National Technical University of Athens
School of Mechanical Engineering
Fluids Section
Laboratory of Thermal Turbomachines
Parallel CFD & Optimization Unit

The Continuous Adjoint Method with Consistent Discretization Schemes for Transitional Flows and the Use of Deep Neural Networks in Shape Optimization in Fluid Mechanics

PhD Thesis

Marina G. Kontou

Supervisor: Kyriakos C. Giannakoglou,
Professor NTUA

Athens, 2023

National Technical University of Athens
School of Mechanical Engineering
Fluids Section
Laboratory of Thermal Turbomachines
Parallel CFD & Optimization Unit

The Continuous Adjoint Method with Consistent Discretization Schemes for Transitional Flows and the Use of Deep Neural Networks in Shape Optimization in Fluid Mechanics

PhD Thesis

Marina G. Kontou

Examination Committee:

1. Kyriakos Giannakoglou*(Supervisor), Professor, NTUA,
   School of Mechanical Engineering
2. Nikolaos Aretakis*, Associate Professor, NTUA,
   School of Mechanical Engineering
3. Konstantinos Mathioudakis*, Professor, NTUA,
   School of Mechanical Engineering
4. Spyridon Voutsinas, Professor, NTUA,
   School of Mechanical Engineering
5. Francesco Montomoli, Professor, Imperial College London,
   Department of Aeronautics
6. Vasilios Riziotis, Associate Professor, NTUA,
   School of Mechanical Engineering
7. Gilbert Rogé, HDR, Dassault Aviation,
   Département de Mathématiques Appliquées et Informatique Numérique, Sorbonne Université

**Member of the Advisory Committee*

Athens, 2023

Abstract

In this PhD Thesis, methods and tools for shape optimization in fluid dynamics are developed, assessed and applied. Emphasis is laid on (a) the mathematical formulation, programming and assessment of the continuous adjoint method for shape optimization in transitional flows of compressible fluids, (b) the development of consistent discretization schemes for the adjoint PDEs and adjoint boundary conditions involved in continuous adjoint and (c) the use of Deep Neural Networks (DNNs) in flow predictions and demonstration of the expected gain in shape optimization. The two first subjects are related to Gradient-Based (GB) optimization, whereas the last one is associated with the more efficient use of Evolutionary Algorithms (EAs).

Throughout this Thesis, the in-house GPU-enabled (U)RANS solver PUMA, developed by the Parallel CFD & Optimization Unit of the NTUA, is used. The compressible flow solver of PUMA, previously developed for turbulent flows, is extended to transitional flows. The $\gamma-\tilde{R}e_{\theta t}$ transition model is coupled with both the $k-\omega$ SST and the Spalart-Allmaras turbulence models. A series of 2D or 3D cases, in internal and external aerodynamics, are used for the purpose of comparison either with experimental or numerical results. These cases include the transitional flow over flat-plates and the flow through a highly loaded transonic turbine blade which is evaluated using aerothermal criteria. Regarding external aerodynamics, an isolated airfoil and two wings, all with Natural Laminar Flow are used.

The adjoint method is used to compute the Sensitivity Derivatives (SDs) of any objective function with respect to (w.r.t.) the design variables in the most efficient way, in a GB optimization. Herein, the continuous adjoint method is mathematically developed for the $\gamma-\tilde{R}e_{\theta t}$ transition model coupled with the Spalart-Allmaras turbulence model, for the first time in the corresponding literature. The "frozen transition" adjoint (i.e. skipping the adjoint to the transition model equations) is evaluated as a possible cheaper alternative and is shown to noticeably damage the accuracy of SDs.

The main advantages of the continuous adjoint method is the clear physical insight into the adjoint PDEs and their boundary conditions and, in particular, the low memory footprint of the adjoint code. The discretization schemes used for the adjoint equations can significantly affect the accuracy of SDs; ideally these should be consistent with the discretized flow (primal) problem. In discrete adjoint, consistency is not an issue; the main weaknesses are the excessive memory footprint of the adjoint code and the lack of a clear interpretation of the adjoint equations. This PhD Thesis bridges the gap between the two adjoint variants by proposing consistent discretization schemes (inspired by discrete adjoint) for the continuous adjoint PDEs and their boundary conditions, with a clear physi-

cal meaning. The new *Think-Discrete Do-Continuous* (*TDDC*) adjoint is developed for the time-marching hyperbolic-type solver of compressible fluid flows of the PUMA code. The *TDDC* adjoint computes SDs as accurate as discrete adjoint, ensuring though low memory footprint and physical insight into the adjoint problem. The *TDDC* adjoint outperforms the continuous adjoint that makes use of the "standard" discretization schemes and the more accurate SDs result in improved optimization behavior.

The continuous adjoint method for transitional flows, that also makes use of the developed consistent discretization schemes, is used to carry out two industrial optimizations, that of a wing and that of a high aspect-ratio wing business jet configuration.

In its last part, which focuses on cost reduction in shape optimization problems, this PhD Thesis develops and uses DNNs for flow prediction and shape optimization. A first approach is to train DNNs and use them to replace the numerical solution of the turbulence and/or transition model PDEs by providing the turbulent viscosity field in each pseudo–time iteration of the RANS equations' solver. The optimal DNN configuration (architecture and input data) is found from a Metamodel–Assisted Evolutionary Algorithm (MAEA). The resulting cost–effective solver is used as an evaluation tool during the shape optimization of an isolated airfoil, a turbomachinery cascade and a car model using the MAEA of the EASY (Evolutionary Algorithm SYstem) platform.

A second approach is to use DNNs in multi-disciplinary and multi-row turbomachinery shape optimization problems. The use of the two-branch $\lambda-$DNN architecture is proposed; its distinguishing feature is the separate processing of input data of different nature. Consequently, the $\lambda-$DNN is ideal for use in multi-disciplinary problems by replacing one or more disciplines. Herein, it is used in Conjugate Heat Transfer (CHT) shape optimization problems by replacing the numerical solution of the heat conduction equation solver in the solid material. The new, less computationally expensive solver is used as an evaluation tool in two CHT shape optimizations: an S-Shaped cooling channel and an internally cooled turbine blade-airfoil. Finally, for multi-row turbomachinery optimization problems, a DNN surrogate to the mixing plane technique is proposed. Overall, the use of DNN-assisted solvers reduces the cost of optimizations using MAEAs and the EASY platform, in specific.

Keywords: Aerodynamic Shape Optimization, Continuous Adjoint, Discretization Schemes for the Adjoint Equations, Transition Modeling, Deep Neural Networks, Surrogates for Turbulence Closure, Evolutionary Algorithms

Acknowledgments

The development related to the *Think-Discrete Do-Continuous* adjoint was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "2nd Call for H.F.R.I. Research Projects to support Faculty Members & Researchers" (Project Number: 3821).

Special thanks to Dassault Aviation, and personally to Dr. Gilbert Rogé, for providing the data and for the close cooperation in the framework of the NEXTAIR project funded by the European Union under Grant Agreement No. 101056732.

Research on the implementation of DNNs for the mixing plane technique took place in the framework of the REGALE project co-funded by the General Secreteriat for Research and Innovation of the Ministry of Development and Investments under the EuroHPC JU, EU Grant Agreement No. 956560. Data were provided by Andritz Hydro and, in specific, Dr. Evgenia Kontoleontos.

I would also like to thank the Greek Research and Technology Network High Performance Computing Services for providing the computational means to conduct the research related to the $\lambda-$DNN, through the CGT-DNN Project.

Thanks

I would like to express my gratitude to those people who supported me during my PhD Thesis. First of all, I would like to sincerely thank my supervisor, Kyriakos Giannakoglou, Professor NTUA, for his passion, knowledge and unique way of teaching, as well as his continuous guidance and support over the last years. I am grateful to him for the faith he showed in me at the very beginning.

In addition, I would like to thank Konstantinos Mathioudakis, Professor NTUA, and Nikolaos Aretakis, Associate Professor NTUA, for their comments on the Thesis presentation and their participation in the advisory committee of my Thesis.

I would also like to thank Dr. Varvara Asouti and Dr. Xenofon Trompoukis for the guidance and support they offered my throughout my Thesis. Their knowledge, their willing to share it and their insightful suggestions and priceless advice were of paramount importance for me, especially the moments when things were far from what they were expected to be. Moreover, I would like to express my gratitude to Dr. Konstantinos Tsiakas and Dr. Dimitrios Kapsoulis for their guidance and assistance at the beginning of my PhD Thesis. I would also like to thank Dr. Evangelos Papoutsis-Kiachagias for all our fruitful discussions. I am grateful to Dr. Themistoklis Skamagkis, Andreas Margetis, Dr. Morteza Monfaredi, Christos Karliampas and Nikolaos Galanos for their company and support. Special thanks to Konstantina Kovani for the trust she showed in me.

I would also like to thank my brother for the clear and unclouded way he approaches life, my mother for her persistence and sincerity and my father for the naturalism and practicality he faces every situation. Special thanks to my friends and close to me people for the light they shed even without knowing it. Last but not least, I would like to thank Dimitris for being next to me.

We live everything as it comes,
without warning, [...]

The Unbearable Lightness of Being,
Milan Kundera

Abbreviations

ABC	Adjoint Boundary Conditions
ANN	Artificial Neural Network
CFD	Computational Fluid Dynamics
CHT	Conjugate Heat Transfer
DB_{DNN}	DataBase of the Deep Neural Network
DNN	Deep Neural Network
DPA	Double Precision Arithmetics
EA	Evolutionary Algorithm
EASY	Evolutionary Algorithms SYstem
FAE	Field Adjoint Equations
FDs	Finite Differences
FI	Field Integral
FSI	Fluid Solid Interface
GB	Gradient-Based
GPU	Graphics Processing Unit
l.h.s	Left-Hand Side
LTT	Lab of Thermal Turbomachines
MAEA	Metamodel Assisted EA
ML	Machine Learning
MPA	Mixed Precision Arithmetics
NTUA	National Technical University of Athens
NURBS	Non-Uniform Rational B-Splines
PCOpt	Parallel CFD & Optimization (Unit)
PDE	Partial Differential Equation
PUMA	Parallel Unstructured Adjoint Multi-row
RANS	Reynolds-Averaged Navier-Stokes
RANS-TM	RANS equations coupled with Turbulence/Transition Model
RANS-DNN	RANS equations coupled with Deep Neural Network
r.h.s	Right-Hand Side
SDs	Sensitivity Derivatives
SLSQP	Sequential Least SQuares Programming
SPA	Single Precision Arithmetics
TDDC	Think-Discrete Do-Continuous
w.r.t.	with respect to

Contents

Contents

Chapter 1

Introduction

Air transportation is among the most important segments of the transportation sector and is expected to witness significant growth in the upcoming years. However, its growth is concomitant with increased fuel consumption and, as a result, increased greenhouse gases. Although the aeronautical sector has taken actions towards environmental sustainability, these are not enough for meeting the Sustainability Development Goals (SDGs) for affordable and green energy by the year of 2030[1] and climate neutrality by the year of 2050[2].

During the last decades, there has been a turn in the wing's design for improved efficiency and reduced environmental footprint that includes the so–called Natural Laminar Flow (NLF) wings, [87]. The idea behind the NLF wings is to extend the area over the wing where the flow is laminar leading to reduced drag and, as a consequence, environmental sustainability. In the direction of green aircraft, in the field of turbomachines, the ability to correctly capture transition effects is essential in order to accurately compute losses and heat transfer on engine components. A critical study of laminar-turbulent transition phenomena and its role in aerodynamics and heat transfer in modern and future gas turbine engines is presented in [93].

Consequently, the development of a computational tool that can accurately capture the transitional phenomena with low computational cost -in terms of time and resources- is essential. These phenomena should also be taken into account and, thus, incorporated into optimization loops used for new designs.

On the other hand, Artificial Intelligence (AI) has penetrated into the everyday life due to its ability to detect input data properties and patterns and make accurate predictions with low computational cost, [13]. Automatic translation [145, 148, 125], image recognition [79, 97] and autonomous driving [115, 12, 85] are some of the common aspects of AI, nowadays. Moreover, over the recent years,

[1] https://undocs.org/A/RES/70/1

[2] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52019DC0640

AI has also been utilized by the medical sector significantly improving diagnosis, treatment planning and patient care [123, 89]. It can be seen that the use of AI by the Computational Fluid Dynamics (CFD) community is a natural step in the direction of cost reduction (without, though, damaging accuracy) in engineering problems, [131].

This PhD Thesis addresses the aforementioned topics and aims at developing accurate and cost–efficient numerical methods and computational tools capable of carrying out real–world analysis and optimization tasks.

1.1 Laminar-to-Turbulent Transition Modeling

Turbulence modeling is an everyday routine in the majority of the modern CFD codes. Various turbulence models are capable of capturing the effect of turbulence and can successfully be applied to a wide range of applications. This is not, though, the case for transition modeling. Apart from the extra computational cost, the other reason for the absence of transition modeling in common CFD simulations is that laminar-to-turbulent transition occurs through different mechanisms in different applications; including all these effects in a single transition model is challenging.

There are three main mechanisms of transition: natural, bypass and separation -induced transition. The natural transition is due to the so–called Tollmien-Schlichting (TS) instabilities [120]; these are developed in the laminar boundary layer and then proceed to fully turbulent flow. Practically, 2D TS instabilities are initially developed and then are amplified to 3D instabilities and vortices. Natural transition is usually present in wings. In turbomachinery applications, and in specific in gas turbines, large disturbances in the external flow, caused by high turbulence intensity levels, bypass the natural transition and the TS instabilities. This is described as the bypass transition [100]. In the last mechanism, the separation-induced transition [88], the transition occurs in a separated laminar boundary layer and may or may not include TS instabilities. Occasionally, the flow can re-attach as turbulent by forming a laminar-separation/turbulent-reattachment "bubble" on the surface. This type of transition is usually present in gas turbines, compressors and low-pressure turbines.

In the literature, there are various ways to model laminar-to-turbulent transition, i.e. algebraic expressions [27], boundary layer transition modeling [126], transport equations [72], Large Eddy Simulations (LES) [91] and Direct Numerical Simulations (DNS) [35].

Regarding the algebraic approach, the intermittency is usually determined as an analytical/empirical expression. Based on the theoretical work of Emmons in 1951, [39], there is a turbulence spot production function that is related to the probability of the flow to be turbulent; a first analytical expression of this function was proposed by Dhawan in 1958, [27]. Different approaches were presented in

[10, 40]; the latter was applied for the design of wind turbine airfoils. Recently, an algebraic model relying exclusively on local flow variables was proposed in [15] where the production term of the Spalart–Allmaras turbulence model was multiplied with an intermittency distribution function. Based on that, the production of turbulence was damped until some turbulence onset criteria were satisfied. The model was calibrated in a series of flat plates and used in 2D and 3D internal and external aerodynamic applications. An algebraic transition model for use with the $k-\omega$ SST turbulence model was proposed in [68, 69] and applied in a 3D turbomachinery case.

Stability theory is a common alternative for transition modeling; the most widely used method is the so–called e^N of Smith [126] and van Ingen [137]. Based on that, the transition onset occurs where the amplitude of the most unstable frequency exceeds the initial unstable amplitude by the factor of e^N. The value of N, though, is user-defined and usually depends on the freestream conditions; a typical range is between 8 and 10 with 9 being widely used for simulating natural transition phenomena. A thorough review regarding the e^N method can be found in [138]. Although it has been successfully applied in a wide range of applications [30, 29], the necessity to have access to information regarding the boundary layer development is its main flaw. In order to compute this quantity, an integration in the normal direction at every wall node is required. The need for non-local information increases the computational cost and makes its parallelization difficult.

In this direction, there was a significant effort in the corresponding literature to develop transition models that could predict the laminar-to-turbulent transition exclusively based on local quantities (quantities available in every CFD code, i.e. nodal density, velocity components, pressure etc.). Practically, this involved the development and assessment of transition models in the form of transport equations that can be solved in a similar manner to the mean flow and the turbulence model equations.

The first and most widely used transition model that ensures compatibility with modern CFD codes (unstructured grids and parallel execution) was the $\gamma-\tilde{Re}_{\theta t}$ model proposed by Langtry and Menter, [72, 73, 71]. The model solves for two transport equations, that of the intermittency γ and that of the transition momentum-thickness Reynolds number $\tilde{Re}_{\theta t}$ and is coupled with the $k-\omega$ SST turbulence model. The $\gamma-\tilde{Re}_{\theta t}$ was accepted by the research community and various modifications were proposed for the expression of the empirical correlations for the $k-\omega$ SST model, [86, 94], and for its extension to the Spalart–Allmaras turbulence model [23, 22, 113]. A deficiency of the $\gamma-\tilde{Re}_{\theta t}$ transition model is that it is not Galilean invariant due to the explicit use of the velocity vector. In order to overcome this, while also avoiding the solution of the $\tilde{Re}_{\theta t}$ equation and simplifying the empirical correlations, the one-equation γ model was proposed in [96]. An additional approach is the Amplification Factor Transport (AFT) model, [19, 20, 18], based on which an additional transport equation for the amplification

factor for streamwise instabilities is proposed. The model relies on the stability theory of the e^N (as proposed in [32]) and is fully compatible with modern CFD codes as it is based on local information.

As already stated, higher fidelity modeling, such as LES [91, 82, 110] and DNS [35, 149, 82, 102] can be also used to capture transitional effects. The DNS computation requires the solution of the time–dependent Navier–Stokes equations, without the need of Reynolds averaging. Consequently, there is no need for turbulence/transition closure and all the corresponding phenomena are captured due to the grid fineness. These methods are associated with high computational cost which is often prohibitive for use in real–world large–scale problems. In addition, the development of the continuous adjoint method (which is one of the targets of this PhD Thesis) is out of question for LES in order to be routinely used by the industry.

A thorough review regarding transition modeling and their compatibility with CFD codes can be found in [28, 31].

1.2 Shape Optimization Methods in Fluid Dynamics

Aerodynamic shape optimization problems can be carried out using either stochastic, population–based or deterministic (gradient–based) optimization methods. The former search for the optimal solution(s) in a systematically randomized way while the latter exploit the gradient of the objective function(s) in order to locate the optimal solution(s). A literature survey for both optimization methods follows.

Evolutionary Algorithms

Evolutionary Algorithms (EAs) are the more representative population–based, gradient–free stochastic optimization methods and are inspired from the theory of evolution of species proposed by Darwin [24]. EAs handle populations of candidate solutions (rather than a single solution in each optimization step) which evolve from generation to generation using evolution operators such as crossover, mutation and elitism. The individuals (i.e. candidate solutions) of the current generation which are best adapted in their environment (i.e. have better performance) are more likely to be selected to reproduce and the offspring of the new generation (i.e. new candidate solutions) may be even better adapted. The main advantage of EAs is that they can solve any optimization problem, without requiring knowledge of the problem or access to its source code. It suffices to have a software for evaluating the candidate solutions; this is treated as a black–box and to distinguish it, this will be referred to as the Problem Specific Model (PSM). EAs are also advantageous compared to other optimization methods as they can easily deal with constrained and Multi–Objective Optimization (MOO) problems to

compute (Pareto) fronts of non–dominated solutions, [146]. Moreover, EAs can converge to global extrema, regardless of the initial population, on condition that a sufficient number of candidate solutions is evaluated on the PSM.

On the other hand, optimization problems carried out with EAs are associated with high turnaround time due to the excessive amount of calls to the PSM which is needed to reach the global extrema. In large scale optimization problems, such as the CFD-based ones, where the PSM is computationally demanding, the cost of an EA optimization can be prohibitively expensive. To overcome this, Artificial Neural Networks (ANNs) or other regression models are used as metamodels, giving raise to the so–called Metamodel-Assisted EAs (MAEAs). Metamodels are surrogate models that can approximate the response of the PSM at negligible cost. They are categorized as off-line or on–line trained ones [42], depending on whether they are not or are connected to the evolution.

MAEAs with off–line trained metamodels make use of a single metamodel, valid for the entire search space. This is trained on a group of patterns generated based on a Design of Experiment (DoE) technique, [99, 14], and evaluated on the PSM. The EA is then exclusively based on the off–line trained metamodel to evaluate the candidate solutions at negligible cost. The "optimal" solution(s) are re-evaluated on the PSM and the optimization is either terminated or resumed (if necessary, the metamodel is updated) based on convergence criteria.

Regarding MAEAs with on–line trained metamodels, in each population and for each population member a "local/personalized" metamodel is trained. The MAEA relies on the synergistic use of the on–line metamodel and the PSM tool. PCOpt/LTT has proposed a MAEA based on Low–Cost Pre–Evaluation (LCPE), [61]. In LCPE, the on–line metamodel is used to pre–evaluate all population members and then the PSM tool is used to evaluate only the most promising among them.

Techniques to reduce the computational cost of EAs involve distributed (DEAs) and hierarchical (HEAs) search schemes. In DEAs, a number of sub–populations evolve in semi–isolation both aiming at improved exploration of the design space and exchange information. HEAs utilizes more than one evaluation tool, i.e. CFD solvers of different accuracy and computational cost. For example, in a two–level configuration, the low–level uses the low–cost, low–fidelity evaluation tool to undertake the exploration of the design space; then, the most promising individuals are forwarded to the high–level where are evaluated on the high–cost, high–fidelity tool and are further evolved. The LCPE phase has already been extended to Distributed and Hierarchical EAs/MAEAs (DEAs/DMAEs and HEAs/HMAEs) or to a combination of them (DHEAs/DHMAEAs), [56, 55].

In general, the presence of many design variables slows the evolution due to the so–called "curse of dimensionality". In MAEAs, the cost for training the metamodel(s) becomes higher, the start of the LCPE phase is delayed and overall, the quality of the metamodel predictions is questionable. To overcome this, unsupervised learning techniques based on linear, [70], and kernel, [59, 58], Principal

Component Analysis (PCA) are used in this context. The PCA can be used both for applying the evolution operators in the feature space and for truncating the metamodel's inputs during the LCPE phase. The aforementioned variations are available in the general-purpose optimization platform EASY (Evolutionary Algorithm SYstem), [4], developed by the PCOpt/NTUA and, also, used in part of this PhD Thesis.

Gradient–Based Methods

Gradient–Based (GB) methods require the gradient of the objective function, also known as Sensitivity Derivatives (SDs), while, occasionally, additional information is needed regarding the (exact or approximated) second derivatives of the objective function with respect to (w.r.t.) the design variables for use in efficient (quasi)–Newton methods, [105]. The main advantage of deterministic methods, compared to the stochastic ones, is that they are more computationally efficient, [41]. Their main drawback, though, is that they are more likely to be trapped into local extrema, depending on the starting point of the optimization.

The most significant part in GB optimization methods is the accurate computation of the SDs. In optimization problems governed by Partial Differentiable Equations (PDEs), the SDs can be computed using Finite Differences (FDs), the Complex Variable Method (CVM), Direct Differentiation (DD) as well as the adjoint method. The simpler method is the FDs where each design variable $b_i,\ i=1,\ldots,N$ is perturbed by a small quantity $\pm\epsilon$ and, for each perturbation, the PSM is called to compute the objective function value J. Then, $\frac{\delta J}{\delta b_i}$ is computed with second–order accuracy as

$$\frac{\delta J}{\delta b_i}=\frac{J\left(b_1,b_2,\ldots,b_i+\epsilon,\ldots,b_N\right)-J\left(b_1,b_2,\ldots,b_i-\epsilon,\ldots,b_N\right)}{2\epsilon}$$

Based on the above equation, the computation of all the N components of the gradient of J requires $2N$ (or N for first–order accuracy) calls to the PSM (i.e. the CFD solver). In addition, the accuracy of the SDs depends on the selected value of ϵ; large values lead to decreased accuracy on the SDs, whereas small ones may result in numerical errors due to the denominator approaching zero. If the CVM, [5], or DD [37], is used for the computation of the gradient, this is again at a cost proportional to N.

In industrial optimization problems with many design variables, the ability to compute the N derivatives of J at a cost that is independent of N is of paramount importance. In this direction, during the last three decades, great attention has been paid to the development and assessment of adjoint methods. The optimization problem is treated as a constrained one in which J is minimized (or maximized) under the constraint that the state equations are satisfied ($R_n=0$).

Then, a dual or adjoint problem is formulated by introducing the dual or adjoint variables Ψ into the Lagrangian function $J_{\text{aug}} = J + \Psi_n R_n$. Differentiating J_{aug} w.r.t. the design variables b_i and eliminating the derivatives of state (flow) quantities w.r.t. b_i yields a set of adjoint equations. The cost of solving the adjoint equations is similar to that of the PSM; making the cost of computing the objective function gradient independent of N. The adjoint-based optimization method was introduced by Lions in 1971, [78]. In the field of aerodynamics, the adjoint method was applied for potential flow problems by Pironneau in 1974, [114], and extended to the compressible Euler equations by Jameson, [53, 116, 54].

In the literature, there are two approaches of the adjoint method, namely the discrete and the continuous one. In discrete adjoint, the primal/flow equations are first discretized and then differentiated and, consequently, the adjoint equations are directly derived in discrete form. The discretized residuals of the flow problem can be differentiated either "by hand" or by code transformation tools and/or operator overloading. The latter approach is referred to as Algorithmic Differentiation (AD), [46]. Some well-known tools that transform a code that computes the flow residuals and solves the flow equations to a code that solves the adjoint ones are the TAPENADE [2], ADIFOR [3] and ADOL-C [142]. The main advantage of discrete adjoint is the high accuracy of computing $\frac{\delta J}{\delta b_i}$ due to the consistency with the discretized flow problem. In addition, the adjoint code inherits the convergence rate of the flow equations, [44]. On the other hand, the "hand-differentiated" discrete adjoint requires large investment in code development, while in AD the memory requirements for the solution of the adjoint equations increase up to $3-5$ times compared to solving the flow ones, [36].

In continuous adjoint, the rule of first-differentiate and then-discretize is followed. As a result, the adjoint equations are similar to the flow PDEs and, thus, similar techniques can be used for their numerical solution. The main advantage of the continuous adjoint is the clear physical insight into the adjoint problem and the low memory footprint of the adjoint code. On the other hand, the discretization schemes used for the adjoint PDEs can significantly affect the accuracy of computing $\frac{\delta J}{\delta b_i}$ as these are not always consistent with the discretized flow problem. In specific, the accuracy with which $\frac{\delta J}{\delta b_i}$ is computed with the continuous adjoint converges to that of the discrete adjoint as the grid becomes finer, [45, 101].

The continuous adjoint method can further be categorized in three approaches depending on the final expression for computing $\frac{\delta J}{\delta b_i}$. In the first approach, referred to as the severed-Surface Integral (severed-SI), [106], the expression of $\frac{\delta J}{\delta b_i}$ depends only on quantities computed along the domain boundaries neglecting the boundary terms containing the residuals of the flow equations. The second approach, referred to as the Field Integral (FI), [108], involves field integrals of grid sensitivities in the expression of $\frac{\delta J}{\delta b_i}$. The FI approach leads to more accurate SDs compared to the severed-SI, with more complex expressions though, [64]. Finally, in [62], a third approach was introduced as a treatment for the severed-SI,

namely the Enhanced-Surface Integral (E-SI). Based on this approach, the adjoint grid displacement equations were incorporated in the adjoint PDEs resulting in expression of $\frac{\delta J}{\delta b_i}$ that exclusively relies on boundary integrals, matching, at the same time, the accuracy of the FI approach. In this PhD Thesis, the FI approach is employed.

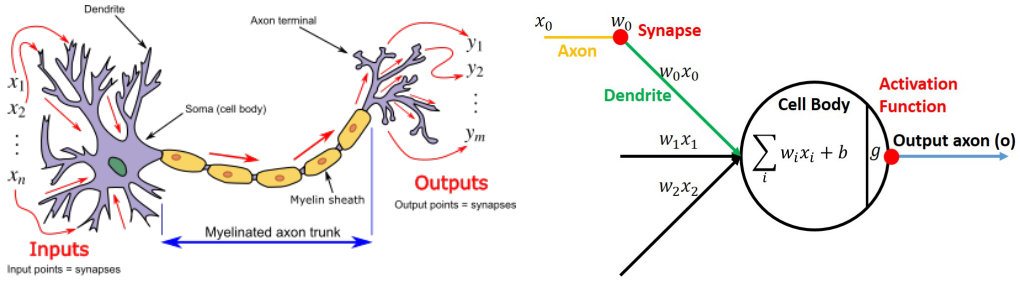
In turbulent flows, the adjoint to the turbulence models should be developed, programmed and solved in order to accurately compute $\frac{\delta J}{\delta b_i}$. In discrete adjoint, the differentiation of the turbulence model is a common approach [6, 75, 104] even if there are still studies that refrain from solving the extra equations [141]. On the other hand, regarding the continuous adjoint, the "frozen turbulence" assumption (implying that the differentiation of the turbulence model is omitted) was commonly used and was overcome by PCOpt/NTUA in [152]. In specific, the continuous adjoint method to the k–ε and k–ω turbulence models for incompressible flows [109, 63] and to the Spalart–Allmaras turbulence model for compressible flows [134] have been formulated and presented by the PCOpt/NTUA.

In transitional flows, the corresponding literature regarding the differentiation of transition models is restricted. The aerodynamic shape optimization of the S809 wind turbine blade was performed based on the discrete adjoint method; transition was modeled by means of the $k-\omega$ SST turbulence model coupled with the $\gamma-\tilde{Re}_{\theta t}$ model in [65]. In [47], discrete adjoint-based shape optimizations of subsonic and transonic airfoils were performed; these included the differentiation of the AFT transition model. The discrete adjoint method to the SA-sLM2015 transition model was developed and used for the shape optimization of isolated airfoils and an infinite swept wing for drag minimization in [112]. Regarding the adjoint method and the transition modeling, all the above studies were based on discrete adjoint. To the author knowledge, the development of the continuous adjoint method for transitional flows appears in this PhD Thesis for the first time in the corresponding literature.

1.3 Deep Neural Networks in Fluid Dynamics

Deep Neural Networks (DNNs) were introduced to the Machine Learning (ML) community by Dechter, [25], and have been applied in fields including computer vision, speech/audio/image recognition, natural language processing etc. Deep Learning is a subfield of ML which is based on ANN architectures. ANNs are a set of algorithms inspired by the human brain and mammal nervous system that mimic the way information is processed in biological systems.

A biological neuron consists of synapses, dendrites, the cell body (soma) and the axon, Fig. 1.1a. The neurons are responsible to process and transmit the information. Each biological neuron receives signals from other neurons via the dendrites, the cell body sums all the incoming signals and, if this summation exceeds a threshold, it is sent as an output to the axon which transmits the



(a) Biological Neuron **(b)** Artificial Neuron

Figure 1.1: (a) Biological neuron model. By Egm4313.s12 (Prof. Loc Vu-Quoc) - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=72816083 (b) Artificial neuron model.

signals to other cells.

The same structure is followed for an artificial neuron, Fig. 1.1b. The synapses of the biological neuron correspond to the weighted signals received from other neurons, these signals are carried to the cell body by the dendrites. The main operation of the cell body is the summation of the weighted signals; if this signal exceeds a threshold value, this is sent to the output down the axon. The threshold is modeled by using activation functions introducing non-linearities that assist the model to handle complex problems.

Over the recent years, DNNs have also been widely used in the context of aerodynamic optimization as an alternative to computationally expensive evaluation tools. In the PhD Thesis of [58] performed at the PCOpt/NTUA, DNNs were used to predict entire 2D and 3D subsonic/transonic flow fields around airfoils and wings based either on image or raw data processing. The DNNs had acceptably low error and negligible cost compared to the CFD simulation; for the latter, the DNNs were faster by several orders of magnitude, excluding of course their training cost. The DNNs were also used as off–line trained metamodels in a MAEA-based shape optimization improving its performance; the objective function was computed by post-processing the predicted flow field. In [77], DNNs were used to predict the Reynolds stress anisotropy tensor based on high fidelity simulation data. The architecture included a multiplicative layer with an invariant tensor basis and was implemented to predict a turbulent duct flow and a flow over a wavy wall. The idea of extracting high fidelity data in order to train ML for turbulence closure was also investigated in [90]; there, emphasis was laid on internal turbomachinery cooling flows, namely the flow through a turbulent channel and a stationary serpentine passage. The target was to use ANNs for modifying the Boussinesq assumption based on DNS simulations. A thorough review for the use of ML in turbomachinery applications can be found in [48]. In [34], a data–driven approach for modeling turbulent and transitional flows, based on inverse modeling and ML, was proposed. The aim was to quantify and account for deficiencies in turbu-

lence and transition modeling using data from simulations and measurements. The inverse modeling provided direct information on the model inadequacy and DNNs were used to convert the inferred information into modeling knowledge. The method was applied in turbulent (over a convex wall) and transitional (over a flat plate) flows by modeling a non-equilibrium turbulent boundary layer and the bypass transition, respectively. In [26], a DNN-optimal sensor placement was proposed for the ensemble Kalman filter-based data assimilation; a feature importance layer was designed for the DNN in order to obtain the spatial sensitivity of the velocity with respect to the $k-\varepsilon$ and Spalart–Allmaras turbulence model constants. This was implemented in a free round jet flow as well as a separated and re-attached flow around a blunt plate. [92] used neural networks to predict the turbulent viscosity, using data generated by the Spalart–Allmaras, $k-\varepsilon$ and $k-\omega$ SST turbulence models, in incompressible flows.

Over and above to the use of DNNs in CFD analysis or in any other problem, finding and efficiently training the optimal network architecture is of utmost importance. An optimization procedure based on genetic algorithms, for identifying the hyperparameters of convolutional layers (max. pooling, batch normalization, activation functions, dropout and number of kernels) and fully connected layers (number of nodes, batch normalization, activation functions, dropout, optimizer), was presented in [124]. In [150], a training benchmark for DNNs was presented including several ML applications, from image classification to reinforcement learning, and implemented in different frameworks and hardware configurations. Performance analysis, including the selection of performance metrics, ways for analyzing the results as well as tools for determining memory consumption by different data structures, were developed. Regarding the training phase, [74] discussed and tested three principles/steps for the DNN training, namely the pre-training of one layer at a time, the use of unsupervised learning and the fine-tuning of the entire network based on the criterion of interest.

1.4 Thesis Outline

This PhD Thesis lays emphasis on aero/hydro-dynamic shape optimization. During this Thesis, a GPU-enabled CFD software that takes into account transitional effects is developed, the continuous adjoint method for transitional flows is formulated, consistent discretization schemes for the continuous adjoint equations are proposed and DNNs are used for flow prediction in shape optimizations.

Chapter 2 presents the extensions made on the flow solver of PUMA (Parallel Unstructured Adjoint Multi-row) of PCOpt/NTUA for capturing transitional flows. The governing equations, including the $\gamma-\tilde{R}e_{\theta t}$ transition model and its variants, are presented. Additional information regarding the discretization, the numerical solution and the GPU implementation of the PUMA solver is described. Finally, a series of validation/verification cases of a wide range (2D/3D, internal/external

aerodynamics) is used to demonstrate the prediction capabilities of the new PUMA software for capturing transition effects.

Chapter 3 is concerned with the mathematical development, assessment and validation of the continuous adjoint method for transitional flows using the $\gamma - \tilde{R}e_{\theta t}$ transition model. The adjoint solver of the PUMA software is extended by accommodating the adjoint to three transition model variants. The field adjoint equations and the derivation of the adjoint boundary conditions is presented. At the end, the continuous adjoint method for transitional flows is validated and used in shape optimization studies; the impact of the "frozen transition" assumption is also investigated.

In Chapter 4, consistent discretization schemes for the continuous adjoint equations and boundary conditions, developed in Chapter 3, with a clear physical insight are proposed under the so–called *Think-Discrete Do-Continuous* (*TDDC*) adjoint. The accuracy of the SDs computed based on discretization schemes proposed by the *TDDC* adjoint is compared with that resulted from "standard" discretization schemes widely used in the literature. The impact on the optimization convergence is also examined in a series of 2D and 3D validation cases.

The continuous adjoint method for transitional flows with consistent discretization schemes, developed in Chapters 3 and 4, is used for the shape optimization of industrial applications, that of a wing and that of an aircraft configuration, in Chapter 5.

The next two chapters investigate the use of DNNs for flow prediction in shape optimization. In Chapter 6, DNNs are used to replace the numerical solution of the turbulence and/or transition model equations by providing the turbulent viscosity field in each pseudo-iteration of the flow solver. The DNN configuration is optimized by means of a MAEA. The new cost-effective solver is used in three MAEA-based shape optimization problems (including that of a car model) and the gain in terms of computational cost is demonstrated.

In Chapter 7, DNNs are used in multi-disciplinary and multi-row turbomachinery shape optimization problems. The use of the two-branch $\lambda-$DNN architecture in conjugate heat transfer shape optimization problems by replacing the heat conduction equation solver is proposed. Then, a DNN surrogate to the mixing plane technique is proposed for multi-row turbomachinery problems.

Finally, conclusions are drawn and recommendations for future work are proposed in Chapter 8.

Chapter 2

The Navier–Stokes Equations and Their Numerical Solution

The main focus of this Thesis is the development and assessment of computational methods and numerical tools for use in CFD shape optimization problems. The optimizations that are carried out rely on both gradient-based techniques and evolutionary algorithms; for the former the continuous adjoint method with consistent discretization schemes is used, while for the latter, DNNs are used to reduce the computational cost of the evaluation tool. Before proceeding to the development of the continuous adjoint method and other tools required by the optimization workflow, the flow model (equations, discretization schemes, numerical solution) used to evaluate the performance of each candidate solution resulting during the optimization procedure is presented. The flow model consists of the RANS equations (or URANS for unsteady problems) for compressible flows, the Spalart–Allmaras, $k-\omega$ SST turbulence models and the $\gamma-\tilde{Re}_{\theta t}$ transition model (in order to include turbulent and transition effects, respectively). In this Thesis, the in–house software PUMA, a CFD solver together with its adjoint counterpart, is used and further enhanced. The development of PUMA started about two decades ago in the framework of a number of PhD theses of PCOpt/LTT ([147],[8],[133],[135],[98]) and, during the last years, it has been ported to GPUs and enriched with new features and capabilities. This Thesis relies (and extends) exclusively upon the GPU-enabled variant of PUMA.

In this chapter, the governing flow equations and their boundary conditions, along with their discretization and numerical methods to solve the discretized system of equations are presented.

2.1 Governing Equations for Compressible Flows

2.1.1 The URANS Equations for Compressible Flows

Let $O(x_1\ x_2\ x_3)$ be a coordinate system which rotates at a constant angular velocity $\omega_m\ (m=1,2,3)$. In this system, the Navier–Stokes equations for compressible fluid flows can be expressed as

$$R_n^{\text{MF}} = \underbrace{\frac{\partial U_n}{\partial t}}_{\text{MF}^t} + \underbrace{\frac{\partial f_{nk}^{\text{inv}}}{\partial x_k}}_{\text{MF}^{inv}} - \underbrace{\frac{\partial f_{nk}^{\text{vis}}}{\partial x_k}}_{\text{MF}^{vis}} + \underbrace{S_n}_{\text{MF}^s} = 0 \tag{2.1}$$

Eq. 2.1 is solved for the conservative flow variables $U_n = \left[\rho\ \ \rho v_1^A\ \ \rho v_2^A\ \ \rho v_3^A\ \ \rho E\right]^T$, where ρ stands for the fluid density, $v_m^A\ (m=1,2,3)$ being the velocity Cartesian components w.r.t the absolute/inertial frame of reference and E the total energy per unit mass. The inviscid f_{nk}^{inv} and viscous fluxes f_{nk}^{vis} as well as source terms S_n are given by

$$f_{nk}^{\text{inv}} = \begin{bmatrix} \rho v_k^R \\ \rho v_k^R v_1^A + p\delta_{1k} \\ \rho v_k^R v_2^A + p\delta_{2k} \\ \rho v_k^R v_3^A + p\delta_{3k} \\ \rho v_k^R h_t + v_k^F p \end{bmatrix} \quad f_{nk}^{\text{vis}} = \begin{bmatrix} 0 \\ \tau_{1k} \\ \tau_{2k} \\ \tau_{3k} \\ v_\ell^A \tau_{\ell k} + q_k \end{bmatrix} \quad S_n = \begin{bmatrix} 0 \\ \rho\varepsilon_{1\ell k}\omega_\ell v_k^A \\ \rho\varepsilon_{2\ell k}\omega_\ell v_k^A \\ \rho\varepsilon_{3\ell k}\omega_\ell v_k^A \\ 0 \end{bmatrix} \tag{2.2}$$

where p stands for the static pressure. In Eqs. 2.1-2.2 and the rest of the Thesis, the Einstein convention, according to which summation is indicated by repeated indices in the same term, is used. The relative velocity components v_m^R are linked to the absolute ones v_m^A through the equation $v_m^A = v_m^R + v_m^F$, with $v_m^F = \varepsilon_{m\ell k}\omega_k\left(x_k - x_k^C\right)$ being the rotating/non–inertial frame velocity and x_k^C the position vector of the center of rotation. In Eqs. 2.2, τ_{km} is the viscous stress tensor for a Newtonian fluid defined by

$$\tau_{km} = \frac{\mu + \mu_t}{\text{Re}_0}\left(\frac{\partial v_k^A}{\partial x_m} + \frac{\partial v_m^A}{\partial x_k} - \frac{2}{3}\delta_{km}\frac{\partial v_\ell^A}{\partial x_\ell}\right) \tag{2.3}$$

q_k is the $k-th$ component of the heat flux

$$q_k = \frac{\mathcal{C}_p}{\text{Re}_0}\left(\frac{\mu}{\text{Pr}} + \frac{\mu_t}{\text{Pr}_t}\right)\frac{\partial T}{\partial x_k} \tag{2.4}$$

where $\mathcal{C}_p$ is the specific heat under constant pressure and T is the static temperature. Since the fluid is assumed to be a perfect gas, T is related to pressure and density through the equation of state $p = \rho R_g T$, with R_g being the specific gas constant. Pr, Pr_t stand for the Prandtl and turbulent Prandtl numbers, respectively. The specific heat ratio γ is computed as $\gamma = \frac{\mathcal{C}_p}{\mathcal{C}_v}$ where $\mathcal{C}_v$ is the specific heat under constant volume. The total or stagnation enthalpy (h_t) is expressed as $h_t = E + \frac{p}{\rho}$; for a perfect gas, h_t is linked to pressure (p), density (ρ) and velocity $\left(v_\ell^A, \ell = 1, \ldots, 3\right)$ through

$$h_t = \frac{\gamma p}{\rho\left(\gamma - 1\right)} + \frac{1}{2} v_\ell^A v_\ell^A \tag{2.5}$$

The absolute Mach number (M_A) is defined as $M_A = \frac{\sqrt{v_\ell^A v_\ell^A}}{c}$ where c is the local speed of sound which, for perfect gases, is given as $c = \sqrt{\gamma R_g T}$. The total temperature (T_t) is given by

$$T_t = T + \frac{v_\ell^A v_\ell^A}{2\mathcal{C}_p} \tag{2.6}$$

and the total pressure (p_t) for perfect gases in compressible flows is defined by

$$p_t = p\left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}} \tag{2.7}$$

The dynamic viscosity (μ) is (optionally) related to temperature through the Sutherland's formula [130]

$$\mu = \mu_0 \frac{T_0 + \mathrm{Su}}{T + \mathrm{Su}} \left(\frac{T}{T_0}\right)^{\frac{3}{2}} \tag{2.8}$$

where Su is an effective temperature value usually called Sutherland temperature, T_0, μ_0 are the reference temperature and reference dynamic viscosity, respectively and μ is linked to the kinematic viscosity ν through $\mu = \rho\nu$. For air, $\mu_0 = 1.716 \times 10^{-5}\,\mathrm{kg\,m^{-1}\,s^{-1}}$, $T_0 = 273.15\,\mathrm{K}$ and $\mathrm{Su} = 110.4\,\mathrm{K}$.

