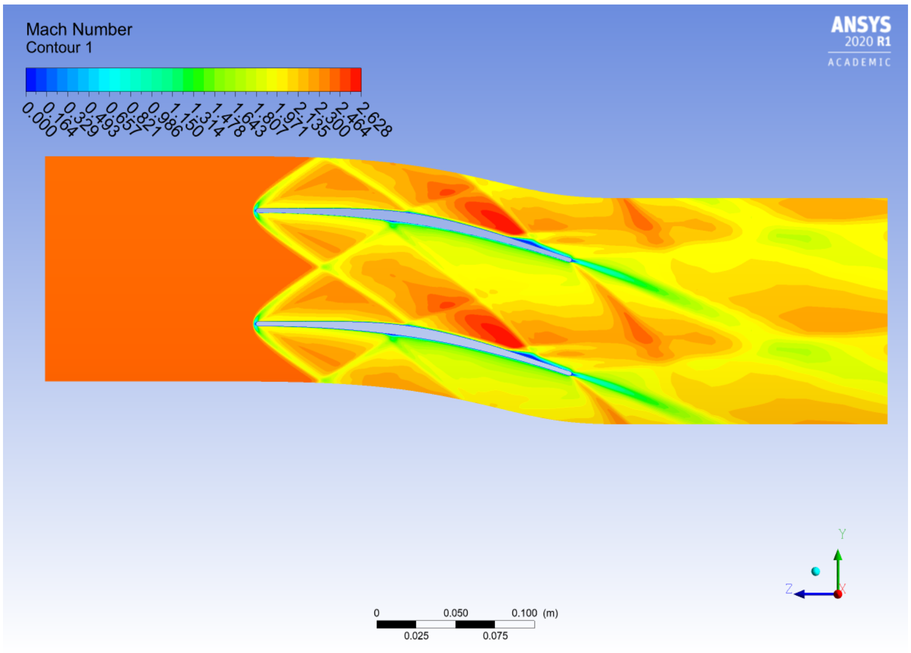


Figure 6.28. Mach number contours. Inlet $M = 2.4$

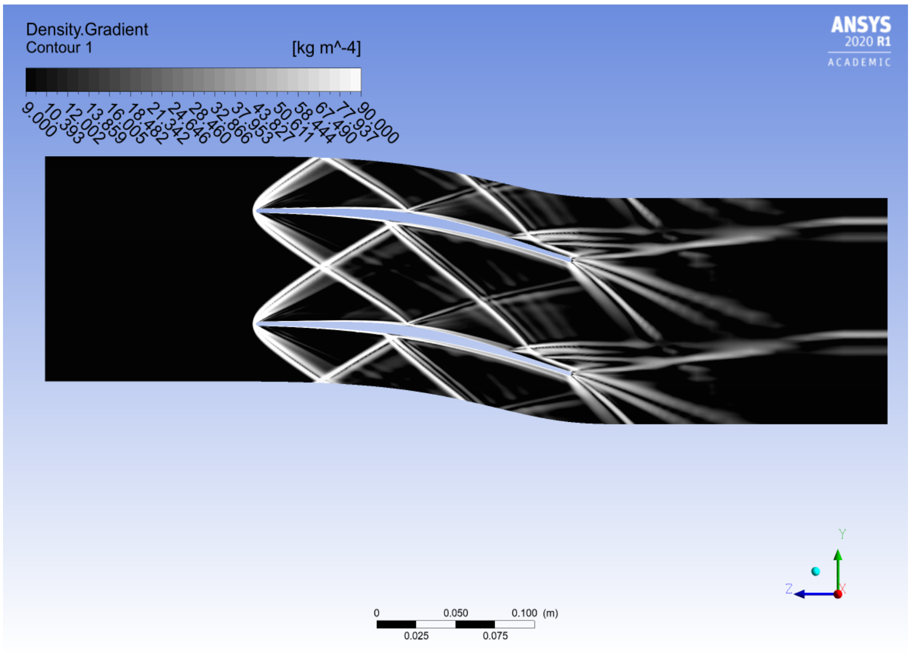


Figure 6.29. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 2.4$. Note that the shock lattice is more *stretched* than the one resulting from simulation in nominal condition (Fig. 5.15). This is the effect of the higher inlet Mach number, resulting in stronger and more oblique bow shock.

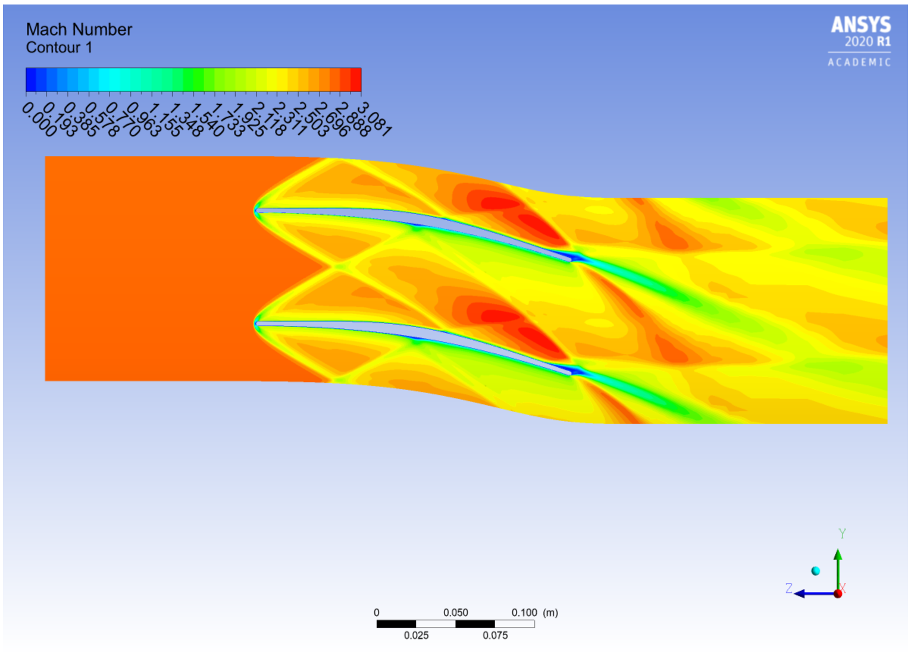


Figure 6.30. Mach number contours. Inlet $M = 2.8$

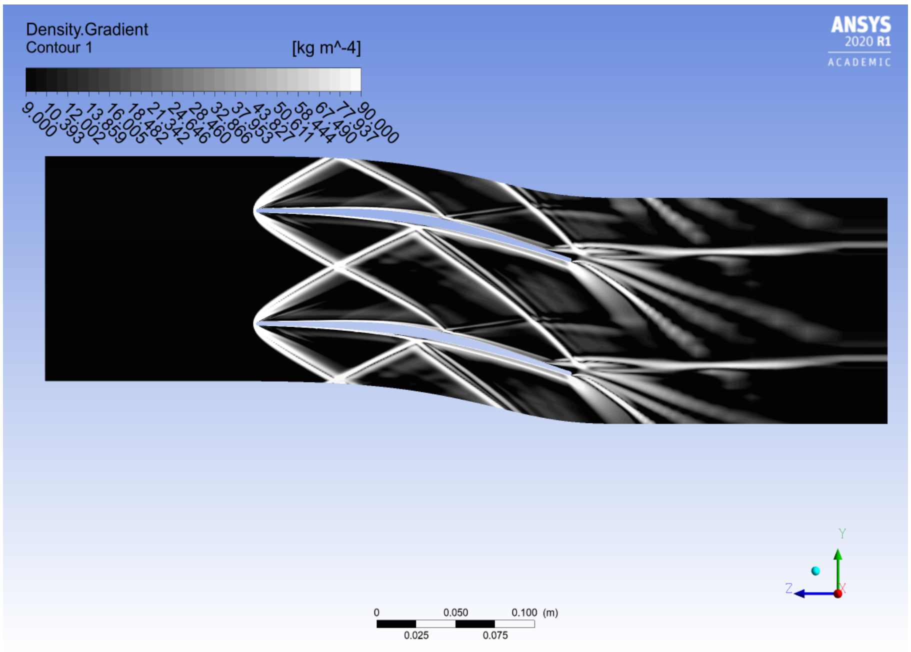


Figure 6.31. Density gradient contours in arbitrary log scale (i.e. numerical-Schlieren visualization). Inlet $M = 2.8$

Chapter 7

High-fidelity shape optimization by surrogate-based evolutionary techniques

In chapter 2, the vortex-flow method, a special method of characteristics developed by Louis J. Goldman [15], was implemented in the mean-line code to design the supersonic blade profile. Then the lower and upper surface were shifted to obtain a finite-thickness blade, where the shifting value was selected after a parametric analysis which aimed to the minimization of the losses generated by the leading edge shock wave. After that in chapter 5, a completely original automated optimization process was developed to find the optimal value of the solidity, which again aims to minimize the entropy produced by the turbine. Finally, in this chapter the design of the profile is further enhanced through a high-fidelity shape optimization by surrogate-based evolutionary techniques. In the vortex-flow method, the flow is considered two dimensional, steady and inviscid, which means that the lower and upper surface of the profile are drawn without considering the boundary layer and its interaction with the impinging oblique shock wave. For this reasons an optimization based on high-fidelity simulations was chosen to complete the design of the profile.

The shape optimization was performed through FORMA (Fluid-dynamic OptimizeR for turbo-Machinery Airfoils): an evolutionary shape-optimization code developed by the laboratory of fluid machines at Politecnico di Milano, based on a geometrical parametrization technique built on B-Splines, a high-fidelity and experimentally validated CFD solver and a surrogate-based evolutionary algorithm [53].

In the first part of this chapter, a detailed but brief description is given for the main components of the FORMA package: geometrical parametrization (subsection 7.1.1), genetic algorithms (subsection 7.1.2), surrogate model and design of experiments (subsection 7.1.3), surrogate-based optimization (subsection 7.1.4) and how it is coded (subsection 7.1.5).

In the second part, it is described how FORMA was employed for the shape optimization of the stator (section 7.2): actually two different cases are presented and each of them generates an optimal profile that has its strengths and weaknesses (subsection 7.2.1 and subsection 7.2.2). Finally, the chapter is ended with some considerations on the shape optimization and with some ideas for future developments on the topic (section 7.3).

7.1 Detailed description of the FORMA package

7.1.1 Geometrical parametrization

The first and fundamental step in the optimization process is the geometrical parametrization of the blade profile, but before describing this topic in detail it is important to understand why this step is necessary. Optimization algorithms work to find a minimum of an objective function, which for our application will be the entropy generated by the supersonic blade row; these algorithms perform properly when the number of variables to be determined is limited, but on the other hand the shape of a blade is usually described with hundreds or thousands of points: this guarantees accuracy, but at the same time it is inconceivable for the algorithms to move so many points and optimize the shape with this type of profile representation. Hence a parametrization of the blade profile through a limited number of variables is a necessary component for the optimization and FORMA employs B-spline curves to achieve this.

B-splines, or basis splines, are defined as sequence of polynomial curve segments of degree "n" that provide local support: they are linear combination of polynomial functions weighted by coefficients represented by control points [60] [22]. To completely define these curves, the following informations are required:

- an interval I=[a, b] over which the B-spline is defined;
- the degree "n" of the polynomials;
- a set of N + 1 control points CP_i with i=[0,N];
- a knot sequence $w = [w_0 = a, \ \dots \ , \ w_i, \ \dots \ , w_m = b]$ with m = (N + 1) + n.

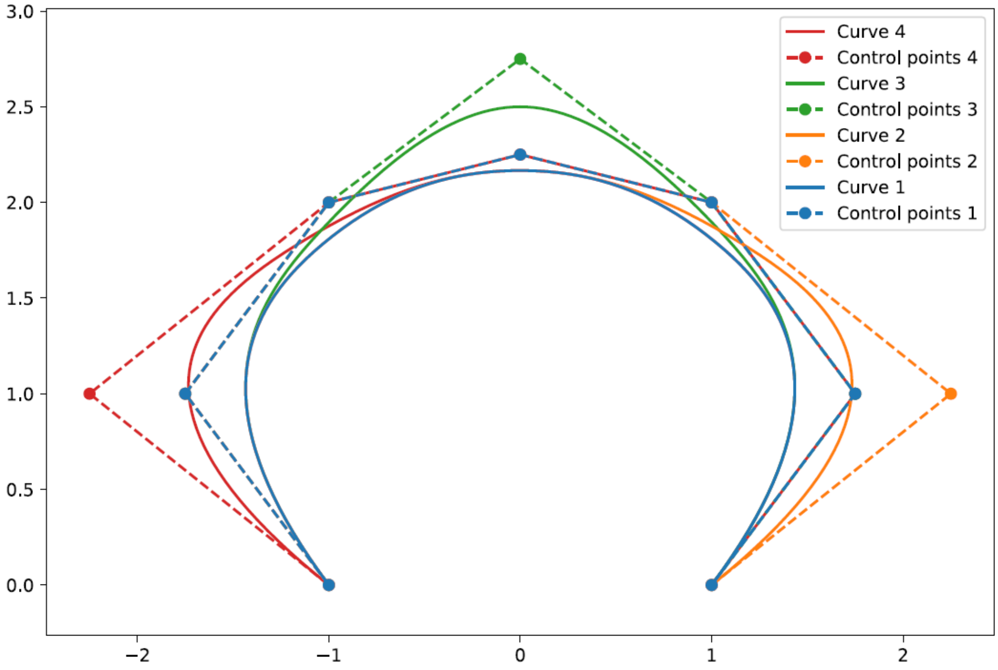


Figure 7.1. B-spline moving a control point (Figure from [22]).

B-splines were chosen by FORMA because they provide excellent properties for this type of application:

- **smoothness**: the curves are piecewise polynomials, which means that they are infinitely differentiable in all the points except the breakpoints; the two polynomial curve segments are at least "n-r" times differentiable in the intersection point, where "r" is the multiplicity of the knot. Therefore a B-spline is a C^{n-r} curve and its smoothness can be controlled through the knot sequence and the associated multiplicity;

- **local controllability**: changing the position of a single control point will result in a local variation of the curve without affecting the points that are far from the moved control point (see Figure 7.1). This property is vital to properly interpolate both the regions with small and high curvature (for example in a supersonic blade profile the centre of the blade has a much smaller curvature compared to the leading edge).

Building a B-spline is not a trivial process, but a detailed description can be easily found in literature and as a reference the textbook by Farin contains a step by step procedure on how to interpolate using B-splines [60]; for this reason, in the next paragraphs only the main points regarding these curves will be treated without reporting the complete theory.

The number of control points is significantly lower than the points to interpolate and because of this there is not a single exact B-spline interpolating the profile; among the infinite interpolating curves available, the code selects the one that minimizes the error, defined in equation 7.1 as the distance between the points and the curve:

$$\epsilon = \sum_{j=0}^{P} ||\mathbf{Y}_j - \mathbf{x}(w_j)|| \tag{7.1}$$

where $\mathbf{Y}_j$ are the P+1 points to interpolate, $\mathbf{x}(w_j)$ are the interpolated points and w_j are the associated knots.

Then a least square approach is employed to minimize the error ϵ and after some mathematical steps, the set of equations is rewritten in the compact form of equation 7.2.

$$\mathbf{M} \cdot \mathbf{CP} = \mathbf{b} \tag{7.2}$$

The Cholesky decomposition is applied to solve equation 7.2 because matrix **M** is symmetric and ill conditioned; finally the matrix **CP** is obtained and it is very important to remember that usually the control points do not belong to the interpolating curve for a B-spline that has a higher degree than 2.

As stated previously, the knot sequence plays a vital role in the generation of a B-spline because it does not only affects the smoothness, but also how precise is the interpolating curve compared to the original data. Usually a completely analytical formulation is not feasible because it is a non-linear problem with several constraints; therefore the knot sequence is generated a priori and different solutions are available in literature on how to perform this. In FORMA the centripetal parametrization and the rational parametrization were both tested, but finally the rational one was chosen because it guarantees overall a much more accurate result for this type of application, as shown by Rossi in Figure 7.2.

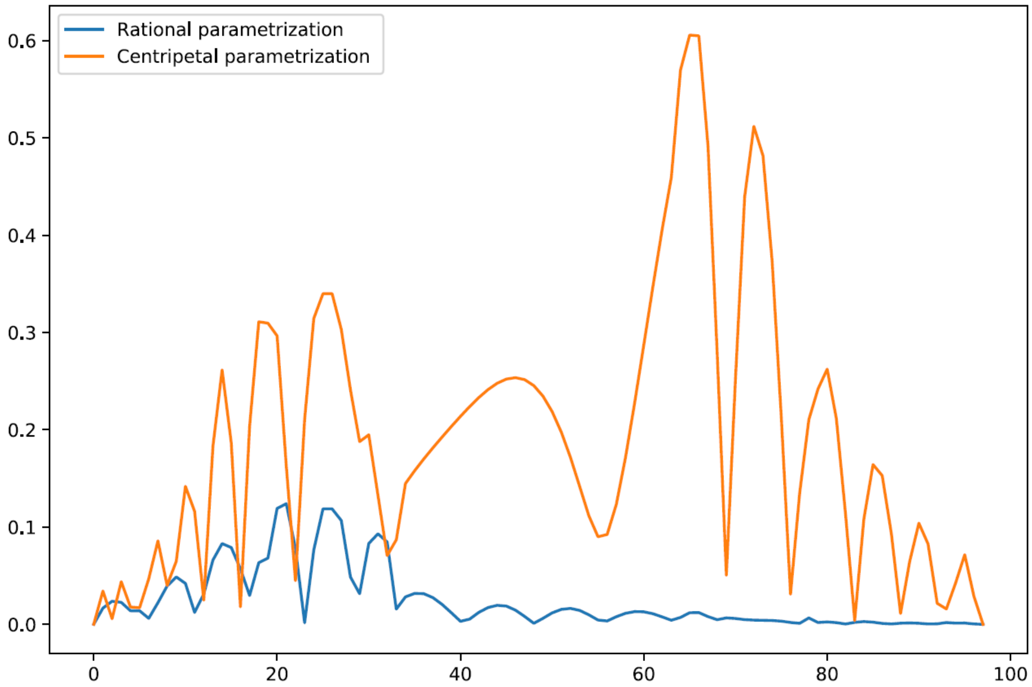


Figure 7.2. Point by point error comparison between centripetal and rational parametrization (Figure from [22]).

At the trailing edge, the blades usually are thinner and present a much higher curvature, which would require many control points and an adequate knot sequence from a B-spline point of view. To simplify these operations, the trailing edge is not managed using B-splines, but instead the profile is reconstructed using a section of a circle or an ellipse. If the circle option is selected, the trailing edge is built using the same theory and equations already presented in subsection 2.4.5.

The ellipse is presented as an alternative to the circular shape because it allows to impose the tangency to both the upper and lower surface; even if all four constraints are satisfied (the passage and the tangency to both the upper and lower surface), there is still one degree of freedom because an ellipse is defined by 5 parameters. As last constraint, the code selects between the infinite ellipse satisfying the previous constraints the one that minimizes the eccentricity, which intrinsically means that the code selects the ellipse that is closer to be a circle.

Now that the interpolation phase of the blade is completed, the next step is to define the design space and in particular to instruct the optimizer the way the control points are moved. Since the interpolation is performed in 2D, hypothetically the points can be moved independently in the x and the y directions, but that would definitely overcomplicate the problem; a smarter choice is to move the control points of a specified distance d in a normal direction respect to the curve, where as a convention the normal is pointing outwards. In equation 7.3 are reported the relations to move the control points, where m is the tangent to the curve, CP is the original control point and Q is the new one.

$$\begin{cases} \theta = atan(m) \\ x_Q = x_{CP} - d \cdot sin\theta \\ y_Q = y_{CP} + d \cdot cos\theta \end{cases} \tag{7.3}$$

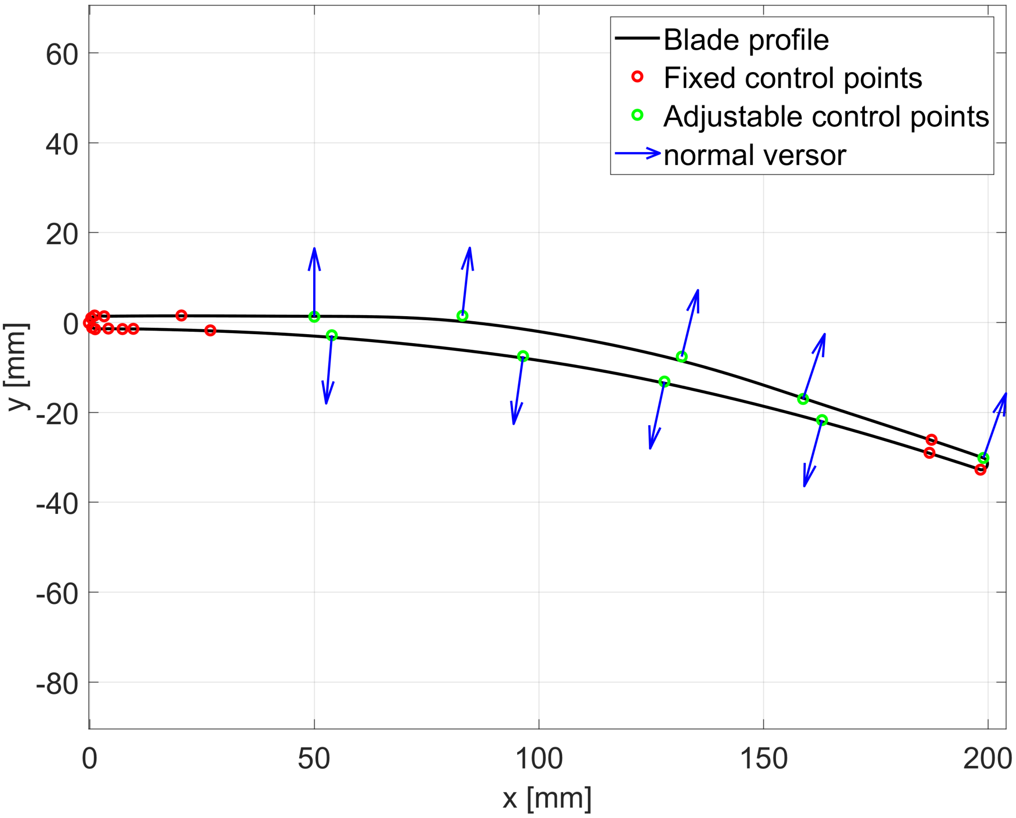


Figure 7.3. Example of the control points and their associated normals used to perform a B-spline parametrization.

Finally for the modification of the trailing edge, which is defined by the first two and the last two control points (Figure 7.3), the code offers three different options:

- a completely fixed trailing edge, for which no control point moves during the optimization phase;
- the first control point is adjustable, while the last one moves to preserve the trailing edge thickness or vice-versa;
- the first control point is adjustable, while the other three points move rigidly to preserve the trailing edge shape.

7.1.2 Genetic algorithms

Genetic algorithms belong to the family of nature-inspired search methods, also known as evolutionary algorithms or metaheuristic methods because they make no assumptions about the optimization problem and can search very large spaces for candidate solutions [61]. Genetic algorithms and in general this whole family of methods are really powerful for the following properties:

- the restrictions on their application are really limited, as they can be employed in smooth problems, non-smooth problems, discrete/integer programming problems, network optimization problems and problems with noisy functions;
- the derivatives are not required, which is really useful in some cases where the calculation of the derivatives is extremely difficult or even not possible;
- they converge in the defined design space to the global minimum of the objective function.

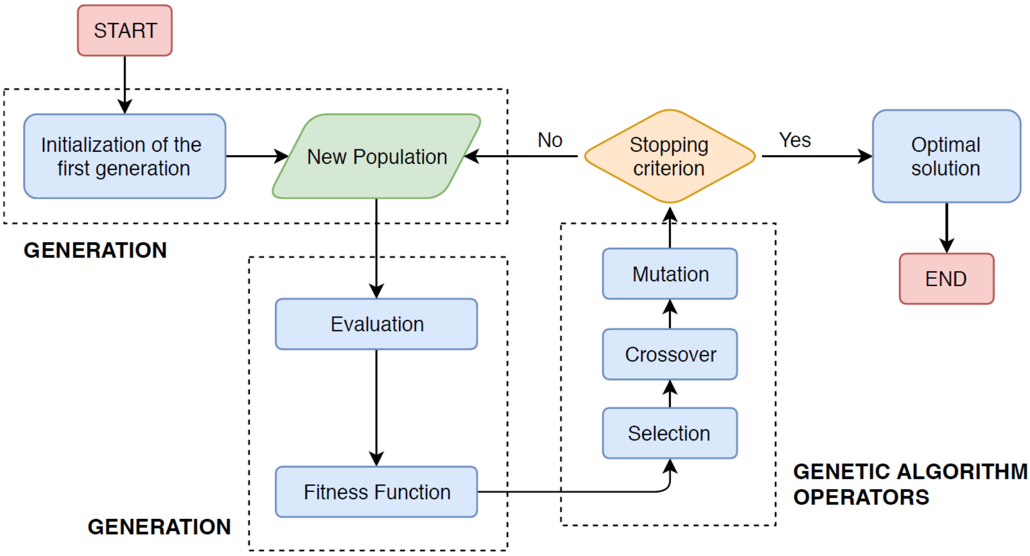


Figure 7.4. Genetic algorithms logic (Figure from [22]).

Genetic algorithms are based on Darwin's theory of natural selection: the process starts with the random generation of a population of N individuals by assigning to each of them allowable values for each design variable. Each individual is then evaluated using the fitness function, which for our case will be the increase of entropy from stator inlet to stator outlet. From the current population, the fittest members are selected to generate new individuals through some specific operations that later will be described in more detail; therefore the new generation of population, whose size is fixed, has a higher probability of obtaining better fitness values. Finally the same process is repeated from generation to generation till a stopping criterion is satisfied.

Before proceeding further with the discussion, it is necessary to introduce some basic terminology adopted from the genetic theory:

- the **population** is the complete set of design points (individuals) in the current iteration;
- the **generation** is each iteration of the genetic algorithm;

- a **chromosome** contains the values of all the variables associated to a design point;
- a **gene** represents the value of a design variable and therefore each chromosome is composed of a complete series of genes.

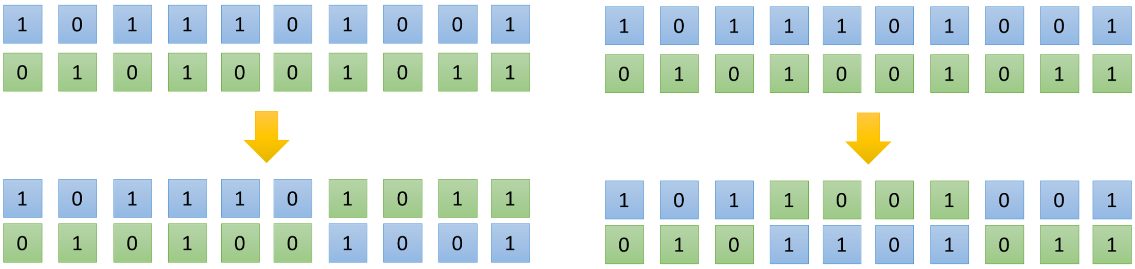
Then it is necessary to find a method to represent each design point in the population and, even though there are other possibilities, the most used approach is the binary encoding.
Now that all the fundamental informations have been introduced, it is finally possible to explain how the new generation is obtained through the operations of selection, crossover and mutation.

Selection

The selection is the process through which the average fitness function of the next generation is improved in order to converge towards the optimal solution. This is achieved by simply assigning a larger probability of selection to the individuals that have a higher fitness value.

Crossover

The crossover is the process of combining and mixing two different chromosomes of the population. Although there are many methods for performing crossover, the most common ones are the one-cut point and two-cut point methods [61]. The amount of crossovers must be tuned properly and usually is a small number because too many crossovers can result in designs that are really far from the optimal solution.



(a) *one-cut point* (b) *two-cut point*

Figure 7.5. Two different methods of the crossover process.

Mutation

Mutation is a fundamental process to avoid a premature convergence of the algorithm due to loss of valuable genetic material during the reproduction and crossover steps. Its implementation is simplified in the binary encoding because the optimizer just randomly switch some zeros into ones or vice versa. Differently than crossover, mutation changes a design point without getting too far from the original values and for this reason a higher amount of mutation can be selected.

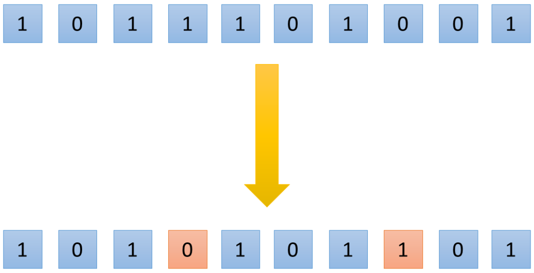


Figure 7.6. Example of the mutation process.

As previously stated, genetic algorithms have many advantages, but they have also an important drawback: they require a really large amount of function evaluations even when the population size is limited. For our application, this factor becomes even more critical because to each function evaluation is associated a high-fidelity CFD simulation and the associated computational cost would be prohibitive even with high performance computing (HPC); therefore the optimization process will not be performed directly on the high-fidelity function, but on a surrogate based on the Kriging model (for more details see subsection 7.1.3).
In conclusion, FORMA implements the genetic algorithm through the JEGA library, which makes available both a single-objective genetic algorithm (SOGA) and a multi-objective genetic algorithm (MOGA).

7.1.3 Surrogate model and design of experiments (DOE)

Surrogate modelling is defined as a non-linear inverse procedure which aims to determine a continuous function of a set of design variables from a limited amount of available data [62]. As explained in the previous section, a direct application of the genetic algorithm to high fidelity simulations is unfeasible due to the enormous computational cost; therefore, FORMA implements a surrogate-based optimization procedure to improve the shape of the blade profile.
In literature there is an extensive theory regarding surrogate models and each of them performs better with certain type of applications. In FORMA several options including artificial neural networks, radial basis functions, multi-variate adaptive regression splines were tried in the preliminary stage, but at the end the universal Kriging formulation with non-zero covariance, based on a Gaussian correlation function, was selected as standard [53].
The Kriging model was developed in the field of geostatistics and the basic idea is to create an interpolating function which is able to predict the value of an unknown point using a stochastic process. It is very important to understand that the relation generated by the model has no real physical meaning: it is just a mathematical expression that tries to link together the inputs and the outputs.
In order to obtain an acceptable prediction, the model requires a significant amount of data associating the inputs with the outputs: for our case the inputs are the geometrical parameters that generate the shape, while the outputs are the entropy generated from stator inlet to stator outlet estimated using high fidelity simulations.
The initial training of the model must be carefully considered to maximize the amount of information acquired in order to develop a model that is as reliable as possible;

this sampling procedure of the design space is called Design Of Experiments (DOE). Again in literature there are many ways to execute the DOE and one of them is the Latin Hypercube Sampling (LHS) employed by FORMA.
The LHS is based on the concept of stratification of the cumulative distribution function into intervals of equal probability and then one and only one sample is taken from each interval. To understand this concept, let's analyse the name of this DOE method: in two dimensions a square grid is a "latin" square if there is only one sample in each row and each column (Figure 7.7); then the word "hypercube" is added to generalize the same idea to an arbitrary number of dimensions, whereby each sample is the only one in each axis-aligned hyperplane containing it.

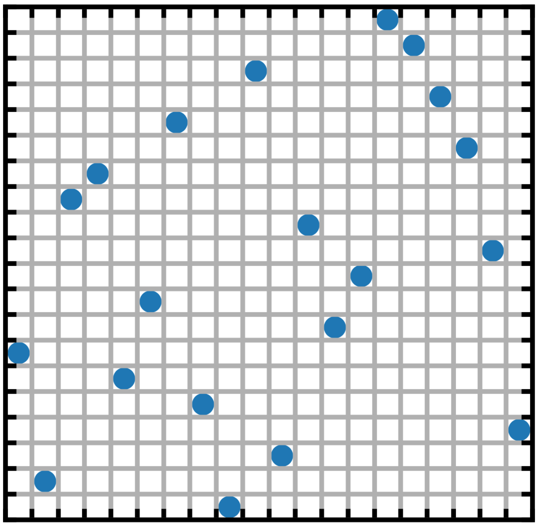


Figure 7.7. Example of latin hypercube sampling in two dimensions (latin square).

7.1.4 Surrogate-based optimization

Now that the various components of the FORMA package have been described, is finally time to put all the pieces together. The entire process starts with the design of experiments to properly sample the entire design space and to generate the initial database required by the Kriging surrogate model. Since the initial data is obviously limited and during the optimization new high-fidelity simulations are performed, the surrogate model is updated each iteration to improve its reliability; there are actually two different strategies to execute this in FORMA:

- Surrogate Based Local Optimization (SBLO): the DOE is generated at each iteration, which requires a significant amount of computational cost. The main advantage is a better scalability when a lot of computational power is available;
- Surrogate Based Global Optimization (SBGO): the DOE is generated only once and after that the optimization proceeds in a serial manner; this option is relatively less expensive from a computational point of view.