The Reynolds number (Re_0) arises when the equations are solved in non-dimensional form. Let the velocity field be normalized by a reference velocity (v_{ref}), the density by a reference density (ρ_{ref}), all lengths by a reference length (l_{ref}) and the molecular viscosity by a reference viscosity (μ_{ref}). By doing so, the following Reynolds number

$$\mathrm{Re}_0 = \frac{\rho_{\mathrm{ref}}\, v_{\mathrm{ref}}\, l_{\mathrm{ref}}}{\mu_{\mathrm{ref}}}$$

appears into the equations. This must not be confused with the Reynolds number of the flow $\mathrm{Re}=\frac{\rho v l}{\mu}$, where l is a reference length.

In this Thesis, turbulent viscosity μ_t is computed either by employing the one-equation Spalart–Allmaras [128] or the two-equation Menter's $k-\omega$ SST [95] turbulence models.

2.1.2 The Spalart–Allmaras Turbulence Model

According to the Spalart–Allmaras model, [128], an additional PDE (also in non - dimensional form) is solved for the turbulence field $\tilde{\nu}$, namely

$$
\begin{aligned}
R^{\tilde{\nu}} = & \underbrace{\frac{\partial\left(\rho\tilde{\nu}\right)}{\partial t}}_{\mathrm{SA}^t} + \underbrace{\frac{\partial\left(\rho\tilde{\nu}v_k^R\right)}{\partial x_k}}_{\mathrm{SA}^c} - \underbrace{\frac{\rho}{\mathrm{Re}_0\,\sigma}\left\{\frac{\partial}{\partial x_k}\left[\left(\nu+\tilde{\nu}\right)\frac{\partial\tilde{\nu}}{\partial x_k}\right] + c_{b_2}\frac{\partial\tilde{\nu}}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\right\}}_{\mathrm{SA}^d} \\
& \underbrace{- \rho c_{b_1}\left(1-f_{t_2}\right)\tilde{S}\tilde{\nu} + \frac{\rho}{\mathrm{Re}_0}\left(c_{w_1}f_w - \frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\left(\frac{\tilde{\nu}}{\Delta}\right)^2}_{\mathrm{SA}^s} = 0
\end{aligned}
\tag{2.9}
$$

where Δ stands for the distance of each point within the flow domain from the closest wall boundary. Solving Eq. 2.9, μ_t is computed from $\tilde{\nu}$ as $\mu_t=\rho\tilde{\nu}f_{v_1}$. Eq. 2.9 is supplemented by the following relations and constants [128]

$$
\begin{aligned}
& \chi=\frac{\tilde{\nu}}{\nu},\quad f_{v_1}=\frac{\chi^3}{\chi^3+c_{v_1}^3},\quad f_{v_2}=1-\frac{\chi}{1+\chi f_{v_1}},\quad \zeta=\sqrt{\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_m^A}{\partial x_\ell}\frac{\partial v_r^A}{\partial x_q}}, \\
& \tilde{S}=\zeta+\frac{\tilde{\nu}f_{v_2}}{\mathrm{Re}_0\,\kappa^2\Delta^2},\quad f_w=g\left(\frac{1+c_{w_3}^6}{g^6+c_{w_3}^6}\right)^{\frac{1}{6}},\quad g=r+c_{w_2}\left(r^6-r\right), \\
& r=\min\left(10,\frac{\tilde{\nu}}{\mathrm{Re}_0\,\tilde{S}\kappa^2\Delta^2}\right),\ \tilde{\mu}=\rho\tilde{\nu},\quad f_{t_2}=c_{t_3}e^{-c_{t_4}\chi^2}, \\
& c_{v_1}=7.1,\quad c_{b_1}=0.1355,\quad c_{b_2}=0.622,\quad c_{w_1}=\frac{c_{b_1}}{\kappa^2}+\frac{1+c_{b_2}}{\sigma}, \\
& c_{w_2}=0.3,\quad c_{w_3}=2.0,\quad \sigma=\frac{2}{3},\quad \kappa=0.41,\quad c_{t_3}=1.2,\quad c_{t_4}=0.5
\end{aligned}
\tag{2.10}
$$

2.1.3 The $k-\omega$ SST Turbulence Model

According to Menter's $k-\omega$ SST turbulence model, [95], two PDEs are solved for the turbulent kinetic energy k and the specific dissipation rate ω, namely

$$
\begin{aligned}
R^k &= \underbrace{\frac{\partial(\rho k)}{\partial t}}_{\mathrm{k}^t} + \underbrace{\frac{\partial\left(\rho k v_k^R\right)}{\partial x_k}}_{\mathrm{k}^c} - \underbrace{\frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[(\mu+\sigma_k\mu_t)\frac{\partial k}{\partial x_k}\right]}_{\mathrm{k}^d} \underbrace{-\tilde{P}_\kappa + \beta^*\rho k\omega}_{\mathrm{k}^s} = 0 \\
R^\omega &= \underbrace{\frac{\partial(\rho\omega)}{\partial t}}_{\omega^t} + \underbrace{\frac{\partial\left(\rho\omega v_k^R\right)}{\partial x_k}}_{\omega^c} - \underbrace{\frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[(\mu+\sigma_\omega\mu_t)\frac{\partial\omega}{\partial x_k}\right]}_{\mathrm{k}^d} \\
&\underbrace{-\gamma\frac{P_\kappa}{\nu_t} + \beta\rho\omega^2 - 2(1-F_1)\frac{\rho\sigma_{\omega_2}}{\omega}\frac{\partial k}{\partial x_k}\frac{\partial\omega}{\partial x_k}}_{\omega^s} = 0
\end{aligned}
\tag{2.11}
$$

The $k-\omega$ SST turbulence model combines the standard $k-\omega$ and $k-\varepsilon$ models, having the accuracy of $k-\omega$ near the walls and the freestream independence of $k-\varepsilon$ model at the far-field. The constants are blended using function F_1 which is equal to 1.0 close to rigid walls and 0.0 far from them. Any model's constant ϕ is computed through $\phi = F_1\phi_1 + (1-F_1)\phi_2$. Solving Eqs. 2.11, μ_t is computed as $\mu_t = \mathrm{Re}_0\frac{\rho a_1 k}{\max(a_1\omega, SF_2)}$. The Menter's $k-\omega$ SST model is supplemented by the following relations and constants

$$
\begin{aligned}
&F_1 = \tanh\left(\min\left(arg_{F1}^4, 10\right)\right), \quad F_2 = \tanh\left(\min\left(arg_{F2}^2, 100\right)\right), \\
&arg_{F1} = \min\left(\max\left(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{\omega y^2}\frac{1}{\mathrm{Re}_0}\right), \frac{4\rho k\sigma_{\omega 2}}{CD_{k\omega}y^2}\right), \\
&arg_{F2} = \max\left(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{\omega y^2}\frac{1}{\mathrm{Re}_0}\right), \\
&CD_{k\omega} = \max\left(\frac{2\rho\sigma_{\omega 2}}{\omega}\frac{\partial k}{\partial x_k}\frac{\partial\omega}{\partial x_k}, e^{-10}\right), \quad \tilde{P}_\kappa = \min\left(P_\kappa, 10\beta^*\rho k\omega\right), \\
&P_\kappa = \frac{1}{\mathrm{Re}_0}\mu_t S^2 - \frac{2}{3}\rho k\delta_{km}\frac{\partial v_m^A}{\partial x_k}, \quad S = \sqrt{2S_{km}S_{km}}, \quad S_{km} = \frac{1}{2}\left(\frac{\partial v_m^A}{\partial x_k} + \frac{\partial v_k^A}{\partial x_m}\right), \\
&\sigma_{k1} = 0.85034, \quad \sigma_{k2} = 0.5, \quad \beta_1 = 0.075, \quad \gamma_1 = 5/9, \quad \alpha_1 = 0.31, \\
&\sigma_{k2} = 1.0, \quad \sigma_{\omega 2} = 0.85616, \quad \beta_2 = 0.0828, \quad \gamma_2 = 0.44, \quad \beta^* = 0.09
\end{aligned}
\tag{2.12}
$$

2.1.4 The $\gamma - \tilde{Re}_{\theta t}$ Transition Model

In order to take transition effects into account, the turbulence models are coupled with the two equation $\gamma - \tilde{Re}_{\theta t}$ transition model, [73]. According to that, two additional PDEs are solved for the transition intermittency γ and the transition

momentum-thickness Reynolds number $\tilde{Re}_{\theta t}$, namely

$$
\begin{aligned}
R^{\gamma} &= \underbrace{\frac{\partial\left(\rho\gamma\right)}{\partial t}}_{\gamma^{t}} + \underbrace{\frac{\partial\left(\rho v_{k}^{A}\gamma\right)}{\partial x_{k}}}_{\gamma^{c}} - \underbrace{\frac{1}{\mathrm{Re}_{0}}\frac{\partial}{\partial x_{k}}\left[\left(\mu+\frac{\mu_{t}}{\sigma_{f}}\right)\frac{\partial\gamma}{\partial x_{k}}\right]}_{\gamma^{d}} \underbrace{-P_{\gamma}+E_{\gamma}}_{\gamma^{s}} = 0 \\
R^{\tilde{Re}_{\theta t}} &= \underbrace{\frac{\partial\left(\rho\tilde{Re}_{\theta t}\right)}{\partial t}}_{\tilde{Re}_{\theta t}^{t}} + \underbrace{\frac{\partial\left(\rho v_{k}^{A}\tilde{Re}_{\theta t}\right)}{\partial x_{k}}}_{\tilde{Re}_{\theta t}^{c}} - \underbrace{\frac{1}{\mathrm{Re}_{0}}\frac{\partial}{\partial x_{k}}\left[\sigma_{\theta,t}\left(\mu+\mu_{t}\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_{k}}\right]}_{\tilde{Re}_{\theta t}^{d}} \underbrace{-P_{\theta,t}-D_{SCF}}_{\tilde{Re}_{\theta t}^{s}} = 0
\end{aligned}
\tag{2.13}
$$

The $\gamma-\tilde{Re}_{\theta t}$ transition model is not Galilean invariant, due to its explicit use of the velocity vector (see Secs. 2.1.4.1-2.1.4.4). For this reason, its equations are expressed at the absolute frame of reference, in contrast to the mean flow and the turbulence models equations which were expressed in the relative frame of reference.

The original $\gamma-\tilde{Re}_{\theta t}$ transition model proposed by Langtry and Menter in [73] and extended for capturing stationary crossflow effects in [71] interacts with the $k-\omega$ SST turbulence model. Modifications to the $\gamma-\tilde{Re}_{\theta t}$ source terms were made in [23, 22, 113] so as these to be coupled with the Spalart–Allmaras turbulence model. The expressions of P_{γ}, E_{γ}, $P_{\theta,t}$ and D_{SCF} for all four variants as well as their interaction with the turbulence models (by modifying their source terms) follow. In particular, Sec. 2.1.4.1 presents the SST-2003-LM2015 transition model coupled with the $k-\omega$ SST turbulence model and Secs. 2.1.4.2, 2.1.4.3 and 2.1.4.4 the SA-noft2-Gamma-Retheta, SA-LM2015 and SA-sLM2015 transition models solved along with the Spalart–Allmaras turbulence model.

2.1.4.1 The SST-2003-LM2015 Transition Model

Based on the original $\gamma-\tilde{Re}_{\theta t}$ transition model, [73, 71], which is coupled with the $k-\omega$ SST turbulence model, the source terms in the γ equation are

$$
\begin{aligned}
P_{\gamma} &= \rho c_{\alpha_1} S\sqrt{\gamma F_{onset}}\left(1-c_{e_1}\gamma\right)F_{length} \\
E_{\gamma} &= \rho c_{\alpha_2}\zeta\gamma F_{turb}\left(c_{e_2}\gamma-1\right)
\end{aligned}
\tag{2.14}
$$

where S and ζ are the strain rate and the vorticity magnitude, respectively. Regarding the production term P_γ, F_{onset} is obtained as

$$\begin{aligned}
&F_{onset} = \max\left(F_{onset2} - F_{onset3}, 0\right), \quad F_{onset2} = \min\left[\max\left(F_{onset1}, F_{onset1}^4\right), 2\right] \\
&F_{onset3} = \max\left[\left(1 - \left(\frac{R_T}{2.5}\right)^3\right), 0\right], \quad F_{onset1} = \frac{Re_\nu}{2.193 Re_{\theta c}} \\
&Re_\nu = \frac{\rho S \Delta^2}{\mu} \mathrm{Re}_0, \quad R_T = \frac{\rho k}{\mu \omega}
\end{aligned} \tag{2.15}$$

F_{length} and $Re_{\theta c}$ are functions of $\tilde{R}e_{\theta t}$ based on the following empirical correlations

$$\begin{aligned}
&F_{length} = F_{length1}(1 - F_{sublayer}) + 40 F_{sublayer}, \quad F_{sublayer} = e^{-\left(\frac{R_\omega}{0.4}\right)^2}, \quad R_\omega = \frac{\rho \Delta^2 \omega}{500 \mu} \\
&F_{length1} = \begin{cases}
398.189e^{-1} - 119.270e^{-4}\tilde{R}e_{\theta t} - 132.567e^{-6}\tilde{R}e_{\theta t}^2 & \text{, if } \tilde{R}e_{\theta t} < 400 \\
263.404 - 123.939e^{-2}\tilde{R}e_{\theta t} + 194.548e^{-5}\tilde{R}e_{\theta t}^2 & \\
\quad -101695e^{-8}\tilde{R}e_{\theta t}^3 & \text{, if } 400 \le \tilde{R}e_{\theta t} < 596 \\
0.5 - \left(\tilde{R}e_{\theta t} - 596\right) 3e^{-4} & \text{, if } 596 \le \tilde{R}e_{\theta t} < 1200 \\
0.3188 & \text{, if } 1200 \ge \tilde{R}e_{\theta t}
\end{cases} \\
&Re_{\theta c} = \begin{cases}
\tilde{R}e_{\theta t} - 396.035e^{-2} + 120.656e^{-4}\tilde{R}e_{\theta t} - 868.23e^{-6}\tilde{R}e_{\theta t}^2 & \\
+696.506e^{-9}\tilde{R}e_{\theta t}^3 - 174.105e^{-12}\tilde{R}e_{\theta t}^4 & \text{, if } \tilde{R}e_{\theta t} \le 1870 \\
\tilde{R}e_{\theta t} - 593.11 - \left(\tilde{R}e_{\theta t} - 1870\right) 0.482 & \text{, if } \tilde{R}e_{\theta t} > 1870
\end{cases}
\end{aligned} \tag{2.16}$$

Regarding the destruction term E_γ, F_{turb} is defined as

$$F_{turb} = \exp\left[-\left(\frac{R_T}{4}\right)^4\right] \tag{2.17}$$

The source terms $P_{\theta,t}$ and D_{SCF} in the $\tilde{R}e_{\theta t}$ equation are expressed as

$$\begin{aligned}
P_{\theta,t} &= \rho \frac{c_{\theta,t}}{\mathcal{T}} \left(Re_{\theta,t}^{eq} - \tilde{R}e_{\theta t}\right)(1 - F_{\theta t}) \\
D_{SCF} &= c_{\theta,t} \frac{\rho}{\mathcal{T}} c_{crossflow} \min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right) F_{\theta t2}
\end{aligned} \tag{2.18}$$

The blending function $F_{\theta t}$ and the timescale $\mathcal{T}$ are defined as

$$\begin{aligned} F_{\theta t} &= \min\left[\max\left[F_{wake}\exp\left[-\left(\frac{\Delta}{\delta}\right)^4\right], 1-\left(\frac{\gamma-1/c_{e_2}}{1-1/c_{e_2}}\right)^2\right], 1\right] \\ F_{wake} &= \exp\left[-\left(\frac{Re_\omega}{10^5}\right)^2\right], \quad Re_\omega = \frac{\rho\omega\Delta^2}{\mu}\mathrm{Re}_0 \\ \mathcal{T} &= \frac{500\mu}{\rho\left|\mathbf{U}\right|^2}\frac{1}{\mathrm{Re}_0}, \quad \delta = \frac{375\zeta\mu\tilde{R}e_{\theta t}\Delta}{\rho\left|\mathbf{U}\right|^2}\frac{1}{\mathrm{Re}_0}, \quad \left|\mathbf{U}\right| = \sqrt{v_\ell^A v_\ell^A}, \ (\ell=1,\ldots,3) \end{aligned} \tag{2.19}$$

$Re_{\theta,t}^{eq}$ is a function of the turbulence intensity (Tu) and the pressure gradient parameter λ_θ

$$\begin{aligned} Tu &= 100\frac{\sqrt{2k/3}}{\left|\mathbf{U}\right|} \\ \lambda_\theta &= \frac{\rho\theta^2}{\mu}\frac{d\left|\mathbf{U}\right|}{ds}\mathrm{Re}_0, \quad \frac{d\left|\mathbf{U}\right|}{ds} = \frac{v_m^A v_n^A}{\left|\mathbf{U}\right|^2}\frac{\partial v_m^A}{\partial x_n} \end{aligned} \tag{2.20}$$

$$Re_{\theta,t}^{eq} = \begin{cases} \left(1173.51 - 589.428Tu + 0.2196/Tu^2\right)F\left(\lambda_\theta\right) & \text{, if } Tu \le 1.3 \\ 331.5\left(Tu - 0.5668\right)^{-0.671}F\left(\lambda_\theta\right) & \text{, if } Tu > 1.3 \end{cases} \tag{2.21}$$

where

$$F\left(\lambda_\theta\right) = \begin{cases} 1+\left[12.986\lambda_\theta+123.66\lambda_\theta^2+405.689\lambda_\theta^3\right]\exp\left[-\left(\frac{Tu}{1.5}\right)^{1.5}\right] & \text{, if } \lambda_\theta \le 0 \\ 1+0.275\left[1-\exp\left(-35\lambda_\theta\right)\right]\exp\left(\frac{-Tu}{0.5}\right) & \text{, if } \lambda_\theta > 0 \end{cases} \tag{2.22}$$

The expression for $Re_{\theta,t}^{eq}$ is an implicit function of θ through the presence of λ_θ since $Re_{\theta,t}^{eq} = \frac{\rho|\mathbf{U}|\theta}{\mu}\mathrm{Re}_0$. Equations 2.20-2.22 can be solved by iterating on the value of θ. For numerical robustness, λ_θ, Tu and $Re_{\theta,t}^{eq}$ should be limited as follows

$$\begin{aligned} -0.1 \le \lambda_\theta &\le 0.1 \\ Tu &\ge 0.027 \\ Re_{\theta,t}^{eq} &\ge 20 \end{aligned}$$

The stationary crossflow instabilities are present mainly due to the surface roughness (h_{rms}) and can be important in aeronautical applications at lower freestream turbulence levels, [71]. Regarding the source term related to the stationary crossflow effects D_{SCF}, $F_{\theta t2}$ stands for a crossflow sink term, active only inside the laminar boundary layer. Timescale $\mathcal{T}$ is limited for robustness reasons

based on the local grid length (L)

$$
\begin{aligned}
Re_{SCF} &= \frac{\rho\theta_t\left|\mathbf{U}\right|}{0.82\mu}\,\mathrm{Re}_0 = -35.088\ln\left(\frac{h_{rms}}{\theta_t}\right) + 319.51 + f\left(\mathcal{D}H_{CF+}\right) - f\left(\mathcal{D}H_{CF-}\right)\\
f\left(\mathcal{D}H_{CF+}\right) &= 6200\mathcal{D}H_{CF+} + 5000\left(\mathcal{D}H_{CF+}\right)^2\\
f\left(\mathcal{D}H_{CF-}\right) &= 75\tanh\left(\frac{\mathcal{D}H_{CF-}}{0.0125}\right)\\
\mathcal{D}H_{CF+} &= \max\left[+\left(0.1066 - \mathcal{D}H_{CF}, 0\right)\right]\\
\mathcal{D}H_{CF-} &= \max\left[-\left(0.1066 - \mathcal{D}H_{CF}, 0\right)\right]\\
\mathcal{D}H_{CF} &= H_{CF}\left[1 + \min\left(\frac{\mu_t}{\mu}, 0.4\right)\right]\\
H_{CF} &= \frac{\zeta_{streamwise}\Delta}{\left|\mathbf{U}\right|},\quad \zeta_{streamwise} = \left|\frac{v_k^A}{\left|\mathbf{U}\right|}\frac{\partial v_m^A}{\partial x_\ell}\varepsilon_{k\ell m}\right|
\end{aligned}
\tag{2.23}
$$

$$
\begin{aligned}
F_{\theta t2} &= \min\left(F_{wake}\exp\left(-\left({}^{\Delta}/_{\delta}\right)^4\right), 1\right)\\
\mathcal{T} &= \min\left(\frac{500\mu}{\rho\left|\mathbf{U}\right|^2}, \frac{\rho L^2}{\left(\mu+\mu_t\right)}\right)\frac{1}{\mathrm{Re}_0}
\end{aligned}
\tag{2.24}
$$

The expression for Re_{SCF} is an implicit function of θ_t so it must be solved in a similar way to $Re_{\theta,t}^{eq}$.

The model constants are $c_{\alpha_1}=2$, $c_{\alpha_2}=0.06$, $c_{\epsilon_1}=1$, $c_{\epsilon_2}=50$, $c_{\theta,t}=0.03$, $\sigma_f=1$, $\sigma_{\theta,t}=2$, $c_{crossflow}=0.6$. The modification to the intermittency which allows to predict transition induced by flow separation is

$$
\begin{aligned}
\gamma_{sep} &= \min\left(2\max\left[0, \left(\frac{Re_\nu}{3.235Re_{\theta c}}\right) - 1\right]F_{reattach}, 2\right)F_{\theta t}\\
F_{reattach} &= \exp\left[-\left(\frac{R_T}{20}\right)^4\right],\quad \gamma_{eff} = \max\left(\gamma, \gamma_{sep}\right)
\end{aligned}
\tag{2.25}
$$

To interact with the $k-\omega$ SST turbulence model, the following modifications to the source terms of the turbulence model are made

$$
\begin{aligned}
\tilde{P}_k &= \gamma_{eff}P_K,\quad \tilde{D}_k = \min\left(\max\left(\gamma_{eff}, 0.1\right), 1\right)D_k\\
R_y &= \frac{\rho\Delta\sqrt{k}}{\mu},\quad F_3 = e^{-\left(\frac{R_y}{120}\right)^8},\quad F_1 = \max\left(F_{1orig}, F_3\right)
\end{aligned}
\tag{2.26}
$$

where P_k, D_k, F_{1orig} are the original production and destruction terms and the blending function for $k-\omega$ SST model.

This model is denoted as SST-2003-LM2015. For 2D cases (where crossflow effects are not present) $D_{SCF} = 0$.

2.1.4.2 The SA-noft2-Gamma-Retheta Transition Model

D' Alessandro et al., [23, 22], proposed slight modifications to the original $\gamma-\tilde{Re}_{\theta t}$ transition model, Sec. 2.1.4.1, in order to couple it with the Spalart–Allmaras turbulence model; the new model is denoted as SA-noft2-Gamma-Retheta. These modifications refer to the F_{onset}, F_{length} and $Re_{\theta c}$ correlations (present in the γ PDE) and the $F_{\theta t}$ term (present in the $\tilde{Re}_{\theta t}$ PDE). A constant value of Tu is considered for the entire computational domain.

$$\begin{gathered} F_{onset} = \max\left(F_{onset2} - F_{onset3}, 0\right), \quad F_{onset2} = \min\left[\max\left(F_{onset1}, F_{onset1}^4\right), 4\right] \\ F_{onset3} = \max\left[\left(2 - \left(\frac{R_T}{2.5}\right)^3\right), 0\right], \quad F_{onset1} = \frac{Re_\nu}{2.193 Re_{\theta c}}, \quad R_T = \frac{\mu_t}{\mu} \end{gathered} \tag{2.27}$$

$$F_{length} = \min\left(\exp\left(7.168 - 0.01173\tilde{Re}_{\theta t}\right) + 0.5, 3000\right) \tag{2.28}$$

$$Re_{\theta c} = \min\left(0.615\tilde{Re}_{\theta t} + 61.5, \tilde{Re}_{\theta t}\right) \tag{2.29}$$

$$F_{\theta t} = \min\left[\max\left[\exp\left[-\left(\frac{\Delta}{\delta}\right)^4\right], 1 - \left(\frac{\gamma - 1/c_{e_2}}{1 - 1/c_{e_2}}\right)^2\right], 1\right] \tag{2.30}$$

In order to interact with the Spalart–Allmaras turbulence model, the production term of Eq. 2.9 is multiplied with γ_{eff}, f_{t_2} is set to zero and the following modifications to the Spalart–Allmaras source terms are made

$$\tilde{P}_{\tilde{\nu}} = \gamma_{eff} \rho c_{b_1} \tilde{S} \tilde{\nu} \tag{2.31}$$

$$\tilde{D}_{\tilde{\nu}} = \frac{\rho}{\mathrm{Re}_0} c_{w_1} f_w \left(\frac{\tilde{\nu}}{\Delta}\right)^2 \tag{2.32}$$

$$\tilde{S} = \left[\zeta + \min\left(S - \zeta\right)\right] + \frac{\tilde{\nu} f_{v_2}}{\mathrm{Re}_0 \kappa^2 \Delta^2} \tag{2.33}$$

$$r = \begin{cases} 10 & , \text{if } \frac{\tilde{\nu}}{\mathrm{Re}_0 \tilde{S} \kappa^2 \Delta^2} < 0 \\ \min\left(10, \frac{\tilde{\nu}}{\mathrm{Re}_0 \tilde{S} \kappa^2 \Delta^2}\right) & , \text{if } \frac{\tilde{\nu}}{\mathrm{Re}_0 \tilde{S} \kappa^2 \Delta^2} \geq 0 \end{cases} \tag{2.34}$$

It should be noted that crossflow effects are not present. The rest of terms are similar to those described in Sec. 2.1.4.1.

2.1.4.3 The SA-LM2015 Transition Model

In [113], the original $\gamma - \tilde{R}e_{\theta t}$ transition model proposed by Langtry and Menter (described in Sec. 2.1.4.1) was modified by Piotrowski and Zingg and coupled with the Spalart–Allmaras turbulence model. A significant modification was the replacement of the strain rate magnitude (S) with vorticity (ζ), in the production term of the intermittency equation, in order to enhance the stability near the laminar separation bubbles, [113]. This model is denoted by SA-LM2015. The production and destruction terms for the intermittency equation read

$$\begin{aligned} P_\gamma &= \rho c_{\alpha_1} F_{length} F_{onset} \zeta \sqrt{\gamma} \left(1 - c_{\epsilon_1}\gamma\right) \\ E_\gamma &= \rho c_{\alpha_2} F_{turb} \zeta \gamma \left(c_{\epsilon_2}\gamma - 1\right) \end{aligned} \tag{2.35}$$

For the γ equation, the following modifications are made compared to the SST-2003-LM2015 transition model

$$\begin{aligned} &F_{onset} = \max\left(F_{onset2} - F_{onset3}, 0\right), \quad F_{onset2} = \min\left[\max\left(F_{onset1}, F_{onset1}^4\right), 4\right] \\ &F_{onset3} = \max\left[\left(2 - \left(\frac{R_T}{2.5}\right)^3\right), 0\right], \quad F_{onset1} = \frac{Re_S}{2.193 Re_{\theta c}}, \\ &Re_S = \frac{\rho \Delta^2 S}{\mu} \mathrm{Re}_0, \quad R_T = \frac{\mu_t}{\mu} \end{aligned} \tag{2.36}$$

while, for the $\tilde{R}e_{\theta t}$ equation

$$F_{\theta t} = F_{wake} \exp\left[-\left(\frac{\Delta}{\delta}\right)^4\right], \quad F_{wake} = \exp\left[-{}^{Re_S}/_{1.e^6}\right] \tag{2.37}$$

In order to interact with the Spalart–Allmaras turbulence model, the production term of Eq. 2.9 is multiplied with γ, f_{t_2} is set to zero and the following modifications to the Spalart–Allmaras model source terms are made

$$\tilde{P}_{\tilde{\nu}} = \gamma \rho c_{b_1} \tilde{S} \tilde{\nu} \tag{2.38}$$

$$D_{\tilde{\nu}} = \frac{\rho}{\mathrm{Re}_0} c_{w_1} f_w \left(\frac{\tilde{\nu}}{\Delta}\right)^2 \tag{2.39}$$

For the Reynolds number Re_0, the reference velocity is proposed to be the sound speed according to [113]. The rest of terms are similar to those described in Sec. 2.1.4.1.

2.1.4.4 The SA-sLM2015 Transition Model

The expressions of the source terms in the transition models presented in Secs. 2.1.4.1, 2.1.4.2, 2.1.4.3 include min. and max. operators as well as conditional statements. These non-smooth functions can lead to discontinuities, not only during the numerical solution of the primal equations (see Sec. 2.4) but also during the formulation of the adjoint problem. In [113], to overcome this, smooth approximations to the min./max. operators and simplifications to the conditional statements were introduced by Piotrowski and Zingg based on the SA-LM2015 model, giving rise to the SA-sLM2015 transition model.

The smooth min./max. operator function ϕ_p ($p>0$ stands for the max. operator and $p<0$ for the min.) for two variables x_1, x_2 is described as

$$\begin{aligned}
\text{for p>0:}\quad \phi_p\left(x_1,x_2\right) &= \begin{cases} \alpha & \text{, if } |\alpha-\beta| > -\frac{\log(|p|p_{switch})}{|p|} \\ \alpha+\frac{\log(1+\exp\left[p(\beta-\alpha)\right])}{p} & \text{, else} \end{cases} \\
\text{for p<0:}\quad \phi_p\left(x_1,x_2\right) &= \begin{cases} \beta & \text{, if } |\alpha-\beta| > -\frac{\log(|p|p_{switch})}{|p|} \\ \beta+\frac{\log(1+\exp\left[p(\alpha-\beta)\right])}{p} & \text{, else} \end{cases} \\
\alpha = \max\left(x_1,x_2\right),&\quad \beta = \min\left(x_1,x_2\right),\quad p_{switch}=10^{-15}
\end{aligned} \tag{2.40}$$

Then, the production and destruction terms of the γ PDE read

$$\begin{aligned}
P_\gamma &= \rho c_{\alpha_1} F_{length} F_{onset} \left[\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\right] \sqrt{\gamma}\left(1-c_{\epsilon_1}\gamma\right) \\
E_\gamma &= \rho c_{\alpha_2} F_{turb} \left[\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\right] \gamma\left(c_{\epsilon_2}\gamma-1\right)
\end{aligned} \tag{2.41}$$

F_{onset}, F_{turb}, F_{length} and $Re_{\theta c}$ are expressed as

$$
\begin{aligned}
F_{onset} &= \frac{\tanh\left[6\left(F_{onset1}-1.35\right)\right]+1}{2}, \quad F_{onset1} = \sqrt{\left(\frac{Re_S}{2.6Re_{\theta c}}\right)^2 + \left(R_T\right)^2} \\
F_{length} &= 44 - \frac{44 - \left(0.50 - 3\cdot 10^{-4}\left(\tilde{R}e_{\theta t} - 596\right)\right)}{\left(1+F_{length1}\right)^{1/6}}, \quad F_{length1} = \exp\left(-3\cdot 10^{-2}\left(\tilde{R}e_{\theta t} - 460\right)\right) \\
Re_{\theta c} &= 0.67\tilde{R}e_{\theta t} + 24\sin\left(\frac{\tilde{R}e_{\theta t}}{240} + 0.5\right) + 14 \\
F_{turb} &= \left(1 - F_{onset}\right)\exp\left(-R_T\right)
\end{aligned}
\tag{2.42}
$$

Regarding the $\tilde{R}e_{\theta t}$ equation, the expression of $F\left(\lambda_\theta\right)$ is

$$
\begin{aligned}
F\left(\lambda_\theta\right)_1 &= 1 + 0.275\left[1 - \exp^{[-35\lambda_\theta]}\right]\exp^{-\frac{Tu}{0.5}} \\
F\left(\lambda_\theta\right)_2 &= \phi_{300}\left(F\left(\lambda_\theta\right)_1, 1\right) \\
F\left(\lambda_\theta\right)_3 &= 1 + \left[12.986\lambda_\theta + 123.66\lambda_\theta^2 + 405.689\lambda_\theta^3\right]\exp^{\left[-(Tu/1.5)^{1.5}\right]} \\
F\left(\lambda_\theta\right) &= \phi_{-300}\left(F\left(\lambda_\theta\right)_2, F\left(\lambda_\theta\right)_3\right)
\end{aligned}
\tag{2.43}
$$

The coupling with the Spalart–Allmaras turbulence model is the same as for the SA-LM2015 model

$$\tilde{P}_{\tilde{\nu}} = \gamma\rho c_{b_1}\tilde{S}\tilde{\nu} \tag{2.44}$$

$$D_{\tilde{\nu}} = \frac{\rho}{\mathrm{Re}_0}c_{w_1}f_w\left(\frac{\tilde{\nu}}{\Delta}\right)^2 \tag{2.45}$$

2.1.5 The Hamilton–Jacobi Equation

The distance Δ field from the closest solid walls which is necessary for the source terms of the turbulence and transition models is computed based on the Hamilton–Jacobi (Eikonal) equation as

$$R^\Delta = \frac{\partial}{\partial x_k}\left(\Delta\frac{\partial\Delta}{\partial x_k}\right) - \Delta\frac{\partial}{\partial x_k}\left(\frac{\partial\Delta}{\partial x_k}\right) - 1 = 0 \tag{2.46}$$

2.2 Boundary Conditions

Eqs. 2.1, 2.9, 2.11 and 2.13 must be associated with a set of appropriate boundary conditions. These are imposed in a strong or weak way depending on the boundary condition type. For more details, the reader is referred to [135].

Wall Boundary Conditions

- Slip condition: the no-penetration rule applies; the normal component of the velocity relative to the wall is set equal to the normal component of the wall velocity. This is expressed as $v_k^A \mathsf{n}_k = v_k^W \mathsf{n}_k$, where $v_k^W, (k = 1, 2, 3)$ is the known wall velocity (i.e. $v_k^W = 0$ for stationary walls and non-zero otherwise).

- Non-slip condition: the absolute velocity is set equal to the wall velocity, namely $v_k^A = v_k^W$,

In addition, for the Spalart–Allmaras turbulence model, $\tilde{\nu}$ is set to zero $\tilde{\nu} = 0$, while for the $k-\omega$ SST model, $k = 0$ and $\omega = \frac{60\tilde{\nu}}{\beta_1 \Delta^2} \frac{1}{\mathrm{Re}_0}$. For the $\gamma - \tilde{Re}_{\theta t}$ transition model $\frac{\partial \gamma}{\partial n} = 0$ and $\frac{\partial \tilde{Re}_{\theta t}}{\partial n} = 0$

Regarding the thermal conditions for the wall boundaries, these can be **a)** adiabatic, $q_k \mathsf{n}_k = 0$ **b)** constant temperature (T^W), $T = T^W$ or **c)** constant heat flux (q^W), $q_k \mathsf{n}_k = q^W$.

Inlet Boundary Conditions For subsonic inlet boundaries, the total pressure (p_t), total temperature (T_t) and the flow angles (θ_1, θ_2) (in 3D) are specified. The velocity components are then defined as

$$\begin{aligned} v_1^A &= |\mathbf{U}| \sin\theta_1 \\ v_2^A &= |\mathbf{U}| \cos\theta_1 \sin\theta_2 \\ v_3^A &= |\mathbf{U}| \cos\theta_1 \cos\theta_2 \end{aligned} \tag{2.47}$$

Four quantities are specified and a fifth one has to be extrapolated from the flow domain. Either the velocity magnitude, the local Mach number or the static pressure is extrapolated from the domain interior. Then Eqs. 2.6, 2.7 along with the expression of the Mach number are used to compute all the necessary flow quantities. In the Spalart–Allmaras turbulence model, the inlet viscosity ratio $\left(\frac{\nu_t}{\nu}\right)^{IN}$ is specified. In the $k-\omega$ SST model, the inlet viscosity ratio and the turbulence intensity (Tu) are specified and the turbulence model variables are computed as $k^{IN} = 1.5 Tu^2 |\mathbf{U}|^2$ and $\omega^{IN} = k/\nu_t$. For the $\gamma - \tilde{Re}_{\theta t}$ transition model, $\gamma^{IN} = 1$ and $\tilde{Re}_{\theta t}$ is computed based on Eq. 2.21.

Outlet Boundary Conditions For subsonic outlet boundaries, a single flow variable is specified. This can be **a)** the outlet static pressure distribution, **b)** the outlet mean static pressure or **c)** the outlet mass flow rate. For the last two options, values for pressure and the normal to the outlet velocity are computed iteratively by uniformly correcting those extrapolated from the fluid domain, so as to achieve the imposed integral quantity. The remaining four quantities are extrapolated from the interface of the fluid domain. For the turbulence and transition models, the outlet is assumed to be convective and zero Neumann boundary conditions are imposed to $\tilde{\nu},\ k,\ \omega,\ \gamma$ and $\tilde{Re}_{\theta t}$.

Far–field Boundaries The far–field boundaries are treated as a combination of inlet and outlet boundaries, depending on the local velocity field; if flow enters the domain is locally treated as inlet, otherwise as outlet.

Symmetry Boundaries Symmetry conditions are imposed: $\frac{\partial U_m}{\partial x_k}\hat{\mathsf{n}}_k^{\mathrm{sym}} = 0$ and $\hat{\mathsf{n}}_k^{\mathrm{sym}}$, $(m = 1, 2, 3)$ are the unit normal to the symmetry plane components.

Periodic Boundaries Periodic conditions are imposed along pairs of periodic boundaries. For linear cascades, two points are periodically paired if they have two of their coordinates equal and differ on the third by the cascade pitch. All flow variables (scalar and vector ones) must be equal between periodically paired points. In case of peripheral blade rows, two points are periodically paired if their projections on the meridional plane coincide and their circumferential position differs by the blade row pitch. Between them, the scalar quantities are the same, while the vector and tensor quantities of one periodic boundary are equal to that of its pair rotated by the row pitch.

2.3 Discretization of the Governing Equations

Eqs. 2.1, 2.9, 2.11 and 2.13 are solved using a time–marching technique on unstructured grids, consisting of tetrahedra, pyramids, prisms and/or hexahedra. They are integrated over vertex–centered finite volumes. At each real time–step, a finite volume is formed around each grid node by connecting the edge midpoints, face centers and cell centers of the edges, faces and cells attached to this node. An example of the finite volume formed around node P is shown in Fig. 2.1, in a 2D case, for demonstration purposes.

2.3.1 Discretization of the Inviscid Terms

Applying the Green-Gauss theorem to the integral of the inviscid terms for the finite volume of node P and discretizing the result one obtains

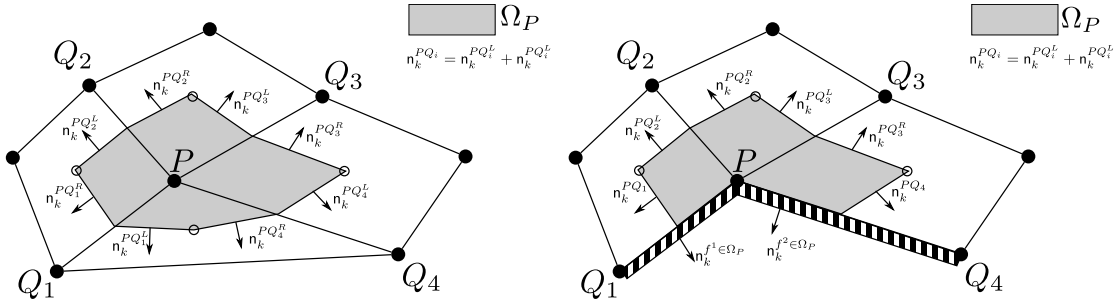


Figure 2.1: Vertex–centered finite volumes formed around an internal node P (left) and node P lying over the domain boundary (right). The normal vectors ($\mathbf{n}$) on the finite volume interface have a magnitude equal to the area (or length) of the interface between two adjacent finite volumes (dimensional normal vector).

$$\int\limits_{\Omega_P^{t_{i+1}}} \frac{\partial f_{nk}^{\text{inv}}}{\partial x_k} \mathrm{d}\Omega = \int\limits_{\partial\Omega_P^{t_{i+1}}} f_{nk}^{\text{inv}} \hat{\mathsf{n}}_k \mathrm{d}\left(\partial\Omega\right) \simeq \sum_{Q\in\mathcal{N}(P)} \Phi_n^{\text{inv},PQ} + \sum_{f\in\mathcal{B}(P)} \Phi_n^{\text{inv},f\in\Omega_P} \tag{2.48}$$

where $\partial\Omega_P^{t_{i+1}}$ is the boundary of the finite volume formed around node P at real-time step t_{i+1} and $\hat{\mathsf{n}}_k\left(t\right)$, $(k=1,2,3)$ the corresponding unit normal components, pointing outwards. $Q\in\mathcal{N}\left(P\right)$ is a neighboring node connected to P through a grid edge, $f\in\mathcal{B}\left(P\right)$ is a boundary face (if any) emanating from node P.

According to the Roe's upwind scheme [117], the flux crossing the face separating the finite volumes of nodes P and Q is

$$\Phi_n^{\text{inv},PQ} = \frac{1}{2}\left(A_{nmk}^P U_m^P + A_{nmk}^Q U_m^Q\right)\mathsf{n}_k^{PQ} - \frac{1}{2}\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|\left(U_m^R - U_m^L\right) \tag{2.49}$$

where $A_{nmk} = \frac{\partial f_{nk}^{\text{inv}}}{\partial U_m}$ stands for the flux Jacobian, $f_{nk}^{\text{inv}} = A_{nmk}U_m$ and $\mathsf{n}_k^{PQ,t_{i+1}}$ is the normal vector to the finite volume interface between nodes P and Q at time t_{i+1}, with magnitude equal with the area of the interface, Fig. 2.1. $\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k\right| = P_{n\ell}\left|\lambda_{\ell r}\right|P_{rm}^{-1}$ where Λ is the diagonal matrix containing the absolute eigenvalues of $A_{nmk}\mathsf{n}_k$ computed using the Roe–averaged quantities between the left (L) and right (R) states. The flow variables at L and R are computed with extrapolation from P and Q using second-order accuracy or appropriate limiting functions [139, 11, 140]. For second-order accuracy, the primitive flow variables V at P, Q are extrapolated to the finite volume interface using the spatial gradients computed at P, Q as follows

$$V_m^L = V_m^P + \frac{1}{2}\mathsf{t}_\ell^{PQ}\frac{\partial V_m^P}{\partial x_\ell}, \qquad \frac{\partial V_m^P}{\partial x_\ell} = D_\ell^P V_m^P + \sum_{\Lambda\in\mathcal{N}(P)} Z_\ell^{P\Lambda} V_m^\Lambda \tag{2.50a}$$

$$V_m^R = V_m^Q - \frac{1}{2}\mathsf{t}_\ell^{PQ}\frac{\partial V_m^Q}{\partial x_\ell}, \qquad \frac{\partial V_m^Q}{\partial x_\ell} = D_\ell^Q V_m^Q + \sum_{K\in\mathcal{N}(Q)} Z_\ell^{QK} V_m^K \tag{2.50b}$$

where coefficients D_ℓ and Z_ℓ are based on geometrical data. For more details, the reader is referred to [147]. Limiting functions can be applied to eliminate non-physical oscillations in areas of high gradients. The van Leer-van Albada [139] limiting function $\mathscr{L}$ can be used, according to which

$$V_m^L = V_m^P + \frac{1}{2}\mathscr{L}\left[\left(2\mathsf{t}_\ell^{PQ}\frac{\partial V_m^P}{\partial x_\ell} - \left(V_m^Q - V_m^P\right)\right), \left(V_m^Q - V_m^P\right)\right] \tag{2.51a}$$

$$V_m^R = V_m^Q - \frac{1}{2}\mathscr{L}\left[\left(2\mathsf{t}_\ell^{PQ}\frac{\partial V_m^Q}{\partial x_\ell} - \left(V_m^Q - V_m^P\right)\right), \left(V_m^Q - V_m^P\right)\right] \tag{2.51b}$$

$$\mathscr{L}(\alpha,\beta) = \begin{cases} \frac{\left(a^2+\varepsilon\right)\beta + \left(\beta^2+\varepsilon\right)\alpha}{a^2+\beta^2+2\varepsilon} & \text{, if } \alpha\beta > 0 \\ 0 & \text{, if } \alpha\beta \leq 0 \end{cases} \tag{2.52}$$

where ε is a small positive value. Nodal values (instead of L and R states) are used in the first term of Eq. 2.49. According to [6], this maintains the second-order accuracy of the scheme and facilitates the use of a similar scheme for solving the continuous adjoint equations.

The boundary flux (last term in Eq. 2.48) is computed differently for the wall, the inlet/outlet and the far-field boundaries. For the wall boundaries, $f \in \mathcal{B}^{\mathcal{W}}(P)$,

$$\Phi_n^{\text{inv},f\in\Omega_P} = f_{nk}^{\text{inv},f\in\Omega_P}\mathsf{n}_k \tag{2.53}$$

where the flux $f_{nk}^{\text{inv},f\in\Omega_P}$ is computed by Eq. 2.2 using the flow quantities at node P and taking into account the appropriate slip wall conditions, as described in Sec. 2.2. In Eq. 2.53, n_k are the components of the normal vector to the boundary f, with magnitude equal to the area of f contained in the finite volume formed around node P.

At inlet/outlet boundaries $f_{nk}^{\text{inv},f\in\Omega_P}$ is computed by Eq. 2.49, where n_k is replaced by $\mathsf{n}_k^{f\in\Omega_P^{t_{i+1}}}$ and, node Q is a halo (fake) node. The flow variables at node Q are set using the boundary conditions as described in Sec. 2.2.

At the far-field boundaries, the corresponding boundary flux is computed

based on the Flux Vector Scheme (FVS), [129], as

$$\Phi_n^{\text{inv},f\in\Omega_P} = A_{nmk}^{+P}\mathsf{n}_k U_m^P + A_{nmk}^{-P}\mathsf{n}_k U_m^Q \tag{2.54}$$

where node Q corresponds to a halo (fake) node with defined fluid's far–field density, velocity and pressure and

$$\begin{aligned} A_{nmk}^{+P}\mathsf{n}_k &= \frac{1}{2}\left(A_{nmk}^P + \left|A_{nmk}^P\mathsf{n}_k\right|\right) \\ A_{nmk}^{-P}\mathsf{n}_k &= \frac{1}{2}\left(A_{nmk}^P - \left|A_{nmk}^P\mathsf{n}_k\right|\right) \end{aligned}$$

Both Eqs. 2.49 and 2.53, contain terms of the form $v_k^R\mathsf{n}_k$ along the finite volume boundaries; these can be computed as $v_k^R\mathsf{n}_k = v_k^A\mathsf{n}_k - v_k^F\mathsf{n}_k$. The computation of $v_k^F\mathsf{n}_k$ must satisfy the Geometric Conservation Law (GCL), [76].

$$\frac{\mathrm{d}}{\mathrm{d}t}\int\limits_{\Omega_P^{t_{i+1}}}\mathrm{d}\Omega = \int\limits_{\partial\Omega_P^{t_{i+1}}} v_k^G\hat{\mathsf{n}}_k\mathrm{d}\left(\partial\Omega\right) \tag{2.55}$$

where v_k^G is the velocity of the grid. In case of grid movement, the finite volume $\Omega_P^{t_{i+1}}$ is computed using a 2^{nd} order backward difference formula (BDF2). When solving the flow equations in a relative frame of reference with a non–deformable grid, one may substitute v_k^F for v_k^G into Eq. 2.55. Taking into account that, in such a case, the volume of a cell remains constant, Eq. 2.55 becomes

$$\int\limits_{\partial\Omega_P^{t_{i+1}}} v_k^F\hat{\mathsf{n}}_k\mathrm{d}\left(\partial\Omega\right) \simeq \sum_{Q\in\mathcal{N}(P)}\left(v_k^{F,PQ}\mathsf{n}_k^{PQ}\right)^{t_{i+1}} + \sum_{f\in\mathcal{B}(P)}\left(v_k^{F,f\in\Omega_P}\mathsf{n}_k^{f\in\Omega_P}\right)^{t_{i+1}} = 0 \tag{2.56}$$

2.3.2 Discretization of the Viscous Terms

Applying the Green-Gauss theorem to the integral of the viscous terms and discretizing the result one gets

$$\int\limits_{\Omega_P^{t_{i+1}}}\frac{\partial f_{nk}^{\text{vis}}}{\partial x_k}\mathrm{d}\Omega = \int\limits_{\partial\Omega_P^{t_{i+1}}} f_{nk}^{\text{vis}}\hat{\mathsf{n}}_k\mathrm{d}\left(\partial\Omega\right) \simeq \sum_{Q\in\mathcal{N}(P)}\Phi_n^{\text{vis},PQ} + \sum_{f\in\mathcal{B}(P)}\Phi_n^{\text{vis},f\in\Omega_P} \tag{2.57}$$

where $\Phi_{m+1}^{\text{vis},PQ}$ and $\Phi_5^{\text{vis},PQ}$ are the momentum and energy viscous numerical fluxes crossing the finite volume interface between nodes P and Q. These are expressed

as $\Phi_{m+1}^{\text{vis},PQ} = \tau_{mk}^{PQ} \mathsf{n}_k^{PQ,t_{i+1}}$, $(m = 1,2,3)$ and $\Phi_5^{\text{vis},PQ} = v_\ell^{A,PQ} \left(\tau_{\ell k}^{PQ} + q_k^{PQ}\right) \mathsf{n}_k^{PQ,t_{i+1}}$ with τ_{mk}^{PQ} and q_k^{PQ} being the stress tensor and the heat flux components, respectively. The absolute velocity component at the interface PQ is computed as the half sum between the nodal values at P and Q as $v_\ell^{A,PQ} = \frac{1}{2}\left(v_\ell^{A,P} + v_\ell^{A,Q}\right)$. The same formula is used to compute μ^{PQ} and μ_t^{PQ} that appear at the viscous stress and the heat flux. Computing τ_{mk}^{PQ}, q_k^{PQ} at the interface between P and Q involves the computation of the spatial gradients of velocity and temperature there. In order to avoid odd-even decoupling issues, these are not computed as the mean of the corresponding spatial gradients at P and Q; instead, these are computed as [144]

$$\left.\frac{\partial \phi}{\partial x_m}\right|_{PQ} = \left.\overline{\frac{\partial \phi}{\partial x_m}}\right|_{PQ} - \left[\left.\overline{\frac{\partial \phi}{\partial x_\ell}}\right|_{PQ} \hat{\mathsf{t}}_\ell - \frac{\phi^Q - \phi^P}{\sqrt{\left(x_\ell^Q - x_\ell^P\right)\left(x_\ell^Q - x_\ell^P\right)}}\right] \hat{\mathsf{t}}_m \tag{2.58}$$

where ϕ can be any velocity Cartesian component or the temperature,

$$\hat{\mathsf{t}}_m = \frac{x_m^Q - x_m^P}{\sqrt{\left(x_\ell^Q - x_\ell^P\right)\left(x_\ell^Q - x_\ell^P\right)}} \tag{2.59}$$

and

$$\left.\overline{\frac{\partial \phi}{\partial x_m}}\right|_{PQ} = \frac{1}{2}\left[\left(\frac{\partial \phi}{\partial x_m}\right)_P + \left(\frac{\partial \phi}{\partial x_m}\right)_Q\right]$$

Along the boundaries, the viscous fluxes for the momentum and energy equations are computed as $\Phi_{m+1}^{vis,f\in\Omega_P} = \tau_{mk}^P \mathsf{n}_k^{f\in\Omega_P^{t_{i+1}}}$, $(m = 1,2,3)$ and $\Phi_5^{vis,f\in\Omega_P} = \left[\left(\tau_{\ell k} v_\ell^A\right)^{f\in\Omega_P} + q_k^{f\in\Omega_P}\right] \mathsf{n}_k^{f\in\Omega_P^{t_{i+1}}}$, by also taking the boundary conditions defined in Sec. 2.2 into account.

2.3.3 Discretization of the Temporal Terms

Applying the Reynolds transport theorem to the integral of the temporal term of Eq. 2.1 one obtains

$$\int_{\Omega_P^{t_{i+1}}} \frac{\partial U_n}{\partial t} \mathrm{d}\Omega = \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega_P^{t_{i+1}}} U_n \mathrm{d}\Omega - \int_{\partial\Omega_P^{t_{i+1}}} U_n v_k^G \hat{\mathsf{n}}_k \mathrm{d}\left(\partial\Omega\right) \tag{2.60}$$

By discretizing the first integral on the right-hand-side (r.h.s.) of Eq. 2.60 us-

ing a second–order accurate backward difference formula (BDF2), for constant time-step Δt, together with the discretization of the second integral, Eq. 2.60 is expressed as

$$\begin{aligned}\int_{\Omega_P^{t_{i+1}}} \frac{\partial U_n}{\partial t} \mathrm{d}\Omega \simeq & \frac{1}{2\Delta t}\left(3U_n^{P,t_{i+1}}V_P^{t_{i+1}} - 4U_n^{P,t_i}V_P^{t_i} + U_n^{P,t_{i-1}}V_P^{t_{i-1}}\right) \\ & - \sum_{Q\in\mathcal{N}(P)} \Phi_n^{grid,PQ} - \sum_{f\in\mathcal{B}(P)} \Phi_n^{grid,f\in V_P}\end{aligned} \tag{2.61}$$

with $\Phi_n^{grid,PQ} = U_n^{PQ}\left(v_k^G \mathsf{n}_k\right)^{PQ,t_{i+1}}$ and $\Phi_n^{grid,f\in V_P} = U_n^{P}\left(v_k^G \mathsf{n}_k\right)^{f\in V_P,t_{i+1}}$

The grid velocities v_k^G are computed so as to satisfy the GCL, Eq. 2.55. Employing a BDF2 scheme, this leads to the following constraint

$$\sum_{Q\in\mathcal{N}(P)} \left[v_k^G \mathsf{n}_k\right]^{PQ,t_{i+1}} + \sum_{f\in\mathcal{B}(P)} \left[v_k^G \mathsf{n}_k\right]^{f\in\Omega_P,t_{i+1}} = \frac{1}{2\Delta t}\left(3\Omega_P^{t_{i+1}} - 4\Omega_P^{t_i} + \Omega_P^{t_{i-1}}\right) \tag{2.62}$$

which must be satisfied by the discretization scheme. Employing the same scheme for the time-integration of the flow quantities and for computing the grid velocities automatically satisfies the GCL.

2.3.4 Discretization of the Source Terms

The source value S_n^P at node P is assumed to remain constant within the finite volume, yielding

$$\int_{\Omega_P^{t_{i+1}}} S_n \mathrm{d}\Omega \simeq S_n^P \Omega_P^{t_{i+1}} \tag{2.63}$$

2.3.5 Discretization of the Turbulence/Transition Model Terms

Concerning the Spalart–Allmaras, the k–ω SST turbulence models and the γ–$\tilde{Re}_{\theta t}$ transition model, for the temporal, diffusion and source terms, the same discretization schemes as in the mean flow equations are used. However, the discretization of the convection term is different. Applying the Green–Gauss theorem to this term leads to the following discretized form

$$\int\limits_{\Omega_P^{t_{i+1}}} \frac{\partial\left(\rho\varphi v_k^R\right)}{\partial x_k} d\Omega = \int\limits_{\partial\Omega_P^{t_{i+1}}} \rho\varphi v_k^R \hat{\mathsf{n}}_k d\left(\partial\Omega\right) \simeq \sum_{Q\in\mathcal{N}(P)} \Phi_{conv}^{\varphi,PQ} + \sum_{f\in\mathcal{B}(P)} \left(\rho v_k^R \varphi\right)^f \mathsf{n}_k^{f\in\Omega_P^{t_{i+1}}} \tag{2.64}$$

where φ stands for $\tilde{\nu}, k, \omega, \gamma$ or $\tilde{Re}_{\theta t}$. $\Phi_{conv}^{\varphi,PQ}$ can be discretized using either the FVS, [129], or the Roe's, [117], upwind scheme. Based on the FVS upwind scheme

$$\Phi_{conv}^{\varphi,PQ} = \left(\rho\varphi\right)^{PQ}\left(v_k^R \mathsf{n}_k\right)^{PQ,t_{i+1}} \quad \text{with} \quad \left(\rho\varphi\right)^{PQ} = \begin{cases} \left(\rho\varphi\right)^P, & \text{for } \left(v_k^R \mathsf{n}_k\right)^{PQ,t_{i+1}} > 0 \\ \left(\rho\varphi\right)^Q, & \text{for } \left(v_k^R \mathsf{n}_k\right)^{PQ,t_{i+1}} < 0 \end{cases} \tag{2.65}$$

whereas with the Roe's upwind scheme

$$\Phi_{conv}^{\varphi,PQ} = \frac{1}{2}\left[\left(\rho v_k^R\varphi\right)^P + \left(\rho v_k^R\varphi\right)^Q\right]\mathsf{n}_k^{PQ,t_{i+1}} - \frac{1}{2}\left|\left(v_k^R\mathsf{n}_k\right)^{PQ,t_{i+1}}\right|\left[\left(\rho\varphi\right)^Q - \left(\rho\varphi\right)^P\right] \tag{2.66}$$

Due to the non Galilean invariance of the γ–$\tilde{Re}_{\theta t}$ transition model, relative velocity v_k^R must be substituted with the absolute one v_k^A.

2.4 Numerical Solution of the Discretized Equations

The unsteady residuals of the discretized equations can be expressed as

$$\begin{aligned} \mathcal{R}_n^{\text{GE},P} = & \sum_{Q\in\mathcal{N}(P)} \left(\Phi_n^{\text{GE,conv}} - \Phi_n^{\text{GE,diff}}\right)^{PQ,t_{i+1}} \\ & + \sum_{f\in\mathcal{B}(P)} \left(\Phi_n^{\text{GE,conv}} - \Phi_n^{\text{GE,diff}}\right)^{f\in V_P,t_{i+1}} \\ & + S_n^P\Omega_P^{t_{i+1}} + \frac{1}{2\Delta t}\left(3\mathcal{Q}_n^{P,t_{i+1}}\Omega_P^{t_{i+1}} - 4\mathcal{Q}_n^{P,t_i}\Omega_P^{t_i} + \mathcal{Q}_n^{P,t_{i-1}}\Omega_P^{t_{i-1}}\right) \end{aligned} \tag{2.67}$$

where $\mathcal{Q}$ stands for either U_n $(n = 1,\ldots,5)$ or $(\tilde{\mu}, \rho k, \rho\omega, \rho\gamma, \rho\tilde{Re}_{\theta t})$ and GE (Governing Equation) for either MF or SA, k–ω SST, γ–$\tilde{Re}_{\theta t}$, As mentioned, Eqs. 2.67 are solved using a dual time-stepping technique, with the correction of the flow variables as unknown, namely

$$\frac{\Omega_P}{\Delta\tau_P}\Delta\mathcal{Q}_n^P = -\mathcal{R}_n^{\text{GE},P} \tag{2.68}$$

where τ is the pseudo-time step. Using a point–implicit scheme and denoting the

real-time iteration by i and the pseudo–time one by j, the system of equations reads

$$\left[\frac{\Omega_P}{\Delta\tau_P}\delta_{nm}+\left(\frac{\partial\mathcal{R}_n^{\mathrm{GE},P}}{\partial\mathcal{Q}_m}\right)_{i+1,j}\right]\left(\Delta\mathcal{Q}_m^P\right)_{i+1,j+1}=-\left(\mathcal{R}_n^{\mathrm{GE},P}\right)_{i+1,j} \tag{2.69}$$

and by splitting the left-hand-side (l.h.s.) term of these equations in diagonal and off–diagonal terms, these are written as

$$\left(\mathcal{D}_{nm}^{\mathrm{GE},P}\right)_{i+1,j}\left(\Delta\mathcal{Q}_m^P\right)_{i+1,j+1}+\sum_{Q\in\mathcal{N}(P)}\left(\mathcal{Z}_{nm}^{\mathrm{GE},PQ}\right)_{i+1,j}\left(\Delta\mathcal{Q}_m^Q\right)_{i+1,j+1}=-\left(\mathcal{R}_n^{\mathrm{GE},P}\right)_{i+1,j} \tag{2.70}$$

where

$$\left(\mathcal{D}_{nm}^{\mathrm{GE},P}\right)_{i+1,j}=\frac{\partial\left(\mathcal{R}_n^{\mathrm{GE},P}\right)_{i+1,j}}{\partial\mathcal{Q}_m^P},\quad\left(\mathcal{Z}_{nm}^{\mathrm{GE},PQ}\right)_{i+1,j}=\frac{\partial\left(\mathcal{R}_n^{\mathrm{GE},P}\right)_{i+1,j}}{\partial\mathcal{Q}_m^Q}$$

A multi-stage Runge-Kutta scheme is employed for the integration of the governing equations in the pseudo-time. For more details regarding the solution of the discretized equations and the computation of the pseudo–time step, one should refer to [135].

2.5 GPU Implementation of the Flow Solver

The in–house software PUMA was initially a CFD solver running on CPUs and during the last years it has been transferred to GPUs. This Thesis relies on the GPU variant of the PUMA code; programming and development is performed in CUDA C/C++. The PCOpt/LTT GPU cluster consists of 9 M2050, 4 K20, 4 K40, 8 P100, 4 V100, 6 NVIDIA A100 GPUs and 3 RTX 2070 GPUs; these GPUs are used to carry on all simulations presented in this Thesis.

The different architecture and hardware capabilities of GPUs compared to CPUs raise some issues concerning the discretization and numerical algorithms used for the solution of the flow equations. GPUs are shared memory processors, meaning that all GPU threads which are executed in parallel, access the same RAM. This may lead to thread race conditions which, if not resolved properly, can make the numerical solution process unpredictable. This issue, is more profound in scatter–add algorithms, employed in the computation of the numerical fluxes and the corresponding numerical flux Jacobians. In addition, the amount of cache memory of GPUs (even the latest ones) is still limited compared to CPUs,

demanding a more delicate memory handling, for minimizing the overall memory latency.

In this section, some of the GPU–specific techniques, originating from the PhD Thesis of X. Trompoukis [133] and also included in the flow solver used in this PhD Thesis, are briefly presented. In [133], the GPU variant of the flow solver was 45 times faster compared to the CPU variant of the same flow solver. It must be noted that the speed-up figures may vary depending on the selected GPU and CPU hardware.

2.5.1 Computation of Numerical Fluxes and Flux Jacobians

The most efficient way to compute the numerical fluxes and their Jacobians in a vertex-centered CPU flow solver is via an edge–based algorithm. Based on this algorithm, a sweep over all grid edges is performed, the numerical fluxes and approximated Jacobians are computed at the finite volume interface associated with each edge and, then, these are added to each of the edge's end-nodes, forming r.h.s. (residuals) and l.h.s. (Jacobian) terms. The GPU equivalent of this approach is to associate each GPU thread with a grid edge; this may lead to a race condition among threads executed in parallel. Though atomic operations can be used to avoid racing conditions, this is not preferable since an implicit synchronization among the parallel thread is implied. To overcome this problem, three different techniques have been developed in [133], namely the edge coloring method, the one–kernel and the two–kernel scheme. These techniques are presented, in brief, below.

Edge Coloring

Edge coloring technique was the first method developed to avoid using atomic operations. Based on it, grid edges are colored/grouped so that threads belonging to the same warp do not access the same memory positions; the simplest way is to group edges in a way so that edges of the same color do not share common grid nodes. The kernel computing the r.h.s. $(\mathcal{R})$ and l.h.s. $(\mathcal{D},\ \mathcal{Z})$ terms is called consecutively for each color group. On unstructured grids, a large number of groups is created and, thus, the parallel efficiency is deteriorated by this method. For this reason, the edge coloring technique is not used in the scope of this Thesis.

One-Kernel Scheme

In this scheme, each grid node is associated with a GPU thread that computes the numerical fluxes and Jacobians at all grid edges emanating from this node and add them to the memory position corresponding to it. By doing so, the problems of race condition are resolved, at the expense though of extra computations, since

the numerical fluxes and the corresponding Jacobian matrices are computed twice for all internal grid edges. Despite the need for extra computations, the one–kernel scheme outperforms the edge coloring technique, due to the increase in the GPU occupancy achieved by the kernel through a more efficient memory access. Moreover, no intermediate data are stored (in contrast to the two–kernel technique to be presented right next) resulting in low memory requirements. Overall, it offers the best compromise between execution time and memory consumption and, for that, it is used for most scatter–add operations in the flow solver.

Two-Kernel Scheme

This scheme was proposed in [9] to overcome the excessive computations associated with the one–kernel technique. Based on the two–kernel technique, two steps (and, thus, two GPU kernels) are used for the scatter–add operation. Firstly, a kernel is launched, assigning each grid edge to a GPU thread to compute the numerical fluxes and Jacobian matrices at the finite volume interface corresponding to every edge and store them using intermediate memory positions. Then, a second kernel associates every GPU thread with a grid node and sweeps all edges emanating from this node and adds flux and Jacobian contributions to $\mathcal{R}$, $\mathcal{D}$ and $\mathcal{Z}$. The computation of the fluxes and Jacobian matrices is performed once for each grid edge, making the technique computationally efficient and with the highest speed-up. However, the higher memory requirements of the two–kernel technique restricts its use, particularly in cases with huge 3D grids.

2.5.2 GPU Memory Handling

The patterns used to store and access data in memory directly affect the efficiency of the GPU code due to the limited cache memory offered by the -even latest- GPUs, compared to CPUs. In this section, the patterns of storing the l.h.s. coefficients are described, since accessing them represents the majority of the memory accesses performed during the numerical solution of the flow equations.

In the CPU implementation of the flow solver, the diagonal terms $\mathcal{D}$ are stored as a list of 5×5 matrices. This is far from an optimal implementation on a GPU since, in order to minimize cache miss memory access, each memory segment transferred through the bus to each multi-processor must contain as many useful data for the thread of the current warp as possible. Therefore, a different pattern is followed. The $(0,0)$ element of matrix $\mathcal{D}$ for the node associated with thread 0 is stored, followed by the $(0,0)$ matrix element of the node associated with thread 1 and so on, up to thread 31, completing the first thread warp (out of 32 threads which are executed simultaneously). Then, the second element of the matrix $\mathcal{D}$ for node of thread 0 follows, and so on. After the whole matrix is stored for all the nodes associated with the first warp, storage continues similarly for the next

warps. If the number of grid nodes is not a multiple of 32, and, in order for the last warp to follow the same memory access pattern, a small amount of extra memory is used. This method ensures that the memory accesses performed by the threads of a warp are as few as possible. As a result, the memory bandwidth achieved is close to the hardware nominal value.

Storing the off–diagonal terms $\mathcal{Z}$ is more challenging. The number of $\mathcal{Z}$ matrices to be stored is twice the number of grid edges. In a serial code, matrices $\mathcal{Z}$ could be stored similarly as matrices $\mathcal{D}$, one after another, following now the ordering of grid edges. Such an approach, though, does not guarantee the minimum memory accesses needed by the threads of a warp on a GPU code. For this reason, the $\mathcal{Z}$ matrices for the first edges emanating from the nodes associated with the warp 0 are first stored, followed by the $\mathcal{Z}$ matrices for the second edges and so on; storage of each matrix $\mathcal{Z}$ follows the same pattern used for storing matrices $\mathcal{D}$. However, since the number of neighbors per node is not fixed, threads of the same wrap may need to access different number of $\mathcal{Z}$ matrices. To eliminate redundant memory accesses, the grid nodes are ordered w.r.t. the number of their neighbors. This ensures that threads of the same warp need to access approximately the same number of $\mathcal{Z}$ matrices. To keep the same memory pattern for all the threads of a warp, regardless of whether the number of neighbors differ slightly among them, the memory allocated for each node is equal to the one needed from the warp thread with the greater number of neighbors.

2.5.3 Mixed Precision Arithmetics

The time spent to access the memory can be reduced not only by improving the memory access patterns (as described in Sec. 2.5.2) but also by reducing the amount of data to be assessed. This is achieved by the so–called Mixed Precision Arithmetics (MPA) [57]. In MPA, different arithmetics are used for storing the l.h.s. and r.h.s. terms. The r.h.s. terms $(\mathcal{R})$ contain the residuals of the flow equations representing the physics of the equations, while the l.h.s. terms ($\mathcal{D}$ and $\mathcal{Z}$ matrices) are linked to the convergence properties of the numerical solution. This means that not using the exact values for the l.h.s. terms (but slightly different) will slightly affect the convergence history of the numerical solution without jeopardizing the solution accuracy. According to the MPA, all quantities are computed with double precision arithmetics (DPA), the r.h.s. terms are stored in DPA but the l.h.s. terms are stored using single precision arithmetics (SPA). Apart from reducing the number of memory accesses and, consequently, the cache-miss operations needed for accessing the l.h.s. terms, the memory requirements for storing the demanding l.h.s. terms is noticeably reduced; this might become crucial in applications involving grids with a large number of nodes.

2.6 Validation and Verification Test Cases for Transition Modeling

In this section, a series of validation and verification test cases is presented. PUMA has extensively been validated for aerodynamic cases in terms of the implementation of the various numerical schemes using the Spalart–Allmaras turbulence model (without a transition model) in [135]. Here, code validation/verification focuses on the $\gamma - \tilde{Re}_{\theta t}$ transition model (all of its variants) and its coupling with the $k-\omega$ SST and the Spalart–Allmaras turbulence models, four variants in total. The purpose is to examine the accuracy of the transition models implementation in comparison with other computational results and experiments.

Five test cases, from 2D to 3D and external to internal aerodynamics, are considered: a series of flat-plate cases, the LS89 turbine blade, the NLF(1)–0416 isolated airfoil, the NLF(2)–0415 infinite swept wing and the CRM-NLF Wing, presented in Secs. 2.6.1-2.6.5, respectively. The flat plate series and the NLF(1)–0416 case are validated using all transition models (SST-2003-LM2015, SA-noft2-Gamma-Retheta, SA-LM2015 and SA-sLM2015). The LS89 turbine blade is validated only with the SST-2003-LM2015 model since, as an internal aerodynamic case, $k-\omega$ SST is more suitable. Moreover, all the SA-based transition models have been developed and validated mainly for external aerodynamics. The investigation of the NLF(2)–0415 swept wing focuses on crossflow instabilities due to surface roughness, thus the SA-noft2-Gamma-Retheta model cannot be applied since crossflow phenomena are not included. Finally, the computationally demanding CRM-NLF wing is validated using only the SA-sLM2015 model. The above are also summarized in Table 2.1.

	Flat-Plate	**LS89**	**NLF(1)-0416**	**NLF(2)-0415**	**CRM-NLF**
SST-2003-LM2015	✓	✓	✓	✓	✗
SA-noft2-Gamma-Retheta	✓	✗	✓	✗	✗
SA-LM2015	✓	✗	✓	✓	✗
SA-sLM2015	✓	✗	✓	✓	✓

Table 2.1: Summary of the validation and verification test cases and the transition model variants used per case.

2.6.1 Flat-Plate Test Cases

The ERCOFTAC T3 series [119] and the Schubauer and Klebanoff [122] flat plate experiment are often used as validation test cases for transition models. The T3A,T3B and T3A- cases are zero pressure gradient flows with different freestream

turbulence intensity levels corresponding to the transition in bypass regime. The Schubauer and Klebanoff test case has a low freestream turbulence intensity and corresponds to natural transition. The computational grid of $\sim 17K$ nodes is presented in Fig. 2.2 along with the patches where the boundary conditions are imposed. The inlet boundary conditions, for each flat plate, are summarized in Table 2.2. The non-dimensional first distance of the first nodes off the wall is $y^+ < 1$ for all test cases. A single CFD run takes ~ 6 min and ~ 5 min on an NVIDIA P100 GPU for the SST- and the SA-based models, respectively.

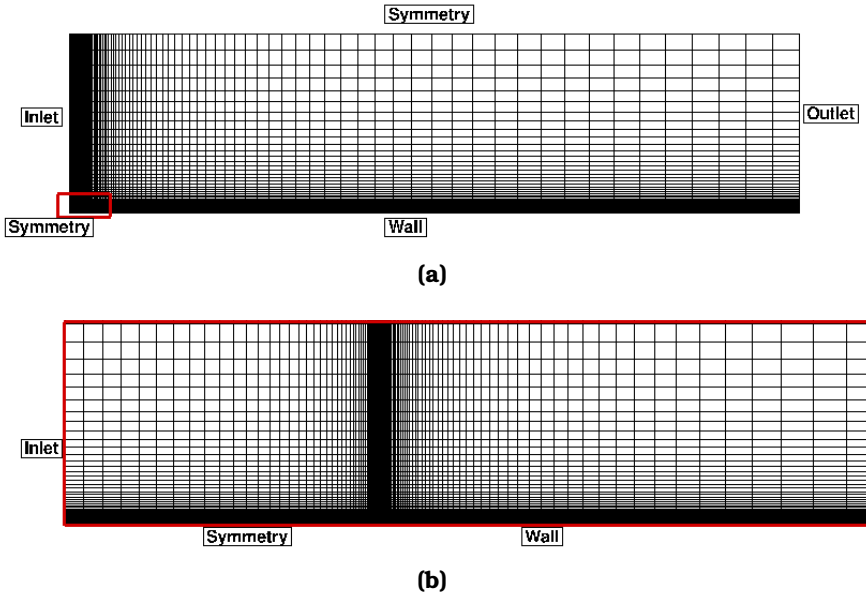


Figure 2.2: *Flat Plate Series*: (a) Structured grid and identification of boundary patches of the flat plate. (b) Close-up view to the leading edge.

Quantity	Symbol	S&K	T3A	T3B	T3A-
Velocity $(\mathrm{m\,s^{-1}})$	U	50.1	5.4	9.4	19.8
Turbulence Intensity (%)	Tu	0.18	3.3	6.5	0.874
Viscosity Ratio	(μ_t/μ)	1	12	100	8.72
Density $(\mathrm{kg\,m^{-3}})$	ρ	1.2	1.2	1.2	1.2
Dynamic Viscosity $(10^{-5}\,\mathrm{kg\,m^{-1}\,s^{-1}})$	μ	1.8	1.8	1.8	1.8

Table 2.2: *Flat Plate Series*: Inlet conditions for the flat-plate cases; the inlet is located at 0.25 m upstream of the leading edge.

The simulations are performed with the $k-\omega$ SST and the Spalart–Allmaras turbulence models coupled with the $\gamma-\tilde{Re}_{\theta t}$ transitional model, four variants in total denoted as SST-2003-LM2015, SA-noft2-Gamma-Retheta, SA-LM2015, SA-sLM2015, respectively. The skin friction coefficient along the flat plates are compared with experimental data and, also, numerical results obtained by Langtry

and Menter [73] (for the $k-\omega$ SST), Fig. 2.3. A very good agreement with the experimental data is obtained for all transition variants for the S&K and the T3A test cases. Some small discrepancies are present at high turbulence intensity (T3B) flat plate due to the very high inlet viscosity ratio value. For the SST-2003-LM2015 and the SA-sLM2015, the laminar skin friction coefficient is over-estimated while, for the SA-noft2-Gamma-Retheta and SA-sLM2015 models, the transition location is well predicted. Finally, for the T3A- test case, all but the SA-sLM2015 model match the experimental data. Comparing the original and the smoothed terms of SA-LM2015 and SA-sLM2015 models, the smoothing of F_{turb} was found to be responsible for the different behavior of the two models, in this specific case. The intermittency field is presented for the T3A case and the SST-2003-LM2015 simulation in Fig. 2.4.

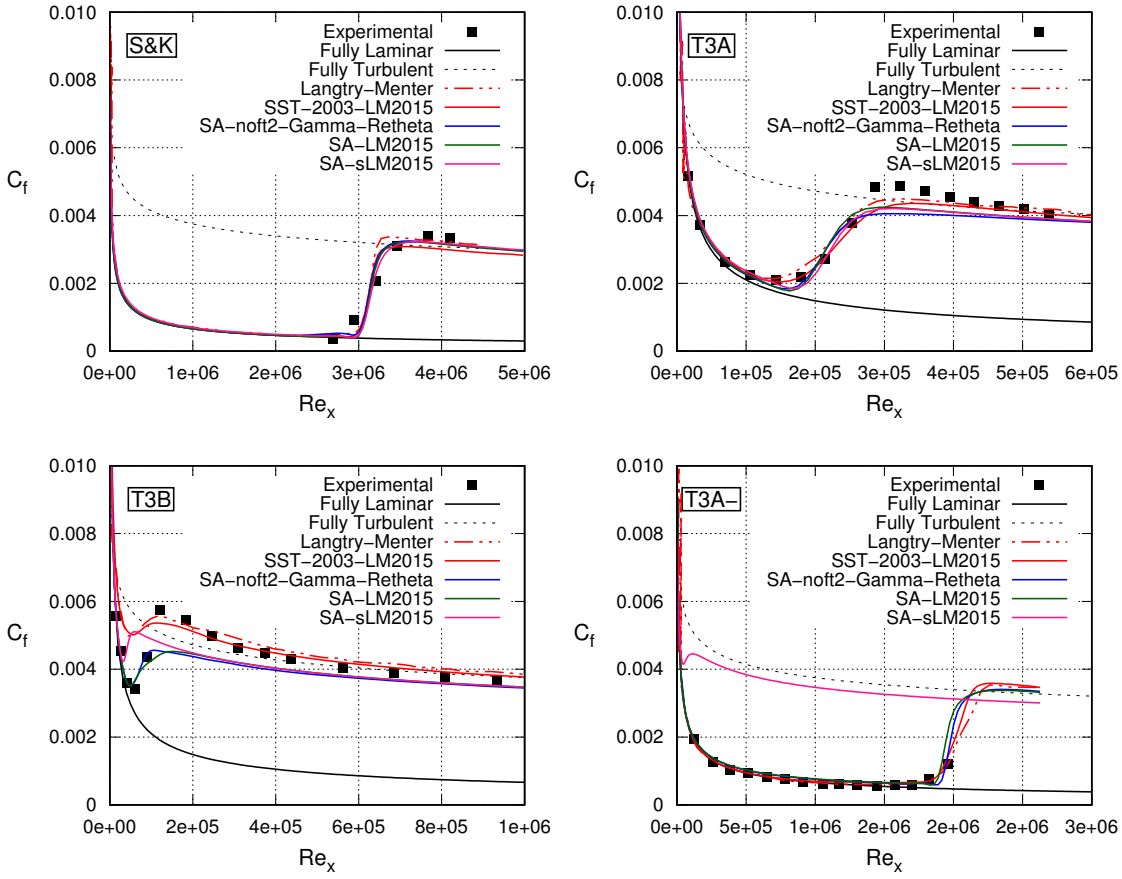


Figure 2.3: *Flat Plate Series*: Skin friction coefficient as computed by the programmed add-ons in PUMA for four transition models for the S&K, T3A, T3B and T3A- flat plates compared with experimental [119, 122] and other numerical [73] (labelled as Langtry-Menter) results. Fully laminar and turbulent results are also included.

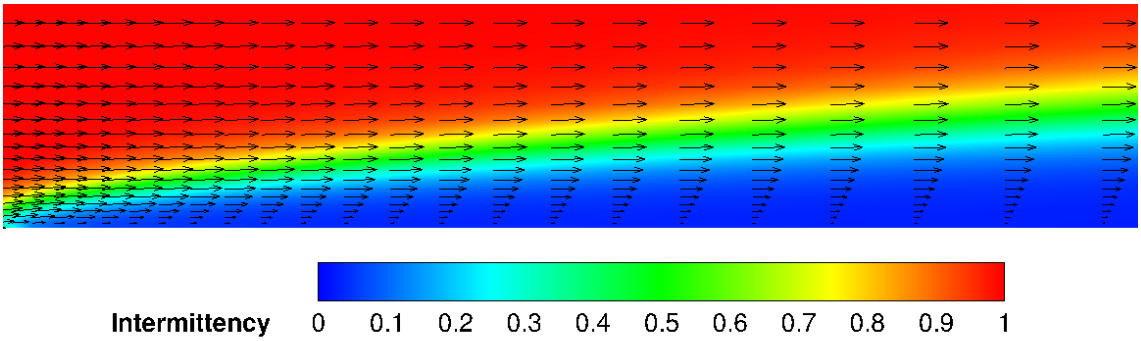


Figure 2.4: *Flat Plate Series*: Intermittency (γ) field for part of the inlet (close to the solid wall) of the T3A test case.

2.6.2 Flow through the LS89 Transonic Turbine Blade

The highly loaded transonic turbine LS89 of the von Karman Institute (VKI), [7], is validated regarding its aerothermal behavior. Two operating points with low (MUR116) and high (MUR241) inlet turbulence intensity and viscosity ratio values are considered; the flow conditions are summarized in Table 2.3. The blade's chord and pitch are $c = 67.674\,\text{mm}$ and $g = 57.50\,\text{mm}$, respectively. A block-structured grid of $\sim 50K$ nodes is generated with $y^+ < 0.22$ for the first nodes off the wall, Fig. 2.5. The SST-2003-LM2015 transition model is used and a single CFD run takes $\sim$20 min on an NVIDIA V100 GPU.

Quantity	**Symbol**	**MUR116**	**MUR241**
Inlet Total Pressure (bar)	p_t^{in}	3.269	3.257
Inlet Total Temperature (K)	T_t^{in}	418.90	416.40
Outlet Static Pressure (bar)	p^{out}	1.550	1.574
Wall Static Temperature (K)	T^{w}	297.55	299.75
Inlet Flow Angle (°)	α^{in}	0.0	0.0
Inlet Turbulence Intensity (%)	Tu^{in}	0.8	6
Inlet Viscosity Ratio	$(\mu_t/\mu)^{\text{in}}$	11	1000

Table 2.3: *LS89 Turbine Blade*: Inlet, outlet and wall conditions of the highly loaded LS89 turbine blade for Cases 116 and 241. The dynamic viscosity is computed based on the Sutherland's law.

Experimental data for the heat transfer coefficient (HTC) are available in [7]. This coefficient, Eq. 2.71, is defined as the ratio of the wall heat flux (q^w) and the difference between the total freestream (T_{01}) and the local wall temperatures (T^w) as

$$HTC = \frac{q^w}{T_{01} - T^w} \tag{2.71}$$

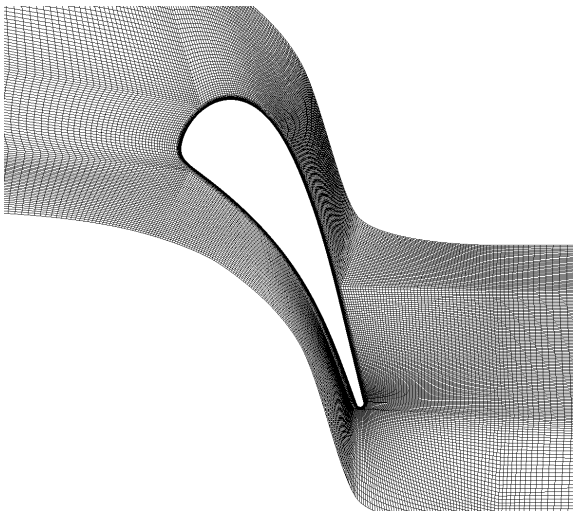


Figure 2.5: *LS89 Turbine Blade*: Block-structured CFD grid around the blade.

where $T_{01} = 420.00\,\mathrm{K}$. The HTC along the blade surface resulted from simulations on the $k-\omega$ SST with and without the $\gamma-\tilde{R}e_{\theta t}$ transition model (SST-2003-LM2015) is compared with experimental data [7] and with numerical results of the original model, [73] in Fig. 2.6. In this case, transition modeling turns out to be absolutely necessary. Regarding the pressure side of MUR116, the computed HTC with transition modeling is in excellent agreement with experimental data. For the suction side, the flow is laminar until right before the trailing edge; a difference is observed between the computed and the experimental heat transfer at this area, but this also exists at the numerical results of the original model. A difference along the suction side can also be seen in the MUR241 case. In [73], this is attributed to the effect of the Mach number on the transition length. The Mach number field and isolines is presented in Fig. 2.7; the intense of the shock wave can be seen in both blades.