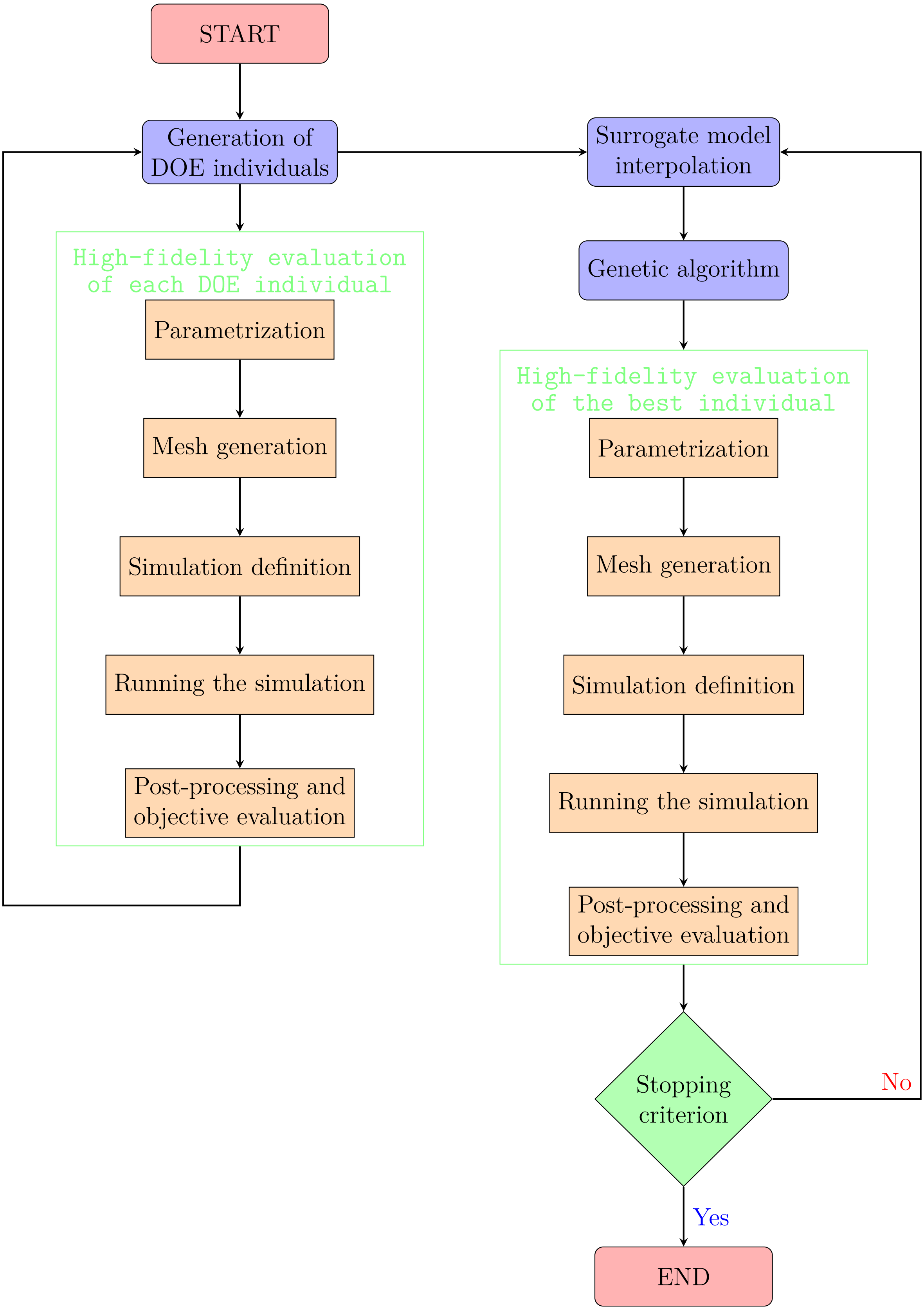

START
Generation of DOE individuals
High-fidelity evaluation of each DOE individual
Parametrization
Mesh generation
Simulation definition
Running the simulation
Post-processing and objective evaluation
Surrogate model interpolation
Genetic algorithm
High-fidelity evaluation of the best individual
Parametrization
Mesh generation
Simulation definition
Running the simulation
Post-processing and objective evaluation
Stopping criterion
No
Yes
END

Figure 7.8. Flowchart to illustrate the surrogate-based global optimization algorithm.

In this thesis a detailed description will be given only for the SBGO because only this option was used in the optimization procedure, but if the reader is interested a complete discussion about both strategies can be found in [22] and [53].
In the SBGO the design of experiments is generated only once and for this reason the size of the initial population is a key parameter: if a too small number is selected, the surrogate model is not able to create a reliable input-output relation, while on the other hand if a too large number is chosen valuable time is wasted generating the initial database; using the latin hypercube sampling technique, the number of samples for the DOE are fixed using equation 7.4:

$$size\ of\ DOE = (m)^{n-1} \tag{7.4}$$

where "m" is the number of intervals in which each variable is divided and "n" is the number of variables (for the optimizer the variables are the adjustable control points selected while defining the design space for the shape optimization of the blade).
Once the DOE has been generated, the genetic algorithm is applied to the surrogate function to find the global optimum. Then a high-fidelity simulation is executed using the new optimum set of variables; the following operations are carried out for each evaluation: parametrization, mesh generation, setting of the simulation, running the simulation, post-processing and objective evaluation.
The new high-fidelity evaluation is added to the initial database to increase the accuracy and reliability of the surrogate interpolating function, especially in the region close to the optimum. The same procedure is repeated at each iteration and finally the optimization process stops when either the convergence tolerance is met or the maximum number of iterations has been reached.

7.1.5 Code implementation of FORMA

In the previous sections a detailed description was given for each of the main components of the FORMA package, but no information was provided on how all these operations have been implemented from a coding point of view. Three main software work together to perform a high-fidelity shape optimization:

- **Python**: most of the code is written using this open source language; from the many tasks executed, the most important are the geometrical parametrization, the high-fidelity evaluations by interfacing with the CFD software, failure management and communication with DAKOTA for the input-output management;

- **Ansys CFX**: the high-fidelity simulations were performed using this commercial tool made available by Politecnico di Milano; all the details regarding the mesh, the solver, the setting of the simulation and the post-processing have been thoroughly discussed in chapters 3, 5 and 6;

- **Dakota**: this software is a command line multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis [63]. In FORMA, Dakota can be considered as the brain of the optimization procedure: it starts with the design of experiments and communicates with Python for the high-fidelity simulation of each sample;

after completing this step, it moves to the actual search of the optimal shape by building at each iteration a new surrogate function using the Kriging model and then finds the minimum of the objective function through a genetic algorithm; the procedure is finally completed when the convergence tolerance is met or the maximum number of iterations has been reached. The logic of the optimization is implemented by Dakota through a structure composed of different types of blocks (environment, interface, method, model, variable and responses) and the one implemented in FORMA is shown in Figure 7.9.

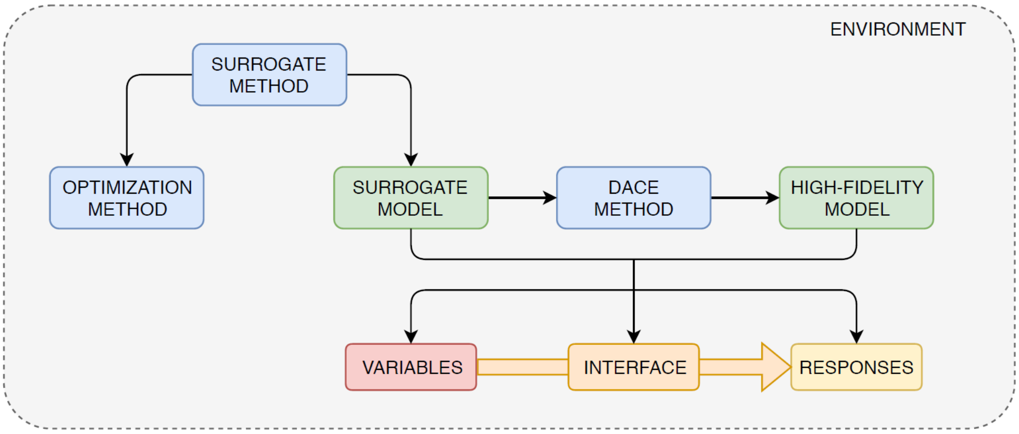


Figure 7.9. Block-structure implemented in Dakota for FORMA (Figure from [22]).

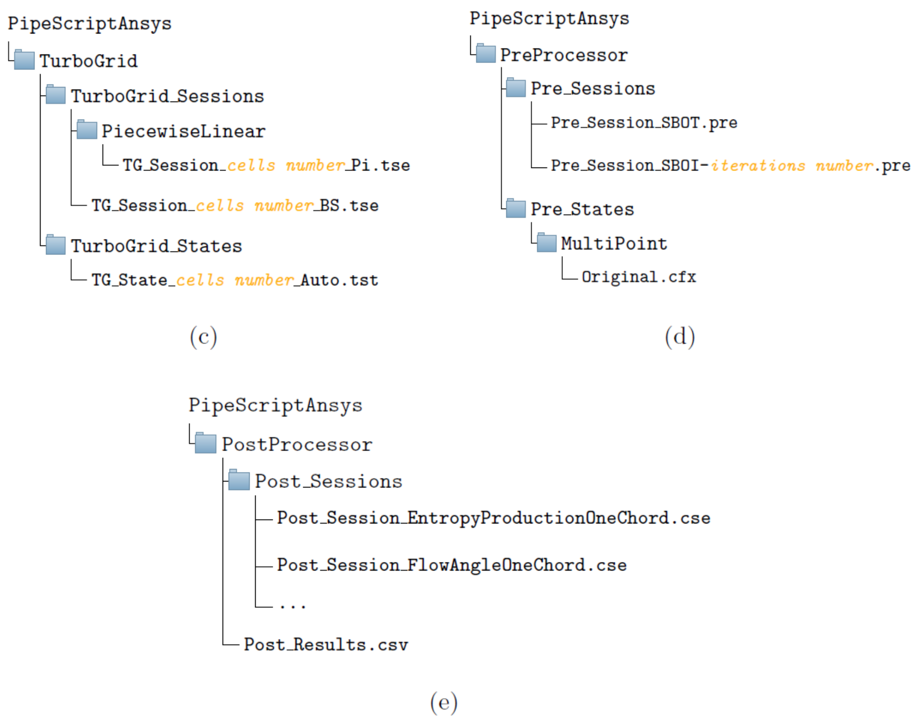
7.2 Stator profile shape optimization by FORMA

In chapter 2, the stator and rotor profile was designed using the vortex-flow method, a specific type of method of characteristics developed by Louis J. Goldman [15]. Then in chapter 5, a completely original automated process was developed to find the optimal solidity: this led to a reduction of entropy production Δs of 25% in the stator and of 5% in the rotor; the improvement in the stator is more substantial than in rotor because the losses is this blade row are significantly larger due to a higher inlet Mach number, which results in a stronger bow-shock wave and its reflections in the channel.
In this section the design of the supersonic turbine is further enhanced through a high-fidelity shape optimization by surrogate-based evolutionary techniques: this procedure will only be applied to the stator, since the losses in the stator are still larger than in the rotor, even after the previous optimization.
As stated in the introduction of the chapter, the FORMA code was used as an advanced user: new features were not developed, but a complete understanding of the source code was necessary to modify and adapt some pieces of the code to apply FORMA for our case.
The shape optimization of a blade profile is an expensive procedure from a computational point of view and requires a powerful CPU; for this reason the code was run on a node of CFDHub, a cluster at Politecnico di Milano. The node has 126 GB of random access memory (RAM) and the CPU is an Intel Xeon E5-2650 v2: it is an octa-core CPU (sixteen threads) with a base frequency of 2.60 GHz and a

maximum frequency of 3.40 GHz. Each complete optimization attempt required on average 48 hours, considering that every high-fidelity simulation for this case takes around 7 minutes: at first glance this could seem really expensive, but considering the widespread availability of High Performance Computing (HPC) at cheaper cost, these types of optimization process have become really common in the engineering field and they are really pushing the limits to find better and better designs.
First of all, it is necessary to prepare all the instructions to automatically mesh the domain, set the simulation, run the simulation and post-process it to evaluate the objective function; even though all this operations have been already recorded in the optimization of chapter 5, those macros were Workbench Journals which can be executed only by Ansys Workbench. Meanwhile, FORMA calls CFX directly in batch mode and then executes the prepared CFX macros; the scripting feature of Ansys CFX is really similar to the one of Ansys Workbench and it is based on two different type of files:

- **states**: these are file in which, as the name suggests, a specific state is saved; this state contains all the parameters that are necessary to define the mesh or to pre-process the simulation;
- **sessions**: the session are the real instructions that the software has to execute; the first command is to load the associated state and after that CFX is instructed to change only those parameters and files that change for each high-fidelity CFD simulation.

FORMA has a very organized folder structure and all the sessions and states must be saved in the correct location, as shown in Figure 7.10.



(c)

(d)

(e)

Figure 7.10. Folder structure for Ansys CFX (Figure from [22]).

The details on the mesh and the setting of the simulation have been thoroughly discussed in chapters 3, 5 and 6; to briefly summarise, the main selected parameters are:

- 100k mesh adopted after a grid independence analysis C;
- the turbulence is described by the $k\!-\!\omega$ SST equations, which require a y^+ lower than 1 near the wall;
- the solver run is initialized using the baseline for a faster convergence;
- the maximum number of iterations are 700, but usually the residuals convergence established at $toll = 2 \cdot 10^{-6}$ is reached before the maximum;
- the blades walls are considered completely adiabatic;
- a high resolution scheme has been used for the advection;
- the minimum possible temperature in the domain has been fixed to 100 K: this option facilitates convergence, especially when working with high velocity flows; obliviously in the post-processing it has been checked that there are no points with this temperature in the domain due this manually fixed limit.

A fundamental aspect to observe is that the geometrical parametrization performed in FORMA works with two dimensional profiles; this is not a real issue for our application because the mean-line code described in chapter 2 also generates a finite thickness two dimensional blade profile. Once the profile has been parametrized, FORMA performs a coordinate conversion because Ansys TurboGrid is built to deal with annular geometries, so the blade to blade planar channel must be converted into an annular channel (for more details on coordinate conversion, see appendix D). To completely eliminate this problem, even if it was not strictly necessary after the coordinate conversion, the stator profile has been shifted to a much higher radius while maintaining exactly the same solidity; using the coordinate conversion equations, it was checked that at this higher radius the profile on the 2D plane and on the cylindrical plane are indistinguishable.

The first step in the optimization procedure are the high-fidelity simulations of each sample of the design of experiments; this operation is highly parallelizable because from a theoretical point of view, given enough computational power, all the CFD simulations could be launched at the same time, since they are independent. Obviously our resources are limited, so it is necessary to find the optimal number of concurrent simulations that minimizes the overall DOE generation time. If the reader is interested, Rossi carried a complete scalability study of FORMA by considering not only the case of a single simulation at a time, but also the many combinations of concurrent simulation for the DOE [22]. Considering that the tests by Rossi were performed on a CPU which is really similar to the one used for our optimization, it was decided for the DOE phase to perform 4 concurrent simulations and to assign to each of them 4 thread for a total of 16 threads.

This complex parallel process was possible thanks to the asynchronous local parallelism option offered by Dakota (Figure 7.11): the term asynchronous refers to the fact that it is not required that all the simulations in a set have been completed to start a new set, but whenever an evaluation process is finished, an additional new one is immediately started [63]. For this reason, a particular care must be taken to maintain the simulations independent to protect the parameters and results files of one simulation from the concurrent ones, which have exactly the same name; a very effective approach offered again by Dakota and employed by FORMA is to create a tagged working subdirectory for each simulation instance.

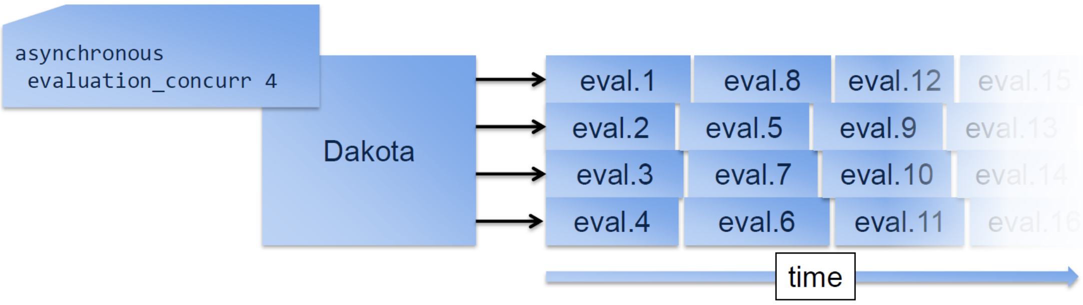


Figure 7.11. Asynchronous local parallelism logic in Dakota.