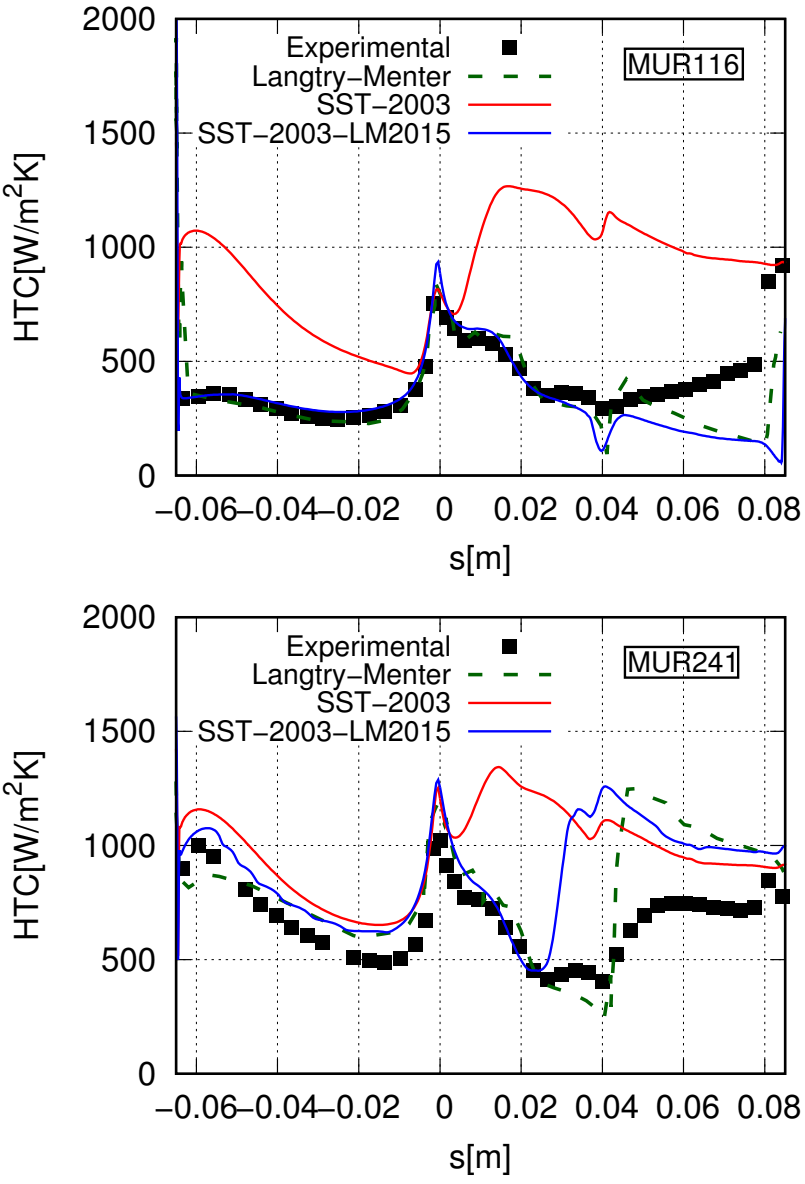
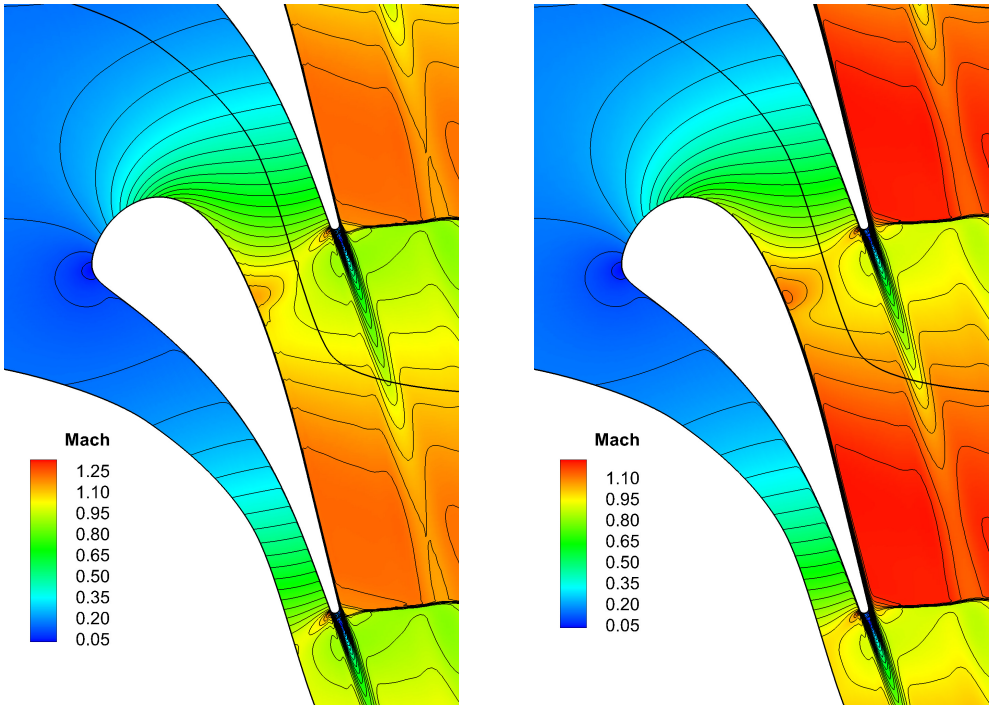


Figure 2.6: *LS89 Turbine Blade*: Comparison of the heat transfer coefficient computed using PUMA with and without transition model, with experimental data [7] and the CFD results of Langtry and Menter, [73]. Omitting the transition model leads to noticeable deviations from the experimental data.



(a) MUR116 **(b)** MUR241

Figure 2.7: *LS89 Turbine Blade*: Mach number field isolines for (a) MUR116 and (b) MUR241 cases.

2.6.3 Flow around the NLF(1)–0416 Airfoil

The NLF(1)–0416 is a low speed Natural Laminar Flow (NLF) airfoil for general aviation applications. The airfoil was tested in various Angles of Attack (AoA) between -17° to 17° and experimental data for pressure, lift, drag and moment coefficient are available in [127]. The flow conditions are $Re = 4 \cdot 10^6$, $M_{\infty} = 0.1$ and $Tu = 0.15\%$.

NLF(1)–0416 is widely used for transition modeling validation and, thus, instructions for grid generation (tiny, coarse, medium, fine, extra fine and ultra fine) can be found in [17]. The notation regarding the grid resolution is also based on [17]. A grid resolution study regarding C_L, C_D and C_m values is performed for AoA $= 0^{\circ}$ and all transition models, six grids are considered from tiny to ultra fine, Fig. 2.8. The results are summarized in Table 2.4. Based on Table 2.4, these can be categorized in two groups according to the convergence of the C_L, C_D and C_m values; the first one includes the tiny-coarse-medium grids while the second includes the medium-fine-extra fine-ultra fine grids. At the first group, Figs. 2.8a-2.8c, the difference between successive grids is up to 2%, 12% and 2% for the C_L, C_D and C_m values, respectively. For the second group, Figs. 2.8c-2.8f, the corresponding difference significantly decreases below 0.1%, 0.7% and 0.9% for the C_L, C_D and C_m values, respectively. As a result, the grid independence is achieved for grids at least of medium quality, here 705×97 or more nodes.

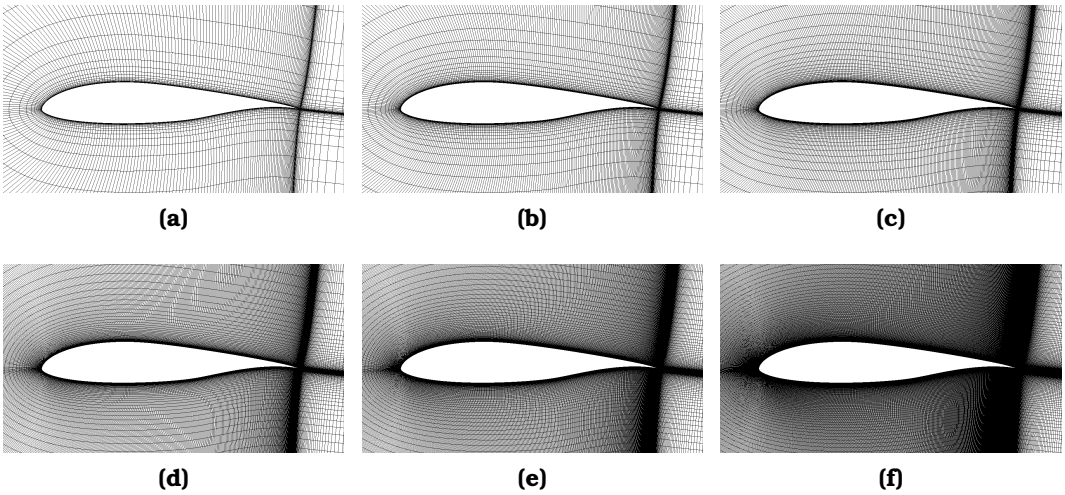


Figure 2.8: *NLF(1)-0416 Airfoil*: C–type structured grids of different quality around the airfoil. (a) Tiny (353×49), (b) Coarse (529×73), (c) Medium (705×97), (d) Fine (1057×145), (e) Extra Fine (1409×193), (f) Ultra Fine (2113×289).

The convergence histories of the residuals of the $\gamma - \tilde{Re}_{\theta t}$ transition model variants are presented in Fig. 2.9 for the fine grid (Fig. 2.8d) and AoA $= 0^{\circ}$. Regarding the γ equation, the residuals of the SST-2003-LM2015, SA-noft2-Gamma-

	$C_L = 0.447$					
	Tiny	**Coarse**	**Medium**	**Fine**	**Extra**	**Ultra**
SST-2003-LM2015	0.4809	0.4721	0.4819	0.4825	0.4824	0.4824
SA-noft2-Gamma-Retheta	0.4858	0.4814	0.4877	0.4877	0.4879	0.4882
SA-LM2015	0.4850	0.4810	0.4814	0.4877	0.4879	0.4883
SA-sLM2015	0.4859	0.4807	0.4876	0.4881	0.4884	0.4888

	$C_D = 0.0059$					
	Tiny	**Coarse**	**Medium**	**Fine**	**Extra**	**Ultra**
SST-2003-LM2015	0.005557	0.005855	0.005228	0.005188	0.005196	0.005216
SA-noft2-Gamma-Retheta	0.005715	0.005874	0.005382	0.005344	0.005332	0.005288
SA-LM2015	0.005763	0.005940	0.005441	0.005405	0.005385	0.005372
SA-sLM2015	0.005670	0.005958	0.005406	0.005353	0.005329	0.005308

	$C_m = 0.104$					
	Tiny	**Coarse**	**Medium**	**Fine**	**Extra**	**Ultra**
SST-2003-LM2015	0.1068	0.1052	0.1071	0.1071	0.1070	0.1070
SA-noft2-Gamma-Retheta	0.1079	0.1071	0.1083	0.1083	0.1083	0.1083
SA-LM2015	0.1078	0.1069	0.1081	0.1081	0.1082	0.1082
SA-sLM2015	0.1079	0.1069	0.1082	0.1082	0.1083	0.1083

Table 2.4: *NLF(1)-0416 Airfoil*: Grid independence study for the four transition models regarding C_L, C_D and C_m values at AoA $= 0.01°$. Numerical results are compared with the experimental data, [127], $C_L = 0.447$, $C_D = 0.0059$ and $C_m = 0.104$.

Retheta and SA-LM2015 models drop by $\sim$3 orders of magnitude while, for the SA-sLM2015, a deep convergence is achieved with a drop of $\sim$12 orders of magnitude. The improved numerical behavior of the SA-sLM2015 model is due to the smooth approximations that replaced the min./max. operators and the conditional statements. A deep convergence of the $\tilde{Re}_{\theta t}$ equation is obtained for all transition models; the SST-2003-LM2015 model drops by $\sim$12 orders of magnitude, while a similar behavior between all the SA-based models is present which converge by $\sim$20 orders of magnitude. It should be noted that the comparison of the convergence histories aims at demonstrating the numerical behavior and robustness of the $\gamma-\tilde{Re}_{\theta t}$ transition model variants; the differences in the convergence rates do not affect the quality of results since all transition models converge, more or less, to the same C_L and C_D values. It is important to mention that, in view of the impending differentiation and development of the continuous adjoint, the difference in convergence is significant as the adjoint method requires residuals that vanish. A comparison of their performance against experimental data follows.

Fig. 2.10 presents the C_L value w.r.t. the AoA, as well as C_L vs. C_D, C_L vs. C_m polar diagrams for the four transition models, SST-2003-LM2015, SA-noft2-Gamma-Retheta, SA-LM2015 and SA-sLM2015. Results from fully turbulent runs (Spalart–Allmaras and $k-\omega$ SST) are also included. In all cases, the use of transition model improves the quality of the results and a good agreement with the ex-

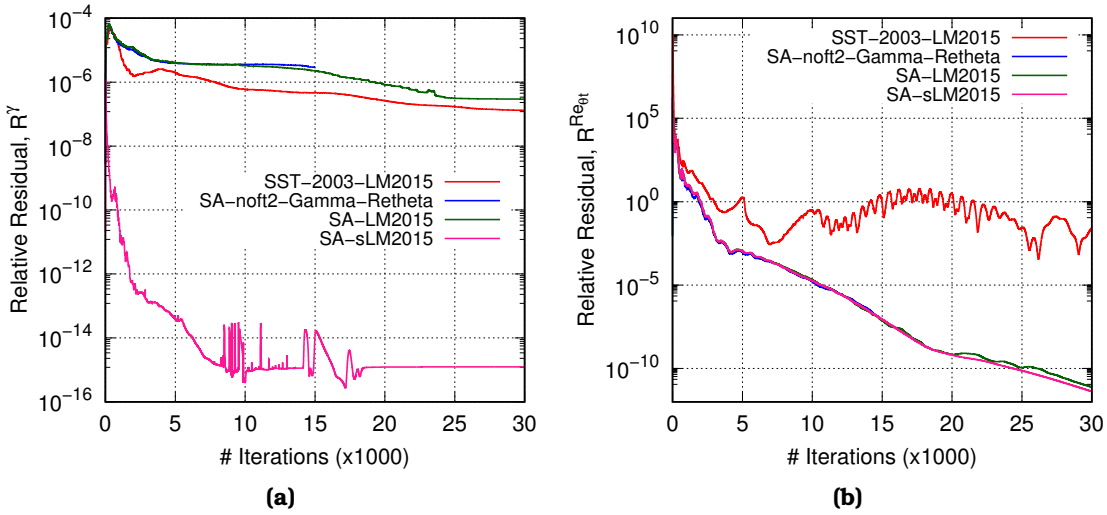


Figure 2.9: *NLF(1)-0416 Airfoil*: Relative residual convergence histories for the four transition models on the fine grid (Fig. 2.8d) and AoA$=0^{\circ}$. (a) γ and (b) $\tilde{R}e_{\theta t}$ equations.

perimental data is obtained. The skin friction coefficient along the airfoil surface, as well as the range of the transition point location (gray area) based on the experimental data, is presented in Fig. 2.11. All transition models accurately predict the transition location over the suction side, small differences are present over the pressure side where the SST-2003-LM2015 and SA-noft2-Gamma-Retheta models slightly delay the transition onset. Regarding the pressure coefficient distribution, Fig. 2.12, all transition (and turbulence) models are in good agreement with the experimental data.

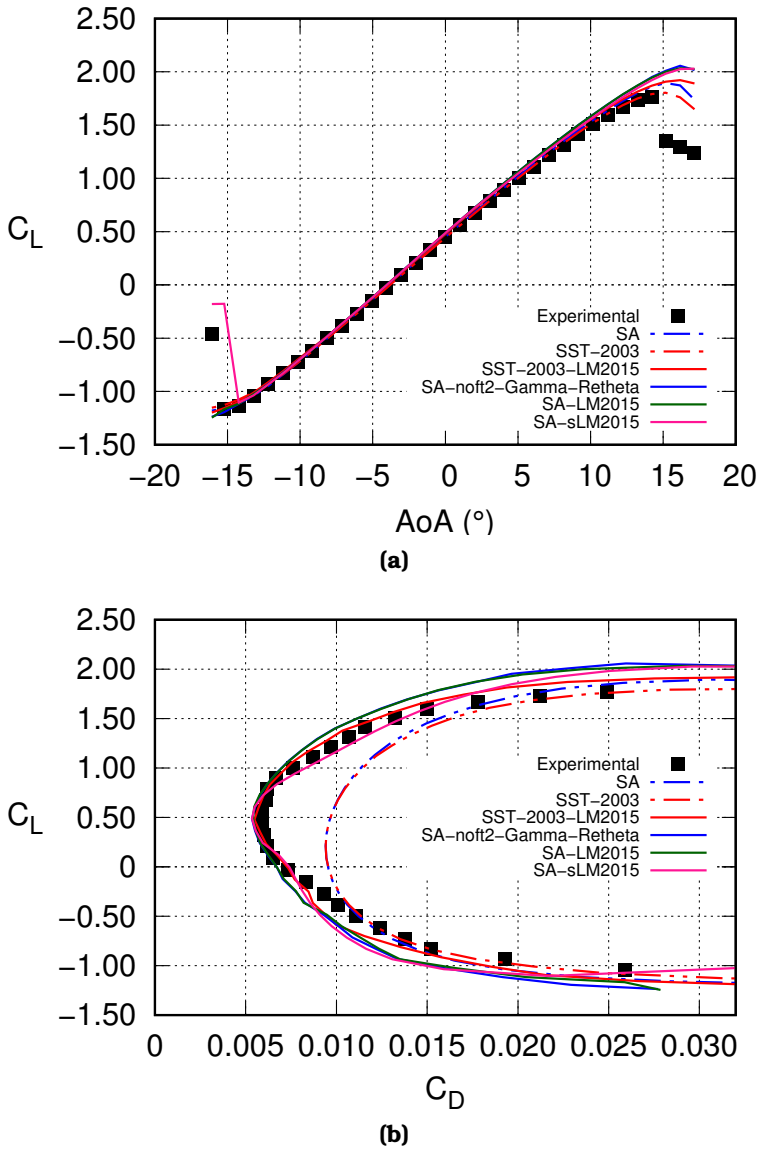
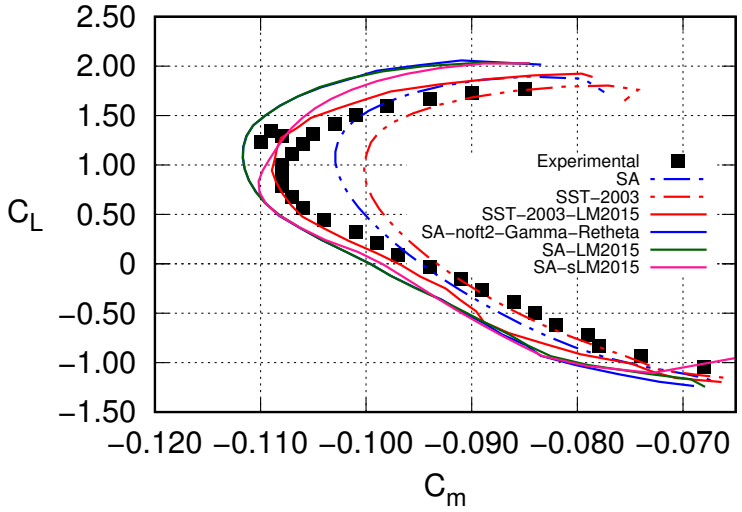


Figure 2.10: *NLF(1)-0416 Airfoil*: Comparison of the (a) C_L for several angles of attack, polar diagram (b) (C_L vs. C_D) and (c) (C_L vs. C_m). The use of a transition model with both the k–ω SST and the Spalart–Allmaras turbulence models improves the accuracy of the computed results. Results are compared with experimental data [127].



(c)

Figure 2.10: Concluded

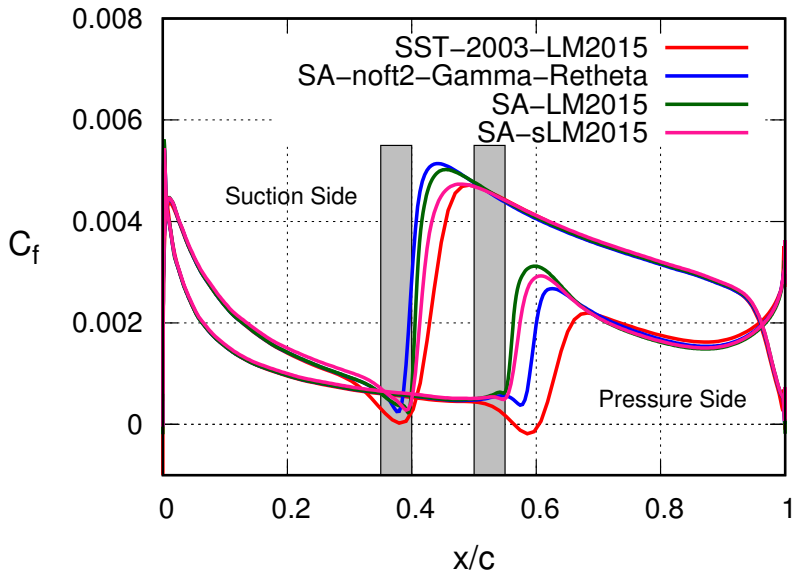


Figure 2.11: *NLF(1)-0416 Airfoil*: Skin friction coefficient distribution along the pressure and suction sides of the airfoil as computed using PUMA with various transition models, on the fine grid. The area in gray represents the range of the transition location based on the experimental data, [127].

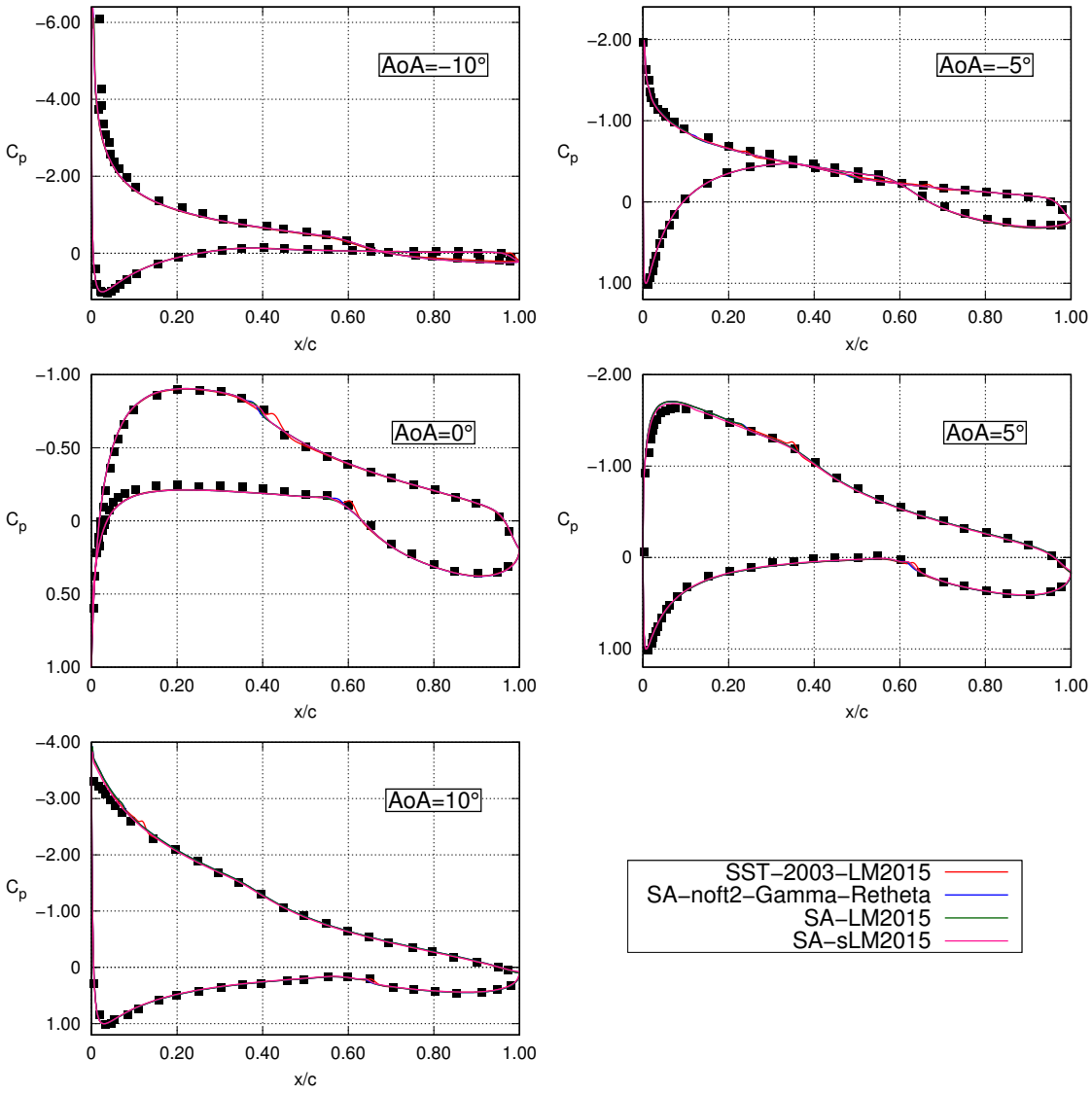


Figure 2.12: *NLF(1)-0416 Airfoil*: Comparison of the pressure coefficient distribution around the airfoil for different angles of attack; the abscissa is normalized by the airfoil's chord (c). Results are compared with experimental data, [127].

2.6.4 Flow around the NLF(2)–0415 Infinite Swept Wing

The NLF(2)–0415 infinite swept wing is widely used for validation of the stationary crossflow effects which are dominated by the surface roughness. Experiments reported by Dagenhart and Saric, [21], provide data regarding the transition location for AoA$=-4^\circ$ and for a wide range of Mach and Reynolds numbers. The simulations performed here are for $M=0.15$ and $Re=[1.92, 2.19, 2.37, 2.73, 3.27, 3.79]\cdot 10^6$, the turbulence intensity level is $Tu=0.20\%$. The NLF(2)–0415 infinite swept wing consists of the NLF(2)-0415 airfoil extruded with a 45° sweep angle. The computational grid consists of $\sim 204K$ nodes, three similar sections are created with $\sim 68K$ nodes each. The maximum y^+ value of the first off the wall nodes is $y^+ = 0.7$. In order to simulate the infinite swept wing, periodic boundaries are imposed. A single CFD run takes $\sim$17 min and $\sim$14 min on an NVIDIA V100 GPU for the SST- and the SA-based models, respectively.

The transition locations for a roughness level of $h_{rms} = 3.3\,\mu$m as resulted from the simulations with the SST-2003-LM2015, SA-LM2015 and SA-sLM2015 models are compared with the experimental ([21]) and numerical ([71, 113]) data in Fig. 2.13. All transition models predict accurately the transition location for all Reynolds numbers except for the lowest one. This discrepancy is also present in the numerical results of the original models, [113]. The skin friction coefficient along the suction side (on which the calculation of the transition location was based) is presented in Fig. 2.14 for the three transition models.

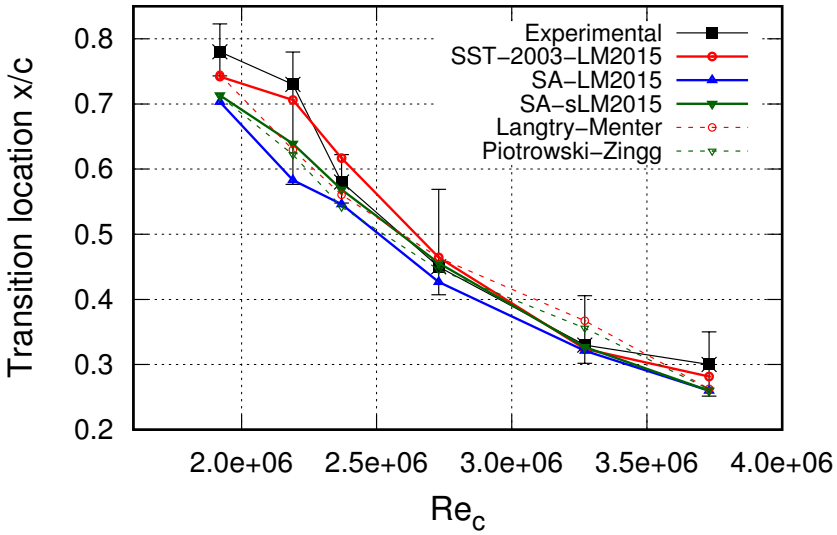


Figure 2.13: *NLF(2)-0415 Swept Wing*: Comparison of the transition location for several Reynolds numbers as computed with PUMA, Langtry and Menter [71] and Piotrowski and Zingg [113] and as experimentally measured, [21].

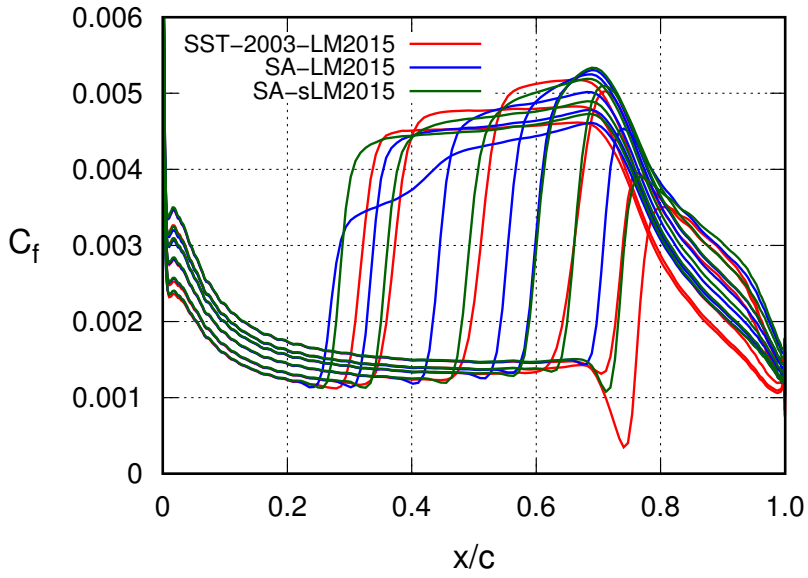
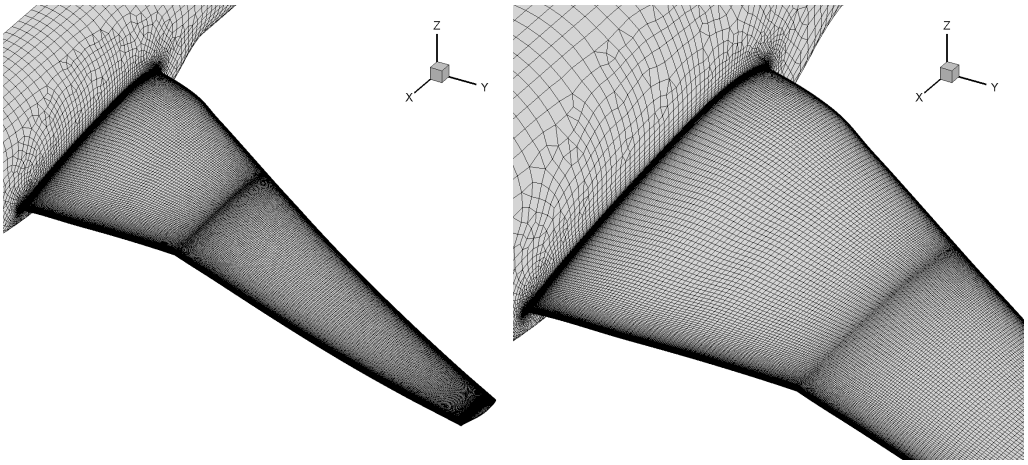


Figure 2.14: *NLF(2)-0415 Swept Wing*: Comparison of the skin friction coefficient along the suction side of the wing as computed with PUMA with various transition models. For each transition model, six curves are plotted which correspond to the six Reynolds numbers of the simulations.

2.6.5 Flow around the NASA CRM-NLF Wing

The NASA Common Research Model with Natural Laminar Flow (CRM-NLF) wing is used as a 3D validation test case. Experimental data regarding lift (C_L), drag (C_D) and moment (C_m) coefficients, as well as pressure coefficient (C_p) distributions over a range of angle of attacks are available, [83, 84]. Whereas the nominal conditions are $M_\infty = 0.86$ and $Re = 15 \cdot 10^6$, the exact test conditions, provided by the First AIAA Transition Modeling and Prediction Workshop, [1], (Point 2523, 2524, 2525, 2526) are summarized in Table 2.5. For all operation points, the turbulence intensity and surface roughness are $Tu = 0.24\%$ and $h_{rms} = 0.0254\,\mu\text{m}$, respectively. An unstructured grid of $\sim 14M$ nodes is generated with maximum $y^+ = 1.01$ of the first nodes off the walls, Fig. 2.15; this corresponds to a medium-grid level ($L1$), [127]. A single CFD run takes $\sim 2.5\,\text{h}$ on an NVIDIA A100 GPU. Computations are carried out for the half aircraft using symmetry conditions.

Simulations using the SA-sLM2015 transition model are performed for all test points of Table 2.5. The resulted C_L, C_D and C_m values and the C_p distributions for different spanwise cuts across the wing are compared with experimental data, [83], in Table 2.6 and Figs. 2.16, 2.17, 2.18 and 2.19, respectively. Overall, the computed results are in good agreement with experimental data; small discrepancies appear at the spanwise cut close to the wing's tip ($\eta = 0.910$) for all test points where the simulations under-predict the transition onset. One should keep in



(a) **(b)**

Figure 2.15: *CRM-NLF Wing*: (a) Structured grid on the wing surface. Part of the unstructured grid on the fuselage surface is also shown. (b) Close up view to the wing's crank and the joint with the fuselage.

mind that compressibility corrections are not included at the transition model's PDEs. The skin friction and the pressure coefficients on the CRM-NLF surface are plotted for each test point in Figs. 2.20 and 2.21.

Quantity	**Symbol**	**2523**	**2524**	**2525**	**2526**
Freestream Mach Number	M_∞	0.856489	0.856491	0.856051	0.855801
Angle of Attack $(^\circ)$	AoA	1.44848	1.98031	2.46141	2.93787
Reynolds Number $(\cdot 10^6)$	Re	14.97197	14.94591	14.90909	14.85308

Table 2.5: *CRM-NLF Wing*: Flow conditions for the four test points.

Point	**2523**			**2524**		
	C_L	C_D	C_m	C_L	C_D	C_m
Experimental	0.312	0.0174	-0.083	0.378	0.0192	-0.091
SA-sLM2015	0.311	0.0167	-0.081	0.365	0.0189	-0.085

Point	**2525**			**2526**		
	C_L	C_D	C_m	C_L	C_D	C_m
Experimental	0.439	0.0212	-0.099	0.500	0.0238	-0.109
SA-sLM2015	0.405	0.0213	-0.087	0.523	0.0238	-0.085

Table 2.6: *CRM-NLF Wing*: Comparison of C_L, C_D and C_m, computed by PUMA with experimental data, [83], for several test points.

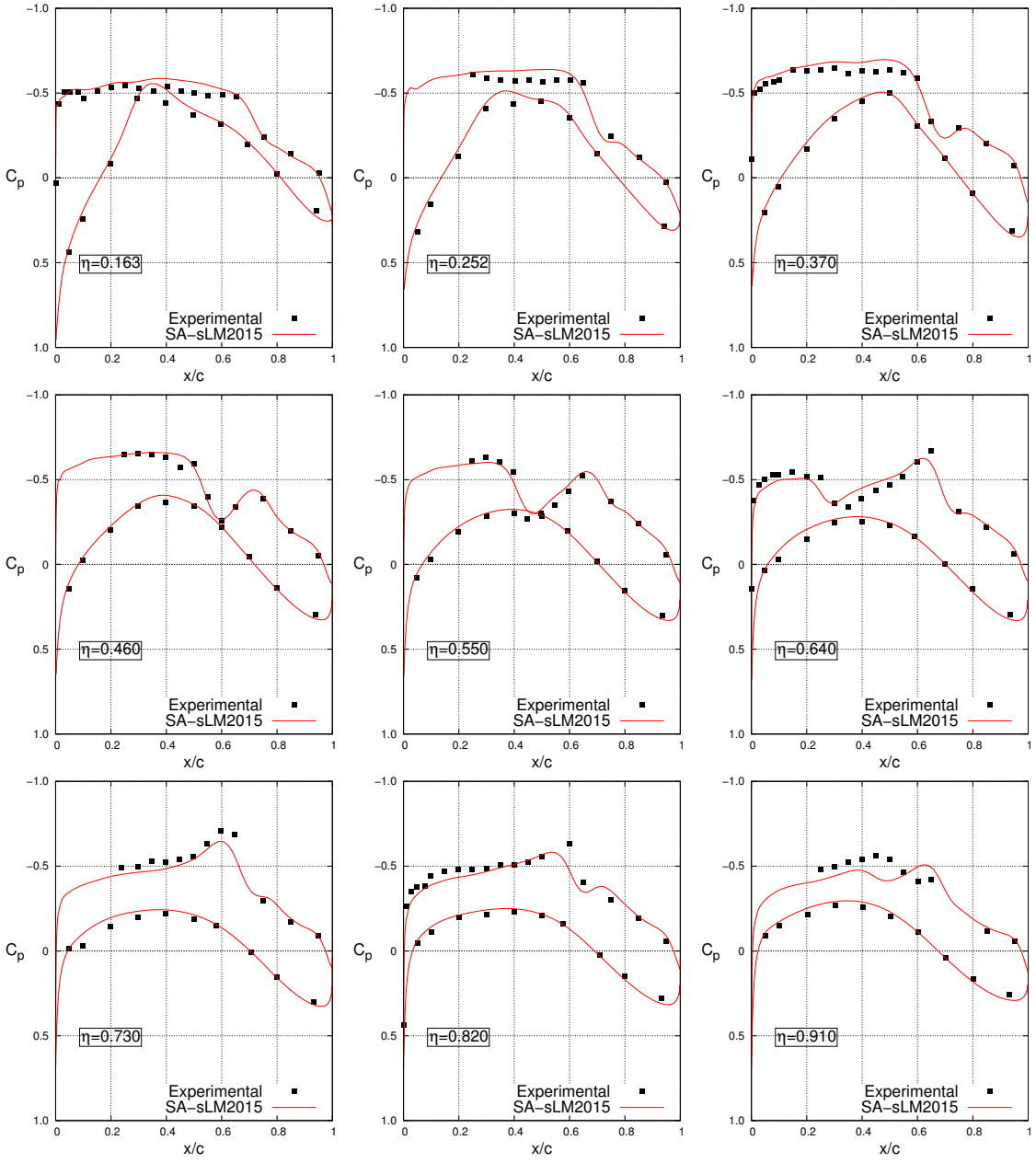


Figure 2.16: *CRM-NLF Wing*: Pressure coefficient distributions at different spanwise cuts (η) compared with experimental data, [83], for Point 2523 (AoA $\simeq 1.5°$).

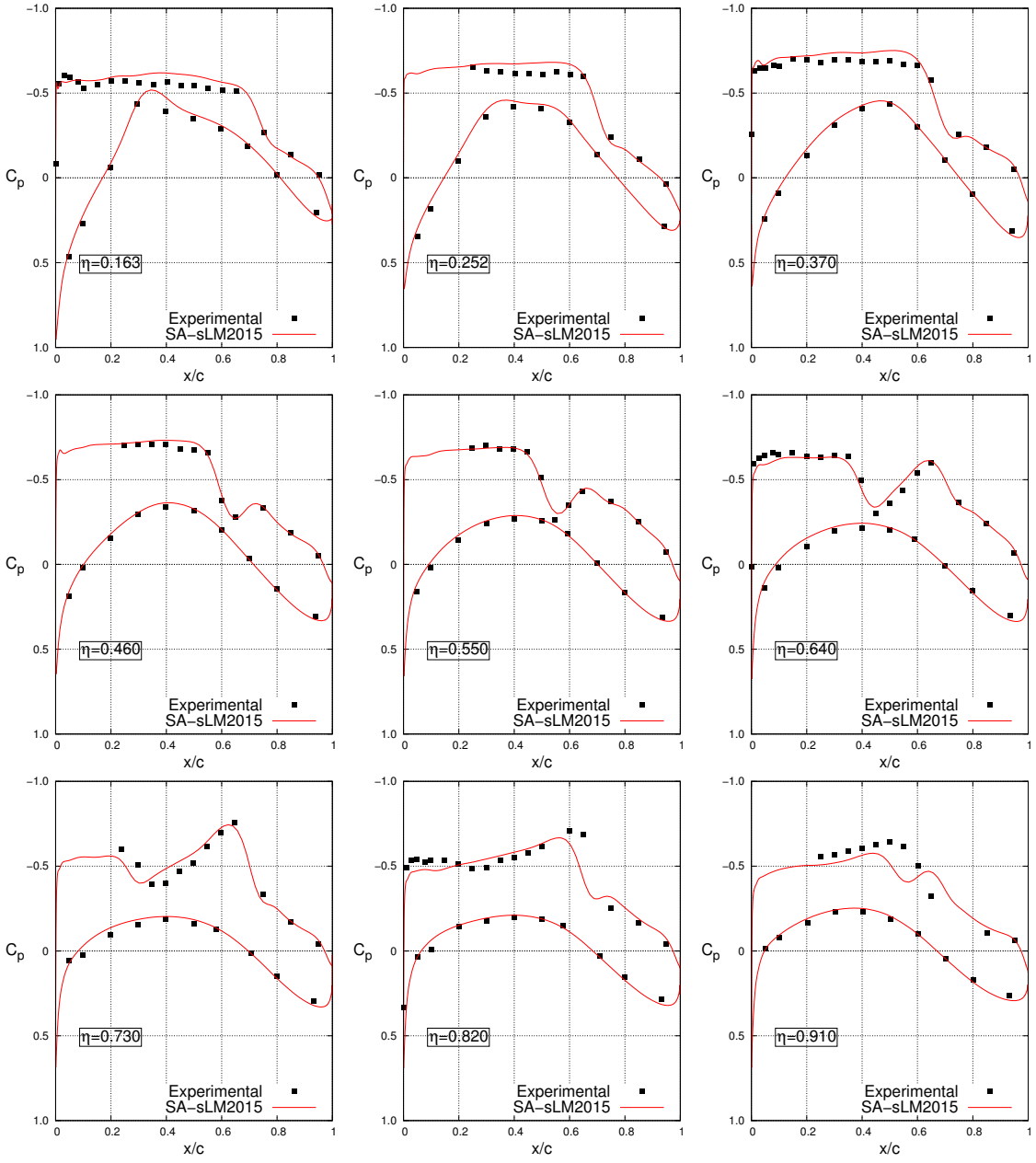


Figure 2.17: *CRM-NLF Wing*: Pressure coefficient distributions at different spanwise cuts (η) compared with experimental data, [83], for Point 2524 (AoA $\simeq 2.0°$).

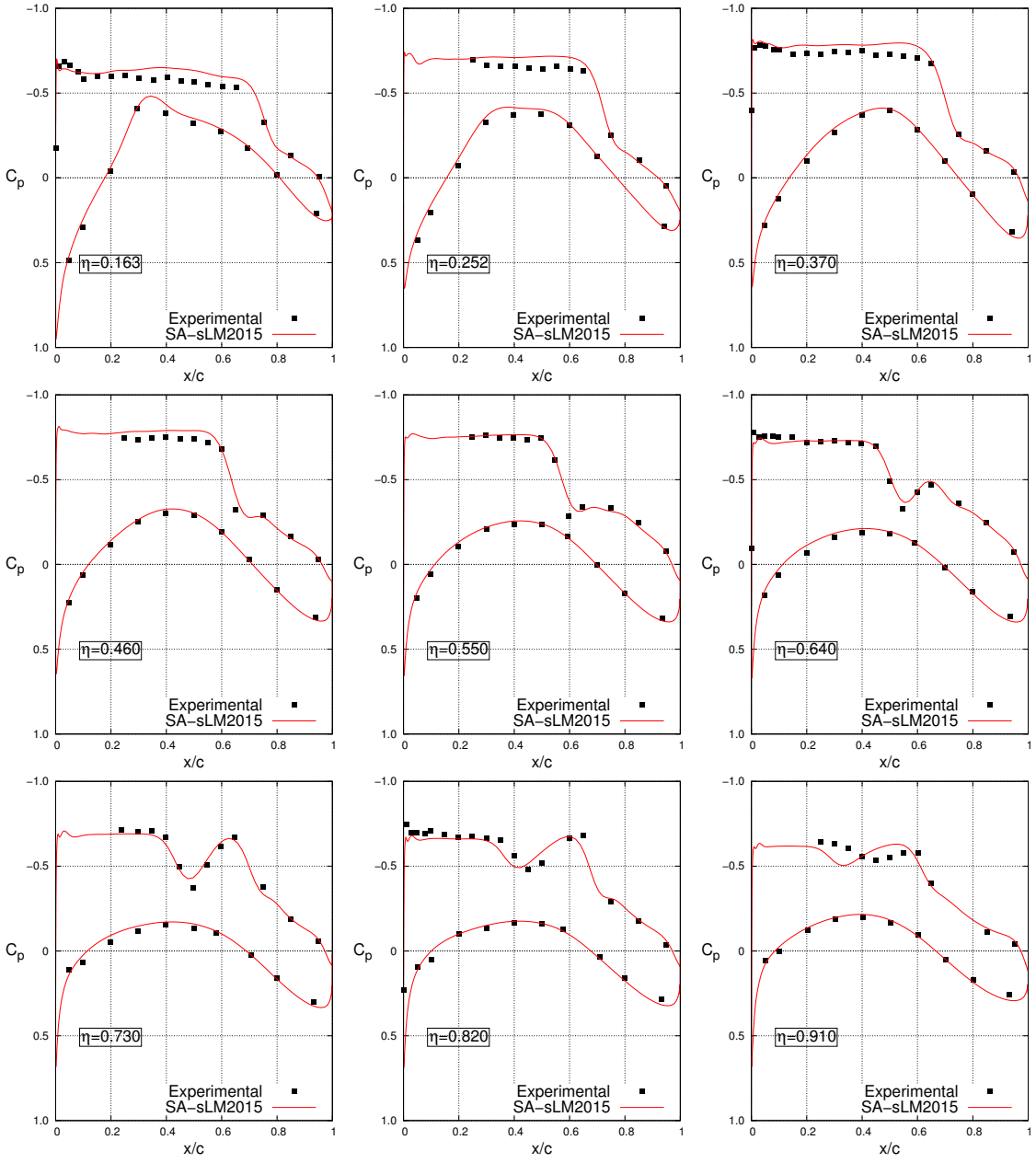


Figure 2.18: *CRM-NLF Wing*: Pressure coefficient distributions at different spanwise cuts (η) compared with experimental data, [83], for Point 2525 (AoA $\simeq 2.5°$).

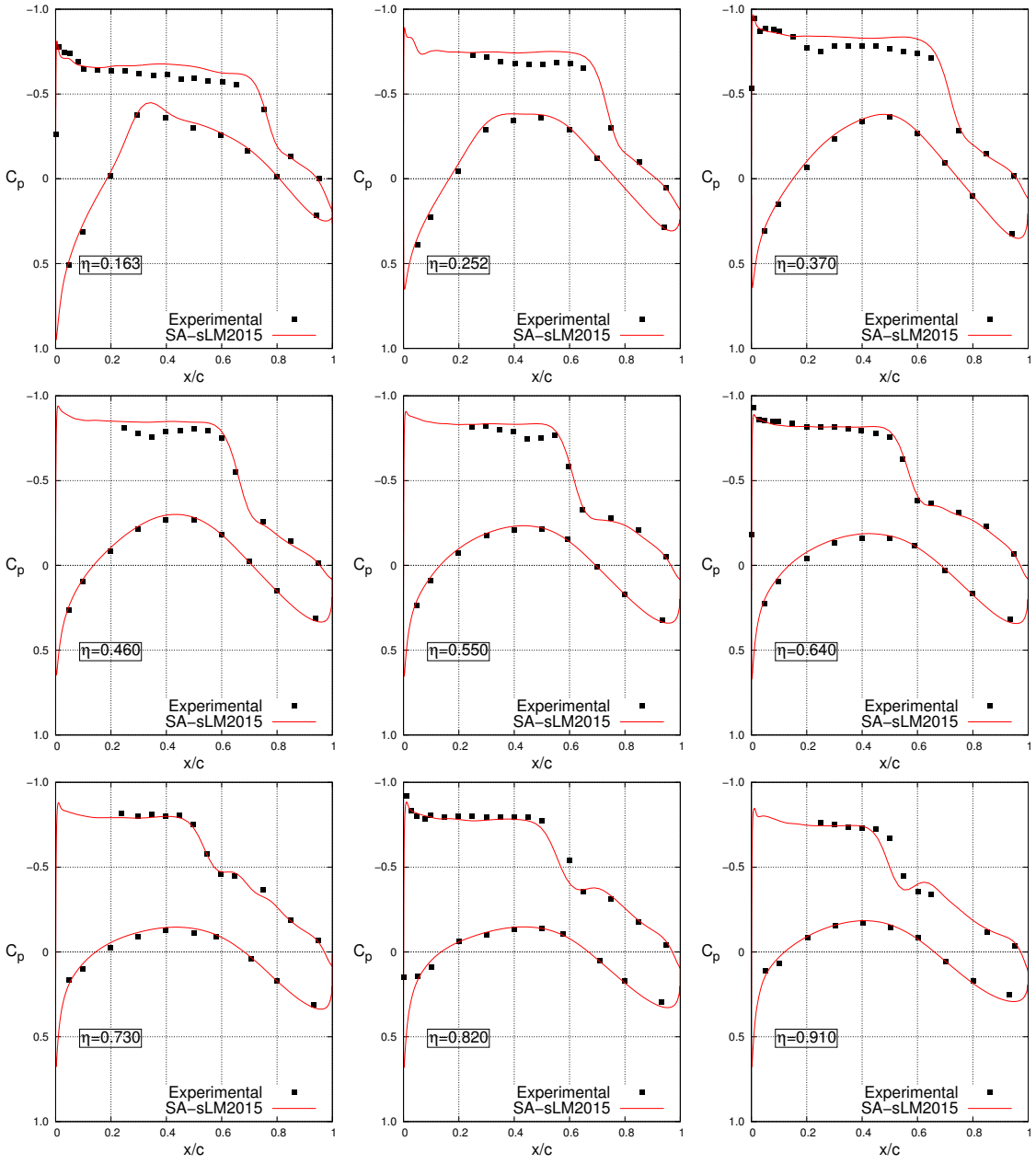
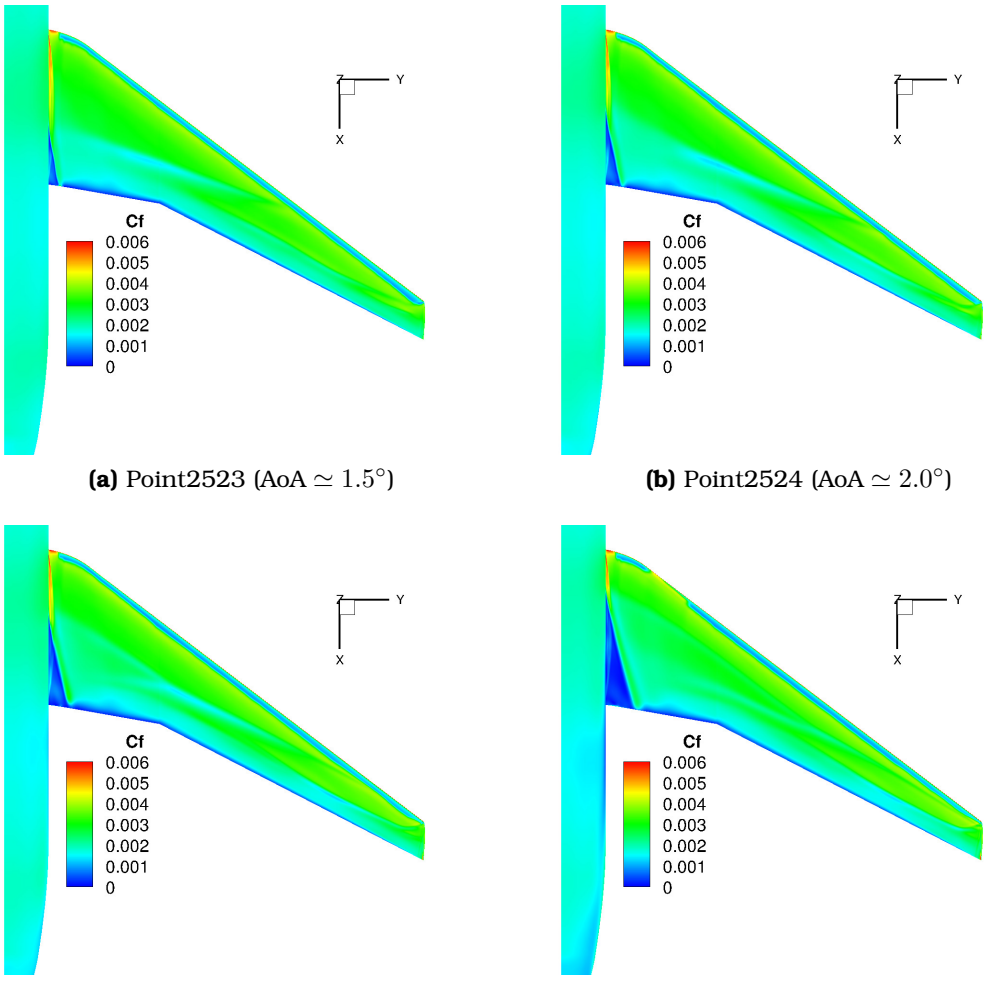


Figure 2.19: *CRM-NLF Wing*: Pressure coefficient distributions at different spanwise cuts (η) compared with experimental data, [83], for Point 2526 (AoA $\simeq 3.0°$).



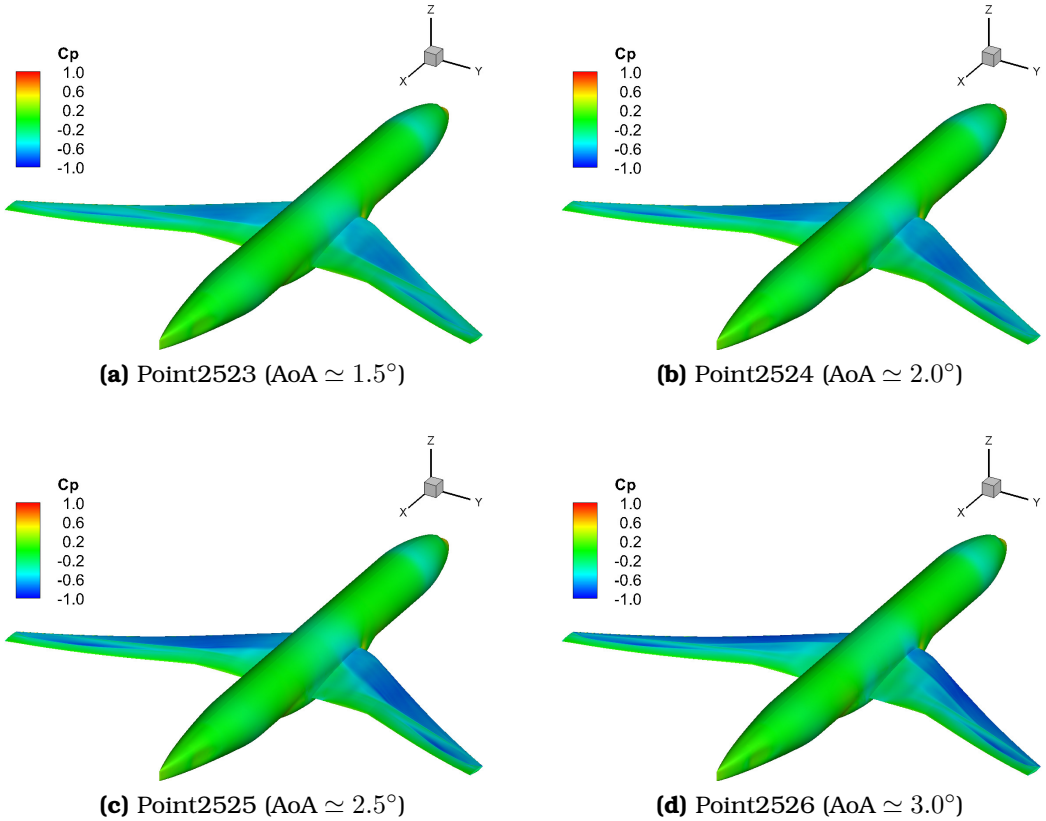
(a) Point2523 (AoA $\simeq 1.5°$)

(b) Point2524 (AoA $\simeq 2.0°$)

(c) Point2525 (AoA $\simeq 2.5°$)

(d) Point2526 (AoA $\simeq 3.0°$)

Figure 2.20: *CRM-NLF Wing*: Skin friction coefficient fields for points (a) 2523, (b) 2524, (c) 2525 and (d) 2526.



(a) Point2523 (AoA $\simeq 1.5°$)

(b) Point2524 (AoA $\simeq 2.0°$)

(c) Point2525 (AoA $\simeq 2.5°$)

(d) Point2526 (AoA $\simeq 3.0°$)

Figure 2.21: *CRM-NLF Wing*: Pressure coefficient fields for points (a) 2523, (b) 2524, (c) 2525 and (d) 2526.

2.7 Conclusions

In this chapter, the governing equations for transitional flows of compressible fluids have been presented. The system of the primal mean flow equations was closed by the Spalart–Allmaras or the $k-\omega$ SST turbulence model and the $\gamma-\tilde{Re}_{\theta t}$ transition model. To this end, the in-house GPU-enabled solver PUMA was extended with the implementation of 4 transition model variants, namely the SST-2003-LM2015, SA-noft2-Gamma-Retheta, SA-LM2015 and SA-sLM2015. Modifications to the pre-existing turbulence models (Spalart–Allmaras and $k-\omega$ SST) were also made for them to be coupled with the transition model. The programming was performed in CUDA C/C++. The discretization of the governing equations was described in detail along with the method for their numerical solution. Discussion regarding the implementation of the discretization and the numerical solution algorithm due to the characteristics of the GPUs was also performed.

The new features of the PUMA code were extensively validated and verified against experimental data and numerical results. A variety of test cases widely used in the literature was selected (internal and external aerodynamics, subsonic and transonic, 2D and 3D). A good agreement was obtained between the numerical results of PUMA and the experimental data. Although all models were able to capture the transitional phenomena of the validation and verification cases with high accuracy, the robustness of the SA-sLM2015 model should be mentioned. The improved numerical behavior of the SA-sLM2015 model, and not its predictive capability which is equivalent to that of the rest transition model variants, should be taken under consideration in view of the development of the continuous adjoint.

Based on the work presented in this chapter, the programmed software, i.e. the new PUMA code with the capabilities to also simulate transitional flows, is considered a reliable tool to be implemented in shape optimization cases, using evolutionary algorithms and gradient-based optimization, provided that the adjoint to the transition model equations will be available too (see Chapter 3).

Chapter 3

The Continuous Adjoint Method for Shape Optimization

In this chapter, the development of the continuous adjoint method for use in aerodynamic shape optimization problems is presented. The continuous adjoint method for turbulent flows in the PUMA code was developed and presented in the PhD Thesis of K. Tsiakas, [135], and herein is extended to transitional flows by differentiating the $\gamma - \tilde{Re}_{\theta t}$ transition model. Gradient–based shape optimization involves the computation of the derivatives of the objective function J w.r.t. the design variables $b_i,\ i = 1, \ldots, N$, in each optimization cycle. This gradient $\frac{\delta J}{\delta b_i}$ will also be referred to as sensitivity derivatives.

Let J be an integral quantity,

$$J = \int_{S_{\text{Obj}}} J_S \mathrm{d}S \tag{3.1}$$

where S_{Obj} usually is a subset of the boundary of the flow domain Ω. Any field quantity Φ can be expressed as a function of the flow variables $\mathcal{U}_n$, $(n{=}1,\ldots,9$ with $\mathcal{U}_m \coloneqq U_m$ for $m{=}1,\ldots,5,\ \mathcal{U}_6 \coloneqq \tilde{\nu},\ \mathcal{U}_7 \coloneqq \gamma,\ \mathcal{U}_8 \coloneqq \tilde{Re}_{\theta t},\ \mathcal{U}_9 \coloneqq \Delta\Big)$ and the position in space $x_k,\ (k{=}1,\ldots,3)$. A distinction between the partial derivative of a field quantity Φ with respect to b_i $\left(\dfrac{\partial \Phi}{\partial b_i}\right)$ and the corresponding total derivative $\left(\dfrac{\delta \Phi}{\delta b_i}\right)$ has to be made. The partial derivative refers to the change in Φ caused exclusively by changes in the flow variables $\mathcal{U}_n$, while the total derivative includes also the change in Φ due to the changed location x_k of a grid node. The partial and total derivatives are linked through

$$\frac{\delta \Phi}{\delta b_i} = \frac{\partial \Phi}{\partial b_i} + \frac{\partial \Phi}{\partial x_k}\frac{\delta x_k}{\delta b_i} \tag{3.2}$$

Using Eq. 3.2 and the Leibniz rule, Eq. 3.1 can be differentiated as follows

$$\frac{\delta J}{\delta b_i} = \int\limits_{S_{\text{Obj}}} \frac{\partial \left(J_S \mathrm{d}S\right)}{\partial \mathcal{U}_n} \frac{\delta \mathcal{U}_n}{\delta b_i} + \int\limits_{S_{\text{Obj}}} \frac{\partial \left(J_S \mathrm{d}S\right)}{\partial \left(\mathsf{n}_k \mathrm{d}S\right)} \frac{\delta \left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} + \int\limits_{S_{\text{Obj}}} \frac{\partial \left(J_S \mathrm{d}S\right)}{\partial x_k} \frac{\delta x_k}{\delta b_i} \tag{3.3}$$

The second and third integral on the r.h.s. of Eq. 3.3 can be computed by taking the parameterization of the surface into account. However, the first integral contains derivatives of the flow variables $\mathcal{U}_n$ w.r.t. b_i, the computation of which has a high cost. For instance, if central finite differences (FDs) are used, the cost amounts to two equivalent flow solutions (EFS) for each design variable (two geometries resulting per design variable by modifying it by a small number $\pm\epsilon$). Thus, FDs become very expensive in problems with many design variables ($N >>$) as their cost scales with N. With the adjoint method, however, the cost of computing $\frac{\delta J}{\delta b_i}$ is independent of the number of design variables.

The adjoint method can be developed in its discrete or continuous form. The continuous adjoint method itself can be formulated as either a surface integral (SI) [106, 151] or a field integral (FI) [81, 107] adjoint. Though mathematically equivalent, they lead to different sensitivity derivative expressions and likely different accuracies, especially in turbulent flow cases. Here, the continuous FI adjoint method developed in [135] for turbulent flows will be extended to transitional flows by differentiating the SA-noft2-Gamma-Retheta, the SA-LM2015 and the SA-sLM2015 transition models.

In order to avoid the computationally expensive terms of Eq. 3.3, the objective function J is augmented by the field integrals of the products of the primal equations' residuals and the adjoint variables over Ω, resulting in the augmented objective function J_{aug}

$$J_{\text{aug}} = J + \int\limits_{\Omega} \Psi_n R_n^{\text{MF}} \mathrm{d}\Omega + \int\limits_{\Omega} \tilde{\nu_a} R^{\tilde{\nu}} \mathrm{d}\Omega + \int\limits_{\Omega} \gamma_a R^{\gamma} \mathrm{d}\Omega + \int\limits_{\Omega} \tilde{Re}_a R^{\tilde{Re}_{\theta t}} \mathrm{d}\Omega + \int\limits_{\Omega} \Delta_a R^{\Delta} \mathrm{d}\Omega \tag{3.4}$$

In Eq. 3.4 $\Psi_n (n = 1, ..., 5)$ are the mean flow adjoint fields and $\tilde{\nu_a}$, γ_a, $\tilde{Re}_a$ and Δ_a are the adjoint Spalart–Allmaras model variable, the adjoint intermittency, the adjoint transition momentum-thickness Reynolds number and the adjoint distance, respectively. These act as Lagrange multipliers since they multiply the equality constraints of the flow equations in the problem of minimizing J_{aug}. Upon convergence of the flow equations, their residuals vanish so $J_{\text{aug}} \equiv J$ and the sought SDs are given by $\frac{\delta J_{\text{aug}}}{\delta b_i}$.

Next step is to differentiate J_{aug} w.r.t. the design variables b_i. The volume integrals of J_{aug} are differentiated by passing the operator $\frac{\delta}{\delta b_i}$ inside the integral.

For this development, the following equation, [107, 62],

$$\frac{\partial}{\partial x_\ell}\left(\frac{\delta\Phi}{\delta b_i}\right) = \frac{\partial}{\partial x_\ell}\left(\frac{\partial\Phi}{\partial b_i}\right) + \frac{\partial^2\Phi}{\partial x_k \partial x_\ell}\frac{\delta x_k}{\delta b_i} + \frac{\partial\Phi}{\partial x_k}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_k}{\delta b_i}\right) \tag{3.5}$$

where Φ is any flow-related quantity, must be used. From Eq. 3.2 one can also get an expression for the total derivative of the spatial gradient of Φ as

$$\frac{\delta}{\delta b_i}\left(\frac{\partial\Phi}{\partial x_\ell}\right) = \frac{\partial}{\partial b_i}\left(\frac{\partial\Phi}{\partial x_\ell}\right) + \frac{\partial^2\Phi}{\partial x_k \partial x_\ell}\frac{\delta x_k}{\delta b_i} \tag{3.6}$$

Subtracting Eq. 3.5 from Eq. 3.6, the total derivative of the spatial gradient of Φ w.r.t. b_i is linked to the spatial gradient of the total derivative of Φ through

$$\frac{\delta}{\delta b_i}\left(\frac{\partial\Phi}{\partial x_\ell}\right) = \frac{\partial}{\partial x_\ell}\left(\frac{\delta\Phi}{\delta b_i}\right) - \frac{\partial\Phi}{\partial x_k}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_k}{\delta b_i}\right) \tag{3.7}$$

Eq. 3.7 is extensively used in the remainder of this section, in which the FI continuous adjoint is presented. Differentiating J_{aug} results in

$$\frac{\delta J_{\text{aug}}}{\delta b_i} = \frac{\delta J}{\delta b_i} + \underbrace{\int_\Omega \Psi_n \frac{\delta R_n}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{MF}} + \underbrace{\int_\Omega \tilde{\nu_a}\frac{\delta R^{\tilde{\nu}}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{SA}} + \underbrace{\int_\Omega \gamma_a \frac{\delta R^{\gamma}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\gamma}} + \underbrace{\int_\Omega \tilde{Re}_a \frac{\delta R^{\tilde{Re}_{\theta t}}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\tilde{Re}_{\theta t}}} + \underbrace{\int_\Omega \Delta_a \frac{\delta R^{\Delta}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\Delta}} \tag{3.8}$$

During the mathematical development of $\frac{\delta J_{\text{aug}}}{\delta b_i}$, volume and surface integrals containing $\frac{\delta \mathcal{U}_n}{\delta b_i}$ arise, the computation of which should be avoided. To eliminate the expensive volume integrals, the multipliers of $\frac{\delta \mathcal{U}_n}{\delta b_i}$ are set to zero. By doing so, a new set of PDEs arises, the so–called field adjoint equations (FAE). A similar approach is followed for the surface integrals leading to the adjoint boundary conditions (ABC). Surface or volume integrals which include derivatives of geometric quantities w.r.t. b_i contribute to the SDs; their computation is performed by differentiating the implemented grid deformation tool.

In what follows, the six terms on the r.h.s. of Eq. 3.8 are differentiated. This work relies upon the development of the continuous adjoint for turbulent flows performed in the PhD Thesis of K. Tsiakas [135], and extends it to transitional flows. Herein, the mathematical development of the adjoint method is performed for $v^F = 0$. During the mathematical development, arrows ($\rightarrow$) underneath the underbraced terms indicate where these terms contribute to. For instance terms including $\frac{\delta U_m}{\delta b_i}$, $\frac{\delta \tilde{\mu}}{\delta b_i}$, $\frac{\delta(\rho\gamma)}{\delta b_i}$ and $\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}$ are denoted as $\rightarrow$ FAE_MF, $\rightarrow$ FAE_SA, $\rightarrow$ FAE_γ and $\rightarrow$ FAE_$\tilde{Re}_{\theta t}$ and contribute to the FAE of the mean flow, the Spalart–Allmaras, the γ and $\tilde{Re}_{\theta t}$, respectively. Terms denoted as $\rightarrow$ *Suth* and $\rightarrow$ FAE_Δ include $\frac{\delta \mu}{\delta b_i}$ and $\frac{\delta \Delta}{\delta b_i}$. Regarding $\frac{\delta \mu}{\delta b_i}$, the Sutherland formula is differen-

tiated and, then, contributes to the mean flow equations. If μ is constant, this term vanishes. Finally, $\rightarrow\mathcal{S}$ and $\rightarrow$SD stand for the surface (boundary) integrals and the sensitivity derivatives.

3.1 Differentiation of the Mean Flow Equations

In Eq. 3.8, term $\mathcal{I}^{\text{MF}}$, corresponding to the mean flow equations, is expanded as

$$\mathcal{I}^{\text{MF}} = \underbrace{\int_\Omega \Psi_n \frac{\delta}{\delta b_i}\left(\frac{\partial f_{nk}^{inv}}{\partial x_k}\right)\mathrm{d}\Omega}_{\mathcal{I}^{\text{MF\_inv}}} \underbrace{-\int_\Omega \Psi_n \frac{\delta}{\delta b_i}\left(\frac{\partial f_{nk}^{vis}}{\partial x_k}\right)\mathrm{d}\Omega}_{\mathcal{I}^{\text{MF\_vis}}} \tag{3.9}$$

Through Eq. 3.7, the inviscid term $\mathcal{I}^{\text{MF\_inv}}$ becomes

$$\mathcal{I}^{\text{MF\_inv}} = \int_\Omega \Psi_n \frac{\partial}{\partial x_k}\left(\frac{\delta f_{nk}^{inv}}{\delta b_i}\right)\mathrm{d}\Omega \underbrace{-\int_\Omega \Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}} \tag{3.10}$$

The last integral on the r.h.s. of Eq. 3.10 contributes to the SD expression; the first one is further expanded as

$$\int_\Omega \Psi_n \frac{\partial}{\partial x_k}\left(\frac{\delta f_{nk}^{inv}}{\delta b_i}\right)\mathrm{d}\Omega = \underbrace{\int_{\partial\Omega} \Psi_n \mathsf{n}_k \frac{\delta f_{nk}^{inv}}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_1} \underbrace{-\int_\Omega A_{nmk}\frac{\partial \Psi_n}{\partial x_k}\frac{\delta U_m}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}} \tag{3.11}$$

The viscous integral $\mathcal{I}^{\text{MF\_vis}}$ becomes

$$\mathcal{I}^{\text{MF\_vis}} = \underbrace{-\int_{\partial\Omega} \Psi_n \mathsf{n}_k \frac{\delta f_{nk}^{vis}}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_2} \underbrace{+\int_\Omega \frac{\partial \Psi_n}{\partial x_k}\frac{\delta f_{nk}^{vis}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\text{MF\_vis,1}}} \underbrace{+\int_\Omega \Psi_n \frac{\partial f_{nk}^{vis}}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}} \tag{3.12}$$

To further develop $\mathcal{I}^{\text{MF\_vis,1}}$, two new quantities are defined, the adjoint stress tensor τ_{mk}^{adj} and the adjoint heat flux q_k^{adj}, as (in 3D flows)

$$\begin{aligned}
\tau_{mk}^{\text{adj}} &= \frac{\mu+\mu_t}{\text{Re}_0}\left[\frac{\partial\Psi_{m+1}}{\partial x_k}+\frac{\partial\Psi_{k+1}}{\partial x_m}+\frac{\partial\Psi_5}{\partial x_m}v_k^A+\frac{\partial\Psi_5}{\partial x_k}v_m^A-\frac{2}{3}\delta_{mk}\left(\frac{\partial\Psi_{\ell+1}}{\partial x_\ell}+\frac{\partial\Psi_5}{\partial x_\ell}v_\ell^A\right)\right]\\
q_k^{\text{adj}} &= \frac{\mathcal{C}_p}{\text{Re}_0}\left(\frac{\mu}{\text{Pr}}+\frac{\mu_t}{\text{Pr}_\text{t}}\right)\frac{\partial\Psi_5}{\partial x_k}
\end{aligned} \tag{3.13}$$

and $\mathcal{I}^{\text{MF\_vis},1}$ becomes

$$\begin{aligned}
\mathcal{I}^{\text{MF_vis},1} = &\underbrace{-\int_\Omega\left(\frac{\partial\tau_{mk}^{\text{adj}}}{\partial x_k}-\frac{\partial\Psi_5}{\partial x_k}\tau_{km}\right)\frac{\partial v_m^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega-\int_\Omega\frac{\partial q_k^{\text{adj}}}{\partial x_k}\frac{\partial T}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow\text{FAE_MF}}\\
&\underbrace{+\int_{\partial\Omega}\tau_{mk}^{\text{adj}}\mathsf{n}_k\frac{\delta v_m^A}{\delta b_i}\text{d}S}_{\mathcal{S}_3}+\underbrace{\int_{\partial\Omega}q_k^{\text{adj}}\mathsf{n}_k\frac{\delta T}{\delta b_i}\text{d}S}_{\mathcal{S}_4}\underbrace{-\int_\Omega\left(\tau_{mk}^{\text{adj}}\frac{\partial v_m^A}{\partial x_\ell}+q_k^{\text{adj}}\frac{\partial T}{\partial x_\ell}\right)\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\text{d}\Omega}_{\rightarrow\text{SD}}\\
&\underbrace{+\int_\Omega\frac{\partial\Psi_{m+1}}{\partial x_k}\frac{\tau_{km}}{\mu+\mu_t}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\text{d}\Omega}_{\rightarrow\text{FAE_SA}}\\
&\underbrace{+\int_\Omega\frac{\partial\Psi_5}{\partial x_k}v_m^A\frac{\tau_{km}}{\mu+\mu_t}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\text{d}\Omega}_{\rightarrow\text{FAE_SA}}\\
&\underbrace{+\int_\Omega\frac{\mathcal{C}_p}{\text{Re}_0}\frac{\partial\Psi_5}{\partial x_k}\frac{\partial T}{\partial x_k}\frac{1}{\text{Pr}_\text{t}}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\text{d}\Omega}_{\rightarrow\text{FAE_SA}}\\
&\underbrace{+\int_\Omega\frac{\partial\Psi_{m+1}}{\partial x_k}\frac{\tau_{km}}{\mu+\mu_t}\left[1+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow\textit{Suth}}\\
&\underbrace{+\int_\Omega\frac{\partial\Psi_5}{\partial x_k}v_m^A\frac{\tau_{km}}{\mu+\mu_t}\left[1+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow\textit{Suth}}\\
&\underbrace{+\int_\Omega\frac{\mathcal{C}_p}{\text{Re}_0}\frac{\partial\Psi_5}{\partial x_k}\frac{\partial T}{\partial x_k}\left[\frac{1}{\text{Pr}}+\frac{1}{\text{Pr}_\text{t}}\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow\textit{Suth}}
\end{aligned} \tag{3.14}$$

3.2 Differentiation of the Spalart–Allmaras Equation

The term $\mathcal{I}^{\text{SA}}$ in Eq. 3.8 is decomposed in three terms arising from the differentiation of the convection, diffusion and source terms of the Spalart–Allmaras PDE

$$\mathcal{I}^{\text{SA}} = \underbrace{\int_\Omega \tilde{\nu_a} \frac{\delta \text{SA}^c}{\delta b_i} \text{d}\Omega}_{\mathcal{I}^{\text{SA\_conv}}} + \underbrace{\int_\Omega \tilde{\nu_a} \frac{\delta \text{SA}^d}{\delta b_i} \text{d}\Omega}_{\mathcal{I}^{\text{SA\_diff}}} + \underbrace{\int_\Omega \tilde{\nu_a} \frac{\delta \text{SA}^s}{\delta b_i} \text{d}\Omega}_{\mathcal{I}^{\text{SA\_src}}}$$

The convection term ($\mathcal{I}^{\text{SA\_conv}}$) can be developed as

$$\begin{aligned}\mathcal{I}^{\text{SA\_conv}} = &\underbrace{-\int_\Omega \frac{\partial \tilde{\nu_a}}{\partial x_k} \tilde{\mu} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \text{d}\Omega}_{\rightarrow \text{FAE\_MF}} \underbrace{-\int_\Omega \frac{\partial \tilde{\nu_a}}{\partial x_k} v_k^A \frac{\delta \tilde{\mu}}{\delta b_i} \text{d}\Omega}_{\rightarrow \text{FAE\_SA}} \underbrace{+\int_{\partial\Omega} \tilde{\nu_a} \mathsf{n}_k \frac{\delta\left(\tilde{\mu} v_k^A\right)}{\delta b_i} \text{d}S}_{\mathcal{S}_5} \\ &\underbrace{-\int_\Omega \tilde{\nu_a} \frac{\partial\left(\tilde{\mu} v_k^A\right)}{\partial x_\ell} \frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right) \text{d}\Omega}_{\rightarrow \text{SD}}\end{aligned} \tag{3.15}$$

The diffusion term is treated similarly and yields

$$\begin{aligned}\mathcal{I}^{\text{SA\_diff}} = &\underbrace{-\frac{1}{\text{Re}_0\,\sigma}\int_\Omega \tilde{\nu_a}\left\{\frac{\partial}{\partial x_k}\left[\left(\nu+\tilde{\nu}\right)\frac{\partial \tilde{\nu}}{\partial x_k}\right] + c_{b_2}\frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\right\}\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{-\frac{1}{\text{Re}_0\,\sigma}\int_\Omega \rho\tilde{\nu_a}\frac{\delta}{\delta b_i}\left(\frac{\partial}{\partial x_k}\left[\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial \tilde{\nu}}{\partial x_k}\right]\right)\text{d}\Omega}_{\mathcal{I}^{\text{SA\_diff1}}} \underbrace{+\frac{c_{b_2}}{\text{Re}_0\,\sigma}\int_\Omega \rho\tilde{\nu_a}\frac{\delta}{\delta b_i}\left(\tilde{\nu}\frac{\partial^2 \tilde{\nu}}{\partial x_k^2}\right)\text{d}\Omega}_{\mathcal{I}^{\text{SA\_diff2}}}\end{aligned} \tag{3.16}$$

where

$$\begin{aligned}\mathcal{I}^{\text{SA\_diff1}} = &\underbrace{\frac{1}{\text{Re}_0\,\sigma}\int_\Omega \frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \nu}{\partial \rho}\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow \text{FAE\_MF}} + \underbrace{\frac{1}{\text{Re}_0\,\sigma}\int_\Omega \frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \nu}{\partial \mu}\frac{\partial \mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow \textit{Suth}} \\ &\underbrace{-\frac{1}{\text{Re}_0\,\sigma}\int_\Omega \frac{\tilde{\nu}}{\rho}\left[\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\left(1+c_{b_2}\right) - \frac{\partial}{\partial x_k}\left[\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial(\rho\tilde{\nu_a})}{\partial x_k}\right]\right]\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\text{d}\Omega}_{\rightarrow \text{FAE\_MF}}\end{aligned}$$

$$
\begin{aligned}
&\underbrace{+\frac{1}{\mathrm{Re}_0\,\sigma}\int_\Omega \frac{1}{\rho}\left[\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\left(1+c_{b_2}\right)-\frac{\partial}{\partial x_k}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial(\rho\tilde{\nu_a})}{\partial x_k}\right]\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_SA}}\\
&\underbrace{+\frac{1}{\mathrm{Re}_0\,\sigma}\int_\Omega \rho\tilde{\nu_a}\frac{\partial}{\partial x_\ell}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_k}\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\int_\Omega \frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\int_{\partial\Omega}\rho\tilde{\nu_a}\mathsf{n}_k\frac{\delta}{\delta b_i}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_k}\right]\mathrm{d}S}_{\mathcal{S}_6}\\
&\underbrace{+\frac{1}{\mathrm{Re}_0\,\sigma}\int_{\partial\Omega}\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\mathsf{n}_k\frac{\delta\tilde{\nu}}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_7}
\end{aligned}
\tag{3.17}
$$

$$
\begin{aligned}
\mathcal{I}^{\text{SA\_diff2}} =&\underbrace{-\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int_\Omega \frac{\tilde{\nu}}{\rho}\left[\rho\tilde{\nu_a}\frac{\partial^2\tilde{\nu}}{\partial x_k^2}+\frac{\partial}{\partial x_k}\left(\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\right)\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int_\Omega \frac{1}{\rho}\left[\rho\tilde{\nu_a}\frac{\partial^2\tilde{\nu}}{\partial x_k^2}+\frac{\partial}{\partial x_k}\left(\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_SA}}\\
&\underbrace{-\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int_\Omega \tilde{\nu_a}\tilde{\mu}\frac{\partial}{\partial x_\ell}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\right)\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega+\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int_\Omega \frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}}\\
&\underbrace{+\frac{c_{b_2}}{\sigma}\int_{\partial\Omega}\tilde{\nu_a}\tilde{\mu}\mathsf{n}_k\frac{\delta}{\delta b_i}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\right)\mathrm{d}S}_{\mathcal{S}_8}\underbrace{-\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int_{\partial\Omega}\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\mathsf{n}_k\frac{\delta\tilde{\nu}}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_9}
\end{aligned}
\tag{3.18}
$$

The expression of the Spalart–Allmaras source terms depends on whether the $\gamma-\tilde{R}e_{\theta t}$ transition model is present and on which variant of it is used. In order to avoid four different mathematical developments (one when no transition is present and three for the transition models discussed in this PhD Thesis), the differentiation of the source terms will be preformed once and special treatment will be performed for the terms that differ among the four alternatives. For this

reason, γ_{eff} will be introduced even if there are no transitional effects (in this case $\gamma_{eff}=1$). When the SA-noft2-Gamma-Retheta model is used, γ_{eff} is expressed by Eq. 2.25, whereas for the Piotrowski and Zingg models (SA-LM2015, SA-sLM2015) $\gamma_{eff}=\gamma$.

The operator $\mathcal{P}(\alpha, c)$ is introduced which denotes the partial derivative of the function α w.r.t. the quantity c, by assuming that any other quantity is constant. For the differentiation of the source terms, the following definitions are used

$$
\begin{aligned}
\mathcal{C}_{\chi} &= \mathcal{P}\left(\tilde{S}, f_{v_2}\right)\left[\mathcal{P}\left(f_{v_2}, \chi\right)+\mathcal{P}\left(f_{v_2}, f_{v_1}\right)\mathcal{P}\left(f_{v_1}, \chi\right)\right] \\
\mathcal{C}_{r} &= \frac{\rho}{\mathrm{Re}_0} c_{w_1}\left(\frac{\tilde{\nu}}{\Delta}\right)^2 \mathcal{P}\left(f_w, g\right)\mathcal{P}\left(g, r\right) \\
\mathcal{C}_{\tilde{S}} &= -\gamma_{eff} c_{b_1}\left(1-f_{t_2}\right)\tilde{\mu}+\mathcal{C}_r\mathcal{P}\left(r, \tilde{S}\right) \\
\mathcal{C}_{f_{t_2}} &= c_{b_1}\rho\left[\gamma_{eff}\tilde{S}\tilde{\nu}-\frac{1}{\mathrm{Re}_0\,\kappa^2}\left(\frac{\tilde{\nu}}{\Delta}\right)^2\right] \\
\mathcal{C}_{\gamma_{eff}} &= -c_{b_1}\tilde{\mu}\left(1-f_{t_2}\right)\tilde{S}, \quad \mathcal{C}_{\gamma_{sep}} = \mathcal{C}_{\gamma_{eff}}\mathcal{P}\left(\gamma_{eff}, \gamma_{sep}\right) \\
\mathcal{C}^{\mathrm{SA}}_{\zeta} &= \mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S}, \zeta\right)+\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep}, F_{\theta t}\right)\mathcal{P}\left(F_{\theta t}, \zeta\right) \\
\mathcal{C}^{\mathrm{SA}}_{S} &= \mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S}, S\right)+\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep}, Re_{\nu}\right)\mathcal{P}\left(Re_{\nu}, S\right)
\end{aligned}
$$

Then, the differentiation of the source terms leads to

$$
\begin{aligned}
\mathcal{I}^{\mathrm{SA\_src}} = & \underbrace{\int_{\Omega}\frac{\tilde{\nu_a}}{\mathrm{Re}_0}\left[-\left(c_{w_1}f_w-\frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\left(\frac{\tilde{\nu}}{\Delta}\right)^2\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathrm{FAE\_MF}} \\
& \underbrace{+\int_{\Omega}\tilde{\nu_a}\left[-\gamma_{eff}c_{b_1}\left(1-f_{t_2}\right)\tilde{S}+\frac{2}{\mathrm{Re}_0}\left(c_{w_1}f_w-\frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\frac{\tilde{\nu}}{\Delta^2}\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathrm{FAE\_SA}} \\
& \underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_r\mathcal{P}\left(r, \tilde{\mu}\right)\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathrm{FAE\_SA}}+\underbrace{\int_{\Omega}\tilde{\nu_a}\mathcal{C}_r\mathcal{P}\left(r, \rho\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathrm{FAE\_MF}}+\underbrace{\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\tilde{S}}\frac{\delta\tilde{S}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\tilde{S}}}+\underbrace{\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{f_{t_2}}\frac{\delta f_{t_2}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{f_{t_2}}} \\
& \underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{eff}}\frac{\delta\gamma_{eff}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\gamma_{eff}}}+\underbrace{\int_{\Omega}\tilde{\nu_a}\left[\mathcal{C}_r\mathcal{P}\left(r, \Delta\right)-2\frac{\rho}{\mathrm{Re}_0}\left(c_{w_1}f_w-\frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\frac{\tilde{\nu}^2}{\Delta^3}\right]\frac{\delta\Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathrm{FAE\_}\Delta}
\end{aligned}
\tag{3.19}
$$

where $\mathcal{I}^{\tilde{S}}$ is expanded as

$$\begin{aligned}\mathcal{I}^{\tilde{S}} = &\underbrace{\int_\Omega \tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}} + \underbrace{\int_\Omega \frac{\tilde{\nu_a}}{\rho}\mathcal{C}_{\tilde{S}}\left[\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)+\mathcal{C}_\chi\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_SA}} \\ &\underbrace{+\int_\Omega \tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\zeta\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^\zeta} + \underbrace{\int_\Omega \tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},S\right)\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^S} \\ &\underbrace{+\int_\Omega \tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\Delta\right)\frac{\delta\Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\Delta} + \underbrace{\int_\Omega \tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{C}_\chi\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\textit{Suth}}\end{aligned} \tag{3.20}$$

Integrals $\mathcal{I}^{f_{t_2}}$ and $\mathcal{I}^{\gamma_{eff}}$ are developed separately for the Spalart–Allmaras model without transition model (subscript SA) and for the one coupled with the $\gamma-\tilde{R}e_{\theta t}$ transition model (subscript SA-noft2,SA-LM,SA-sLM).