The number of samples for the DOE was calculated using equation 7.4, but there is also a rule of thumb that suggests that the size of the DOE can be fixed equal to 10 times the number of design variables; this rule could reduce the computational cost to almost a third for the DOE, but then it is necessary to perform a randomness analysis to check the sensibility of the final optimal shape to the seed used to randomly generate the samples; this basically translates in multiple runs of the same configuration with a different seed, which in the end would result in a computational cost that is similar to the complete DOE calculated with equation 7.4, at least for our case.
Many attempts were carried out to find the best possible configuration to give as input to FORMA and here are some of the options that were explored:

- different parametrizations by changing the number and spacing of the control points;
- number of adjustable control points;
- design space definition;
- trailing edge treatment: completely fixed or rigid motion;
- no constraints or constraint on the mass flow averaged absolute angle at stator outlet.

In the next two sections two different optimization configurations will be presented to show the best results obtained during this campaign of FORMA runs. The two cases are representative of two distinct logics as they differ in almost all the options presented above and for this reason both carry some advantages and disadvantages.

7.2.1 Case A

Interpolation

A proper interpolation of the stator profile has a central position in the overall performance of the optimization process because all the other steps depend on the quality of this one. There is not one exact theory on how to select the number of control points or their spacing, but from past experience with the code there are two main criteria to follow:

- the number of control points should be as low as possible, while controlling and containing the error between the original profile and the parametrized one;
- in the region where the shape optimization will be performed, the control points should be as further as possible conceding more freedom to the optimizer and producing shapes that are sufficiently smooth; on the other side if the adjustable control points are too close, the generated shapes are quite odd because these points are moved of a random quantity within the limits of the design space.

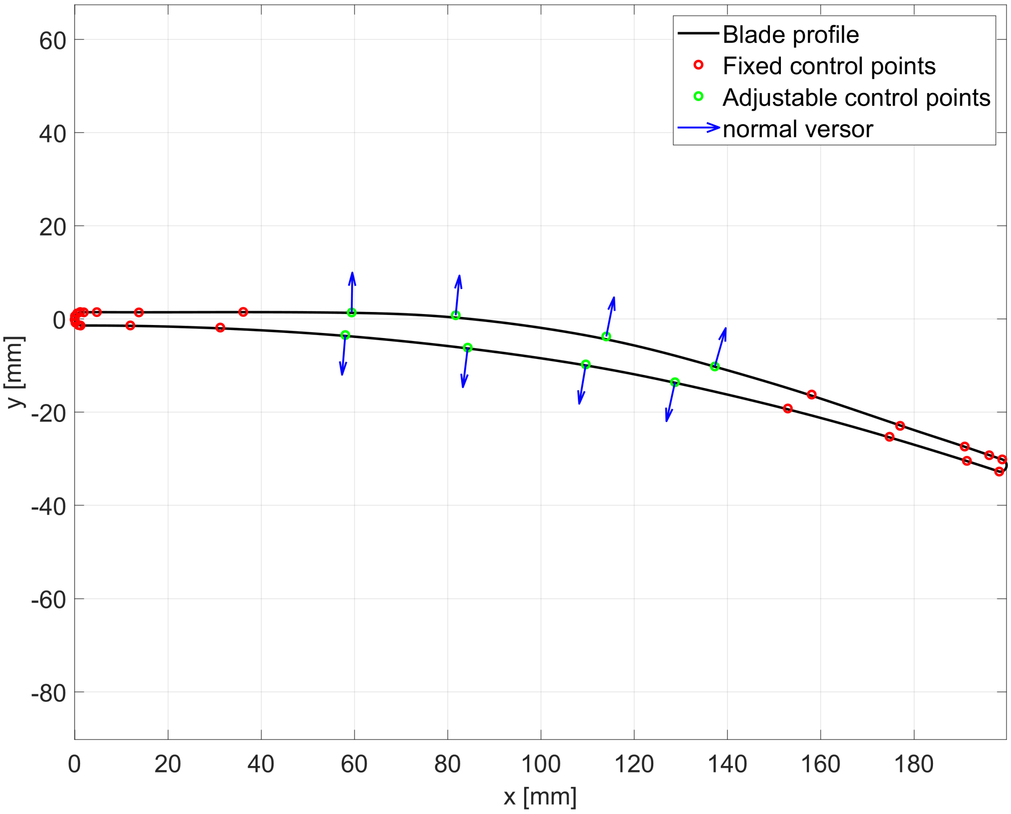


Figure 7.12. Interpolation of the stator profile for case A.

Figure 7.12 reports the interpolation employed for case A. The region selected for the shape optimization is the central part of the profile on both the suction side and the pressure side and for this reason the control points are distant to each other. The leading edge requires many control points because the curvature is much higher in this portion: the use of many control points for this feature is not a real issue because they do not increase the computational cost since they are fixed. For the trailing edge it is selected a circle tangent in the initial point, since this option is exactly the same one that gave finite thickness to the profile in the mean-line code (see subsection 2.4.5). In conclusion a total of 30 control points were used for the geometrical interpolation and the associated mean square error is equal to 0.029.

Selection

The next step is to select the adjustable control points, which are the ones that FORMA will move to find the optimal shape. After that the lower and upper bounds of normal direction motion are defined for each of these points and this whole process creates the design space in which the new optimal shape will be searched. These bounds are carefully decided because if they are too small, FORMA is too restricted and cannot produce significant improvement in the shape, while if they are too large, odd and undesirable shapes are again generated.
Figure 7.13 shows the design space generated in case A, while table 7.1 reports all the numerical data: notice that in case A the trailing edge is completely fixed and this is a very important difference with case B.

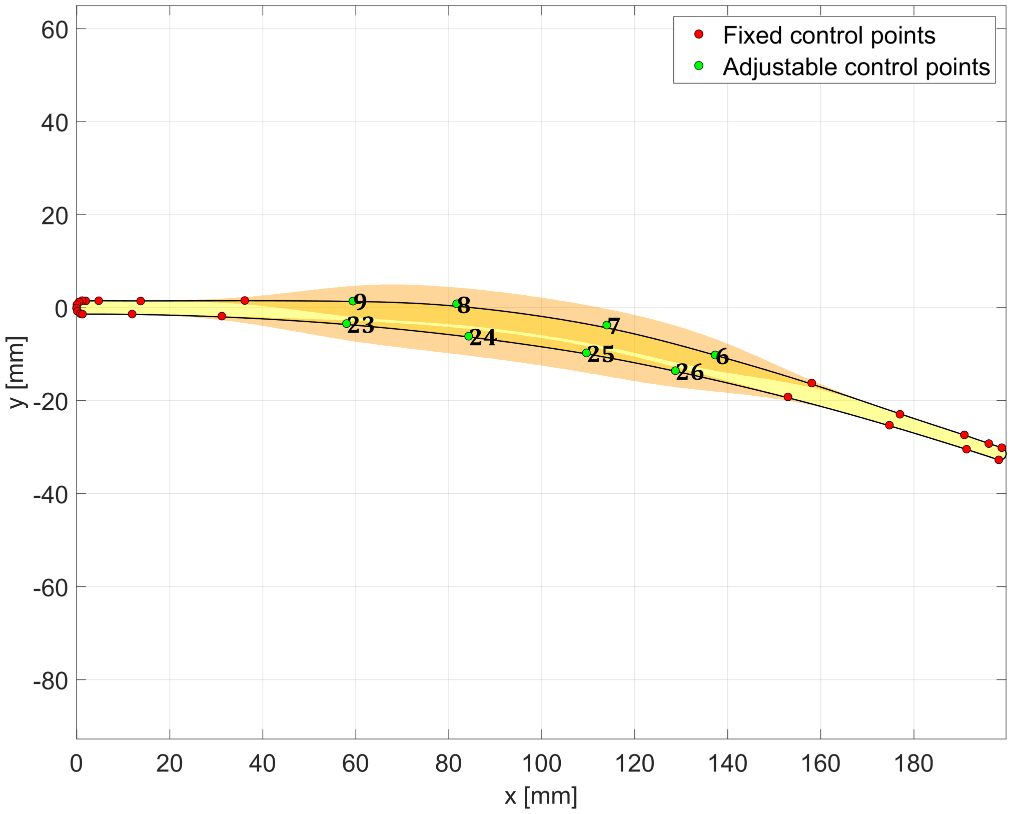


Figure 7.13. Design space definition in case A.

	CP 6	CP 7	CP 8	CP 9	CP 23	CP 24	CP 25	CP 26
lower [mm]	-4	-4	-4	-4	-1	-2	-2	-1
upper [mm]	+4	+4	+4	+4	+4	+4	+4	+4

Table 7.1. Lower and upper bounds for the normal direction motion of the adjustable control points in case A.

Dakota setup

The optimization strategy is the surrogate-based global optimization, which starts with a DOE composed of 256 samples and then continues with 200 iterations of the optimal set of variables given by the genetic algorithm. The objective of the optimization is to minimize the entropy produced in the stator blade row, while no constraint on the outlet flow angle is considered necessary as the trailing edge and the rear portion of the profile are not changed during the optimization. The single-objective genetic algorithm works with a population size of 300 and the maximum number of generations is 500.

Results

Figure 7.14 shows the trend of the objective function with the iterations; the first 256 evaluations are associated to the samples generated by the DOE and already some profiles are performing better than the baseline, as shown by the red line which is the entropy produced by the best individual till that iteration.

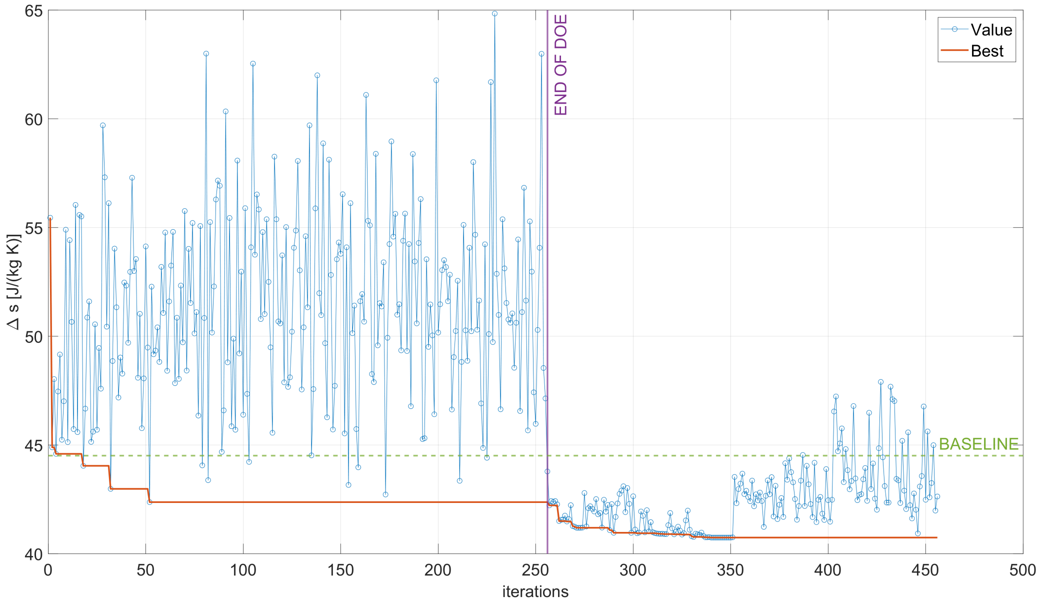


Figure 7.14. Objective function evaluations along iterations for case A.

In the second part the genetic algorithm is applied to the surrogate model and the objective function improves slowly but steadily till iteration 350. After that the

optimizer continues to search if in the design space there is a better profile, by getting even further from the best solution; one can argue that those 100 iterations are a waste of computational cost, but at the same time it is not possible to select a priori the exact number of iterations required to find the optimal profile.
The final optimal profile shown in figure 7.15 produces an entropy increase equal to $\Delta s = 40.74 \ \frac{J}{kg \ K}$, which is **8.16%** lower than the baseline; considering that the shape optimization comes after an advanced blade design through the method of characteristics and a first step of solidity optimization, this improvement is quite good.

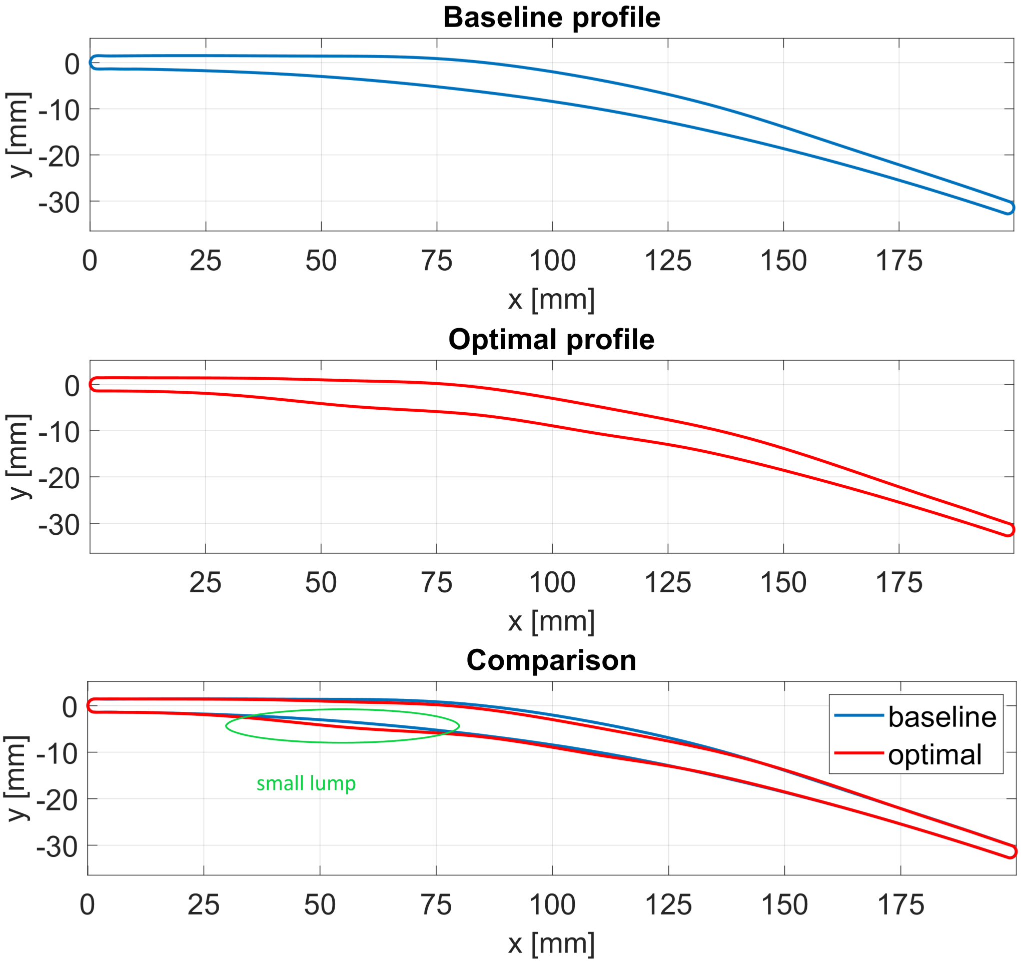
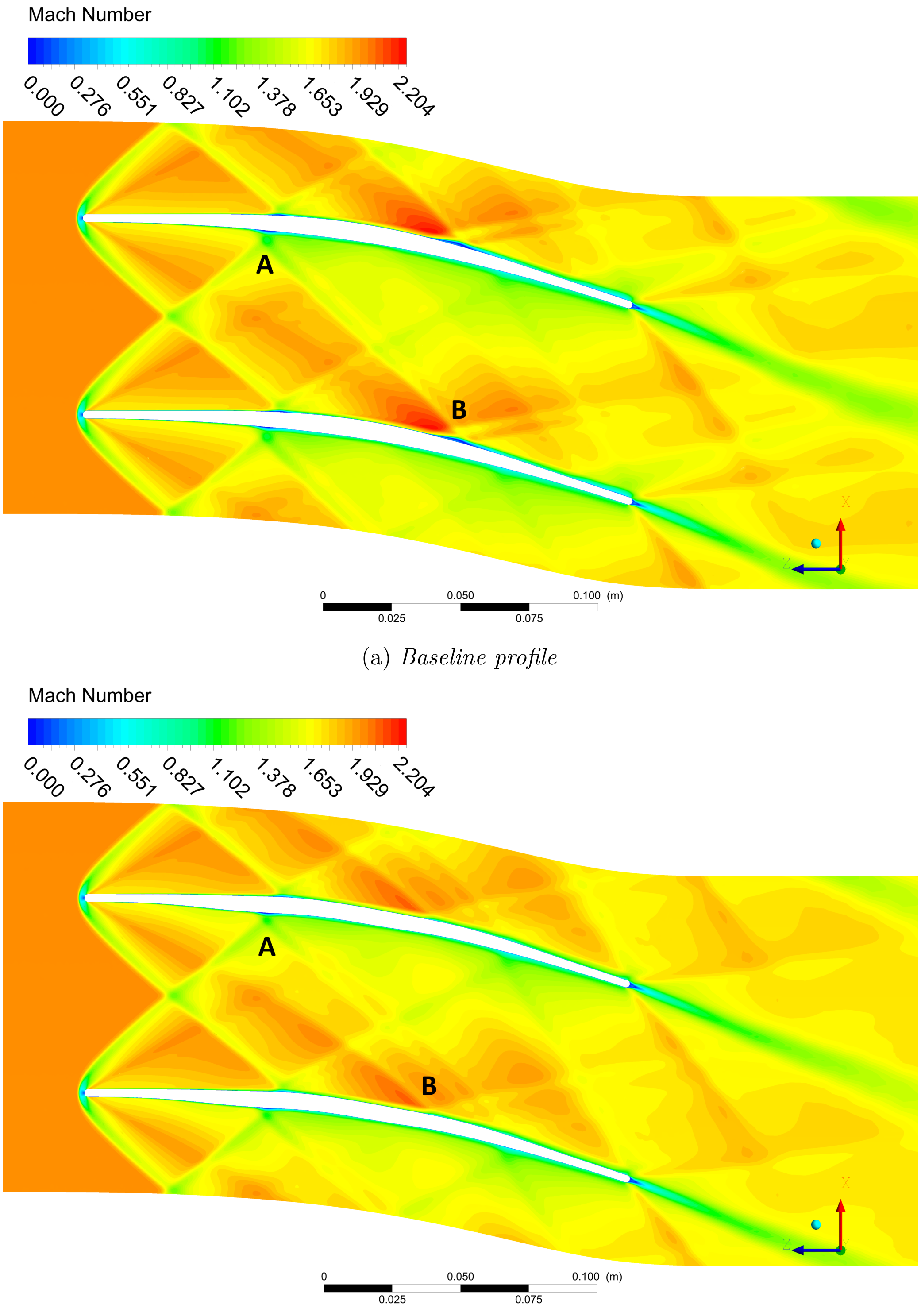


Figure 7.15. Comparison between the baseline profile and the optimal one for case A.