For the Spalart–Allmaras model without transition

$$\begin{aligned}\mathcal{I}^{f_{t_2}}_{\text{SA}} &= \underbrace{\int_\Omega \tilde{\nu_a}\mathcal{C}_{f_{t_2}}\mathcal{P}\left(f_{t_2},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_SA}} + \underbrace{\int_\Omega \tilde{\nu_a}\mathcal{C}_{f_{t_2}}\mathcal{P}\left(f_{t_2},\chi\right)\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\textit{Suth}} \\ \mathcal{I}^{\gamma_{eff}}_{\text{SA}} &= 0\end{aligned} \tag{3.21}$$

When the SA-noft2-Gamma-Retheta transition model is used, then

$$\begin{aligned}\mathcal{I}^{f_{t_2}}_{\text{SA-noft2}} &= 0 \\ \mathcal{I}^{\gamma_{eff}}_{\text{SA-noft2}} = &\underbrace{-\int_\Omega \tilde{\nu_a}\frac{\gamma}{\rho}\left[\mathcal{C}_{\gamma_{eff}}\mathcal{P}\left(\gamma_{eff},\gamma\right)+\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\gamma\right)\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}} \\ &\underbrace{-\int_\Omega \tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\frac{\tilde{R}e_{\theta t}}{\rho}\left[\mathcal{P}\left(\gamma_{sep},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{R}e_{\theta t}\right)+\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\tilde{R}e_{\theta t}\right)\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}} \\ &\underbrace{+\int_\Omega \tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\left[\mathcal{P}\left(\gamma_{sep},Re_\nu\right)\mathcal{P}\left(Re_\nu,\rho\right)+\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\rho\right)\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}\end{aligned}$$

$$
\begin{aligned}
&\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},|\mathbf{U}|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{reattach}\right)\mathcal{P}\left(F_{reattach},\mu_t\right)\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_SA}}\\
&\underbrace{+\int_{\Omega}\frac{\tilde{\nu_a}}{\rho}\left[\mathcal{C}_{\gamma_{eff}}\mathcal{P}\left(\gamma_{eff},\gamma\right)+\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\gamma\right)\right]\frac{\delta\left(\rho\gamma\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\gamma}\\
&\underbrace{+\int_{\Omega}\frac{\tilde{\nu_a}}{\rho}\mathcal{C}_{\gamma_{sep}}\left[\mathcal{P}\left(\gamma_{sep},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{Re}_{\theta t}\right)+\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\tilde{Re}_{\theta t}\right)\right]\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\tilde{Re}_{\theta t}}\\
&\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},Re_\nu\right)\mathcal{P}\left(Re_\nu,\Delta\right)\frac{\delta\Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\Delta}\\
&\underbrace{\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\zeta\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^\zeta}+\underbrace{\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},Re_\nu\right)\mathcal{P}\left(Re_\nu,S\right)\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^S}}\\
&\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{reattach}\right)\mathcal{P}\left(F_{reattach},\mu_t\right)\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}\\
&\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\left[\mathcal{P}\left(\gamma_{sep},Re_\nu\right)\mathcal{P}\left(Re_\nu,\mu\right)+\mathcal{P}\left(\gamma_{sep},F_{reattach}\right)\mathcal{P}\left(F_{reattach},\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}\\
&\underbrace{+\int_{\Omega}\tilde{\nu_a}\mathcal{C}_{\gamma_{sep}}\mathcal{P}\left(\gamma_{sep},F_{\theta t}\right)\mathcal{P}\left(F_{\theta t},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned}
\tag{3.22}
$$

When the SA-LM2015 or SA-sLM2015 transition model is used

$$\mathcal{I}^{f_{t_2}}_{\text{SA-LM}}=\mathcal{I}^{f_{t_2}}_{\text{SA-sLM}}=0$$

$$\mathcal{I}^{\gamma_{eff}}_{\text{SA-LM}} = \mathcal{I}^{\gamma_{eff}}_{\text{SA-sLM}} = \underbrace{-\int_{\Omega}\tilde{\nu}_a \frac{\gamma}{\rho}\mathcal{C}_{\gamma_{eff}}\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}} + \underbrace{\int_{\Omega}\tilde{\nu}_a \frac{1}{\rho}\mathcal{C}_{\gamma_{eff}}\frac{\delta\left(\rho\gamma\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\gamma} \tag{3.23}$$

Finally, terms $\mathcal{I}^{\zeta}$ and $\mathcal{I}^{S}$ are analyzed as

$$\begin{aligned}
\mathcal{I}^{\zeta} =& \underbrace{\int_{\partial\Omega}\tilde{\nu}_a \frac{\mathcal{C}^{\text{SA}}_{\zeta}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v^A_r}{\partial x_q}\mathsf{n}_\ell\frac{\delta v^A_m}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_{10}} \underbrace{-\int_{\Omega}\frac{\partial}{\partial x_\ell}\left[\tilde{\nu}_a \frac{\mathcal{C}^{\text{SA}}_{\zeta}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v^A_r}{\partial x_q}\right]\frac{\partial v^A_m}{\partial U_p}\frac{\delta U_p}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\tilde{\nu}_a \frac{\mathcal{C}^{\text{SA}}_{\zeta}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v^A_r}{\partial x_q}\frac{\partial v^A_m}{\partial x_p}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_p}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}} \\
\mathcal{I}^{S} =& \underbrace{\int_{\partial\Omega}\tilde{\nu}_a \frac{\mathcal{C}^{SA}_{S}}{S}2S_{km}\mathsf{n}_m\frac{\delta v^A_k}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_{11}} \underbrace{-\int_{\Omega}\frac{\partial}{\partial x_m}\left[\tilde{\nu}_a \frac{\mathcal{C}^{\text{SA}}_{S}}{S}2S_{km}\right]\frac{\partial v^A_k}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\tilde{\nu}_a \frac{\mathcal{C}^{\text{SA}}_{S}}{S}2S_{km}\frac{\partial v^A_k}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}}
\end{aligned} \tag{3.24}$$

3.3 Differentiation of the $\gamma-\tilde{Re}_{\theta t}$ Transition Model Equations

The terms $\mathcal{I}^{\gamma}, \mathcal{I}^{\tilde{Re}_{\theta t}}$ in Eq. 3.8 split in three terms arising from the differentiation of the convection, diffusion and source terms of the γ and $\tilde{Re}_{\theta t}$ PDEs.

$$\begin{aligned}
\mathcal{I}^{\gamma} &= \underbrace{\int_{\Omega}\gamma_a\frac{\delta\gamma^c}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\gamma_conv}} + \underbrace{\int_{\Omega}\gamma_a\frac{\delta\gamma^d}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\gamma_diff}} + \underbrace{\int_{\Omega}\gamma_a\frac{\delta\gamma^s}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\gamma_src}} \\
\mathcal{I}^{\tilde{Re}_{\theta t}} &= \underbrace{\int_{\Omega}\tilde{Re}_a\frac{\delta\tilde{Re}^c_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\tilde{Re}_{\theta t}_conv}} + \underbrace{\int_{\Omega}\tilde{Re}_a\frac{\delta\tilde{Re}^d_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\tilde{Re}_{\theta t}_diff}} + \underbrace{\int_{\Omega}\tilde{Re}_a\frac{\delta\tilde{Re}^s_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\tilde{Re}_{\theta t}_src}}
\end{aligned}$$

The convection terms $\left(\mathcal{I}^{\gamma\_conv}, \mathcal{I}^{\tilde{Re}_{\theta t}\_conv}\right)$ can be developed as

$$\begin{aligned}\mathcal{I}^{\gamma\_conv} = &\underbrace{\int_{\partial\Omega} \gamma_a \mathsf{n}_k \frac{\delta\left(\rho v_k^A \gamma\right)}{\delta b_i} \mathrm{d}S}_{\mathcal{S}_{12}} \underbrace{- \int_{\Omega} \frac{\partial \gamma_a}{\partial x_k} \rho\gamma \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \underbrace{- \int_{\Omega} \frac{\partial \gamma_a}{\partial x_k} v_k^A \frac{\delta\left(\rho\gamma\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma} \\ &\underbrace{- \int_{\Omega} \gamma_a \frac{\partial\left(\rho v_k^A \gamma\right)}{\partial x_\ell} \frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right) \mathrm{d}\Omega}_{\rightarrow \text{SD}}\end{aligned} \tag{3.25}$$

$$\begin{aligned}\mathcal{I}^{\tilde{Re}_{\theta t}\_conv} = &\underbrace{\int_{\partial\Omega} \tilde{Re}_a \mathsf{n}_k \frac{\delta\left(\rho v_k^A \tilde{Re}_{\theta t}\right)}{\delta b_i} \mathrm{d}S}_{\mathcal{S}_{13}} \underbrace{- \int_{\Omega} \frac{\partial \tilde{Re}_a}{\partial x_k} \rho \tilde{Re}_{\theta t} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{- \int_{\Omega} \frac{\partial \tilde{Re}_a}{\partial x_k} v_k^A \frac{\delta\left(\rho \tilde{Re}_{\theta t}\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{Re}_{\theta t}} \underbrace{- \int_{\Omega} \tilde{Re}_a \frac{\partial\left(\rho v_k^A \tilde{Re}_{\theta t}\right)}{\partial x_\ell} \frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right) \mathrm{d}\Omega}_{\rightarrow \text{SD}}\end{aligned} \tag{3.26}$$

The diffusion terms $\left(\mathcal{I}^{\gamma\_diff}, \mathcal{I}^{\tilde{Re}_{\theta t}\_diff}\right)$ are treated similarly

$$\begin{aligned}\mathcal{I}^{\gamma\_diff} = &\underbrace{-\frac{1}{\mathrm{Re}_0} \int_{\partial\Omega} \gamma_a \mathsf{n}_k \frac{\delta}{\delta b_i}\left[\left(\mu + \frac{\mu_t}{\sigma_f}\right) \frac{\partial \gamma}{\partial x_k}\right] \mathrm{d}S}_{\mathcal{S}_{14}} \underbrace{+ \frac{1}{\mathrm{Re}_0} \int_{\partial\Omega} \left(\mu + \frac{\mu_t}{\sigma_f}\right) \frac{\partial \gamma_a}{\partial x_k} \mathsf{n}_k \frac{\delta \gamma}{\delta b_i} \mathrm{d}S}_{\mathcal{S}_{15}} \\ &\underbrace{+ \frac{1}{\mathrm{Re}_0} \int_{\Omega} \frac{\gamma}{\rho} \frac{\partial}{\partial x_k}\left[\left(\mu + \frac{\mu_t}{\sigma_f}\right) \frac{\partial \gamma_a}{\partial x_k}\right] \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \underbrace{- \frac{1}{\mathrm{Re}_0} \int_{\Omega} \frac{1}{\rho} \frac{\partial}{\partial x_k}\left[\left(\mu + \frac{\mu_t}{\sigma_f}\right) \frac{\partial \gamma_a}{\partial x_k}\right] \frac{\delta\left(\rho\gamma\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma} \\ &\underbrace{+ \frac{1}{\mathrm{Re}_0} \int_{\Omega} \frac{\partial \gamma_a}{\partial x_k} \frac{\partial \gamma}{\partial x_k} \frac{1}{\sigma_f} \left[\mathcal{P}\left(\mu_t, \tilde{\mu}\right) + \mathcal{P}\left(\mu_t, f_{v_1}\right) \mathcal{P}\left(f_{v_1}, \chi\right) \mathcal{P}\left(\chi, \tilde{\mu}\right)\right] \frac{\delta \tilde{\mu}}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_SA}} \\ &\underbrace{+ \frac{1}{\mathrm{Re}_0} \int_{\Omega} \frac{\partial \gamma_a}{\partial x_k} \frac{\partial \gamma}{\partial x_k} \left[1 + \frac{1}{\sigma_f} \mathcal{P}\left(\mu_t, f_{v_1}\right) \mathcal{P}\left(f_{v_1}, \chi\right) \mathcal{P}\left(\chi, \mu\right)\right] \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \textit{Suth}}\end{aligned}$$

$$\underbrace{+\frac{1}{\mathrm{Re}_0}\int_\Omega\left[\gamma_a\frac{\partial}{\partial x_\ell}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\right]-\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\frac{\partial\gamma}{\partial x_\ell}\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\mathrm{SD}} \tag{3.27}$$

$$\begin{aligned}
\mathcal{I}^{\tilde{Re}_{\theta t}_diff}= &\underbrace{-\frac{1}{\mathrm{Re}_0}\int_{\partial\Omega}\tilde{Re}_a\mathsf{n}_k\frac{\delta}{\delta b_i}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\right]\mathrm{d}S}_{\mathcal{S}_{16}}\underbrace{+\int_{\partial\Omega}\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\mathsf{n}_k\frac{\delta\tilde{Re}_{\theta t}}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_{17}}\\
&\underbrace{+\frac{1}{\mathrm{Re}_0}\int_\Omega\frac{\tilde{Re}_{\theta t}}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\mathrm{FAE_MF}}\\
&\underbrace{+\frac{1}{\mathrm{Re}_0}\int_\Omega\frac{\partial\tilde{Re}_a}{\partial x_k}\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\sigma_{\theta,t}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\mathrm{FAE_SA}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0}\int_\Omega\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\right]\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\mathrm{FAE_}\tilde{Re}_{\theta t}}\\
&\underbrace{+\frac{1}{\mathrm{Re}_0}\int_\Omega\frac{\partial\tilde{Re}_a}{\partial x_k}\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\left[1+\sigma_{\theta,t}\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}\\
&\underbrace{+\frac{1}{\mathrm{Re}_0}\int_\Omega\tilde{Re}_a\frac{\partial}{\partial x_\ell}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\mathrm{SD}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0}\int_\Omega\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\frac{\partial\tilde{Re}_{\theta t}}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\mathrm{SD}}
\end{aligned} \tag{3.28}$$

The differentiation of the source terms is presented separately for the three variants, the SA-noft2-Gamma-Retheta, the SA-LM2015 and the SA-sLM2015 models.

3.3.1 Differentiation of the SA-noft2-Gamma-Retheta Transition Model Source Terms

For the SA-noft2-Gamma-Retheta model, the following definitions are used for the differentiation of the source terms $\left(\mathcal{I}^{\gamma\_src}, \mathcal{I}^{\tilde{Re}_{\theta t}\_src}\right)$

$$
\begin{aligned}
\mathcal{C}_{F_{onset}} &= -\frac{1}{2}\rho c_{\alpha_1} S\sqrt{\gamma F_{onset}}\left(1-c_{\epsilon_1}\gamma\right)\frac{F_{length}}{F_{onset}} \\
\mathcal{C}_{F_{onset3}} &= \mathcal{C}_{F_{onset}}\mathcal{P}\left(F_{onset}, F_{onset3}\right) \\
\mathcal{C}_{F_{onset1}} &= \mathcal{C}_{F_{onset}}\mathcal{P}\left(F_{onset}, F_{onset2}\right)\mathcal{P}\left(F_{onset2}, F_{onset1}\right) \\
\mathcal{C}_{F_{\theta t}} &= \rho\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right) \\
\mathcal{C}_{Re^{eq}_{\theta,t}} &= -\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\mathcal{P}\left(Re^{eq}_{\theta,t}, F\right)\mathcal{P}\left(F,\lambda_\theta\right)}{1-\mathcal{P}\left(Re^{eq}_{\theta,t}, F\right)\mathcal{P}\left(F,\lambda_\theta\right)\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta, Re^{eq}_{\theta,t}\right)} \\
\mathcal{C}_{d|\mathbf{U}|ds} &= \mathcal{C}_{Re^{eq}_{\theta,t}}\mathcal{P}\left(\lambda_\theta, \frac{d\left|\mathbf{U}\right|}{ds}\right) \\
\mathcal{C}^{\gamma-\tilde{Re}_{\theta t}}_{\zeta} &= \gamma_a\rho c_{\alpha_2}\gamma F_{turb}\left(c_{\epsilon_2}\gamma-1\right)+\tilde{Re}_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},\zeta\right) \\
\mathcal{C}^{\gamma-\tilde{Re}_{\theta t}}_{S} &= -\gamma_a\rho c_{\alpha_1}\sqrt{\gamma F_{onset}}\left(1-c_{\epsilon_1}\gamma\right)F_{length}+\gamma_a\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1}, Re_\nu\right)\mathcal{P}\left(Re_\nu, S\right) \\
\mathcal{C}^{\gamma-\tilde{Re}_{\theta t}}_{\mu_t} &= \gamma_a\rho c_{\alpha_2}\zeta\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu_t\right)+\gamma_a\mathcal{C}_{F_{onset3}}\mathcal{P}\left(F_{onset3},\mu_t\right)
\end{aligned}
$$

For equation of γ, $\mathcal{I}^{\gamma\_src} = \mathcal{I}^{\gamma\_prod} + \mathcal{I}^{\gamma\_destr}$

$$
\begin{aligned}
\mathcal{I}^{\gamma\_prod} = & \underbrace{-\int_\Omega \gamma_a c_{\alpha_1} S\sqrt{\gamma F_{onset}}\left(1-c_{\epsilon_1}\gamma\right)F_{length}\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{+\frac{1}{2}\int_\Omega \gamma\gamma_a c_{\alpha_1} S\sqrt{\gamma F_{onset}}F_{length}\left({}^1\!/\!_\gamma-3c_{\epsilon_1}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{+\int_\Omega \tilde{Re}_{\theta t}\gamma_a c_{\alpha_1} S\sqrt{\gamma F_{onset}}\left(1-c_{\epsilon_1}\gamma\right)\mathcal{P}\left(F_{length}, \tilde{Re}_{\theta t}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{-\frac{1}{2}\int_\Omega \gamma_a c_{\alpha_1} S\sqrt{\gamma F_{onset}}F_{length}\left({}^1\!/\!_\gamma-3c_{\epsilon_1}\right)\frac{\delta\left(\rho\gamma\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma}
\end{aligned}
$$

$$\underbrace{-\int_{\Omega}\gamma_a c_{\alpha_1} S\sqrt{\gamma F_{onset}}\left(1-c_{\epsilon_1}\gamma\right)\mathcal{P}\left(F_{length},\tilde{Re}_{\theta t}\right)\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{Re}_{\theta t}}$$
$$\underbrace{-\int_{\Omega}\gamma_a\rho c_{\alpha_1}\sqrt{\gamma F_{onset}}\left(1-c_{\epsilon_1}\gamma\right)F_{length}\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{S}}+\underbrace{\int_{\Omega}\gamma_a\mathcal{C}_{F_{onset}}\frac{\delta F_{onset}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{onset}}} \tag{3.29}$$

$$\mathcal{I}^{\gamma\_destr}=\underbrace{\int_{\Omega}\gamma_a c_{\alpha_2}\zeta\gamma F_{turb}\left(c_{\epsilon_2}\gamma-1\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega-\int_{\Omega}\gamma\gamma_a c_{\alpha_2}\zeta F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_{\Omega}\gamma_a c_{\alpha_2}\zeta F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\frac{\delta\left(\rho\gamma\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma}+\underbrace{\int_{\Omega}\gamma_a\rho c_{\alpha_2}\zeta\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu_t\right)\frac{\delta\mu_t}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathcal{I}^{\mu_t}}$$
$$\underbrace{+\int_{\Omega}\gamma_a\rho c_{\alpha_2}\gamma F_{turb}\left(c_{\epsilon_2}\gamma-1\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\zeta}}+\underbrace{\int_{\Omega}\gamma_a\rho c_{\alpha_2}\zeta\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \mathit{Suth}} \tag{3.30}$$

Regarding $\mathcal{I}^{F_{onset}}$, this takes the form

$$\mathcal{I}^{F_{onset}}=\underbrace{\int_{\Omega}\gamma_a\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_\nu\right)\mathcal{P}\left(Re_\nu,\rho\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{-\int_{\Omega}\gamma_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{Re}_{\theta t}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_{\Omega}\gamma_a\frac{1}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{Re}_{\theta t}\right)\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{Re}_{\theta t}}$$

$$\underbrace{+\int_{\Omega}\gamma_a\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\nu}\right)\mathcal{P}\left(Re_{\nu},S\right)\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{S}}\underbrace{+\int_{\Omega}\gamma_a\mathcal{C}_{F_{onset3}}\mathcal{P}\left(F_{onset3},\mu_t\right)\frac{\delta \mu_t}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\mu_t}}$$

$$\underbrace{+\int_{\Omega}\gamma_a\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\nu}\right)\mathcal{P}\left(Re_{\nu},\Delta\right)\frac{\delta \Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\Delta}$$

$$\underbrace{+\int_{\Omega}\gamma_a\left[\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\nu}\right)\mathcal{P}\left(Re_{\nu},\mu\right)+\mathcal{C}_{F_{onset3}}\mathcal{P}\left(F_{onset3},\mu\right)\right]\frac{\partial \mu}{\partial U_{\ell}}\frac{\delta U_{\ell}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.31}$$

The differentiation of the $\tilde{R}e_{\theta t}$ equation sources leads to

$$\mathcal{I}^{\tilde{R}e_{\theta t}\_src}=\underbrace{-\int_{\Omega}\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\frac{\partial \rho}{\partial U_{\ell}}\frac{\delta U_{\ell}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\int_{\Omega}\rho\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\rho\right)\frac{\partial \rho}{\partial U_{\ell}}\frac{\delta U_{\ell}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\int_{\Omega}\rho\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},|\mathbf{U}|\right)\frac{\partial |\mathbf{U}|}{\partial v^A_k}\frac{\partial v^A_k}{\partial U_{\ell}}\frac{\delta U_{\ell}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{-\int_{\Omega}\tilde{R}e_a\tilde{R}e_{\theta t}\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\partial \rho}{\partial U_{\ell}}\frac{\delta U_{\ell}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}\underbrace{+\int_{\Omega}\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\delta\left(\rho\tilde{R}e_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\tilde{R}e_{\theta t}}$$

$$\underbrace{+\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}\frac{\delta F_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{\theta t}}}\underbrace{-\int_{\Omega}\tilde{R}e_a\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\delta Re^{eq}_{\theta,t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{Re^{eq}_{\theta,t}}}$$

$$\underbrace{+\int_{\Omega}\rho\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\mu\right)\frac{\partial \mu}{\partial U_{\ell}}\frac{\delta U_{\ell}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.32}$$

Regarding $\mathcal{I}^{F_{\theta t}}$, this can be written as

$$\begin{aligned}
\mathcal{I}^{F_{\theta t}} =& \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \rho\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega + \int_\Omega \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{- \int_\Omega \tilde{R}e_a \frac{\gamma}{\rho} \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \gamma\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega - \int_\Omega \tilde{R}e_a \frac{\tilde{R}e_{\theta t}}{\rho} \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \tilde{R}e_{\theta t}\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{+ \int_\Omega \frac{\tilde{R}e_a}{\rho} \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \gamma\right) \frac{\delta\left(\rho\gamma\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_}\gamma} + \underbrace{\int_\Omega \frac{\tilde{R}e_a}{\rho} \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \tilde{R}e_{\theta t}\right) \frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_}\tilde{R}e_{\theta t}} \\
& \underbrace{+ \int_\Omega \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \zeta\right) \frac{\delta \zeta}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^\zeta} + \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned} \tag{3.33}$$

Regarding term $\mathcal{I}^{Re_{\theta,t}^{eq}}$,

$$\begin{aligned}
Re_{\theta,t}^{eq} =& f\left[F\left[\lambda_\theta\left(\rho, \mu, \frac{d|\mathbf{U}|}{ds}, \theta\left(\rho, \mu, |\mathbf{U}|, Re_{\theta,t}^{eq}\right)\right)\right]\right] \\
\frac{\delta Re_{\theta,t}^{eq}}{\delta b_i} =& \mathcal{P}\left(Re_{\theta,t}^{eq}, F\right) \frac{\delta F}{\delta b_i} \\
\frac{\delta Re_{\theta,t}^{eq}}{\delta b_i} =& \mathcal{P}\left(Re_{\theta,t}^{eq}, F\right) \mathcal{P}\left(F, \lambda_\theta\right) \frac{\delta \lambda_\theta}{\delta b_i} \\
\frac{\delta Re_{\theta,t}^{eq}}{\delta b_i} =& \mathcal{P}\left(Re_{\theta,t}^{eq}, F\right) \mathcal{P}\left(F, \lambda_\theta\right) \left[\mathcal{P}\left(\lambda_\theta, \rho\right) \frac{\delta \rho}{\delta b_i} + \mathcal{P}\left(\lambda_\theta, \mu\right) \frac{\delta \mu}{\delta b_i} + \mathcal{P}\left(\lambda_\theta, \frac{d|\mathbf{U}|}{ds}\right) + \mathcal{P}\left(\lambda_\theta, \theta\right) \frac{\delta \theta}{\delta b_i}\right] \\
\underline{\frac{\delta Re_{\theta,t}^{eq}}{\delta b_i}} =& \mathcal{P}\left(Re_{\theta,t}^{eq}, F\right) \mathcal{P}\left(F, \lambda_\theta\right) \left[\mathcal{P}\left(\lambda_\theta, \rho\right) \frac{\delta \rho}{\delta b_i} + \mathcal{P}\left(\lambda_\theta, \mu\right) \frac{\delta \mu}{\delta b_i} + \mathcal{P}\left(\lambda_\theta, \frac{d|\mathbf{U}|}{ds}\right) \frac{\delta}{\delta b_i}\left(\frac{d|\mathbf{U}|}{ds}\right)\right. \\
& \left. + \mathcal{P}\left(\lambda_\theta, \theta\right)\left(\mathcal{P}\left(\theta, \rho\right) \frac{\delta \rho}{\delta b_i} + \mathcal{P}\left(\theta, \mu\right) \frac{\delta \mu}{\delta b_i} + \mathcal{P}\left(\theta, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\delta v_k^A}{\delta b_i} + \mathcal{P}\left(\theta, Re_{\theta,t}^{eq}\right) \underline{\frac{\delta Re_{\theta,t}^{eq}}{\delta b_i}}\right)\right]
\end{aligned}$$

So, $\mathcal{I}^{Re^{eq}_{\theta,t}}$ is expressed as

$$\begin{aligned}\mathcal{I}^{Re^{eq}_{\theta,t}} =& \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{Re^{eq}_{\theta,t}} \left[\mathcal{P}\left(\lambda_\theta,\rho\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\rho\right)\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{Re^{eq}_{\theta,t}} \mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,|\mathbf{U}|\right)\frac{\partial|\mathbf{U}|}{\partial v^A_k}\frac{\partial v^A_k}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} + \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds}\frac{\delta}{\delta b_i}\left(\frac{d|\mathbf{U}|}{ds}\right)\mathrm{d}\Omega}_{\mathcal{I}^{d|\mathbf{U}|ds}} \\ &\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{Re^{eq}_{\theta,t}} \left[\mathcal{P}\left(\lambda_\theta,\mu\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \end{aligned} \tag{3.34}$$

and

$$\begin{aligned}\mathcal{I}^{d|\mathbf{U}|ds} =& \underbrace{\int_{\partial\Omega} \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds}\frac{v^A_k v^A_m}{|\mathbf{U}|^2}\mathsf{n}_m\frac{\delta v^A_k}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_{18}} \underbrace{-\int_\Omega \frac{\partial}{\partial x_m}\left[\tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds}\frac{v^A_k v^A_m}{|\mathbf{U}|^2}\right]\frac{\partial v^A_k}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds}\left[-\frac{2}{|\mathbf{U}|}\frac{d|\mathbf{U}|}{ds}\frac{\partial|\mathbf{U}|}{\partial v^A_k}+\frac{v^A_m}{|\mathbf{U}|^2}\left(\frac{\partial v^A_k}{\partial x_m}+\frac{\partial v^A_m}{\partial x_k}\right)\right]\frac{\partial v^A_k}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{-\int_\Omega \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds}\frac{v^A_k v^A_m}{|\mathbf{U}|^2}\frac{\partial v^A_k}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow \text{SD}} \end{aligned} \tag{3.35}$$

Terms $\mathcal{I}^\zeta$ and $\mathcal{I}^S$ are expanded as

$$\begin{aligned}\mathcal{I}^\zeta =& \underbrace{\int_{\partial\Omega}\frac{\mathcal{C}^{\gamma-\tilde{R}e_{\theta t}}_\zeta}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v^A_r}{\partial x_q}\mathsf{n}_\ell\frac{\delta v^A_m}{\delta b_i}\mathrm{d}S}_{\rightarrow \mathcal{S}_{20}} \underbrace{-\int_\Omega \frac{\partial}{\partial x_\ell}\left[\frac{\mathcal{C}^{\gamma-\tilde{R}e_{\theta t}}_\zeta}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v^A_r}{\partial x_q}\right]\frac{\partial v^A_m}{\partial U_p}\frac{\delta U_p}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{-\int_\Omega \frac{\mathcal{C}^{\gamma-\tilde{R}e_{\theta t}}_\zeta}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v^A_r}{\partial x_q}\frac{\partial v^A_m}{\partial x_p}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_p}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow \text{SD}} \end{aligned}$$

$$\mathcal{I}^{S} = \underbrace{\int_{\partial\Omega} \frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S} 2S_{km}\mathsf{n}_m \frac{\delta v_k^A}{\delta b_i}\mathrm{d}S}_{\rightarrow \mathcal{S}_{21}} \underbrace{-\int_{\Omega} \frac{\partial}{\partial x_m}\left[\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S} 2S_{km}\right]\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{-\int_{\Omega} \frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S} 2S_{km}\frac{\partial v_k^A}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow \text{SD}} \tag{3.36}$$

The jump at the numeration of surface integrals ($\mathcal{S}_{18}$ to $\mathcal{S}_{20}$) is done on purpose, so as to follow the numbering of surface integrals of the rest transition models. Finally, $\mathcal{I}^{\mu_t}$

$$\mathcal{I}^{\mu_t} = \underbrace{\int_{\Omega} \mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\frac{\delta\tilde{\mu}}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_SA}}$$
$$+ \underbrace{\int_{\Omega} \mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}}\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.37}$$

3.3.2 Differentiation of the SA-LM2015 Transition Model Source Terms

For the differentiation of the source terms $\left(\mathcal{I}^{\gamma\_src},\mathcal{I}^{\tilde{Re}_{\theta t}\_src}\right)$ of the SA-LM2015 model, the following definitions are used

$$\begin{aligned}
\mathcal{C}_{F_{onset}} &= -\rho c_{\alpha_1} F_{length}\zeta\sqrt{\gamma}\left(1-c_{\epsilon_1}\gamma\right)\\
\mathcal{C}_{F_{onset3}} &= \mathcal{C}_{F_{onset}}\mathcal{P}\left(F_{onset},F_{onset3}\right)\\
\mathcal{C}_{F_{onset1}} &= \mathcal{C}_{F_{onset}}\mathcal{P}\left(F_{onset},F_{onset2}\right)\mathcal{P}\left(F_{onset2},F_{onset1}\right)\\
\mathcal{C}_{F_{\theta t}} &= \rho\frac{c_{\theta,t}}{\mathcal{T}}\left(Re_{\theta,t}^{eq}-\tilde{Re}_{\theta t}\right) - \rho\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\min\left(Re_{SCF}-\tilde{Re}_{\theta t}\right)\\
\mathcal{C}_{Re_{\theta,t}^{eq}} &= -\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\mathcal{P}\left(Re_{\theta,t}^{eq},F\right)\mathcal{P}\left(F,\lambda_\theta\right)}{1-\mathcal{P}\left(Re_{\theta,t}^{eq},F\right)\mathcal{P}\left(F,\lambda_\theta\right)\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,Re_{\theta,t}^{eq}\right)}\\
\mathcal{C}_{d|\mathbf{U}|ds} &= \mathcal{C}_{Re_{\theta,t}^{eq}}\mathcal{P}\left(\lambda_\theta,\frac{d\left|\mathbf{U}\right|}{ds}\right)\\
\mathcal{C}_{Re_{SCF}} &= -\rho\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\min\left(Re_{SCF}-\tilde{Re}_{\theta t}\right),Re_{SCF}\right)F_{\theta t}\frac{35.088\frac{1}{\theta_t}\frac{1}{\mathrm{Re}_0}}{1-35.088\frac{1}{\theta_t}\frac{0.82\mu}{\rho|\mathbf{U}|}\frac{1}{\mathrm{Re}_0}}
\end{aligned}$$

$$
\begin{aligned}
\mathcal{C}_{\mathcal{D}H_{CF}} &= \mathcal{C}_{Re_{SCF}} \left(\mathcal{P}\left(f\left(\mathcal{D}H_{CF}^{+}\right), \mathcal{D}H_{CF}^{+}\right) \mathcal{P}\left(\mathcal{D}H_{CF}^{+}, \mathcal{D}H_{CF}\right) \right. \\
&\quad \left. - \mathcal{P}\left(f\left(\mathcal{D}H_{CF}^{-}\right), \mathcal{D}H_{CF}^{-}\right) \mathcal{P}\left(\mathcal{D}H_{CF}^{-}, \mathcal{D}H_{CF}\right) \right) \\
\mathcal{C}_{\zeta_{streamwise}} &= \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, H_{CF}\right) \mathcal{P}\left(H_{CF}, \zeta_{streamwise}\right) sign\left(\frac{v_k^A}{|\mathbf{U}|} \frac{\partial v_m^A}{\partial x_\ell} \varepsilon_{k\ell m} \right) \\
\mathcal{C}_{\zeta}^{\gamma\text{–}\tilde{R}e_{\theta t}} &= -\gamma_a \rho c_{\alpha_1} F_{length} F_{onset} \sqrt{\gamma} \left(1 - c_{\epsilon_1}\gamma\right) + \gamma_a \rho c_{\alpha_2} \gamma F_{turb} (c_{\epsilon_2}\gamma - 1) + \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \zeta\right) \\
\mathcal{C}_{S}^{\gamma\text{–}\tilde{R}e_{\theta t}} &= \gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, S\right) \\
&\quad + \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, S\right) \\
\mathcal{C}_{\mu_t}^{\gamma\text{–}\tilde{R}e_{\theta t}} &= \gamma_a \rho c_{\alpha_2} \zeta \gamma \left(c_{\epsilon_2}\gamma - 1\right) \mathcal{P}\left(F_{turb}, \mu_t\right) + \gamma_a \mathcal{C}_{F_{onset3}} \mathcal{P}\left(F_{onset3}, \mu_t\right)
\end{aligned}
$$

For the equation of γ, $\mathcal{I}^{\gamma\_src} = \mathcal{I}^{\gamma\_prod} + \mathcal{I}^{\gamma\_destr}$

$$
\begin{aligned}
\mathcal{I}^{\gamma\_prod} = & \underbrace{-\int_\Omega \gamma_a c_{\alpha_1} F_{length} F_{onset} \zeta \sqrt{\gamma} \left(1 - c_{\epsilon_1}\gamma\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{+\frac{1}{2}\int_\Omega \gamma \gamma_a c_{\alpha_1} F_{length} F_{onset} \zeta \sqrt{\gamma} \left(1/\gamma - 3c_{\epsilon_1}\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{+\int_\Omega \tilde{R}e_{\theta t} \gamma_a c_{\alpha_1} F_{onset} \zeta \sqrt{\gamma} \left(1 - c_{\epsilon_1}\gamma\right) \mathcal{P}\left(F_{length}, \tilde{R}e_{\theta t}\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{-\frac{1}{2}\int_\Omega \gamma_a c_{\alpha_1} F_{length} F_{onset} \zeta \sqrt{\gamma} \left(1/\gamma - 3c_{\epsilon_1}\right) \frac{\delta\left(\rho\gamma\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma} \\
& \underbrace{-\int_\Omega \gamma_a c_{\alpha_1} F_{onset} \zeta \sqrt{\gamma} \left(1 - c_{\epsilon_1}\gamma\right) \mathcal{P}\left(F_{length}, \tilde{R}e_{\theta t}\right) \frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}} \\
& \underbrace{-\int_\Omega \gamma_a \rho c_{\alpha_1} F_{length} F_{onset} \sqrt{\gamma} \left(1 - c_{\epsilon_1}\gamma\right) \frac{\delta \zeta}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\zeta}} + \underbrace{\int_\Omega \gamma_a \mathcal{C}_{F_{onset}} \frac{\delta F_{onset}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{F_{onset}}}
\end{aligned}
\tag{3.38}
$$

$$\mathcal{I}^{\gamma\_destr} = \underbrace{\int_\Omega \gamma_a c_{\alpha_2} \zeta \gamma F_{turb} \left(c_{\epsilon_2}\gamma - 1\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega - \int_\Omega \gamma \gamma_a c_{\alpha_2} \zeta F_{turb} \left(2c_{\epsilon_2}\gamma - 1\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \gamma_a c_{\alpha_2} \zeta F_{turb} \left(2c_{\epsilon_2}\gamma - 1\right) \frac{\delta\left(\rho\gamma\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma} + \underbrace{\int_\Omega \gamma_a \rho c_{\alpha_2} \zeta \gamma \left(c_{\epsilon_2}\gamma - 1\right) \mathcal{P}\left(F_{turb}, \mu_t\right) \frac{\delta \mu_t}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\mu_t}}$$
$$\underbrace{+\int_\Omega \gamma_a \rho c_{\alpha_2} \gamma F_{turb} \left(c_{\epsilon_2}\gamma - 1\right) \frac{\delta \zeta}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\zeta}} + \underbrace{\int_\Omega \gamma_a \rho c_{\alpha_2} \zeta \gamma \left(c_{\epsilon_2}\gamma - 1\right) \mathcal{P}\left(F_{turb}, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \mathit{Suth}} \tag{3.39}$$

Regarding $\mathcal{I}^{F_{onset}}$, this can be written as

$$\mathcal{I}^{F_{onset}} = \underbrace{-\int_\Omega \gamma_a \frac{\tilde{R}e_{\theta t}}{\rho} \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_{\theta c}\right) \mathcal{P}\left(Re_{\theta c}, \tilde{R}e_{\theta t}\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, \rho\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, S\right) \frac{\delta S}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{S}}$$
$$\underbrace{+\int_\Omega \gamma_a \frac{1}{\rho} \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_{\theta c}\right) \mathcal{P}\left(Re_{\theta c}, \tilde{R}e_{\theta t}\right) \frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}}$$
$$\underbrace{\int_\Omega \gamma_a \mathcal{C}_{F_{onset3}} \mathcal{P}\left(F_{onset3}, \mu_t\right) \frac{\delta \mu_t}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\mu_t}} + \underbrace{\int_\Omega \gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, \Delta\right) \frac{\delta \Delta}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\Delta}$$

$$\underbrace{+\int_{\Omega}\gamma_a\left[\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_S\right)\mathcal{P}\left(Re_S,\mu\right)+\mathcal{C}_{F_{onset3}}\mathcal{P}\left(F_{onset3},\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.40}$$

The differentiation of the $\tilde{R}e_{\theta t}$ equation source terms leads to

$$\mathcal{I}^{\tilde{R}e_{\theta t}\_src}=\mathcal{I}^{\tilde{R}e_{\theta t}\_P_{\theta,t}}+\mathcal{I}^{\tilde{R}e_{\theta t}\_D_{SCF}} \tag{3.41}$$

$$\begin{aligned}
\mathcal{I}^{\tilde{R}e_{\theta t}_P_{\theta,t}} = & \underbrace{-\int_{\Omega}\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{+\int_{\Omega}\rho\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\rho\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{+\int_{\Omega}\rho\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},|\mathbf{U}|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\tilde{R}e_a\tilde{R}e_{\theta t}\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}}+\underbrace{\int_{\Omega}\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\delta\left(\rho\tilde{R}e_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_}\tilde{R}e_{\theta t}} \\
& \underbrace{-\int_{\Omega}\tilde{R}e_a\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\delta Re^{eq}_{\theta,t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{Re^{eq}_{\theta,t}}}+\underbrace{\int_{\Omega}\tilde{R}e_a\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\frac{\delta F_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{\theta t}}} \\
& \underbrace{+\int_{\Omega}\rho\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned} \tag{3.42}$$

and

$$\mathcal{I}^{\tilde{R}e_{\theta t}\_D_{SCF}}=\underbrace{-\int_{\Omega}\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\min\left(Re_{SCF}-\tilde{R}e_{\theta t},0\right)F_{\theta t}\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$

$$
\begin{aligned}
&\underbrace{+\int_{\Omega} \tilde{R}e_a \rho \frac{c_{\theta,t}}{\mathcal{T}^2} c_{crossflow} \min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right) F_{\theta t} \mathcal{P}\left(\mathcal{T}, \rho\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\int_{\Omega} \tilde{R}e_a \rho \frac{c_{\theta,t}}{\mathcal{T}^2} c_{crossflow} \min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right) F_{\theta t} \mathcal{P}\left(\mathcal{T}, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\int_{\Omega} \tilde{R}e_{\theta t} \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \mathcal{P}\left(\min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right), \tilde{R}e_{\theta t}\right) F_{\theta t} \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{-\int_{\Omega} \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \mathcal{P}\left(\min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right), \tilde{R}e_{\theta t}\right) F_{\theta t} \frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}} \\
&\underbrace{-\int_{\Omega} \tilde{R}e_a \rho \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right) \frac{\delta F_{\theta t}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{F_{\theta t}}} \underbrace{+\int_{\Omega} \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{\delta Re_{SCF}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{Re_{SCF}}} \\
&\underbrace{+\int_{\Omega} \tilde{R}e_a \rho \frac{c_{\theta,t}}{\mathcal{T}^2} c_{crossflow} \min\left(Re_{SCF} - \tilde{R}e_{\theta t}, 0\right) F_{\theta t} \mathcal{P}\left(\mathcal{T}, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned}
\tag{3.43}
$$

Regarding $\mathcal{I}^{F_{\theta t}}$, this becomes

$$
\begin{aligned}
\mathcal{I}^{F_{\theta t}} = &\underbrace{\int_{\Omega} \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, \rho\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\int_{\Omega} \tilde{R}e_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \rho\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\int_{\Omega} \tilde{R}e_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}
\end{aligned}
$$

$$
\begin{aligned}
&\underbrace{-\int_{\Omega}\tilde{Re}_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{Re}_{\theta t}\right)\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\int_{\Omega}\frac{\tilde{Re}_a}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{Re}_{\theta t}\right)\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\tilde{Re}_{\theta t}}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\zeta\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\zeta}}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,S\right)\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{S}}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,\Delta\right)\frac{\delta\Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\Delta}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned}
\tag{3.44}
$$

Regarding the term $\mathcal{I}^{Re^{eq}_{\theta,t}}$,

$$
\begin{aligned}
\mathcal{I}^{Re^{eq}_{\theta,t}} =&\underbrace{\int_{\Omega}\tilde{Re}_a\mathcal{C}_{Re^{eq}_{\theta,t}}\left[\mathcal{P}\left(\lambda_\theta,\rho\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\rho\right)\right]\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{Re^{eq}_{\theta,t}}\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,|\mathbf{U}|\right)\frac{\partial|\mathbf{U}|}{\partial v_k^A}\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}+\underbrace{\int_{\Omega}\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{\delta}{\delta b_i}\left(\frac{d|\mathbf{U}|}{ds}\right)\mathrm{d}\Omega}_{\mathcal{I}^{d|\mathbf{U}|ds}}\\
&\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{Re^{eq}_{\theta,t}}\left[\mathcal{P}\left(\lambda_\theta,\mu\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\mu\right)\right]\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned}
\tag{3.45}
$$

and

$$\mathcal{I}^{d|\mathbf{U}|ds} = \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds} \left[-\frac{2}{|\mathbf{U}|} \frac{d|\mathbf{U}|}{ds} \frac{\partial |\mathbf{U}|}{\partial v_k^A} + \frac{v_m^A}{|\mathbf{U}|^2} \left(\frac{\partial v_k^A}{\partial x_m} + \frac{\partial v_m^A}{\partial x_k} \right) \right] \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{-\int_\Omega \frac{\partial}{\partial x_m} \left[\tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{v_k^A v_m^A}{|\mathbf{U}|^2} \right] \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} + \underbrace{\int_{\partial\Omega} \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{v_k^A v_m^A}{|\mathbf{U}|^2} \mathsf{n}_m \frac{\delta v_k^A}{\delta b_i} \mathrm{d}S}_{\mathcal{S}_{18}}$$
$$\underbrace{-\int_\Omega \tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{v_k^A v_m^A}{|\mathbf{U}|^2} \frac{\partial v_k^A}{\partial x_\ell} \frac{\partial}{\partial x_m} \left(\frac{\delta x_\ell}{\delta b_i} \right) \mathrm{d}\Omega}_{\rightarrow \text{SD}} \tag{3.46}$$