In the optimal profile the suction side curve is lower than the baseline in order to delay the reflection point of the oblique shock wave, while on the pressure side the profile thickens to anticipate the shock. What is really interesting is that the surrogate function created during the optimization is a pure mathematical and not physical equation, but in the end the profile modifies the blade exactly in the reflections points to change the shock pattern and reduce the losses. The effects of these variations are clearly shown with the Mach contour comparison of figure 7.16 between the baseline profile and the optimal one.



(a) *Baseline profile*

(b) *Optimal profile*

Figure 7.16. Mach contours and flow features comparison between the baseline and the optimal profile of case A.

It is immediately clear that the small bubble of separation present on the rear part of the suction side is significantly reduced; the size and therefore also the losses generated by this feature depend mainly on two factors:

- the natural increase of flow speed on the suction side of the blade;
- the strength of the reflected oblique shock wave.

Notice that the optimal profile works on both of these aspects: first of all it modifies the shape of the profile in the area close to the reflection point to reduce the oblique shock wave strength (point A); then on the suction side the profile is changed to moderate the increase of flow speed, as displayed by the colours of the Mach contours (point B).
Because no constraints were forced in the optimization process, it is very important to verify that the outlet flow angle for the optimal profile is not significantly different than the baseline: in that case the performance of the rotor and therefore of the entire turbine would be compromised. The values for the flow angles of equation 7.5 are calculated as mass flow averages on the outlet section and the variation respect to the baseline is considered within acceptable limits.

$$\begin{cases} \alpha_{2-optimal} = 21.8^\circ \\ \Delta\alpha_2 = \alpha_{2-optimal} - \alpha_{2-baseline} = 0.6^\circ \end{cases} \tag{7.5}$$

The main drawback of this new optimal profile is the formation of a small "lump" on the pressure side (Figure 7.15): the optimizer is behaving correctly because it is necessary to modify the local slope of the profile curve to change the inclination of the reflected oblique shock wave. The main issue is that new optimal profile is reducing the losses in design condition, but in off-design the shock pattern is shifted forward and backwards and the small "lump" can become a weakness and not a strength of the shape. Off-design operation is a really important aspect to consider in the design of a turbine for a rotating detonating combustor because the flow is continuously oscillating around the design condition; for this reason in case B, the optimization configuration will be changed to generate a smoother optimal profile.

7.2.2 Case B

Interpolation

In case A the main issue was the presence of a small "lump" due to the fact that the adjustable control points were too close to each other; hence for case B it was decided to reduce the total number of control points and to increase the distance between the adjustable control points; these variations were made to increase the smoothness of the generated profiles and to give enough freedom to FORMA during the optimization.
The trailing edge is again interpolated using a circle tangent in the initial point, but a fundamental difference is that in case B FORMA can rigidly move the trailing edge while preserving its thickness and shape (see subsection 7.1.1). The region in which the shape optimization is performed is not only the central one, but also the rear

portion and the trailing edge; for this reason it was really important to reduce the number of control points to avoid an enormous computational cost for the high-fidelity simulations of the DOE samples.
The leading edge requires many control points to properly interpolate this high curvature feature because there is not a dedicated treatment as for the trailing edge; as already explained before, this is not a problem because all these points are fixed and they do not affect the computational cost for the optimization.
In conclusion a total of 23 control points were used for the geometrical interpolation and the associated mean square error is equal to 0.096 (Figure 7.17).

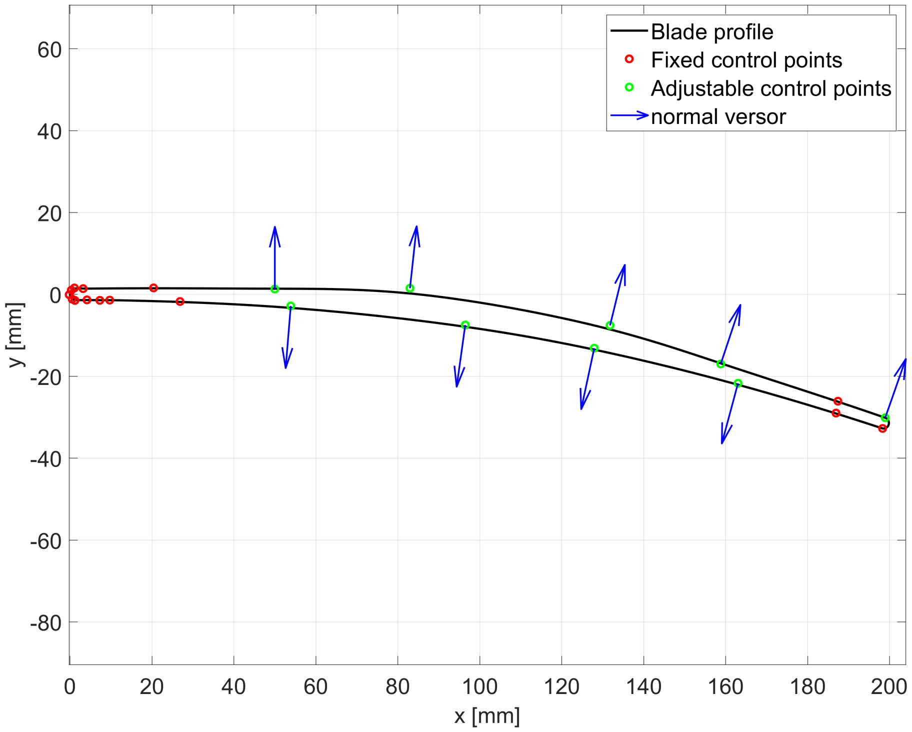


Figure 7.17. Interpolation of the stator profile for case B.

Selection

To generate the design space for case B it was applied the same logic that has already been explained for case A. Some attempts were performed and the final one is shown in figure 7.18, while table 7.2 reports all the numerical data; notice that only one of three points of the trailing edge is actually adjustable, while the other three are rigidly fixed to the motion of the first one to preserve thickness and shape.

CP	1	3	4	5	6	18	19	20	21
lower [mm]	-1.5	-0.75	-1.5	-3	-3	-1.5	-1.5	-2.25	2.25
upper [mm]	+1.5	+1.5	+3	+3	+3	+1.5	+3	+3	1.5

Table 7.2. Lower and upper bounds for the normal direction motion of the adjustable control points in case B.

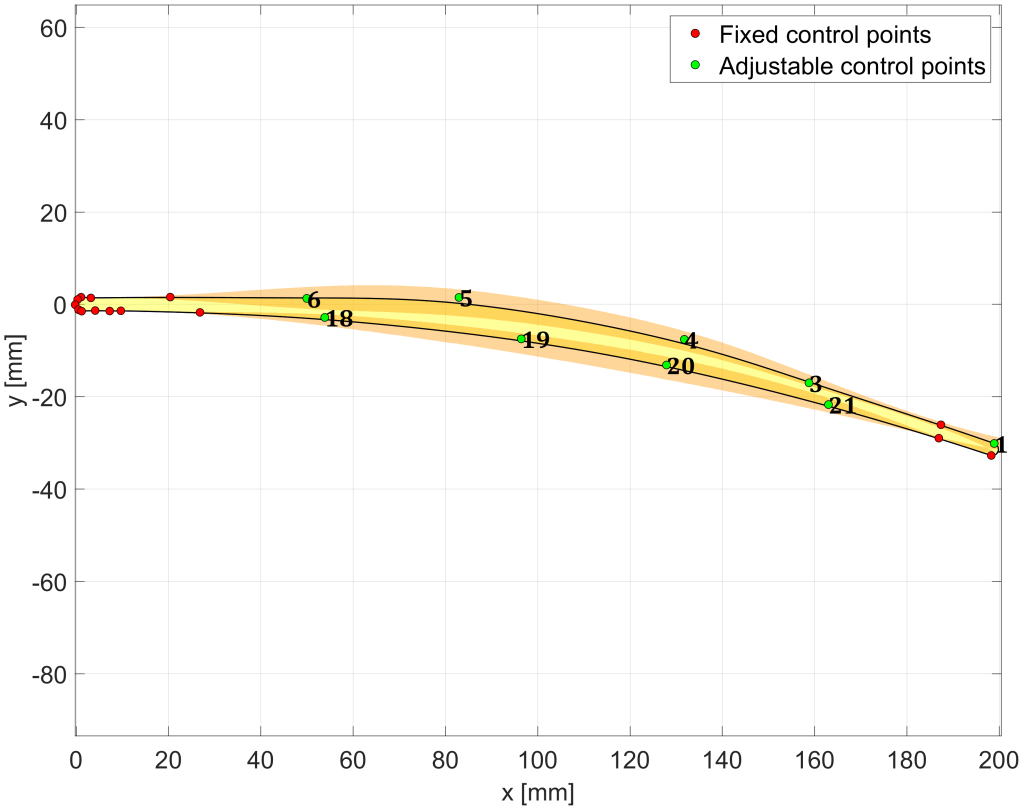


Figure 7.18. Design space definition in case B.

Dakota setup

The setup of Dakota is exactly the same that was used in case A. Actually in the initial attempts, since the trailing edge is no more fixed, it was also imposed a constraint on the outlet flow angle to be within 0.75° of the baseline; observing the results it was noted that during the DOE there were samples that did not respect the constraint, but the final optimal profile had always an outlet flow angle well within range. So in order to get the best possible result, the optimization code was run without any constraint, but during the post-processing it was always verified that the condition on the outlet flow angle was satisfied or not.

Results

Figure 7.19 shows the trend of the objective function with the iterations; as already explained for case A, the first 256 evaluations are associated to the high-fidelity simulations of the samples generated by the DOE, while in the second part the genetic algorithm is applied to the surrogate function to find the optimal set that minimizes the entropy production. Differently than case A, the optimal profile is found in the last iterations: this could mean that if further evaluations were performed, it would have been possible to find an even better profile; considering that in the last 70 iterations the reduction in Δs is really limited, it was finally decided to not relaunch the optimization with a higher number of iterations for the surrogate based global optimization.

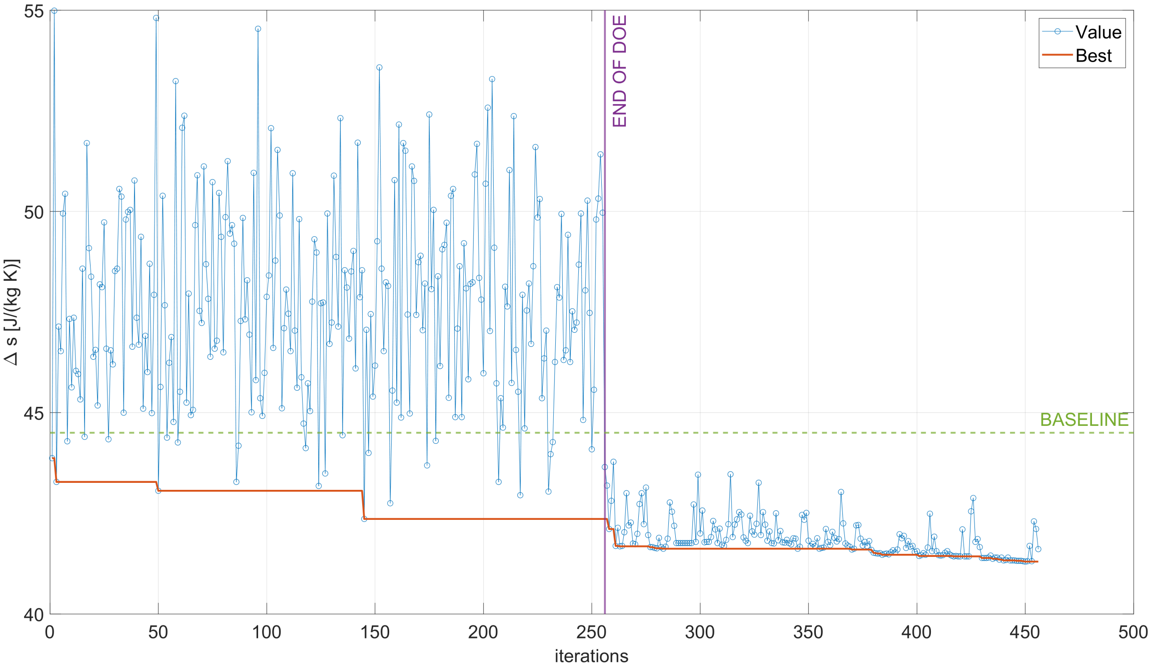


Figure 7.19. Objective function evaluations along iterations for case B.

The final optimal profile, shown in figure 7.20, produces an entropy increase equal to $\Delta s = 41.30 \ \frac{J}{kg \ K}$, which is **7.19%** lower than the baseline; the reduction in losses is lower than case A, but on the other side it is immediately clear that the new optimal profile is definitely smoother than the one of case A.

On the suction side the profile generates a lower curvature as it starts below the baseline profile and finishes above it: this is a beneficial variation because it reduces the increase of flow velocity on the suction side. On the pressure side, the blade is straighter in the area where the bow-shock wave is impinging on the blade in order to anticipate and change the slope of the reflected oblique shock wave.

The most important difference for case B is the trailing edge, as FORMA was allowed to rigidly move it while preserving the shape and thickness. The trailing edge was basically shifted upwards to lower the curvature on both the suction side and the pressure side, while maintaining the inclination of the final portion similar to the baseline: this is really useful to limit the variation of the stator outlet angle respect to the baseline, as proven in the post-processing of the high-fidelity simulation of the optimal profile (Equation 7.6).

$$\begin{cases} \alpha_{2-optimal} = 21.7° \\ \alpha_{2-baseline} = 21.2° \\ \Delta\alpha_2 = \alpha_{2-optimal} - \alpha_{2-baseline} = 0.5° \end{cases} \tag{7.6}$$

The values for the flow angles of equation 7.6 are calculated as mass flow averages on the outlet section and the variation respect to the baseline is considered within acceptable limits.

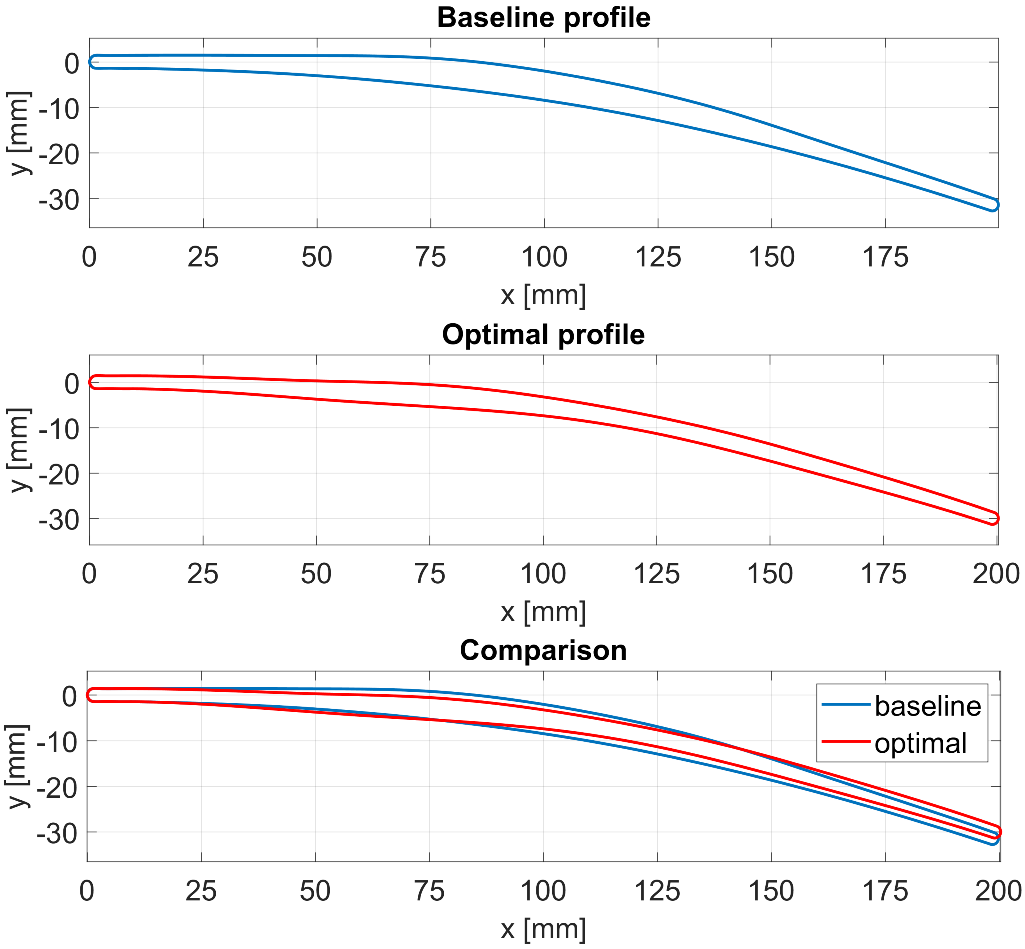
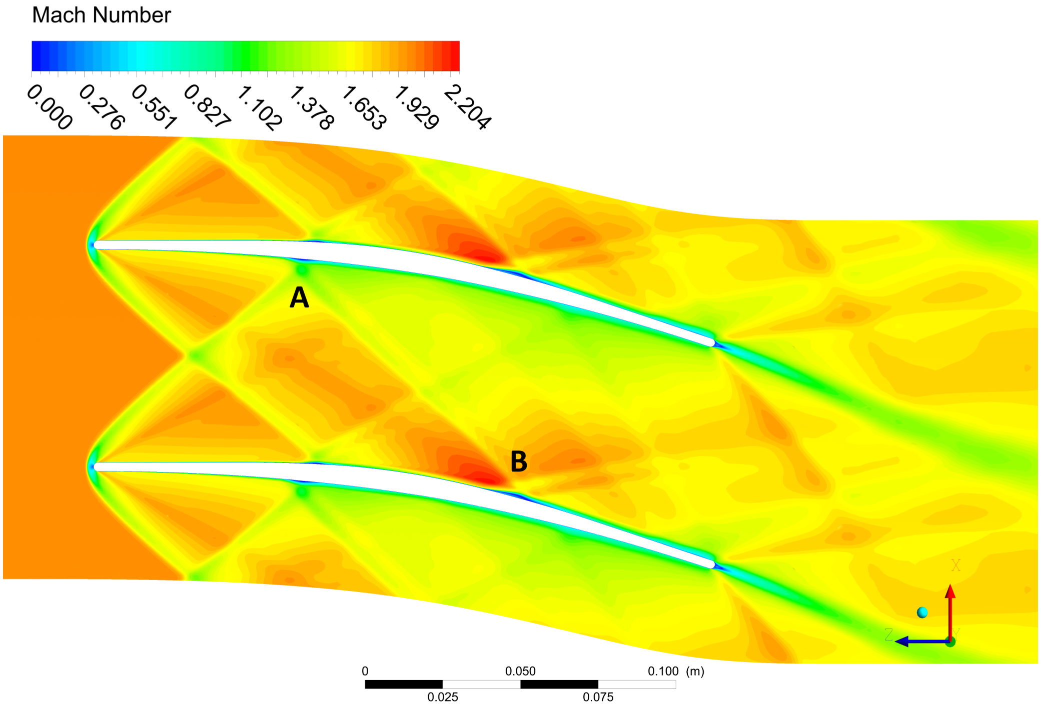


Figure 7.20. Comparison between the baseline profile and the optimal one for case B.

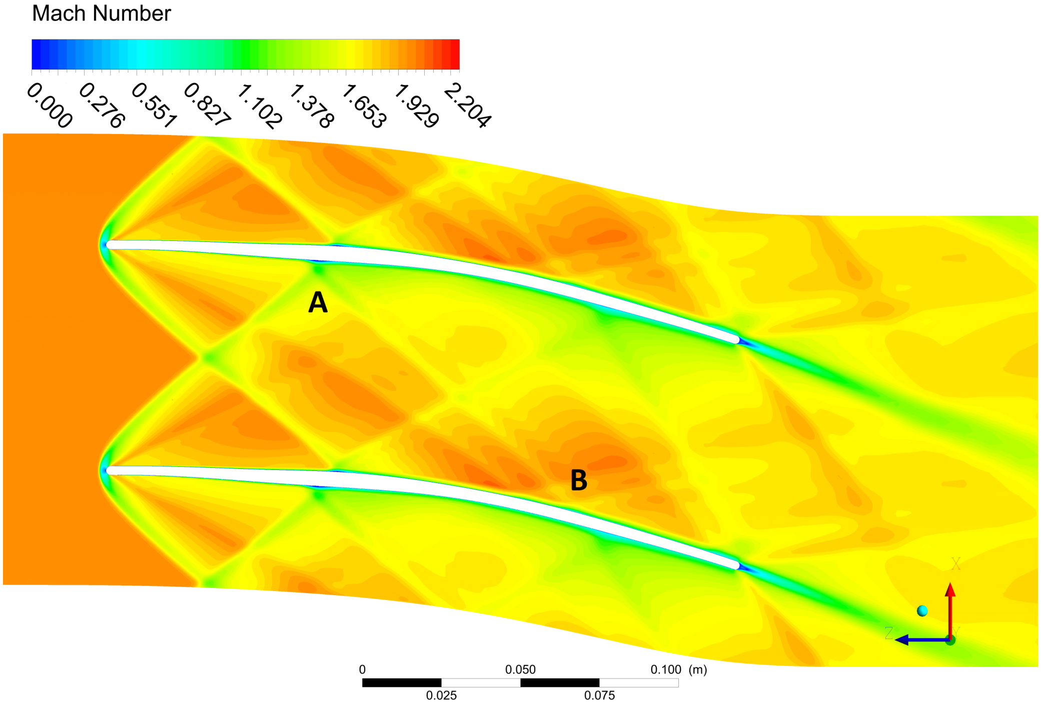
Figure 7.21 shows a Mach contour comparison between the baseline profile and the new optimal one. The small bubble of separation on the suction side of the baseline vanishes for the optimal profile of case B, instead there is just a reasonable increase in the boundary layer; this was again possible thanks to:

- a reduction in flow speed increase with a lower curvature on the suction side (point B);
- a weaker oblique shock wave by locally modifying the profile to obtain a shock with a lower inclination (point A).

The optimal profile of case A was able to guarantee a lower entropy increase than case B, but at the same time the presence of a small lump on the pressure side could have seriously affected the off-design performance of the stator; this is exactly the main strength of the optimal profile of case B, which thanks to smoother curves performs better than the baseline not only in design condition, but works properly also in off-design. The only aspect in the optimal profile of case B that is less desirable is that the central part is thinner, which is not ideal from a mechanical point of view and therefore requires further structural analysis.



(a) *Baseline profile*



(b) *Optimal profile*

Figure 7.21. Mach contours and flow features comparison between the baseline and the optimal profile of case B.

7.3 Some final considerations on the shape optimization

In the previous sections it was noticed how during the shape optimization the shock pattern had a primary role in the profile variation that led to a reduction in losses. Because of this a fundamental question surges: if the shocks are not properly determined and their position is not correct, FORMA is modifying the profile in the wrong areas and all the improvement is completely lost. In the previous chapters, two main actions were taken to avoid this type of error:

- the shock pattern in the CFD simulations were compared to the analytical modelling of the bow-shock wave with Moeckel theory [12] and its first reflection through basic aerodynamics; actually in that occasion, the CFD images were employed to evaluate the reliability of the analytical models (see subsection 2.3.1), but in the same way it is also possible to consider the reliability of the CFD software using the analytical model results;
- the mesh size was selected after a detailed mesh independence analysis (see appendix C).

At this point a final verification was performed to check for a possible inaccuracy on the position of the shocks due to a mesh that is not sufficiently refined. To carry out this validation, the optimal profile of case B was taken as reference; the same profile was tested first with a mesh of 100k cells and then with a more refined one of 300k, while obviously maintaining exactly the same simulation parameters, especially an y^+ lower than 1.

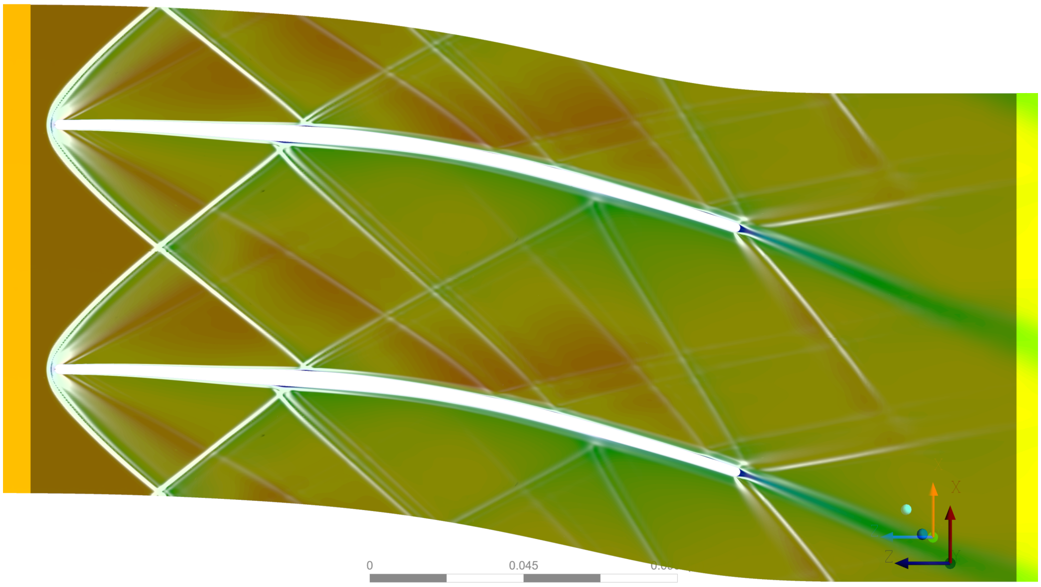


Figure 7.22. Comparison between a 100k mesh Mach contour in background with a 300k mesh density gradient contours in arbitrary linear scale (i.e. numerical-Schlieren visualization) transparent in foreground.

Figure 7.22 is composed of two images: in the background there is a coloured (from blue to red) Mach contour obtained with a 100k cells simulation, while transparent in foreground there is a greyscale (from black to white) density gradient contour from the 300k cells simulation. This image confirms that the mesh used for the shape optimization is perfectly able to determine correctly the shock pattern because even with a much more refined one the shocks remain exactly in the same position.

The shape optimization carried out in this thesis is completed with this final verification, but let's end this chapter with an idea for a possible future development on the topic. The flow arriving from the rotating detonation combustor to the supersonic turbine is oscillating in angle, Mach number, pressure and temperature around the design condition; because these oscillations are symmetric, in this chapter the optimization of the stator was performed in design condition, which is quite reasonable as this is the mean and most frequent work condition of the machine. For an even more advanced shape optimization, a robust multi-point optimization should be considered where the final optimal profile takes account not only of the design condition, but also of the off-design; this new optimal profile will not be as performing as the ones of this chapter in design condition, but the overall losses generated in a complete cycle will be probably lower.

Conclusions

In this thesis work, a supersonic turbine has been designed and a preliminary optimization has been performed. The research field of interest is the development of a Rotating Detonation Engine, a novel technology able to increase the gas turbine performance adopting a detonation combustion mode.

The reason why supersonic turbines are required for this application is that the flow field leaving the RDE is unsteady and supersonic; between the combustor and the turbine there is a diverging nozzle which is vital to obtain better operating conditions by alleviating the thermo-mechanical loading, while favouring a started condition of the supersonic blade row. The outgoing flow is then forced through the turbine to extract work with the maximum efficiency possible. Classical turbine stages are not able to withstand a supersonic field without excessive losses, so novel turbines are needed. The approach followed in this thesis work started by implementing an implicit version of the MOC to design the profile. Actually, this methodology considers expansion and compression characteristic lines in an irrotational, inviscid, adiabatic, steady, two dimensional supersonic flow-field. Because of all these assumptions, the outcomes of the procedures are supersonic blades with sharp leading and trailing edge. Of course, these profiles are not feasible and must be thickened and rounded.

After these operations, a complex shock pattern appears in the real flow-field. A methodology has been implemented and later validated to forecast the strength and the position of the main shock waves. A mean-line code has been written for the design of the full turbine stage in which the shock wave, boundary and mixing losses have been evaluated. In this code the assessment of the starting limit for the supersonic passage is essential, in order to be able to ingest the normal shock wave initially formed in front of the row. For this reason, the rotor blades are designed with a variable height as the outlet annular section is bigger than the inlet one. Afterwards, several parametric analysis have been performed to optimize various turbine parameters, such as the chord length, the rotation speed, the blade thickness, the degree of reaction and so on.

In the second half of the work, an optimization procedure has been implemented to choose the optimal value of Zweifel's tangential force coefficient for both the stator and the rotor. The optimization procedure is based on the Golden-section search algorithm, coded in Python environment, which iteratively evaluate the performance function by running the mean-line code, written in Matlab, importing the output profile curve in Ansys TurboGrid, creating an adequate mesh and, in the end, running the CFD simulation for the blade-to-blade section with Ansys CFX. This approach was necessary because for these pioneering turbines no loading criterium, empirical or semi-empirical, still exists.

After that, the mean-line code has been validated through CFD simulations and then an extensive blade-to-blade simulation campaign has been performed. These simulations allow a better understanding of the complex phenomena involved and give the possibility to assess the turbine performance in off-design condition of both the stator blade alone and the full stage turbine with the mixing-plane technique. The parameters of interest in nominal conditions and in off-design were the entropy production (or an equivalent loss indicator), the incidence angles, the deviation angles and the flow-field variability at the outlet domain section. Unsteady simulations of the rotor blades have been performed by means of the Frozen Gust technique.

Finally, the design of the stator profile has been further enhanced through a high-fidelity shape-optimization by surrogate-based evolutionary technique. The upper and lower curves of the profile were designed with the method of characteristics, but they do not take account of the boundary layer and its interaction with the impinging oblique shock-wave. During the previous extensive campaign of simulations, it was noticed that this interaction can lead to an increase of losses due to the formation of a small bubble of separation behind the shock. Therefore it was decided to further improve the shape of the supersonic profile by employing FORMA: an evolutionary shape-optimization code developed by the laboratory of fluid machines at Politecnico di Milano, based on a geometrical parametrization technique based on B-Splines, a high-fidelity and experimentally validated CFD solver and a surrogate-based evolutionary algorithm. FORMA was applied only to the stator, since the losses in this blade row are still larger than in the rotor, even after the previous optimization. Many attempts were carried out to find the best possible configuration to give as input to FORMA and two different cases were presented as they were representative of two distinct logics: each resulting optimal profile has some strengths and weaknesses, but most importantly the entropy generated by the stator was reduced of around 8%. This final step concludes the preliminary optimization of the turbine and also this thesis.

Many future developments regarding the design of supersonic turbines and their coupling with RDE are possible. All of them are required to make the detonation engines available for improving the gas turbine performances. Starting from this thesis work, a list of hypothetical following steps in this research field is presented below:

- The blade-to-blade simulation campaign must be extended to understand the effect of other inlet condition variations and to find out useful correlations for incidence and deviation angles.

- The mean-line code must be implemented with a deviation angle estimation model. This will allow the code to be more accurate in term of velocity triangles design and efficiency prediction.

- The mean-line code should be adapted to include a spanwise model.