Regarding term $\mathcal{I}^{Re_{SCF}}$,

$$\mathcal{I}^{Re_{SCF}} = \underbrace{-\int_\Omega \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{0.82\mu}{\rho^2 |\mathbf{U}|} \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \underbrace{-\int_\Omega \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{0.82\mu}{\rho |\mathbf{U}|^2} \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \frac{\delta \mathcal{D}H_{CF}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\mathcal{D}H_{CF}}} \underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{0.82 Re_{SCF}}{\rho |\mathbf{U}|} \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.47}$$

and

$$\mathcal{I}^{\mathcal{D}H_{CF}} = \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, H_{CF}\right) \mathcal{P}\left(H_{CF}, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, \mu_t\right) \frac{\delta \mu_t}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\mu_t}} \underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\zeta_{streamwise}} \frac{\delta \zeta_{streamwise}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\zeta_{streamwise}}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, H_{CF}\right) \mathcal{P}\left(H_{CF}, \Delta\right) \frac{\delta \Delta}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\Delta}$$

$$\underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{\Delta H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.48}$$

$$\begin{aligned}
\mathcal{I}^{\zeta_{streamwise}} =& \underbrace{\int_{\partial\Omega}\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_k^A}{|\mathbf{U}|}\varepsilon_{k\ell m}\mathsf{n}_\ell\frac{\delta v_m^A}{\delta b_i}\mathrm{d}S}_{\mathcal{S}_{19}} \\
& \underbrace{-\int_{\Omega}\frac{\partial}{\partial x_q}\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_k^A}{|\mathbf{U}|}\varepsilon_{kqm}\right]\frac{\partial v_m^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{+\int_{\Omega}\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{1}{|\mathbf{U}|}\frac{\partial v_m^A}{\partial x_q}\left[\varepsilon_{kqm}+\frac{u_r}{|\mathbf{U}|}\varepsilon_{rqm}\frac{\partial|\mathbf{U}|}{\partial v_k^A}\right]\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_k^A}{|\mathbf{U}|}\varepsilon_{kqm}\frac{\partial v_m^A}{\partial x_\ell}\frac{\partial}{\partial x_q}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow \text{SD}}
\end{aligned} \tag{3.49}$$

Terms $\mathcal{I}^\zeta$ and $\mathcal{I}^S$ are written as

$$\begin{aligned}
\mathcal{I}^\zeta =& \underbrace{\int_{\partial\Omega}\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_r^A}{\partial x_q}\mathsf{n}_\ell\frac{\delta v_m^A}{\delta b_i}\mathrm{d}S}_{\rightarrow\mathcal{S}_{20}}\underbrace{-\int_{\Omega}\frac{\partial}{\partial x_\ell}\left[\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_r^A}{\partial x_q}\right]\frac{\partial v_m^A}{\partial U_p}\frac{\delta U_p}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_r^A}{\partial x_q}\frac{\partial v_m^A}{\partial x_p}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_p}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow \text{SD}} \\
\mathcal{I}^S =& \underbrace{\int_{\partial\Omega}\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\mathsf{n}_m\frac{\delta v_k^A}{\delta b_i}\mathrm{d}S}_{\rightarrow\mathcal{S}_{21}}\underbrace{-\int_{\Omega}\frac{\partial}{\partial x_m}\left[\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\right]\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\frac{\partial v_k^A}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow \text{SD}}
\end{aligned} \tag{3.50}$$

Finally, $\mathcal{I}^{\mu_t}$

$$\begin{aligned}\mathcal{I}^{\mu_t} = &\underbrace{\int_\Omega \mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}} \left[\mathcal{P}\left(\mu_t, \tilde{\mu}\right) + \mathcal{P}\left(\mu_t, f_{v_1}\right)\mathcal{P}\left(f_{v_1}, \chi\right)\mathcal{P}\left(\chi, \tilde{\mu}\right)\right] \frac{\delta\tilde{\mu}}{\delta b_i} \mathrm{d}\Omega}_{\text{FAE\_SA}} \\ &\underbrace{+\int_\Omega \mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}} \mathcal{P}\left(\mu_t, f_{v_1}\right)\mathcal{P}\left(f_{v_1}, \chi\right)\mathcal{P}\left(\chi, \mu\right) \frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{Suth}\end{aligned} \tag{3.51}$$

3.3.3 Differentiation of the SA-sLM2015 Transition Model Source Terms

For the differentiation of the integral containing the source terms $\left(\mathcal{I}^{\gamma\_src}, \mathcal{I}^{\tilde{Re}_{\theta t}\_src}\right)$ of the SA-sLM2015 model, the following definitions are used. These are the same quantities introduced during the differentiation of the SA-LM2015 model, Sec. 3.3.2, but they are now defined differently as

$$\begin{aligned}
\mathcal{C}_{F_{onset}} &= -\rho c_{\alpha_1} F_{length} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right) \\
&\quad + \rho c_{\alpha_2} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma - 1\right)\mathcal{P}\left(F_{turb}, F_{onset}\right) \\
\mathcal{C}_{F_{onset1}} &= \mathcal{C}_{F_{onset}}\mathcal{P}\left(F_{onset}, F_{onset1}\right) \\
\mathcal{C}_{F_{length}} &= -\rho c_{\alpha_1} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right) \\
\mathcal{C}_{F_{\theta t}} &= \rho\frac{c_{\theta,t}}{\mathcal{T}}\left(Re_{\theta,t}^{eq} - \tilde{Re}_{\theta t}\right) - \rho\frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \min\left(Re_{SCF} - \tilde{Re}_{\theta t}\right) \\
\mathcal{C}_{F(\lambda_\theta)} &= \mathcal{P}\left(F\left(\lambda_\theta\right), F\left(\lambda_\theta\right)_2\right)\mathcal{P}\left(F\left(\lambda_\theta\right)_2, F\left(\lambda_\theta\right)_1\right)\mathcal{P}\left(F\left(\lambda_\theta\right)_1, \lambda_\theta\right) \\
&\quad + \mathcal{P}\left(F\left(\lambda_\theta\right), F\left(\lambda_\theta\right)_3\right)\mathcal{P}\left(F\left(\lambda_\theta\right)_3, \lambda_\theta\right) \\
\mathcal{C}_{Re_{\theta,t}^{eq}} &= -\frac{\rho c_{\theta,t}}{\mathcal{T}} \frac{\left(1 - F_{\theta t}\right)\mathcal{P}\left(Re_{\theta,t}^{eq}, F\right)\mathcal{C}_{F(\lambda_\theta)}}{1 - \mathcal{P}\left(Re_{\theta,t}^{eq}, F\right)\mathcal{C}_{F(\lambda_\theta)}\mathcal{P}\left(\lambda_\theta, \theta\right)\mathcal{P}\left(\theta, Re_{\theta,t}^{eq}\right)} \\
\mathcal{C}_{d|\mathbf{U}|ds} &= \mathcal{C}_{Re_{\theta,t}^{eq}}\mathcal{P}\left(\lambda_\theta, \frac{d\left|\mathbf{U}\right|}{ds}\right) \\
\mathcal{C}_{Re_{SCF}} &= -\rho\frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow}\mathcal{P}\left(\min\left(Re_{SCF} - \tilde{Re}_{\theta t}\right), Re_{SCF}\right) F_{\theta t} \frac{35.088\frac{1}{\theta_t}\frac{1}{\mathrm{Re}_0}}{1 - 35.088\frac{1}{\theta_t}\frac{0.82\mu}{\rho|\mathbf{U}|}\frac{1}{\mathrm{Re}_0}}
\end{aligned}$$

$$
\begin{aligned}
\mathcal{C}_{\mathcal{D}H_{CF}} &= \mathcal{C}_{Re_{SCF}} \left(\mathcal{P}\left(f\left(\mathcal{D}H_{CF}^{+}\right), \mathcal{D}H_{CF}^{+}\right) \mathcal{P}\left(\mathcal{D}H_{CF}^{+}, \mathcal{D}H_{CF}\right) \right. \\
&\quad \left. - \mathcal{P}\left(f\left(\mathcal{D}H_{CF}^{-}\right), \mathcal{D}H_{CF}^{-}\right) \mathcal{P}\left(\mathcal{D}H_{CF}^{-}, \mathcal{D}H_{CF}\right) \right) \\
\mathcal{C}_{\zeta_{streamwise}} &= \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, H_{CF}\right) \mathcal{P}\left(H_{CF}, \zeta_{streamwise}\right) sign\left(\frac{v_k^A}{|\mathbf{U}|} \frac{\partial v_m^A}{\partial x_\ell} \varepsilon_{k\ell m} \right) \\
\mathcal{C}_{\zeta}^{\gamma - \tilde{R}e_{\theta t}} &= -\gamma_a \rho c_{\alpha_1} F_{length} F_{onset} \mathcal{P}\left(\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right), \zeta \right) \sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right) \\
&\quad + \gamma_a \rho c_{\alpha_2} \gamma F_{turb} \mathcal{P}\left(\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right), \zeta \right) \left(c_{\epsilon_2}\gamma - 1\right) + \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, \zeta\right) \\
\mathcal{C}_{S}^{\gamma - \tilde{R}e_{\theta t}} &= \gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, S\right) \\
&\quad + \tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, S\right) \\
\mathcal{C}_{\mu_t}^{\gamma - \tilde{R}e_{\theta t}} &= \gamma_a \rho c_{\alpha_2} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right) \gamma\left(c_{\epsilon_2}\gamma - 1\right) \mathcal{P}\left(F_{turb}, \mu_t\right)
\end{aligned}
$$

For equation of γ, $\mathcal{I}^{\gamma\_src} = \mathcal{I}^{\gamma\_prod} + \mathcal{I}^{\gamma\_destr}$

$$
\begin{aligned}
\mathcal{I}^{\gamma\_prod} = & \underbrace{-\int_\Omega \gamma_a c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right) \sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{+\frac{1}{2}\int_\Omega \gamma \gamma_a c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right) \sqrt{\gamma}\left({}^{1}/_{\gamma} - 3c_{\epsilon_1}\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{-\int_\Omega \gamma_a \frac{\tilde{R}e_{\theta t}}{\rho} \mathcal{C}_{F_{length}} \mathcal{P}\left(F_{length}, F_{length1}\right) \mathcal{P}\left(F_{length1}, \tilde{R}e_{\theta t}\right) \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{-\frac{1}{2}\int_\Omega \gamma_a c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right) \sqrt{\gamma}\left({}^{1}/_{\gamma} - 3c_{\epsilon_1}\right) \frac{\delta\left(\rho\gamma\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\gamma} \\
& \underbrace{+\int_\Omega \gamma_a \frac{1}{\rho} \mathcal{C}_{F_{length}} \mathcal{P}\left(F_{length}, F_{length1}\right) \mathcal{P}\left(F_{length1}, \tilde{R}e_{\theta t}\right) \frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}}
\end{aligned}
$$

$$\underbrace{-\int_{\Omega}\gamma_a\rho c_{\alpha_1}F_{length}F_{onset}\mathcal{P}\left(\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right),\zeta\right)\sqrt{\gamma}\left(1-c_{\epsilon_1}\gamma\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\zeta}}$$

$$\underbrace{-\int_{\Omega}\gamma_a\rho c_{\alpha_1}F_{length}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1-c_{\epsilon_1}\gamma\right)\frac{\delta F_{onset}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{onset}}} \tag{3.52}$$

$$\begin{aligned}
\mathcal{I}^{\gamma_destr} =& \underbrace{\int_{\Omega}\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma F_{turb}\left(c_{\epsilon_2}\gamma-1\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE_MF}} \\
& \underbrace{-\int_{\Omega}\gamma\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE_MF}} \\
& \underbrace{+\int_{\Omega}\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\frac{\delta\left(\rho\gamma\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE_}\gamma} \\
& \underbrace{+\int_{\Omega}\gamma_a\rho c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},F_{onset}\right)\frac{\delta F_{onset}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{onset}}} \\
& \underbrace{+\int_{\Omega}\gamma_a\rho c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu_t\right)\frac{\delta\mu_t}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\mu_t}} \\
& \underbrace{+\int_{\Omega}\gamma_a\rho c_{\alpha_2}\gamma F_{turb}\mathcal{P}\left(\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right),\zeta\right)\left(c_{\epsilon_2}\gamma-1\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\zeta}} \\
& \underbrace{+\int_{\Omega}\gamma_a\rho c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned} \tag{3.53}$$

Regarding $\mathcal{I}^{F_{onset}}$, this can be written in the form

$$\begin{aligned}\mathcal{I}^{F_{onset}} = &\underbrace{-\int_\Omega \gamma_a \frac{\tilde{R}e_{\theta t}}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1}, Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c}, \tilde{R}e_{\theta t}\right)\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{+\int_\Omega \gamma_a \frac{1}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1}, Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c}, \tilde{R}e_{\theta t}\right)\frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}} \\ &\underbrace{+\int_\Omega \gamma_a \mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1}, Re_S\right)\mathcal{P}\left(Re_S, S\right)\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^S} \\ &\underbrace{+\int_\Omega \gamma_a \mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1}, Re_S\right)\mathcal{P}\left(Re_S, \Delta\right)\frac{\delta \Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\Delta} \\ &\underbrace{+\int_\Omega \gamma_a \left[\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1}, Re_S\right)\mathcal{P}\left(Re_S, \mu\right) + \mathcal{C}_{F_{onset3}}\mathcal{P}\left(F_{onset3}, \mu\right)\right]\frac{\partial \mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth}\end{aligned} \tag{3.54}$$

The differentiation of the source terms in the $\tilde{R}e_{\theta t}$ equation leads to

$$\mathcal{I}^{\tilde{R}e_{\theta t}\_src} = \mathcal{I}^{\tilde{R}e_{\theta t}\_P_{\theta,t}} + \mathcal{I}^{\tilde{R}e_{\theta t}\_D_{SCF}} \tag{3.55}$$

$$\begin{aligned}\mathcal{I}^{\tilde{R}e_{\theta t}\_P_{\theta,t}} = &\underbrace{-\int_\Omega \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t} - \tilde{R}e_{\theta t}\right)\left(1 - F_{\theta t}\right)\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{+\int_\Omega \rho \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t} - \tilde{R}e_{\theta t}\right)\left(1 - F_{\theta t}\right)\mathcal{P}\left(\mathcal{T}, \rho\right)\frac{\partial \rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \\ &\underbrace{+\int_\Omega \rho \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t} - \tilde{R}e_{\theta t}\right)\left(1 - F_{\theta t}\right)\mathcal{P}\left(\mathcal{T}, |\mathbf{U}|\right)\frac{\partial |\mathbf{U}|}{\partial v_k^A}\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}\end{aligned}$$

$$\underbrace{-\int_{\Omega}\tilde{Re}_a\tilde{Re}_{\theta t}\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}+\underbrace{\int_{\Omega}\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\tilde{Re}_{\theta t}}$$

$$\underbrace{-\int_{\Omega}\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\frac{\delta Re^{eq}_{\theta,t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{Re^{eq}_{\theta,t}}}+\underbrace{\int_{\Omega}\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right)\frac{\delta F_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{\theta t}}}$$

$$\underbrace{+\int_{\Omega}\rho\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.56}$$

and

$$\mathcal{I}^{\tilde{Re}_{\theta t}\_D_{SCF}}=\underbrace{-\int_{\Omega}\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\int_{\Omega}\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},\rho\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\int_{\Omega}\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},|\mathbf{U}|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k}\frac{\partial v^A_k}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\int_{\Omega}\tilde{Re}_{\theta t}\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big),\tilde{Re}_{\theta t}\right)F_{\theta t}\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{-\int_{\Omega}\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big),\tilde{Re}_{\theta t}\right)F_{\theta t}\frac{\delta\left(\rho\tilde{Re}_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\tilde{Re}_{\theta t}}$$

$$\underbrace{-\int_{\Omega}\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)\frac{\delta F_{\theta t}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{F_{\theta t}}}+\underbrace{\int_{\Omega}\tilde{Re}_a\mathcal{C}_{Re_{SCF}}\frac{\delta Re_{SCF}}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{Re_{SCF}}}$$

$$\underbrace{+\int_{\Omega}\tilde{R}e_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{R}e_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},\mu\right)\frac{\partial\mu}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.57}$$

Regarding $\mathcal{I}^{F_{\theta t}}$, this becomes

$$\begin{aligned}
\mathcal{I}^{F_{\theta t}} =& \underbrace{\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,\rho\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\rho\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,|\mathbf{U}|\right)\frac{\partial|\mathbf{U}|}{\partial v_k^A}\frac{\partial v_k^A}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{-\int_{\Omega}\tilde{R}e_a\frac{\tilde{R}e_{\theta t}}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{R}e_{\theta t}\right)\frac{\partial\rho}{\partial U_\ell}\frac{\delta U_\ell}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\int_{\Omega}\frac{\tilde{R}e_a}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{R}e_{\theta t}\right)\frac{\delta\left(\rho\tilde{R}e_{\theta t}\right)}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_}\tilde{R}e_{\theta t}} \\
&\underbrace{+\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\zeta\right)\frac{\delta\zeta}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{\zeta}} \\
&\underbrace{+\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,S\right)\frac{\delta S}{\delta b_i}\mathrm{d}\Omega}_{\mathcal{I}^{S}} \\
&\underbrace{+\int_{\Omega}\tilde{R}e_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,\Delta\right)\frac{\delta\Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow \text{FAE_}\Delta}
\end{aligned}$$

$$\underbrace{+\int_{\Omega} \tilde{Re}_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth}$$

$$\underbrace{+\int_{\Omega} \tilde{Re}_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.58}$$

$\mathcal{I}^{Re^{eq}_{\theta,t}}$ is expressed as

$$\begin{aligned}
\mathcal{I}^{Re^{eq}_{\theta,t}} = & \underbrace{\int_{\Omega} \tilde{Re}_a \mathcal{C}_{Re^{eq}_{\theta,t}} \left[\mathcal{P}\left(\lambda_\theta, \rho\right) + \mathcal{P}\left(\lambda_\theta, \theta\right) \mathcal{P}\left(\theta, \rho\right)\right] \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{+\int_{\Omega} \tilde{Re}_a \mathcal{C}_{Re^{eq}_{\theta,t}} \mathcal{P}\left(\lambda_\theta, \theta\right) \mathcal{P}\left(\theta, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} + \underbrace{\int_{\Omega} \tilde{Re}_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{\delta}{\delta b_i} \left(\frac{d|\mathbf{U}|}{ds}\right) \mathrm{d}\Omega}_{\mathcal{I}^{d|\mathbf{U}|ds}} \\
& \underbrace{+\int_{\Omega} \tilde{Re}_a \mathcal{C}_{Re^{eq}_{\theta,t}} \left[\mathcal{P}\left(\lambda_\theta, \mu\right) + \mathcal{P}\left(\lambda_\theta, \theta\right) \mathcal{P}\left(\theta, \mu\right)\right] \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth}
\end{aligned} \tag{3.59}$$

and

$$\begin{aligned}
\mathcal{I}^{d|\mathbf{U}|ds} = & \underbrace{\int_{\Omega} \tilde{Re}_a \mathcal{C}_{d|\mathbf{U}|ds} \left[-\frac{2}{|\mathbf{U}|} \frac{d|\mathbf{U}|}{ds} \frac{\partial |\mathbf{U}|}{\partial v_k^A} + \frac{v_m^A}{|\mathbf{U}|^2} \left(\frac{\partial v_k^A}{\partial x_m} + \frac{\partial v_m^A}{\partial x_k}\right)\right] \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} \\
& \underbrace{-\int_{\Omega} \frac{\partial}{\partial x_m} \left[\tilde{Re}_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{v_k^A v_m^A}{|\mathbf{U}|^2}\right] \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE_MF}} + \underbrace{\int_{\partial\Omega} \tilde{Re}_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{v_k^A v_m^A}{|\mathbf{U}|^2} \mathsf{n}_m \frac{\delta v_k^A}{\delta b_i} \mathrm{d}S}_{\mathcal{S}_{18}} \\
& \underbrace{-\int_{\Omega} \tilde{Re}_a \mathcal{C}_{d|\mathbf{U}|ds} \frac{v_k^A v_m^A}{|\mathbf{U}|^2} \frac{\partial v_k^A}{\partial x_\ell} \frac{\partial}{\partial x_m} \left(\frac{\delta x_\ell}{\delta b_i}\right) \mathrm{d}\Omega}_{\rightarrow \text{SD}}
\end{aligned} \tag{3.60}$$

$\mathcal{I}^{Re_{SCF}}$ is expressed as

$$\mathcal{I}^{Re_{SCF}} = \underbrace{-\int_\Omega \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{0.82\mu}{\rho^2 |\mathbf{U}|} \frac{\partial \rho}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}} \underbrace{-\int_\Omega \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{0.82\mu}{\rho |\mathbf{U}|^2} \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \frac{\delta \mathcal{D}H_{CF}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\mathcal{D}H_{CF}}} \underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{Re_{SCF}} \frac{0.82 Re_{SCF}}{\rho |\mathbf{U}|} \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.61}$$

and

$$\mathcal{I}^{\mathcal{D}H_{CF}} = \underbrace{\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, H_{CF}\right) \mathcal{P}\left(H_{CF}, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, \mu_t\right) \frac{\delta \mu_t}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\mu_t}} \underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\zeta_{streamwise}} \frac{\delta \zeta_{streamwise}}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^{\zeta_{streamwise}}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\mathcal{D}H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, H_{CF}\right) \mathcal{P}\left(H_{CF}, \Delta\right) \frac{\delta \Delta}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_}\Delta}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\Delta H_{CF}} \mathcal{P}\left(\mathcal{D}H_{CF}, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow Suth} \tag{3.62}$$

$$\mathcal{I}^{\zeta_{streamwise}} = \underbrace{\int_{\partial\Omega} \tilde{R}e_a \mathcal{C}_{\zeta_{streamwise}} \frac{v_k^A}{|\mathbf{U}|} \varepsilon_{k\ell m} \mathsf{n}_\ell \frac{\delta v_m^A}{\delta b_i} \mathrm{d}S}_{\mathcal{S}_{19}}$$
$$\underbrace{-\int_\Omega \frac{\partial}{\partial x_q} \left[\tilde{R}e_a \mathcal{C}_{\zeta_{streamwise}} \frac{v_k^A}{|\mathbf{U}|} \varepsilon_{kqm} \right] \frac{\partial v_m^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$
$$\underbrace{+\int_\Omega \tilde{R}e_a \mathcal{C}_{\zeta_{streamwise}} \frac{1}{|\mathbf{U}|} \frac{\partial v_m^A}{\partial x_q} \left[\varepsilon_{kqm} + \frac{u_r}{|\mathbf{U}|} \varepsilon_{rqm} \frac{\partial |\mathbf{U}|}{\partial v_k^A} \right] \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{-\int_{\Omega} \tilde{Re}_a \mathcal{C}_{\zeta_{streamwise}} \frac{v_k^A}{|\mathbf{U}|} \varepsilon_{kqm} \frac{\partial v_m^A}{\partial x_\ell} \frac{\partial}{\partial x_q} \left(\frac{\delta x_\ell}{\delta b_i} \right) \mathrm{d}\Omega}_{\rightarrow \mathrm{SD}} \tag{3.63}$$

Terms $\mathcal{I}^{\zeta}$ and $\mathcal{I}^{S}$ are expressed as

$$\begin{aligned}
\mathcal{I}^{\zeta} =& \underbrace{\int_{\partial\Omega} \frac{\mathcal{C}_{\zeta}^{\gamma-\tilde{Re}_{\theta t}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{kqr} \frac{\partial v_r^A}{\partial x_q} \mathsf{n}_\ell \frac{\delta v_m^A}{\delta b_i} \mathrm{d}S}_{\rightarrow \mathcal{S}_{20}} \underbrace{- \int_{\Omega} \frac{\partial}{\partial x_\ell} \left[\frac{\mathcal{C}_{\zeta}^{\gamma-\tilde{Re}_{\theta t}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{kqr} \frac{\partial v_r^A}{\partial x_q} \right] \frac{\partial v_m^A}{\partial U_p} \frac{\delta U_p}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \mathrm{FAE_MF}} \\
& \underbrace{- \int_{\Omega} \frac{\mathcal{C}_{\zeta}^{\gamma-\tilde{Re}_{\theta t}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{kqr} \frac{\partial v_r^A}{\partial x_q} \frac{\partial v_m^A}{\partial x_p} \frac{\partial}{\partial x_\ell} \left(\frac{\delta x_p}{\delta b_i} \right) \mathrm{d}\Omega}_{\rightarrow \mathrm{SD}} \\
\mathcal{I}^{S} =& \underbrace{\int_{\partial\Omega} \frac{\mathcal{C}_{S}^{\gamma-\tilde{Re}_{\theta t}}}{S} 2 S_{km} \mathsf{n}_m \frac{\delta v_k^A}{\delta b_i} \mathrm{d}S}_{\rightarrow \mathcal{S}_{21}} \underbrace{- \int_{\Omega} \frac{\partial}{\partial x_m} \left[\frac{\mathcal{C}_{S}^{\gamma-\tilde{Re}_{\theta t}}}{S} 2 S_{km} \right] \frac{\partial v_k^A}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \mathrm{FAE_MF}} \\
& \underbrace{- \int_{\Omega} \frac{\mathcal{C}_{S}^{\gamma-\tilde{Re}_{\theta t}}}{S} 2 S_{km} \frac{\partial v_k^A}{\partial x_\ell} \frac{\partial}{\partial x_m} \left(\frac{\delta x_\ell}{\delta b_i} \right) \mathrm{d}\Omega}_{\rightarrow \mathrm{SD}}
\end{aligned} \tag{3.64}$$

Finally, $\mathcal{I}^{\mu_t}$ takes the form

$$\begin{aligned}
\mathcal{I}^{\mu_t} =& \underbrace{\int_{\Omega} \mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}} \left[\mathcal{P}\left(\mu_t, \tilde{\mu}\right) + \mathcal{P}\left(\mu_t, f_{v_1}\right) \mathcal{P}\left(f_{v_1}, \chi\right) \mathcal{P}\left(\chi, \tilde{\mu}\right) \right] \frac{\delta \tilde{\mu}}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \mathrm{FAE_SA}} \\
& + \underbrace{\int_{\Omega} \mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}} \mathcal{P}\left(\mu_t, f_{v_1}\right) \mathcal{P}\left(f_{v_1}, \chi\right) \mathcal{P}\left(\chi, \mu\right) \frac{\partial \mu}{\partial U_\ell} \frac{\delta U_\ell}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \mathit{Suth}}
\end{aligned} \tag{3.65}$$

3.4 Differentiation of the Hamilton–Jacobi Equation

Term $\mathcal{I}^{\Delta}$ in Eq. 3.8 is developed as

$$\mathcal{I}^{\Delta} = \underbrace{2\int_{\partial\Omega}\Delta_a\frac{\partial\Delta}{\partial x_k}\mathsf{n}_k\frac{\delta\Delta}{\delta b_i}\mathrm{d}S}_{\rightarrow\mathcal{S}_{22}} \underbrace{-2\int_{\Omega}\frac{\partial}{\partial x_k}\left(\Delta_a\frac{\partial\Delta}{\partial x_k}\right)\frac{\delta\Delta}{\delta b_i}\mathrm{d}\Omega}_{\rightarrow\text{FAE\_}\Delta} \underbrace{-2\int_{\Omega}\Delta_a\frac{\partial\Delta}{\partial x_k}\frac{\partial\Delta}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega}_{\rightarrow\text{SD}} \tag{3.66}$$

3.5 Field Adjoint Equations

Eliminating all volume integrals, i.e. the ensemble of terms denoted as FAE_MF, leads to the mean flow field adjoint equations, while eliminating the FAE_SA, FAE_γ, FAE_$\tilde{Re}_{\theta t}$ and FAE_Δ integrals gives rise to the adjoint Spalart–Allmaras, adjoint $\gamma-\tilde{Re}_{\theta t}$ equations (the SA-sLM2015 transition model is used) and the adjoint Hamilton–Jacobi equation. Terms denoted as *Suth* contribute to the mean flow field adjoint equations only if the Sutherland's law is used for the fluid's dynamic viscosity. The field adjoint equations read

$$R^{\Psi}_m = -A_{nmk}\frac{\partial\Psi_n}{\partial x_k} - \mathcal{K}_m + \mathcal{K}^{\text{SA}}_m + \mathcal{K}^{\gamma-\tilde{Re}_{\theta t}}_m + \mathcal{B}_m + \mathcal{B}^{\text{SA}}_m + \mathcal{B}^{\gamma-\tilde{Re}_{\theta t}}_m = 0 \tag{3.67a}$$

$$R^{\tilde{\nu}_a} = -v^A_k\frac{\partial\tilde{\nu}_a}{\partial x_k} - \mathcal{G}^{\text{SA},diff} + \mathcal{G}^{\text{SA},src} + \left[\mathcal{G}^{\mu_t} + \mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}}\right]\frac{\partial\mu_t}{\partial\tilde{\mu}} = 0 \tag{3.67b}$$

$$R^{\gamma_a} = -v^A_k\frac{\partial\gamma_a}{\partial x_k} - \mathcal{H}^{\gamma-\tilde{Re}_{\theta t},diff}_{\gamma} + \mathcal{H}^{\gamma-\tilde{Re}_{\theta t},src}_{\gamma} + \mathcal{H}^{\text{SA},src}_{\gamma} = 0 \tag{3.67c}$$

$$R^{\tilde{Re}_a} = -v^A_k\frac{\partial\tilde{Re}_a}{\partial x_k} - \mathcal{H}^{\gamma-\tilde{Re}_{\theta t},diff}_{\tilde{Re}_{\theta t}} + \mathcal{H}^{\gamma-\tilde{Re}_{\theta t},src}_{\tilde{Re}_{\theta t}} + \mathcal{H}^{\text{SA},src}_{\tilde{Re}_{\theta t}} = 0 \tag{3.67d}$$

$$R^{\Delta_a} = -2\frac{\partial}{\partial x_k}\left(\Delta_a\frac{\partial\Delta}{\partial x_k}\right) + \mathcal{M}^{\text{SA},src} + \mathcal{M}^{\gamma-\tilde{Re}_{\theta t},src} = 0 \tag{3.67e}$$

$n=1,\ldots,5$, $m=1,\ldots,5$, $k=1,\ldots,3$ where the terms $\mathcal{K}_m$, $\mathcal{K}^{\text{SA}}_m$ and $\mathcal{K}^{\gamma-\tilde{Re}_{\theta t}}_m$ result from the differentiation of the mean flow viscous terms and that of the turbulence and transition model (the SA-sLM2015 model is selected), namely

$$\mathcal{K}_m = \left(\frac{\partial\tau^{\text{adj}}_{kq}}{\partial x_k} - \tau_{kq}\frac{\partial\Psi_5}{\partial x_k}\right)\frac{\partial v^A_q}{\partial U_m} + \frac{\partial q^{\text{adj}}_k}{\partial x_k}\frac{\partial T}{\partial U_m} \tag{3.68}$$

$$\mathcal{K}_m^{\mathrm{SA}} = \mathcal{K}^{\mathrm{SA},\rho}\frac{\partial \rho}{\partial U_m} + \mathcal{K}_k^{\mathrm{SA},v_k^A}\frac{\partial v_k^A}{\partial U_m} \tag{3.69}$$

$$\mathcal{K}_m^{\gamma-\tilde{Re}_{\theta t}} = \mathcal{K}^{\gamma-\tilde{Re}_{\theta t},\rho}\frac{\partial \rho}{\partial U_m} + \mathcal{K}_k^{\gamma-\tilde{Re}_{\theta t},v_k^A}\frac{\partial v_k^A}{\partial U_m} \tag{3.70}$$

For the adjoint mean flow equations,

$$\begin{aligned}
\mathcal{K}^{\mathrm{SA},\rho} = & -\frac{\tilde{\nu_a}}{\mathrm{Re}_0\,\sigma}\left[\frac{\partial}{\partial x_k}\left[(\nu+\tilde{\nu})\frac{\partial \tilde{\nu}}{\partial x_k}\right] + c_{b_2}\frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\right] + \frac{1}{\mathrm{Re}_0\,\sigma}\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\mathcal{P}\left(\nu,\rho\right) \\
& - \frac{1}{\mathrm{Re}_0\,\sigma}\frac{\tilde{\nu}}{\rho}\left[\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\left(1+c_{b_2}\right) - \frac{\partial}{\partial x_k}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\right]\right] \\
& - \frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\frac{\tilde{\nu}}{\rho}\left[\rho\tilde{\nu_a}\frac{\partial^2 \tilde{\nu}}{\partial x_k^2} + \frac{\partial}{\partial x_k}\left(\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\right)\right] + \frac{\tilde{\nu_a}}{\mathrm{Re}_0}\left[-\left(c_{w_1}f_w - \frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\left(\frac{\tilde{\nu}}{\Delta}\right)^2\right] \\
& + \tilde{\nu_a}\left[\mathcal{C}_r\mathcal{P}\left(r,\rho\right)+\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)\right] - \tilde{\nu_a}\frac{\gamma}{\rho}\mathcal{C}_{\gamma_{eff}}
\end{aligned} \tag{3.71}$$

$$\mathcal{K}_k^{\mathrm{SA},v_k^A} = -\tilde{\mu}\frac{\partial \tilde{\nu_a}}{\partial x_k} - \frac{\partial}{\partial x_\ell}\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\mathrm{SA}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\right] - \frac{\partial}{\partial x_m}\left[\tilde{\nu_a}\frac{\mathcal{C}_S^{\mathrm{SA}}}{S}2S_{km}\right]$$

and

$$\begin{aligned}
\mathcal{K}^{\gamma-\tilde{Re}_{\theta t},\rho} = & \frac{1}{\mathrm{Re}_0}\frac{\gamma}{\rho}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial \gamma_a}{\partial x_k}\right] + \frac{1}{\mathrm{Re}_0}\frac{\tilde{Re}_{\theta t}}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial \tilde{Re}_a}{\partial x_k}\right] \\
& -\gamma_a c_{\alpha_1}F_{length}F_{onset}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1-c_{\epsilon_1}\gamma\right) \\
& +\frac{1}{2}\gamma\gamma_a c_{\alpha_1}F_{length}F_{onset}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left({}^1\!/\!_\gamma-3c_{\epsilon_1}\right) \\
& -\gamma_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{length}}\mathcal{P}\left(F_{length},F_{length1}\right)\mathcal{P}\left(F_{length1},\tilde{Re}_{\theta t}\right) \\
& +\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma F_{turb}\left(c_{\epsilon_2}\gamma-1\right) \\
& -\gamma\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right) \\
& -\gamma_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{Re}_{\theta t}\right) \\
& -\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}\left(Re_{\theta,t}^{eq}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)+\rho\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re_{\theta,t}^{eq}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\rho\right)
\end{aligned}$$

$$
\begin{aligned}
&-\tilde{Re}_a\tilde{Re}_{\theta t}\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)-\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\\
&+\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},\rho\right)\\
&+\tilde{Re}_{\theta t}\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big),\tilde{Re}_{\theta t}\right)F_{\theta t}\\
&+\tilde{Re}_a\mathcal{C}_{F_{\theta t}}\left[\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,\rho\right)+\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\rho\right)\right]\\
&-\tilde{Re}_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{Re}_{\theta t}\right)\\
&+\tilde{Re}_a\mathcal{C}_{Re^{eq}_{\theta,t}}\left[\mathcal{P}\left(\lambda_\theta,\rho\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\rho\right)\right]-\tilde{Re}_a\mathcal{C}_{Re_{SCF}}\frac{0.82\mu}{\rho^2\left|\mathbf{U}\right|}
\end{aligned}
\tag{3.72}
$$

$$
\begin{aligned}
\mathcal{K}_k^{\gamma-\tilde{Re}_{\theta t},v_k^A}=&-\rho\left(\frac{\partial\gamma_a}{\partial x_k}\gamma+\frac{\partial\tilde{Re}_a}{\partial x_k}\tilde{Re}_{\theta t}\right)+\rho\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\left|\mathbf{U}\right|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\\
&+\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},\left|\mathbf{U}\right|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\\
&+\tilde{Re}_a\left[\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t,}\delta\right)\mathcal{P}\left(\delta,\left|\mathbf{U}\right|\right)+\mathcal{C}_{Re^{eq}_{\theta,t}}\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\left|\mathbf{U}\right|\right)\right]\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\\
&+\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\left[-\frac{2}{\left|\mathbf{U}\right|}\frac{d\left|\mathbf{U}\right|}{ds}\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}+\frac{v_m^A}{\left|\mathbf{U}\right|^2}\left(\frac{\partial v_k^A}{\partial x_m}+\frac{\partial v_m^A}{\partial x_k}\right)\right]-\frac{\partial}{\partial x_m}\left[\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v_k^Av_m^A}{\left|\mathbf{U}\right|^2}\right]\\
&-\tilde{Re}_a\mathcal{C}_{Re_{SCF}}\frac{0.82\mu}{\rho\left|\mathbf{U}\right|^2}\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}+\tilde{Re}_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},H_{CF}\right)\mathcal{P}\left(H_{CF},\left|\mathbf{U}\right|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\\
&-\frac{\partial}{\partial x_q}\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_m^A}{\left|\mathbf{U}\right|}\varepsilon_{mqk}\right]+\tilde{Re}_a\frac{\mathcal{C}_{\zeta_{streamwise}}}{\left|\mathbf{U}\right|}\frac{\partial v_m^A}{\partial x_q}\left[\varepsilon_{kqm}+\frac{u_r}{\left|\mathbf{U}\right|}\varepsilon_{rqm}\frac{\partial\left|\mathbf{U}\right|}{\partial v_k^A}\right]\\
&-\frac{\partial}{\partial x_\ell}\left[\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\right]-\frac{\partial}{\partial x_m}\left[\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\right]
\end{aligned}
$$

Terms $\mathcal{B}_m,\mathcal{B}_m^{\mathrm{SA}}$ and $\mathcal{B}_m^{\gamma-\tilde{Re}_{\theta t}}$ are present only if the Sutherland's law is differentiated.

Regarding the field adjoint equation to the Spalart–Allmaras model, $\mathcal{G}^{\mathrm{SA},diff}$ and $\mathcal{G}^{\mathrm{SA},src}$ are from the differentiation of the diffusion and source terms of this model. $\mathcal{G}^{\mu_t}$ and $\mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}}$ include the contribution of the mean flow and $\gamma-\tilde{Re}_{\theta t}$ to μ_t. These are expressed as

$$
\begin{aligned}
\mathcal{G}^{\mathrm{SA},diff}=&-\frac{1}{\mathrm{Re}_0}\frac{1}{\sigma}\frac{1}{\rho}\left[\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\left(1+c_{b_2}\right)-\frac{\partial}{\partial x_k}\left\{\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\right\}\right]\\
&-\frac{c_{b_2}}{\mathrm{Re}_0}\frac{1}{\sigma}\frac{1}{\rho}\left[\rho\tilde{\nu_a}\frac{\partial^2\tilde{\nu}}{\partial x_k^2}+\frac{\partial}{\partial x_k}\left(\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\right)\right]
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{G}^{\mathrm{SA},src} &= \tilde{\nu_a}\left[-\gamma c_{b_1}\tilde{S}+\frac{2}{\mathrm{Re}_0}c_{w_1}f_w\frac{\tilde{\nu}}{\Delta^2}\right]+\tilde{\nu_a}\mathcal{C}_r\mathcal{P}\left(r,\tilde{\mu}\right)+\frac{\tilde{\nu_a}}{\rho}\mathcal{C}_{\tilde{S}}\left[\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)+\mathcal{C}_\chi\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\\
\mathcal{G}^{\mu_t} &= \frac{\tau_{km}}{\mu+\mu_t}\left(\frac{\partial\Psi_{m+1}}{\partial x_k}+\frac{\partial\Psi_5}{\partial x_k}v_m^A\right)+\frac{\mathcal{C}_p}{\mathrm{Re}_0}\frac{\partial\Psi_5}{\partial x_k}\frac{\partial T}{\partial x_k}\frac{1}{\mathrm{Pr_t}}\\
\mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}} &= \frac{1}{\mathrm{Re}_0}\left(\frac{\partial\gamma_a}{\partial x_k}\frac{\partial\gamma}{\partial x_k}\frac{1}{\sigma_f}+\frac{\partial\tilde{Re}_a}{\partial x_k}\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\sigma_{\theta,t}\right)+\mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}}
\end{aligned}
\tag{3.73}
$$

In the field adjoint equation to the $\gamma-\tilde{Re}_{\theta t}$ model, terms $\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},diff}/\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},diff}$ $\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},src}/\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},src}$ and $\mathcal{H}_\gamma^{\mathrm{SA},src}/\mathcal{H}_{\tilde{Re}_{\theta t}}^{\mathrm{SA},src}$ are coming from the differentiation of the $\gamma-\tilde{Re}_{\theta t}$ diffusion and source terms and the Spalart–Allmaras source terms. These are defined as

$$
\begin{aligned}
\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},diff} &= \frac{1}{\mathrm{Re}_0}\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\right]\\
\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},src} &= -\frac{1}{2}\gamma_a c_{\alpha_1}F_{length}F_{onset}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left({}^1\!/\!_\gamma-3c_{\epsilon_1}\right)\\
&\quad+\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\\
\mathcal{H}_\gamma^{\mathrm{SA},src} &= -\tilde{\nu_a}c_{b_1}\tilde{\nu}\tilde{S}\\
\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},diff} &= \frac{1}{\mathrm{Re}_0}\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\right]\\
\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},src} &= \gamma_a\frac{1}{\rho}\mathcal{C}_{F_{length}}\mathcal{P}\left(F_{length},F_{length1}\right)\mathcal{P}\left(F_{length1},\tilde{Re}_{\theta t}\right)\\
&\quad+\gamma_a\frac{1}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{Re}_{\theta t}\right)+\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\\
&\quad-\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big),\tilde{Re}_{\theta t}\right)F_{\theta t}\\
&\quad+\frac{\tilde{Re}_a}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{Re}_{\theta t}\right)
\end{aligned}
\tag{3.74}
$$

$$
\mathcal{H}_{\tilde{Re}_{\theta t}}^{\mathrm{SA},src} = 0 \tag{3.75}
$$

Regarding the adjoint Hamilton–Jacobi equation, terms $\mathcal{M}^{\mathrm{SA},src}$ and $\mathcal{M}^{\gamma-\tilde{Re}_{\theta t},src}$ (which come from the differentiation of the Spalart–Allmaras turbulence and the

SA-sLM2015 transition model, respectively) read

$$\begin{aligned}
\mathcal{M}^{\mathrm{SA},src} &= \tilde{\nu_a}\left[\mathcal{C}_r\mathcal{P}\left(r,\Delta\right) - 2\frac{\rho}{\mathrm{Re}_0}c_{w_1}f_w\frac{\tilde{\nu}^2}{\Delta^3}\right] + \tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\Delta\right) \\
\mathcal{M}^{\gamma-\tilde{Re}_{\theta t},src} &= \left[\gamma_a\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_S\right) + \tilde{Re}_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\right]\mathcal{P}\left(Re_S,\Delta\right) \\
&\quad + \tilde{Re}_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},H_{CF}\right)\mathcal{P}\left(H_{CF},\Delta\right) \qquad (3.76)
\end{aligned}$$

A numerical solution procedure similar to the one used to solve the flow equations can be followed to solve Eqs. 3.67. It should also be mentioned that FAE_MF, FAE_SA, FAE_γ and FAE_$\tilde{Re}_{\theta t}$ do not depend on the adjoint distance variable Δ_a; thus, FAE_Δ is numerically solved after solving the aforementioned adjoint PDEs and contributes to the computation of SDs. Moreover, the l.h.s. terms of the system of discretized equations can be computed a priori, as they depend only on flow quantities related to the primal problem which are known. As a result, only the r.h.s. that depends on adjoint variables has to be recomputed during the numerical solution. Since the r.h.s. is less memory demanding, the two-kernel scheme described in Sec. 2.5.1 can be employed.