- A complete 3D simulation should be carried out to provide a better estimation not only of the machine efficiency, but also of the effect of rotor height variation and the importance of end-wall losses, clearance losses and secondary flow losses in a supersonic environment.

- A 3D optimization of the stator and rotor rows can be performed, evaluating the possibility to design swept and leaned blade.

- An internal cooling system must be designed. This is an important problem for this novel turbine, if coupled with RDE. Indeed, the bleed air coming from the last compressor stage is at a lower pressure than the burned gas in the turbine, which means that an external cooling is unfeasible. Alternatively an external compressor will be required.
- The blades integrity must be assured with a thermo-mechanical analysis, starting from scratch in term of material designation.
- 3D conjugate heat transfer simulations can be performed adopting more realistic model for viscosity, heat conduction and internal cooling flows.
- A complete unsteady CFD simulation can be performed while giving as input the time-dependent data extrapolated from a rotating detonation combustor.
- A robust multi-point shape optimization of the supersonic profile should be considered, where the final optimal profile takes account not only of the design condition, but also of the off-design.

In summary, it is clear that we are far from unlocking the potential of detonative combustion and a lot of work has to be done before till a gas turbine equipped with a RDE can be made available for the free market. Nevertheless, the interest for this technology is increasing across the globe and optimistic results from experiments and from numerical simulations are already available for supersonic turbines, discussed in this thesis work, for rotating detonation combustors and for thermodynamic cycle optimization.

Appendix A

k-omega SST equations

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho u_j k)}{\partial x_j} = P_k - 0.09\rho\omega k + \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{k3}}\right)\frac{\partial k}{\partial x_j}\right] \tag{A.1a}$$

$$\frac{\partial(\rho\omega)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j\omega - \left(\mu + \frac{\mu_t}{\sigma_{\omega 3}}\right)\frac{\partial \omega}{\partial x_j}\right) = \frac{\alpha_3\omega}{k}P - \beta_3\rho\omega^2 + \frac{(1-F_1)2\rho}{\omega\sigma_{\omega 2}}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} \tag{A.1b}$$

$$P = \left(\mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) - \frac{2}{3}\rho k\delta_{ij}\right)\frac{\partial u_i}{\partial x_j} \tag{A.1c}$$

$$\mu_t = \frac{\rho a_1 k}{\max(a_1\omega, F_2)\sqrt{2S_{ij}S_{ij}}} \tag{A.1d}$$

$$F_1 = \tanh\left\{\left(\min\left[\max\left(\frac{\sqrt{k}}{0.09\omega d}, \frac{500\nu}{d^2\omega}\right), \frac{4\rho\sigma_{\omega 2}k}{\mathrm{CD}_{k\omega}d^2}\right]\right)^4\right\} \tag{A.1e}$$

$$F_2 = \tanh\left\{\left[\max\left(2\frac{\sqrt{k}}{\beta^*\omega d}, \frac{500\nu}{d^2\omega}\right)\right]^2\right\} \tag{A.1f}$$

$$\mathrm{CD}_{k\omega} = \max\left(\frac{2\rho}{\omega\sigma_{\omega 2}}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20}\right) \tag{A.1g}$$

$$\tag{A.1h}$$

Table A.1. Default Coefficient Values in SST model

$\alpha_1 = 5/9$	$\alpha_2 = 0.44$	$\alpha_3 = F_1\alpha_1 + (1-F_1)\alpha_2$
$\beta_1 = 0.075$	$\beta_2 = 0.0828$	$\beta_3 = F_1\beta_1 + (1-F_1)\beta_2$
$\sigma_{k1} = 1.176$	$\sigma_{k2} = 1$	$\sigma_{k3} = F_1\sigma_{k1} + (1-F_1)\sigma_{k2}$
$\sigma_{\omega 1} = 2$	$\sigma_{\omega 2} = 1/0.856$	$\sigma_{\omega 3} = F_1\sigma_{\omega 1} + (1-F_1)\sigma_{\omega 2}$

Appendix B

Coupled Solver Flow Diagram

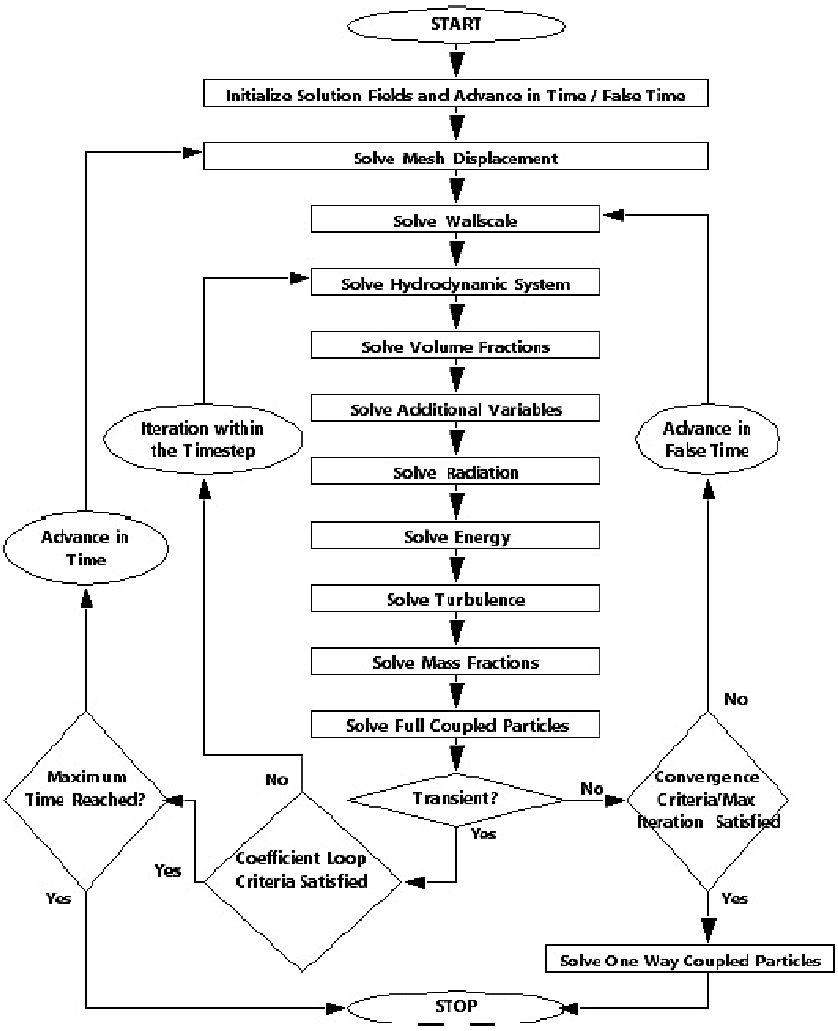

START
Initialize Solution Fields and Advance in Time / False Time
Solve Mesh Displacement
Solve Wallscale
Solve Hydrodynamic System
Solve Volume Fractions
Solve Additional Variables
Solve Radiation
Solve Energy
Solve Turbulence
Solve Mass Fractions
Solve Full Coupled Particles
Iteration within the Timestep
Advance in False Time
Advance in Time
Transient?
No
Yes
Convergence Criteria/Max Iteration Satisfied
No
Yes
Solve One Way Coupled Particles
Coefficient Loop Criteria Satisfied
No
Yes
Maximum Time Reached?
Yes
STOP

Figure B.1. Flow diagram of CFX coupled solver

Appendix C

Mesh Independence Analysis

Discretization error is quantified by systematic refinement of the space and time meshes. In high-quality CFD work, aim is to demonstrate monotonic reduction of the discretization error for target quantities of interest and the flow field as a whole on two or three successive levels of mesh refinement [17]. This process is called *Mesh Independence Analysis*. Sometimes it is possible to accept a coarse assessment of the decrease of a quantity of interest, simply evaluating the reduction in relative error through the mesh refinement process, but in this thesis work a more rigorous methodology have been implemented: the *Grid Convergence Index Method*, described in [64]. It must be specified also that it should not be taken for granted that the actual truncation error in a numerical solution will decay exactly in accordance with the formal order p of accuracy of the basic numerical scheme, since the mesh are not uniform and orthogonal. For this reason, it must be considered an "observed" order of truncation, whose derivation is available [64]. The parameter chosen to monitor the grid convergence is the outlet mass-flow averaged static entropy.
For the stator blade, the mesh analysed are with 21600 (coarse), 34814, 63746, 104486, 188176 (fine) cells. The mesh adopted in this thesis work is the one with 104486 cells (Fig. C.1), which presents a Grid Convergence Index (GCI) equal to 2% and a relative error of 1.4%. The observed order of truncation $\tilde{p}$ is equal to 3.1.

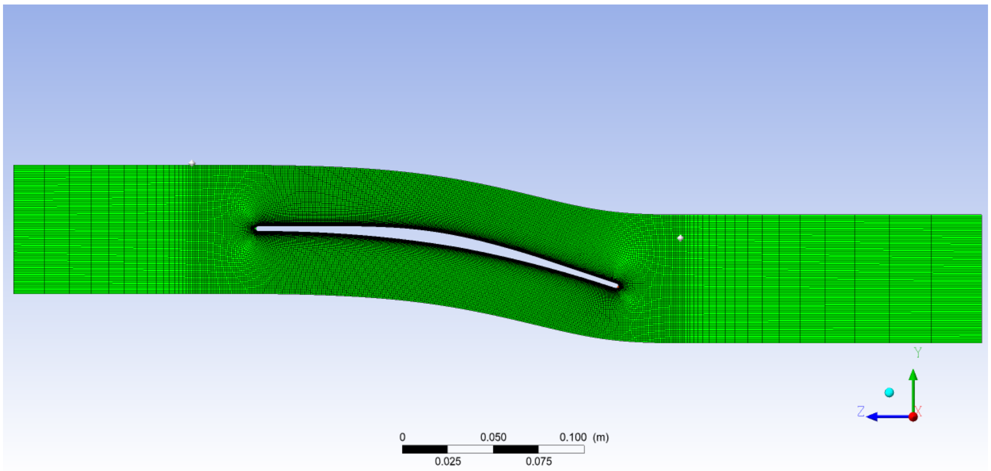


Figure C.1. Stator mesh adopted

For the rotor blade, the mesh analysed are with 30368 (coarse), 53210, 106590, 177906 (fine) cells. The mesh adopted in this thesis work is the one with 106590 cells (Fig. C.2), which presents a GCI equal to 3.67% and a relative error of 2%. The observed order of truncation $\tilde{p}$ is equal to 2.2.

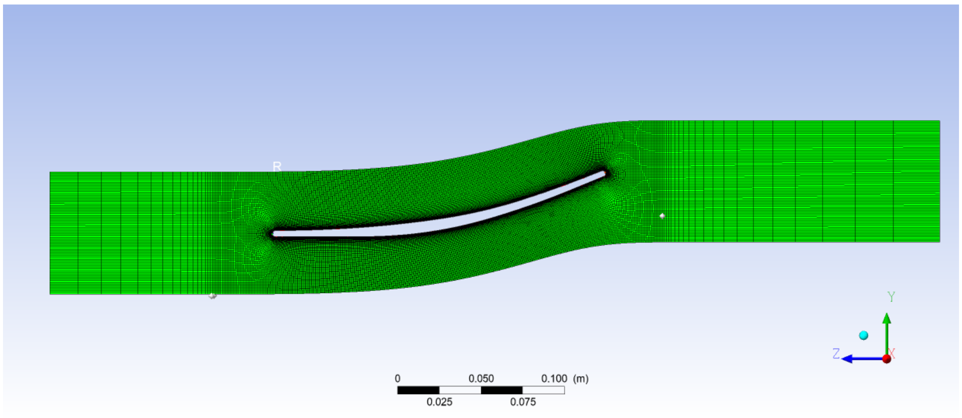


Figure C.2. Rotor mesh adopted

In Fig. C.3 and C.4, it is reported the trend of entropy production parameter for both stator and rotor blades. Monotonic decreasing behaviour of these curves is evident.

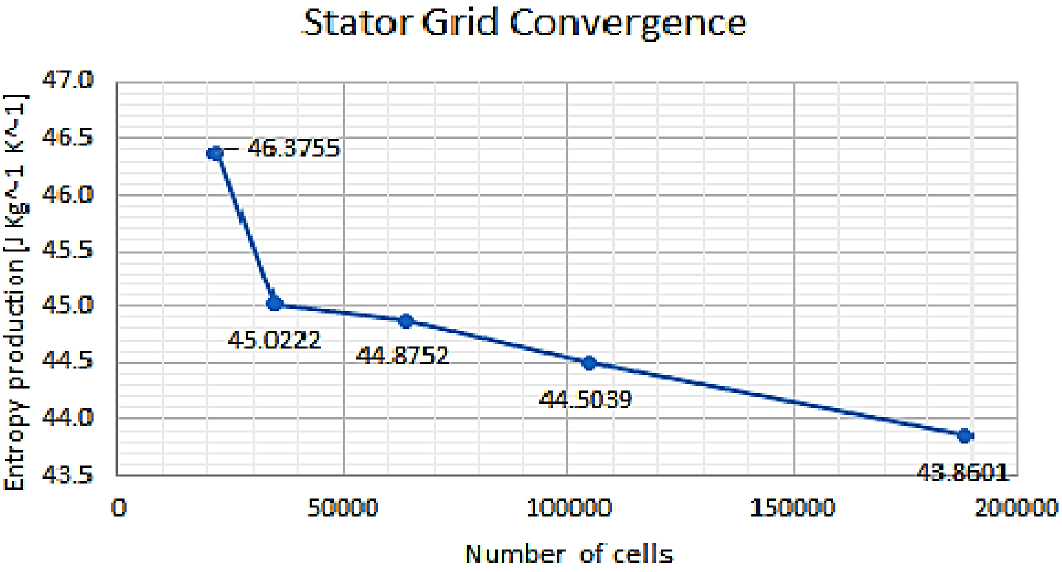


Figure C.3. Trend of monitored parameter for stator

C.1 Mesh refinement in Boundary Layer

Every CFD simulation present in this thesis work, have been carried on, controlling the BL resolution. Indeed, when using the $k-\omega$ SST turbulence model, it is required

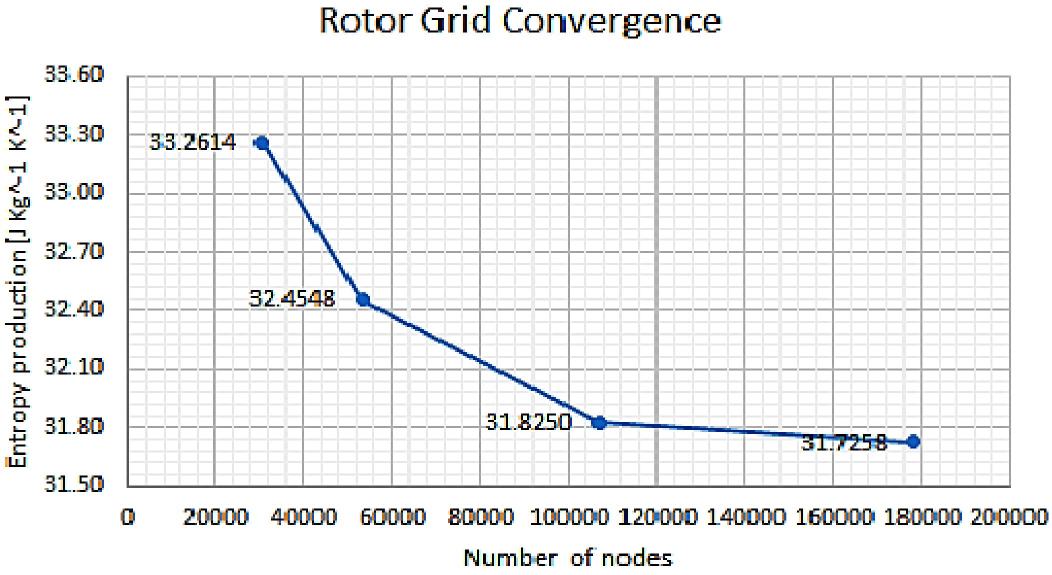
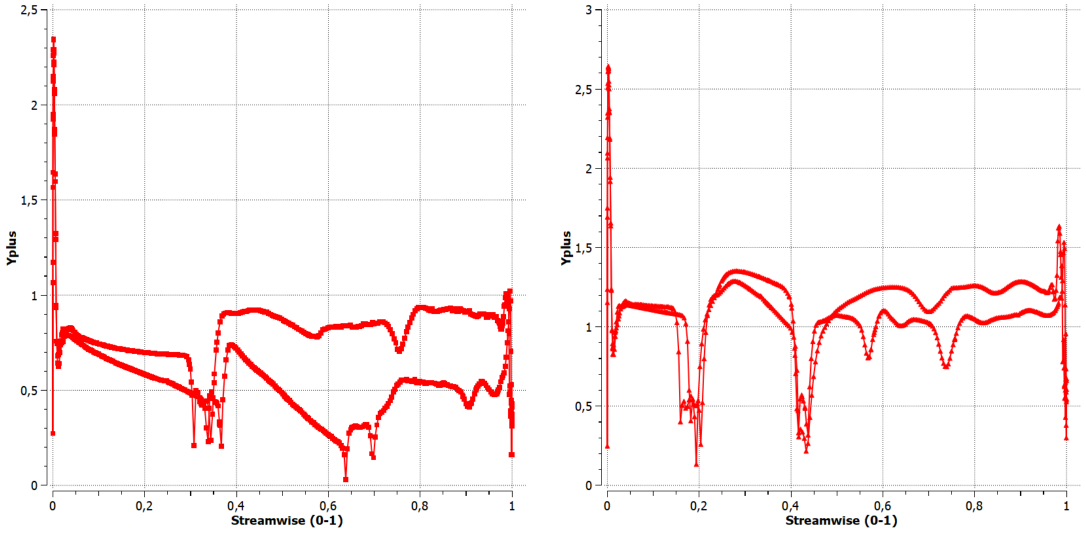


Figure C.4. Trend of monitored parameter for rotor

not to overcome a certain limit of the $y+$ parameter. Although low values of $y+$ lead to mesh elements characterized by an high value of aspect ratio, detrimental for numerics of the CFD code, it is needed to assure at least an average value of $y+$ less than 3. The theoretical background is treated in classical CFD books and in section 3.3. In the following charts, it is reported the value of $y+$ at the blade wall for both stator (Fig. C.6a) and rotor (Fig. C.6b).

Figure C.5. y plus at blade wall vs stream-wise coordinate for (a) stator and (b) rotor.



(a) *y+ mean value equal to 0.668*

(b) *y+ mean value equal to 1.092*

Appendix D

Coordinate conversion

The blade profile is created in a planar surface by means of the MOC and then thickened. This is due to the fact that the MOC itself is intrinsically developed for two dimensional flow and its applications exploit the Cartesian coordinate. Conversely, Ansys TurboGrid is built to deal with annular geometries so the blade to blade planar channel must be transformed into an annular channel with Eq. D.1. In Fig. D.1 it can be appreciate the topological transformation. Actually, the reference radius in these simulations is big enough and the circumferential blade extension small enough to make the difference between the cylindrical and planar profile not relevant.

$$\begin{cases} R = max(y) \\ r\theta = R\arctan\left(\dfrac{x}{y}\right) \\ z = z \end{cases} \tag{D.1}$$

$$\begin{cases} x = R\sin\theta \\ y = R\cos\theta \\ z = z \end{cases} \tag{D.2}$$

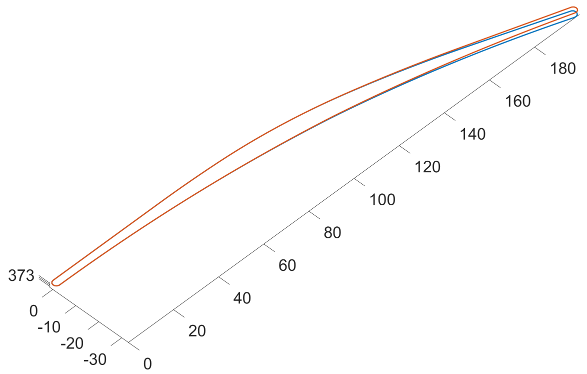


Figure D.1. Stator blade in Cartesian coordinate (red curve) and in cylindrical one (blue curve)

Acronyms

RDE	Rotating Detonation Engine
RDC	Rotating Detonation Combustor
PDE	Pulse Detonation Engine
PGC	Pressure Gain Combustion
GTCC	Gas Turbine Combined Cycle
TIT	Turbine Inlet Temperature
HGP	Hot Gas Path
TBC	Thermal Barrier Coatings
CMC	Ceramic Matrix Composite
CJ	Chapman–Jouguet
FJ	Fickett–Jacobs
CFD	Computational Fluid Dynamic
URANS	Unsteady Reynolds Averaged Navier–Stokes
RANS	Reynolds Averaged Navier–Stokes
MOC	Method of Characteristics
ZND	Zel'dovich - von Neumann - Döring
BL	Bounday Layer
OEM	Original Equipment Manufacturer
FV	Finite Volume
FE	Finite Element
FVM	Finite Volume Method
FEM	Finite Element Method
DNS	Direct Numerical Simulation

LES	Large Eddy Simulation
DES	Detached Eddy Simulation
SAS	Scale-Adaptive Simulation
TKE	Turbulent Kinetic Energy
SST	Shear Stress Transport
RSM	Reynolds Stress Models
EARSM	Explicit Algebraic Reynolds Stress Model
SBES	Stress Blended Eddy Simulation
BSL	Baseline
RNG	renormalisation group
CV	Control Volume
AMG	Algebraic Multigrid
ILU	Incomplete Lower-Upper
HPC	High Performance Computing
DOE	Design Of Experiments
LHS	Latin Hypercube Sampling
SBGO	Surrogate Based Global Optimization
SBLO	Surrogate Based Local Optimization
GCI	Grid Convergence Index

Bibliography

[1] S. C. Gülen, *Gas Turbines for Electric Power Generation*, C. U. Press, Ed., 2019.

[2] P. Wolański, “Detonation engines,” *Journal of KONES*, vol. 18, no. 3, pp. 515–521, 2011.

[3] ——, “Detonative propulsion,” *Proceedings of the Combustion Institute*, vol. 34, no. 1, pp. 125–158, 2013.

[4] G. Paniagua, M. C. Iorio, N. Vinha, and J. Sousa, “Design and analysis of pioneering high supersonic axial turbines,” *International Journal of Mechanical Sciences*, vol. 89, pp. 65–77, 2014.

[5] J. Sousa, G. Paniagua, and E. Collado Morata, “Thermodynamic analysis of a gas turbine engine with a rotating detonation combustor,” *Applied Energy*, vol. 195, pp. 247–256, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2017.03.045

[6] D. Schwer and K. Kailasanath, “Numerical Study of the Effects of Engine Size n Rotating Detonation Engines,” no. January, 2011.

[7] J. Braun, J. Saavedra, and G. Paniagua, “Evaluation of the unsteadiness across nozzles downstream of rotating detonation combustors,” *AIAA SciTech Forum - 55th AIAA Aerospace Sciences Meeting*, no. January, pp. 1–12, 2017.

[8] E. M. Braun, F. K. Lu, D. R. Wilson, and J. A. Camberos, “Airbreathing Rotating Detonation Wave Engine Cycle Analysis,” *AIAA*, no. July, 2015.

[9] J. Sousa, G. Paniagua, and J. Saavedra, “Aerodynamic response of internal passages to pulsating inlet supersonic conditions,” *Computers and Fluids*, vol. 149, pp. 31–40, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.compfluid.2017.03.005

[10] J. Sousa and G. Paniagua, “Entropy minimization design approach of supersonic internal passages,” *Entropy*, vol. 17, no. 8, pp. 5593–5610, 2015.

[11] E. A. Bufi, “Optimisation robuste de turbines pour les cycles organiques de Rankine (ORC),” 2017.

[12] Moeckel, “Approximate method for predicting form and location of detached shock waves ahead of plane or axially symmetric bodies,” *NASA TN 1921*, vol. 211, no. 1, p. 130, 1921.

[13] J. D. Anderson, "Modern compressible flow with historical perspective." 1982.

[14] W. L. Stewart, "Analysis of two-dimensional compressible-flow loss characteristic downstream of turbomachine blade rows in terms of basic boundary-layer characteristics," *NACA Technical note 3515*, 1955.

[15] L. J. Goldman and M. R. Vanco, "Computer Program for Design of Two-Dimensional Sharp-Edged-throat Supersonic Nozzle with Boundary-Layer Correction," *NASA*, 1971.

[16] C. Sun, H. Zheng, Z. Li, N. Zhao, L. Qi, and H. Guo, "Effects of diverging nozzle downstream on flow field parameters of rotating detonation combustor," *Applied Sciences (Switzerland)*, vol. 9, no. 20, 2019.

[17] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics - The Finite Volume Method*, 2nd ed. Pearson, 2007, vol. 6, no. 4.

[18] C. Robinson, S. Street, and L. Ln, "A New Streamline Curvature Throughflow Method for Radial," pp. 1–12, 2020.

[19] Ansys, "ANSYS CFX-Solver Theory Guide," pp. 724–746, 2020.

[20] J. D. Denton and W. N. Dawes, "Computational fluid dynamics for turbomachinery design," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering*, vol. 213, no. 2, pp. 107–124, 1999.

[21] K.-H. Chang, *Design Theory and Methods using CAD / CAE - The Computer Aided Engineering*, 2015.

[22] A. Rossi, "Three-dimensional shape-optimization of a transonic gas-turbine cascade by surrogate-based evolutionary techniques," 2018.

[23] S. C. Gülen, "Pressure gain combustion advantage in land-based electric power generation," *Journal of the Global Power and Propulsion Society*, vol. 1, 2017.

[24] W. Fickett and W. C. Davis, *Detonation - Theory and Experiment*, 1979, vol. 95.

[25] K. Kailasanath, "Review of Propulsion Applications of Detonation Waves," *AIAA JOURNAL*, vol. 38, no. 9, 2000.

[26] I. J. Shaw, J. A.C. Kildare, M. J. Evans, A. Chinnici, C. A.M. Sparks, S. N.H. Rubaiyat, R. C. Chin, and P. R. Medwell, "A Theoretical Review of Rotating Detonation Engines," *IntechOpen*, 2019.

[27] S. M. Frolov, A. V. Dubrovskii, and V. S. Ivanov, "Three-dimensional numerical simulation of operation process in rotating detonation engine," *Progress in Propulsion Physics*, vol. 4, pp. 467–488, 2013.

[28] R. Zhou and J. P. Wang, "Numerical investigation of flow particle paths and thermodynamic performance of continuously rotating detonation engines," *Combustion and Flame*, vol. 159, no. 12, pp. 3632–3645, 2012.

[29] D. A. Schwer and K. Kailasanath, "On reducing feedback pressure in rotating detonation engines," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition 2013*, no. January, 2013.

[30] D. M. Davidenko and A. N. Kudryavtsev, "Hydrogen-Oxygen Detonation With a Detailed," no. November, 2007.

[31] J. Braun, J. Sousa, and G. Paniagua, "Numerical assessment of the convective heat transfer in rotating detonation combustors using a reduced-order model," *Applied Sciences (Switzerland)*, vol. 8, no. 6, 2018.

[32] T. H. Yi, J. Lou, C. Turangan, J. Y. Choi, and P. Wolanski, "Propulsive performance of a continuously rotating detonation engine," *Journal of Propulsion and Power*, vol. 27, no. 1, pp. 171–181, 2011. [Online]. Available: https://doi.org/10.2514/1.46686

[33] S. B. Yao, M. Liu, and J. P. Wang, "The Effect of the Inlet Total Pressure and the Number of Detonation Waves on Rotating Detonation Engines," *Procedia Engineering*, vol. 99, pp. 848–852, 2015.

[34] T. H. Yi, J. Lou, C. Turangan, B. C. Khoo, and P. Wolanski, "Effect of nozzle shapes on the performance of continuously rotating detonation engine," *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, no. January, pp. 1–9, 2010.

[35] J. Braun, B. H. Saracoglu, and G. Paniagua, "Unsteady performance of rotating detonation engines with different exhaust nozzles," *Journal of Propulsion and Power*, vol. 33, no. 1, pp. 121–130, 2017.

[36] F. K. Lu, E. M. Braun, and J. Powers, "Rotating detonation wave propulsion: Experimental challenges, modeling, and engine concepts," *Journal of Propulsion and Power*, vol. 30, no. 5, pp. 1125–1142, 2014.

[37] D. E. Paxson, "Examination of wave speed in rotating detonation engines using simplified computational fluid dynamics," *AIAA Aerospace Sciences Meeting, 2018*, no. 210059, pp. 1–13, 2018.

[38] Z. Liu, J. Braun, and G. Paniagua, "Performance of axial turbines exposed to large fluctuations," *53rd AIAA/SAE/ASEE Joint Propulsion Conference, 2017*, no. July, pp. 1–15, 2017.

[39] J. Sousa, G. Paniagua, and E. Collado-Morata, "Analysis of the aerodynamic losses in a supersonic turbine," *American Society of Mechanical Engineers, Power Division (Publication) POWER*, vol. 1, pp. 1–7, 2017.

[40] J. Sousa, J. Braun, and G. Paniagua, "Development of a fast evaluation tool for rotating detonation combustors," *Applied Mathematical Modelling*, vol. 52, pp. 42–52, 2017. [Online]. Available: https://doi.org/10.1016/j.apm.2017.07.019

[41] Z. Liu, J. Braun, and G. Paniagua, "Characterization of a Supersonic Turbine Downstream of a Rotating Detonation Combustor," *Journal of Engineering for Gas Turbines and Power*, vol. 141, no. 3, pp. 1–13, 2019.

[42] J. Braun, Z. Liu, D. Cuadrado, V. Andreoli, G. Paniagua, J. Saavedra, V. Athmanathan, and T. R. Meyer, “Characterization of an integrated nozzle and supersonic axial turbine with a rotating detonation combustor,” *AIAA Propulsion and Energy Forum and Exposition, 2019*, no. August, pp. 1–11, 2019.

[43] L. J. Goldman and V. J. Scullin, “Analitycal Investigation of Supersonic Turbomachinery Blading - I - Computer Program for Blading Design,” *NASA TN D-4421*, 1968.

[44] L. J. Goldman, “Analytical Investigation of Supersonic Turbomachinery Blading - II - Analysis of Impulse Turbine-Blade Sections,” *NASA TN D-4422*, 1968.

[45] V. J. Scullin and L. J. Goldman, “Computer Program for Design of two-dimensional Rotor Blades with Boundary-Layer Correction,” *NASA TM X-2434*, 1971.

[46] A. J. Glassman, “Turbine Design and Application - Volume 2,” Tech. Rep., 1973.

[47] A. J. Glassman and L. J. Goldman, “Turbine Design and Application - Volume 3,” Tech. Rep., 1975.

[48] B. S. Stratford and G. S. Beavers, “The Calculation of the Compressible Turbulent Boundary Layer in an Arbitrary Pressure Gradient - A Correlation of certain previous Methods,” *Aeronautical Research Council Reports*, no. 3207, 1961.

[49] A. Kantrowitz and C. Donaldson, “Preliminary inversitagtion of supersonic diffusers,” *NACA Wartime reports*, 1945.

[50] W. F. Cope, “Notes and graphs for boundary layer calculations in compressible flow,” *A.R.C Current paper n°89*, 1951.

[51] A. H. Shapiro, *Dynamic and thermodynamics of compressible flow*, 1953, vol. I.

[52] G. B. Schubauer and W. G. Spangenberg, “Forced mixing in boundary layers,” *Journal of Fluid Mechanics*, 1960.

[53] G. Persico, P. Rodriguez-Fernandez, and A. Romei, “High-Fidelity Shape Optimization of Non-Conventional Turbomachinery by Surrogate Evolutionary Strategies,” *Journal of Turbomachinery*, 2019.

[54] P. K. Kundu and I. M. Cohen, *Fluid Mechanics*, 2008.

[55] J. H. Ferziger and M. Peric, *Computational Method for Fluid Dynamics*, 3rd ed. Springer, 2002.

[56] C. Hirsh, *Numerical Computation of Internal & External Flows*, 2007.

[57] Ansys, “ANSYS Fluent Theory Guide,” pp. 1–759, 2020.

[58] R. H. Aungier, *Turbine Aerodynamics - Axial-flow and Radial-inflow turbine design and analysis*, 2005.

[59] M. T. Schobeiri, *Turbomachinery Flow Physics and Dynamic Performance*, 2012.

[60] Gerald Farin, *Curves and Surfaces for CAGD: A Practical Guide*, 2002.

[61] J. Arora, *Introduction to Optimum Design*, 2012.

[62] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. Kevin Tucker, "Surrogate-based analysis and optimization," *Progress in Aerospace Sciences*, vol. 41, no. 1, pp. 1–28, 2005.

[63] B. M. Adams, M. S. Ebeida, M. S. Eldred, J. D. Jakeman, L. P. Swiler, J. A. Stephens, D. M. Vigil, and T. M. Wildey, "DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis Version 5.4 User's Manual SAND2010-2183," Tech. Rep., 2009.

[64] I. B. Celik, U. Ghia, P. J. Roache, C. J. Freitas, H. Coleman, and P. E. Raad, "Procedure for estimation and reporting of uncertainty due to discretization in CFD applications," *Journal of Fluids Engineering, Transactions of the ASME*, vol. 130, no. 7, pp. 780 011–780 014, 2008.