3.6 Adjoint Boundary Conditions

All volume integrals resulting from the differentiation of J_{aug} have been treated (giving rise to the FAE) and only the surface integrals remain. These integrals ($\mathcal{S}_1 - \mathcal{S}_{22}$) may contain derivatives in geometric quantities w.r.t. b_i or in the flow variables along the boundaries; the former contribute directly to the sensitivity derivatives, while, the latter must be eliminated giving rise to the adjoint boundary conditions. Moreover, the primal flow boundary conditions should be taken into account during this treatment, meaning that the derivatives of all imposed quantities w.r.t. b_i are zero. The remaining flow quantity variations are grouped and their multiplier is set to zero.

Here, the development of the adjoint boundary conditions is done considering no–slip stationary boundaries. For the rotating and/or slip wall boundaries, the interested reader should refer to the PhD Theses of K. Tsiakas [135] and M. Monfaredi [98]. Wall boundaries, inlet, outlet and far–field boundaries are considered in the following sections; after their development, terms marked as $\rightarrow$ABC contribute to the adjoint boundary conditions.

3.6.1 Wall Boundaries

Along stationary no–slip walls, the flow boundary condition is expressed as $v_k^A = 0$, (k=1,2,3) leading to $\frac{\delta v_k^A}{\delta b_i} = 0$. For the Spalart–Allmaras and $\gamma - \tilde{Re}_{\theta t}$, $\tilde{\nu} = 0$, $\frac{\partial \gamma}{\partial x_k} \mathsf{n}_k = 0$ and $\frac{\partial \tilde{Re}_{\theta t}}{\partial x_k} \mathsf{n}_k = 0$ are imposed, respectively.

Surface integrals $\mathcal{S}_3$, $\mathcal{S}_{10}$, $\mathcal{S}_{11}$, $\mathcal{S}_{18}$, $\mathcal{S}_{19}$, $\mathcal{S}_{20}$ and $\mathcal{S}_{21}$ vanish since $\frac{\delta v_k^A}{\delta b_i} = 0$, $\mathcal{S}_8$ vanishes since $\tilde{\nu} = 0$, while $\mathcal{S}_7$, $\mathcal{S}_9$ vanish since $\frac{\delta \tilde{\nu}}{\delta b_i} = 0$. Integrals $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_4$ are expanded as follows

$$
\begin{aligned}
\mathcal{S}_1 &= \int_{S_{\text{NoSlip}}} \Psi_n \frac{\delta\left(f_{nk}^{\text{inv}} \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} - \int_{S_{\text{NoSlip}}} \Psi_n f_{nk}^{\text{inv}} \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} \\
&= \int_{S_{\text{NoSlip}}} \Psi_1 \frac{\delta\left(f_{1k}^{\text{inv}} \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} + \int_{S_{\text{NoSlip}}} \Psi_{m+1} \frac{\delta\left(f_{(m+1)k}^{\text{inv}} \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} + \int_{S_{\text{NoSlip}}} \Psi_5 \frac{\delta\left(f_{5k}^{\text{inv}} \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} - \int_{S_{\text{NoSlip}}} \Psi_n f_{nk}^{\text{inv}} \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} \\
&= \cancel{\int_{S_{\text{NoSlip}}} \Psi_1 \frac{\delta\left(\rho v_k^A \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}} + \cancel{\int_{S_{\text{NoSlip}}} \Psi_{m+1} \frac{\delta\left(\rho v_m^A v_k^A \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}} + \underbrace{\int_{S_{\text{NoSlip}}} \Psi_{m+1} \mathsf{n}_m \frac{\delta p}{\delta b_i} \mathrm{d}S}_{\to \text{ABC}^p} \\
&\quad + \underbrace{\int_{S_{\text{NoSlip}}} \Psi_{k+1} p \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}}_{\to \text{SD}} + \cancel{\int_{S_{\text{NoSlip}}} \Psi_5 \frac{\delta\left[\rho h_t v_k^A \mathsf{n}_k \mathrm{d}S\right]}{\delta b_i}} - \underbrace{\int_{S_{\text{NoSlip}}} \Psi_n f_{nk}^{\text{inv}} \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}}_{\to \text{SD}}
\end{aligned}
\tag{3.77}
$$

$$
\begin{aligned}
\mathcal{S}_2 &= -\int_{S_{\text{NoSlip}}} \Psi_n \frac{\delta\left(f_{nk}^{\text{vis}} \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} + \int_{S_{\text{NoSlip}}} \Psi_n f_{nk}^{\text{vis}} \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i} \\
&\quad \underbrace{- \int_{S_{\text{NoSlip}}} \Psi_{m+1} \mathsf{n}_m \frac{\delta\left(\tau_{\ell k} \mathsf{n}_k \mathsf{n}_\ell\right)}{\delta b_i} \mathrm{d}S}_{\to \text{ABC}^{\left(\tau_{\ell k} \mathsf{n}_k \mathsf{n}_\ell\right)}} \underbrace{- \int_{S_{\text{NoSlip}}} \Psi_{m+1} \mathsf{n}_m \frac{\delta\left(\tau_{\ell k} \mathsf{n}_k \mathsf{t}_\ell\right)}{\delta b_i} \mathrm{d}S}_{\to \text{ABC}^{\left(\tau_{\ell k} \mathsf{n}_k \mathsf{t}_\ell\right)}} \\
&\quad \underbrace{- \int_{S_{\text{NoSlip}}} \Psi_{q+1} \mathsf{t}_q \tau_{\ell k} \mathsf{n}_k \mathsf{t}_\ell \mathsf{t}_m \frac{\delta\left(\mathsf{t}_m \mathrm{d}S\right)}{\delta b_i} - \int_{S_{\text{NoSlip}}} \Psi_{m+1} \tau_{\ell k} \mathsf{n}_k \mathsf{n}_\ell \frac{\delta\left(\mathsf{n}_m \mathrm{d}S\right)}{\delta b_i} + \int_{S_{\text{NoSlip}}} \Psi_{q+1} \mathsf{n}_q \tau_{\ell k} \mathsf{n}_k \mathsf{t}_\ell \mathsf{t}_m \frac{\delta(\mathsf{n}_m \mathrm{d}S)}{\delta b_i}}_{\to \text{SD}} \\
&\quad \cancel{- \int_{S_{\text{NoSlip}}} \Psi_5 \frac{\delta\left(v_\ell^A \tau_{\ell k} \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}} \underbrace{- \int_{S_{\text{NoSlip}}} \Psi_5 \frac{\delta\left(q_k \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}}_{\to \text{ABC}^{\left(q_k \mathsf{n}_k\right)}} \underbrace{+ \int_{S_{\text{NoSlip}}} \Psi_n f_{nk}^{\text{vis}} \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}}_{\to \text{SD}}
\end{aligned}
\tag{3.78}
$$

$$\mathcal{S}_3 = \cancel{\int_{S_{\text{NoSlip}}} \tau_{mk}^{\text{adj}} \mathsf{n}_k \frac{\delta v_m^A}{\delta b_i} \mathrm{d}S} \tag{3.79}$$

$$\mathcal{S}_4 = \begin{cases} \underbrace{\int_{S_{\text{NoSlip}}} q_k^{\text{adj}} \mathsf{n}_k \frac{\delta T}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^T} & \text{, for adiabatic or constant heat flux wall.} \\ \cancel{\int_{S_{\text{NoSlip}}} q_k^{\text{adj}} \mathsf{n}_k \frac{\delta T}{\delta b_i} \mathrm{d}S} & \text{, for constant temperature wall.} \end{cases} \tag{3.80}$$

Surface integrals coming from the differentiation of the Spalart–Allmaras model are written as

$$\mathcal{S}_5 = \cancel{\int_{S_{\text{NoSlip}}} \tilde{\nu_a} \frac{\delta\left(\tilde{\mu} v_k^A \mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}} \cancel{- \int_{S_{\text{NoSlip}}} \tilde{\nu_a} \tilde{\mu} v_k^A \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}} \tag{3.81}$$

$$\begin{aligned} \mathcal{S}_6 = & \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \rho \tilde{\nu_a} \left[\nu + (1+c_{b_2})\,\tilde{\nu}\right] \frac{\delta}{\delta b_i} \left(\frac{\partial \tilde{\nu}}{\partial x_k} \mathsf{n}_k \mathrm{d}S\right)}_{\rightarrow \text{ABC}\left(\frac{\partial \tilde{\nu}}{\partial x_k} \mathsf{n}_k\right)} \\ & \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \rho \tilde{\nu_a} \frac{\partial \tilde{\nu}}{\partial x_k} \mathsf{n}_k \mathcal{P}\left(\nu, \rho\right) \mathcal{P}\left(\rho, p\right) \frac{\delta p}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^p} \\ & \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \rho \tilde{\nu_a} \frac{\partial \tilde{\nu}}{\partial x_k} \mathsf{n}_k \left[\mathcal{P}\left(\nu, \mu\right) \mathcal{P}\left(\mu, T\right) + \mathcal{P}\left(\nu, \rho\right) \mathcal{P}\left(\rho, T\right)\right] \frac{\delta T}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^T} \\ & \cancel{-\frac{1}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \rho \tilde{\nu_a} \frac{\partial \tilde{\nu}}{\partial x_k} \mathsf{n}_k \left(1+c_{b_2}\right) \frac{\delta \tilde{\nu}}{\delta b_i} \mathrm{d}S} \\ & \underbrace{+\frac{1}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \rho \tilde{\nu_a} \left[\nu + (1+c_{b_2})\,\tilde{\nu}\right] \frac{\partial \tilde{\nu}}{\partial x_k} \frac{\delta\left(\mathsf{n}_k \mathrm{d}S\right)}{\delta b_i}}_{\rightarrow \text{SD}} \end{aligned} \tag{3.82}$$

$$\mathcal{S}_7 = \cancel{\int_{S_{\text{NoSlip}}} \frac{1}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \frac{\partial\left(\rho \tilde{\nu_a}\right)}{\partial x_k} \left[\nu + (1+c_{b_2})\,\tilde{\nu}\right] \mathsf{n}_k \frac{\delta \tilde{\nu}}{\delta b_i} \mathrm{d}S} \tag{3.83}$$

$$\mathcal{S}_8 = \cancel{\int_{S_{\text{NoSlip}}} \frac{c_{b_2}}{\mathrm{Re}_0\,\sigma} \int_{S_{\text{NoSlip}}} \tilde{\nu_a} \tilde{\mu} \mathsf{n}_k \frac{\delta}{\delta b_i} \left(\frac{\partial \tilde{\nu}}{\partial x_k}\right) \mathrm{d}S} \tag{3.84}$$

$$\mathcal{S}_9 = -\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int\limits_{S_{\mathrm{NoSlip}}} \frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\mathsf{n}_k\frac{\delta\tilde{\nu}}{\delta b_i}\mathrm{d}S \tag{3.85}$$

$$\mathcal{S}_{10} = \int\limits_{S_{\mathrm{NoSlip}}} \tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\mathrm{SA}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_r^A}{\partial x_q}\mathsf{n}_\ell\frac{\delta v_m^A}{\delta b_i}\mathrm{d}S \tag{3.86}$$

$$\mathcal{S}_{11} = \int\limits_{S_{\mathrm{NoSlip}}} \tilde{\nu_a}\frac{\mathcal{C}_S^{SA}}{S}2S_{km}\mathsf{n}_m\frac{\delta v_k^A}{\delta b_i}\mathrm{d}S \tag{3.87}$$

Finally, the surface integrals arising from the differentiation of the $\gamma-\tilde{Re}_{\theta t}$ transition model are treated as

$$\mathcal{S}_{12} = \int\limits_{S_{\mathrm{NoSlip}}} \gamma_a\frac{\delta\left(\rho\gamma v_k^A\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i} - \int\limits_{S_{\mathrm{NoSlip}}} \gamma_a\rho\gamma v_k^A\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i} \tag{3.88}$$

$$\mathcal{S}_{13} = \int\limits_{S_{\mathrm{NoSlip}}} \tilde{Re}_a\frac{\delta\left(\rho\tilde{Re}_{\theta t}v_k^A\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i} - \int\limits_{S_{\mathrm{NoSlip}}} \tilde{Re}_a\rho\tilde{Re}_{\theta t}v_k^A\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i} \tag{3.89}$$

$$\mathcal{S}_{14} = -\int\limits_{S_{\mathrm{NoSlip}}} \gamma_a\frac{\delta}{\delta b_i}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\mathsf{n}_k\mathrm{d}S\right] + \underbrace{\int\limits_{S_{\mathrm{NoSlip}}} \gamma_a\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}}_{\to\mathrm{SD}} \tag{3.90}$$

$$\mathcal{S}_{15} = \underbrace{\int\limits_{S_{\mathrm{NoSlip}}} \left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\mathsf{n}_k\frac{\delta\gamma}{\delta b_i}\mathrm{d}S}_{\to\mathrm{ABC}^\gamma} \tag{3.91}$$

$$\mathcal{S}_{16} = -\int\limits_{S_{\mathrm{NoSlip}}} \tilde{Re}_a\frac{\delta}{\delta b_i}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\mathsf{n}_k\mathrm{d}S\right] + \underbrace{\int\limits_{S_{\mathrm{NoSlip}}} \tilde{Re}_a\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}}_{\to\mathrm{SD}} \tag{3.92}$$

$$\mathcal{S}_{17} = \underbrace{\int\limits_{S_{\mathrm{NoSlip}}} \sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\mathsf{n}_k\frac{\delta\tilde{Re}_{\theta t}}{\delta b_i}\mathrm{d}S}_{\to\mathrm{ABC}^{\tilde{Re}_{\theta t}}} \tag{3.93}$$

$$\mathcal{S}_{18} = \int\limits_{S_{\mathrm{NoSlip}}} \tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v_k^A v_m^A}{|\mathbf{U}|^2}\mathsf{n}_m\frac{\delta v_k^A}{\delta b_i}\mathrm{d}S \tag{3.94}$$

$$\mathcal{S}_{18} = \int\limits_{\partial\Omega} \tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_k^A}{|\mathbf{U}|}\varepsilon_{k\ell m}\mathsf{n}_\ell\frac{\delta v_m^A}{\delta b_i}\mathrm{d}S \tag{3.95}$$

$$\mathcal{S}_{20} = \int_{S_{\text{NoSlip}}} \frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta} \varepsilon_{k\ell m}\varepsilon_{kqr} \frac{\partial v_r^A}{\partial x_q} \mathsf{n}_\ell \frac{\delta v_m^A}{\delta b_i} \mathrm{d}S \tag{3.96}$$

$$\mathcal{S}_{21} = \int_{\partial\Omega} \frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S} 2S_{km}\mathsf{n}_m \frac{\delta v_k^A}{\delta b_i} \mathrm{d}S \tag{3.97}$$

$$\mathcal{S}_{22} = 2\int_{\partial\Omega} \Delta_a \frac{\partial \Delta}{\partial x_k} \mathsf{n}_k \frac{\delta \Delta}{\delta b_i} \mathrm{d}S \tag{3.98}$$

Eliminating terms denotes as ABC^p, the following relation arises

$$\Psi_{m+1}\mathsf{n}_m - \frac{1}{\mathrm{Re}_0\,\sigma}\rho\tilde{\nu_a}\frac{\partial\tilde{\nu}}{\partial x_m}\mathsf{n}_m\mathcal{P}\left(\nu,\rho\right)\mathcal{P}\left(\rho,p\right) + \left.\frac{\partial J_S}{\partial p}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} = 0 \tag{3.99}$$

The elimination of the $\text{ABC}^{\left(\tau_{\ell k}\mathsf{n}_k\mathsf{n}_\ell\right)}$ and $\text{ABC}^{\left(\tau_{\ell k}\mathsf{n}_k\mathsf{t}_\ell\right)}$ terms leads to

$$-\Psi_{m+1}\mathsf{n}_m + \left.\frac{\partial J_S}{\partial\left(\tau_{k\ell}\mathsf{n}_k\mathsf{n}_\ell\right)}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} = 0 \tag{3.100}$$

$$-\Psi_{m+1}\mathsf{t}_m + \left.\frac{\partial J_S}{\partial\left(\tau_{k\ell}\mathsf{n}_k\mathsf{t}_\ell\right)}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} = 0 \tag{3.101}$$

Similarly, for the elimination of terms denoted as $\text{ABC}^{\left(\frac{\partial\tilde{\nu}}{\partial x_k}\mathsf{n}_k\right)}$, ABC^{γ} and $\text{ABC}^{\tilde{Re}_{\theta t}}$ the following relations

$$-\frac{1}{\mathrm{Re}_0\,\sigma}\rho\tilde{\nu_a}\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right] = 0 \Rightarrow \tilde{\nu_a} = 0 \tag{3.102}$$

$$\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\mathsf{n}_k = 0 \Rightarrow \frac{\partial\gamma_a}{\partial x_k}\mathsf{n}_k = 0 \tag{3.103}$$

$$\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\mathsf{n}_k = 0 \Rightarrow \frac{\partial\tilde{Re}_a}{\partial x_k}\mathsf{n}_k = 0 \tag{3.104}$$

must hold. Dirichlet and zero Neumann adjoint boundary conditions are imposed for $\tilde{\nu_a}$, γ_a and $\tilde{Re}_a$, respectively. Since $\Delta = 0$ at wall, $\mathcal{S}_{22}$ is set to zero. In order to obtain a single boundary condition for $\Psi_{m+1}\mathsf{n}_m$, importing Eq. 3.102 into Eq. 3.99 and combining it with Eq. 3.100, the objective function should satisfy the

following

$$\left.\frac{\partial J_S}{\partial p}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} + \left.\frac{\partial J_S}{\partial\left(\tau_{k\ell}\mathsf{n}_k\mathsf{n}_\ell\right)}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} = 0 \tag{3.105}$$

For stationary no-slip adiabatic or constant heat flux walls, where $\frac{\delta\left(q_k\mathsf{n}_k\right)}{\delta b_i} = 0$, terms $\text{ABC}^{\left(q_k\mathsf{n}_k\right)}$ vanish. The elimination of the ABC^T terms leads to the following thermal adjoint boundary condition

$$q_k^{\text{adj}}\mathsf{n}_k - \frac{1}{\text{Re}_0\,\sigma}\rho\tilde{\nu_a}\frac{\partial\tilde{\nu}}{\partial x_k}\mathsf{n}_k\left[\mathcal{P}\left(\nu,\mu\right)\mathcal{P}\left(\mu,T\right)+\mathcal{P}\left(\nu,\rho\right)\mathcal{P}\left(\rho,T\right)\right] + \left.\frac{\partial J_S}{\partial T}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} = 0 \tag{3.106}$$

For stationary no-slip constant temperature walls, where $\frac{\delta T}{\delta b_i} = 0$, the ABC^T terms vanish and the thermal adjoint boundary conditions arise from the elimination of the $\text{ABC}^{\left(q_k\mathsf{n}_k\right)}$ terms, namely

$$-\Psi_5 + \left.\frac{\partial\left(J_S\right)}{\partial\left(q_k\mathsf{n}_k\right)}\right|_{S_{\text{NoSlip}}^{\text{Obj}}} = 0 \tag{3.107}$$

3.6.2 Inlet and Outlet Boundaries

For the inlet and outlet boundaries, surface integrals that arise from the differentiation of the viscous terms are neglected, since the total variation in the viscous stresses and heat flux is zero. Inlet and outlet conditions are presented for subsonic cases. The interested reader must refer to the PhD Thesis of K. Tsiakas, [135], for supersonic cases.

Inlet Boundaries

For inlet boundaries ($S_{I^{\text{sub}}}$), a set of local flow quantities ($V_\ell^{\text{loc}}, \ell = 1,\ldots,5$) is defined; the value of one of them, say V_5^{loc}, is extrapolated from the interior domain while, for the rest, Dirichlet conditions are imposed, and thus $\frac{\delta V_\ell^{\text{loc}}}{\delta b_i} = 0, \ell = 1,\ldots,4$. So

$$\mathcal{S}_1 = \int_{S_{I^{\text{sub}}}} \Psi_n A_{nmk}\mathsf{n}_k \frac{\partial U_m}{\partial V_1^{\text{loc}}}\frac{\delta V_1^{\text{loc}}}{\delta b_i}\mathrm{d}S + \int_{S_{I^{\text{sub}}}} \Psi_n A_{nmk}\mathsf{n}_k \frac{\partial U_m}{\partial V_2^{\text{loc}}}\frac{\delta V_2^{\text{loc}}}{\delta b_i}\mathrm{d}S + \int_{S_{I^{\text{sub}}}} \Psi_n A_{nmk}\mathsf{n}_k \frac{\partial U_m}{\partial V_3^{\text{loc}}}\frac{\delta V_3^{\text{loc}}}{\delta b_i}\mathrm{d}S$$

$$+\cancel{\int_{S_{I^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_4^{\text{loc}}} \frac{\delta V_4^{\text{loc}}}{\delta b_i} \mathrm{d}S} + \underbrace{\int_{S_{I^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_5^{\text{loc}}} \frac{\delta V_5^{\text{loc}}}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^{V_5^{\text{loc}}}} \tag{3.108}$$

Eliminating term $\text{ABC}^{V_5^{\text{loc}}}$ leads to the following equation

$$\Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_5^{\text{loc}}} + \frac{\partial J}{\partial V_5^{\text{loc}}}\bigg|_{S_{I^{\text{sub}}}^{\text{Obj}}} = 0 \tag{3.109}$$

Taking the Dirichlet boundary condition imposed for $\tilde{\nu}$, γ and $\tilde{Re}_{\theta t}$ into account and imposing $\tilde{\nu}_a = 0$, $\gamma_a = 0$ and $\tilde{Re}_a = 0$, the rest terms ($\mathcal{S}_5 - \mathcal{S}_{21}$) coming from the turbulent and transition model vanish. In order to eliminate $\mathcal{S}_{22}$, $\Delta_a = 0$.

Outlet Boundaries

At the outlet boundaries ($S_{O^{\text{sub}}}$), a Dirichlet condition is imposed for V_5^{loc}, so $\frac{\delta V_5^{\text{loc}}}{\delta b_i} = 0$. Term $\mathcal{S}_1$ is expanded as

$$\mathcal{S}_1 = \underbrace{\int_{S_{O^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_1^{\text{loc}}} \frac{\delta V_1^{\text{loc}}}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^{V_1^{\text{loc}}}} + \underbrace{\int_{S_{O^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_2^{\text{loc}}} \frac{\delta V_2^{\text{loc}}}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^{V_2^{\text{loc}}}} + \underbrace{\int_{S_{O^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_3^{\text{loc}}} \frac{\delta V_3^{\text{loc}}}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^{V_3^{\text{loc}}}}$$
$$+ \underbrace{\int_{S_{O^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_4^{\text{loc}}} \frac{\delta V_4^{\text{loc}}}{\delta b_i} \mathrm{d}S}_{\rightarrow \text{ABC}^{V_4^{\text{loc}}}} + \cancel{\int_{S_{O^{\text{sub}}}} \Psi_n A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_5^{\text{loc}}} \frac{\delta V_5^{\text{loc}}}{\delta b_i} \mathrm{d}S} \tag{3.110}$$

Eliminating the $\text{ABC}^{V_\ell^{\text{loc}}}, (\ell = 1, \ldots, 4)$ terms, the following conditions result

$$\Psi_n \mathsf{n}_k A_{nmk} \mathsf{n}_k \frac{\partial U_m}{\partial V_\ell^{\text{loc}}} + \frac{\partial J}{\partial V_\ell^{\text{loc}}}\bigg|_{S_{O^{\text{sub}}}^{\text{Obj}}} = 0 \tag{3.111}$$

Eq. 3.111 is taken into account in the computation of the adjoint flux crossing the outlet bounary. A zero Neumann condition is imposed on $\tilde{\nu}$, γ and $\tilde{Re}_{\theta t}$, which means that $\frac{\delta}{\delta b_i}\left(\frac{\partial \tilde{\nu}}{\partial x_k}\mathsf{n}_k\right) = 0$, $\frac{\delta}{\delta b_i}\left(\frac{\partial \gamma}{\partial x_k}\mathsf{n}_k\right) = 0$ and $\frac{\delta}{\delta b_i}\left(\frac{\partial \tilde{Re}_{\theta t}}{\partial x_k}\mathsf{n}_k\right) = 0$. The remaining integrals from the development the turbulence and transition models are eliminated by setting the multiplier of $\frac{\delta \tilde{\nu}}{\delta b_i}$, $\frac{\delta \gamma}{\delta b_i}$ and $\frac{\delta \tilde{Re}_{\theta t}}{\delta b_i}$ equal to zero. In order to eliminate $\mathcal{S}_{22}$, $\Delta_a = 0$.

3.6.3 Far–Field Boundaries

For the far–field boundaries, the total variation in the viscous stresses and heat flux is assumed to be zero; thus, the surface integrals arising from the differentiation of the viscous terms are neglected. Moreover, the far–field boundary remains fixed during the optimization, zeroing out derivatives of geometric quantities w.r.t. the design variables. According to the treatment that was followed for the primal problem, far–field boundaries are treated by combining the inlet and outlet boundary conditions. Once the flow and adjoint fields are obtained, all terms denoted by SD are used to compute the sensitivity derivatives for use in the gradient-based optimization.

3.7 Expression of the Sensitivity Derivatives

After satisfying the field adjoint equations and their boundary conditions, the remaining field and surface integrals comprise the formula of the gradient of J. The final expression of the SDs reads

$$\frac{\delta J}{\delta b_i} = \mathcal{I}^{\text{SD}}_{\text{MF}} + \mathcal{I}^{\text{SD}}_{\text{SA}} + \mathcal{I}^{\text{SD}}_{\gamma\text{–}\tilde{R}e_{\theta t}} + \mathcal{I}^{\text{SD}}_{\Delta} \tag{3.112}$$

where $\mathcal{I}^{\text{SD}}_{\text{MF}}$, $\mathcal{I}^{\text{SD}}_{\text{SA}}$, $\mathcal{I}^{\text{SD}}_{\gamma\text{–}\tilde{R}e_{\theta t}}$ and $\mathcal{I}^{\text{SD}}_{\Delta}$ are integrals arising from the differentiation of the mean flow equations, the Spalart–Allmaras and the SA-sLM2015 models, and the Hamilton–Jacobi equations; these are defined as

$$\begin{aligned}
\mathcal{I}^{\text{SD}}_{\text{MF}} = &-\int_{\Omega}\left[\Psi_n\left(\frac{\partial f^{inv}_{nk}}{\partial x_\ell}-\frac{\partial f^{vis}_{nk}}{\partial x_\ell}\right)+\left(\tau^{\text{adj}}_{mk}\frac{\partial v^A_m}{\partial x_\ell}+q^{\text{adj}}_k\frac{\partial T}{\partial x_\ell}\right)\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega \\
&-\int_{S_{\text{NoSlip}}}\Psi_n\left(f^{\text{inv}}_{nk}-f^{\text{vis}}_{nk}\right)\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}+\int_{S_{\text{NoSlip}}}\Psi_{k+1}p\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}-\int_{S_{\text{NoSlip}}}\Psi_{m+1}\tau_{\ell k}\mathsf{n}_k\mathsf{n}_\ell\frac{\delta\left(\mathsf{n}_m\mathrm{d}S\right)}{\delta b_i} \\
&-\int_{S_{\text{NoSlip}}}\Psi_{q+1}\mathsf{t}_q\tau_{\ell k}\mathsf{n}_k\mathsf{t}_\ell\mathsf{t}_m\frac{\delta\left(\mathsf{t}_m\mathrm{d}S\right)}{\delta b_i}+\int_{S_{\text{NoSlip}}}\Psi_{q+1}\mathsf{n}_q\tau_{\ell k}\mathsf{n}_k\mathsf{t}_\ell\mathsf{t}_m\frac{\delta(\mathsf{n}_m\mathrm{d}S)}{\delta b_i} \\
\mathcal{I}^{\text{SD}}_{\text{SA}} = &-\int_{\Omega}\tilde{\nu}_a\frac{\partial\left(\tilde{\mu}v^A_k\right)}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega \\
&+\frac{1}{\mathrm{Re}_0\,\sigma}\int_{\Omega}\rho\tilde{\nu}_a\frac{\partial}{\partial x_\ell}\left[\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_k}\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega
\end{aligned}$$

$$
\begin{aligned}
&-\frac{1}{\mathrm{Re}_0\,\sigma}\int_\Omega \frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&+\frac{c_{b_2}}{\mathrm{Re}_0\,\sigma}\int_\Omega\left[\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_\ell}-\tilde{\nu_a}\tilde{\mu}\frac{\partial}{\partial x_\ell}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\right)\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&-\int_\Omega\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\mathrm{SA}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_r^A}{\partial x_q}\frac{\partial v_m^A}{\partial x_p}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_p}{\delta b_i}\right)\mathrm{d}\Omega-\int_\Omega\tilde{\nu_a}\frac{\mathcal{C}_S^{\mathrm{SA}}}{S}2S_{km}\frac{\partial v_k^A}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&+\frac{1}{\mathrm{Re}_0\,\sigma}\int_{S_{\mathrm{NoSlip}}}\rho\tilde{\nu_a}\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_k}\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{I}^{\mathrm{SD}}_{\gamma-\tilde{Re}_{\theta t}}=&-\int_\Omega\gamma_a\frac{\partial\left(\rho v_k^A\gamma\right)}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega-\int_\Omega\tilde{Re}_a\frac{\partial\left(\rho v_k^A\tilde{Re}_{\theta t}\right)}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&+\frac{1}{\mathrm{Re}_0}\int_\Omega\left[\gamma_a\frac{\partial}{\partial x_\ell}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\right]-\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\frac{\partial\gamma}{\partial x_\ell}\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&+\frac{1}{\mathrm{Re}_0}\int_\Omega\left[\tilde{Re}_a\frac{\partial}{\partial x_\ell}\left[\sigma_{\theta,t}(\mu+\mu_t)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\right]-\sigma_{\theta,t}(\mu+\mu_t)\frac{\partial\tilde{Re}_a}{\partial x_k}\frac{\partial\tilde{Re}_{\theta t}}{\partial x_\ell}\right]\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&-\int_\Omega\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v_k^Av_m^A}{|\mathbf{U}|^2}\frac{\partial v_k^A}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega-\int_\Omega\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{kqr}\frac{\partial v_r^A}{\partial x_q}\frac{\partial v_m^A}{\partial x_p}\frac{\partial}{\partial x_\ell}\left(\frac{\delta x_p}{\delta b_i}\right)\mathrm{d}\Omega\\
&-\int_\Omega\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_k^A}{|\mathbf{U}|}\varepsilon_{kqm}\frac{\partial v_m^A}{\partial x_\ell}\frac{\partial}{\partial x_q}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega-\int_\Omega\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\frac{\partial v_k^A}{\partial x_\ell}\frac{\partial}{\partial x_m}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega\\
&+\int_{S_{\mathrm{NoSlip}}}\gamma_a\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}+\int_{S_{\mathrm{NoSlip}}}\tilde{Re}_a\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\frac{\delta\left(\mathsf{n}_k\mathrm{d}S\right)}{\delta b_i}\\
\mathcal{I}^{\mathrm{SD}}_{\Delta}=&-2\int_\Omega\Delta_a\frac{\partial\Delta}{\partial x_k}\frac{\partial\Delta}{\partial x_\ell}\frac{\partial}{\partial x_k}\left(\frac{\delta x_\ell}{\delta b_i}\right)\mathrm{d}\Omega
\end{aligned}
\tag{3.113}
$$

3.8 Verification of the Adjoint Formulation

In this section, the formulation of the adjoint method for the SA-noft2-Gamma-Retheta, SA-LM2015 and SA-sLM2015 transition models is verified. The NLF(1)-0416 airfoil at AoA$=2.03°$, extensively validated against experimental data in Sec. 2.6.3, is used. The airfoil is parameterized using the 8×7 NURBS control lattice of Fig. 3.1; the 12 control points in red are allowed to move in the chordwise and normal-to-the-chord directions, resulting in 24 design variables in total. The

accuracy of the SDs regarding the drag (C_D) and lift coefficient (C_L) is verified against FDs which are assumed to give the reference derivatives. The impact of the "frozen transition" assumption (according to which the adjoint to the transition model equations is not formulated and solved) is also investigated. The verification of the SDs and the shape optimization of the airfoil are performed with each transition model. The two quantities of interest are expressed as

$$J = \frac{F}{\frac{1}{2}\rho_\infty U_\infty^2 c}, \quad F = \int_{S_{\text{Obj}}} (p\mathsf{n}_k - \tau_{km}\mathsf{n}_m)\, r_k dS \tag{3.114}$$

where F stands for the aerodynamic force (drag or lift) and r_k is the direction of the force component, for drag $r_k^D = (-\sin\alpha_\infty, \cos\alpha_\infty)$ and for lift $r_k^L = (\cos\alpha_\infty, \sin\alpha_\infty)$.

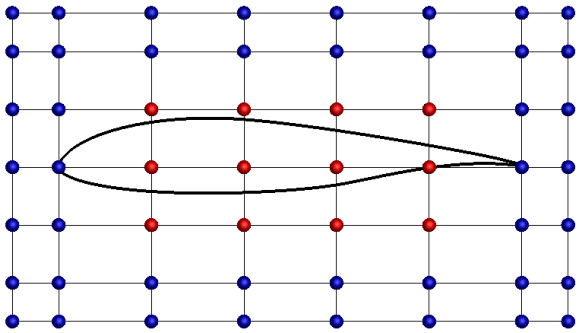
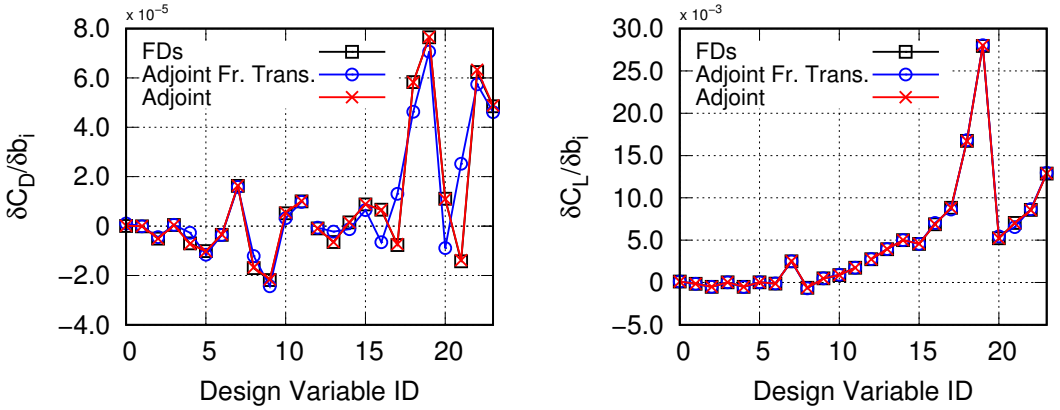
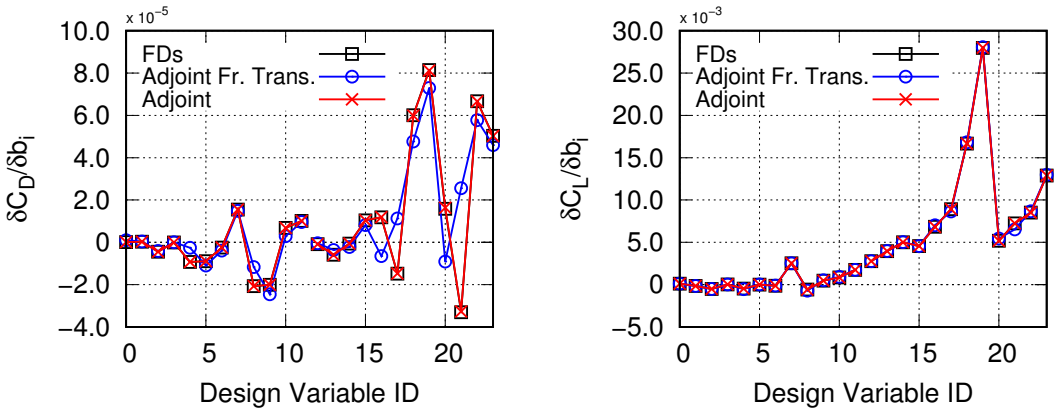


Figure 3.1: *Optimization of the NLF(1)-0416 Airfoil*: Parameterization of the airfoil. Control points in blue remain constant while red ones are allowed to be displaced in the chordwise and the normal-to-the-chord directions.

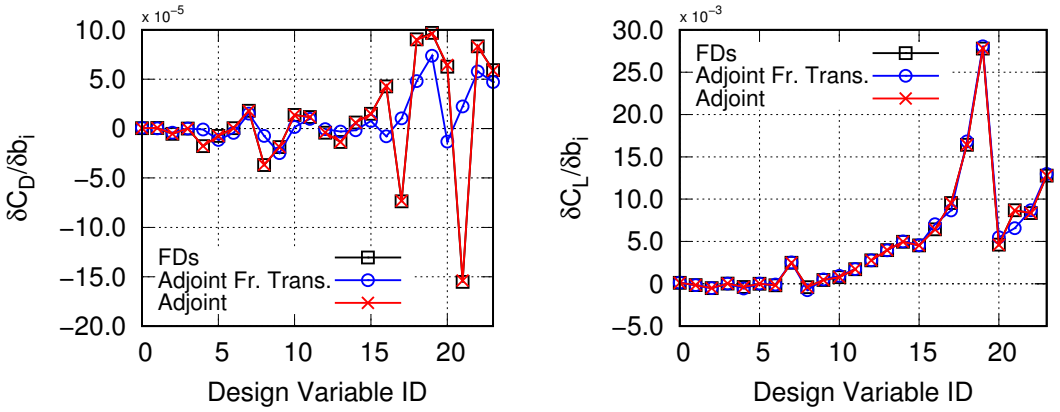
The SDs of C_D and C_L computed by the adjoint method, with and without the "frozen transition" assumption, are compared with FDs in Figs. 3.2a, 3.2b and 3.2c for the SA-noft2-Gamma-Retheta, the SA-LM2015 and the SA-sLM2015 model, respectively. The first half SDs correspond to the x coordinates and the second half to the y coordinates of the control points. The adjoint method reproduces the outcome of the FDs with high accuracy for both C_D and C_L. On the other hand, the "frozen transition" assumption proves to be harmful to the gradient accuracy; higher deviations of the "frozen transition" SDs can be seen for the C_D SDs, as occasionally they even have the opposite sign (i.e. design variable with ID 21). The primal (γ) and adjoint (γ_a) intermittency fields are presented in Fig. 3.3 for the SA-sLM2015 transition model.



(a) SA-noft2-Gamma-Retheta



(b) SA-LM2015



(c) SA-sLM2015

Figure 3.2: *Optimization of the NLF(1)-0416 Airfoil*: Comparison of the C_D (left) and C_L (right) SDs computed by the continuous adjoint method, with and without the "frozen transition" assumption, with FDs. SDs based on the (a) SA-noft2-Gamma-Retheta, (b) SA-LM2015 and (c) SA-sLM2015 transition model.

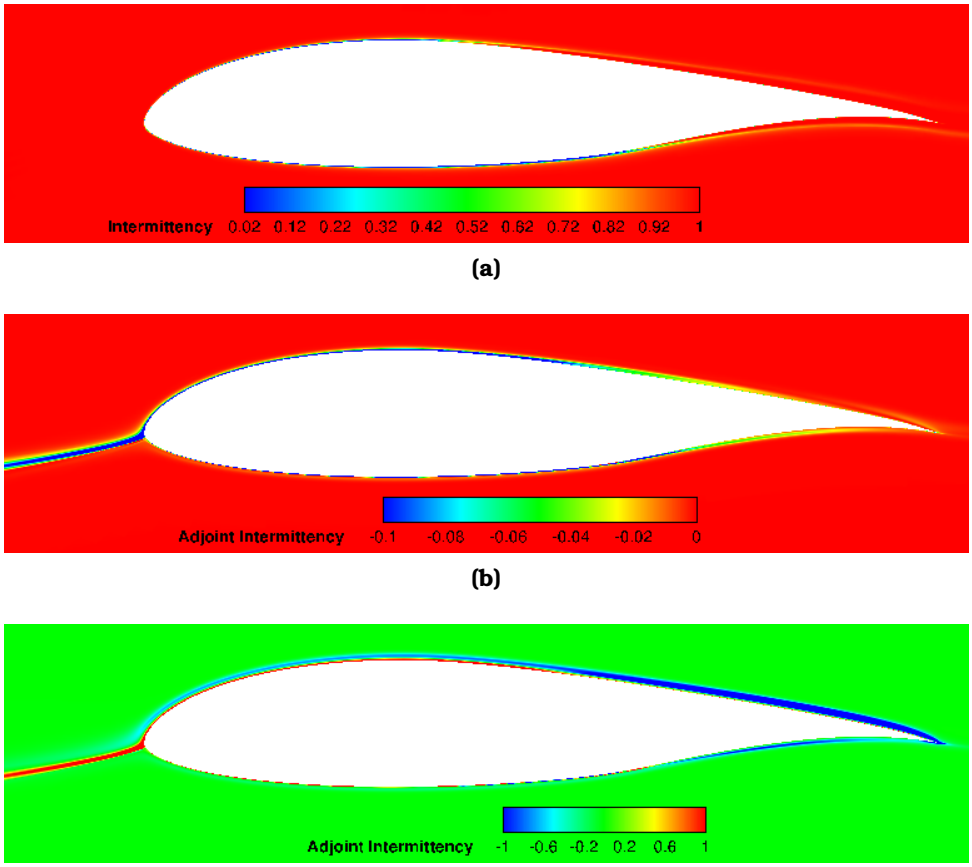
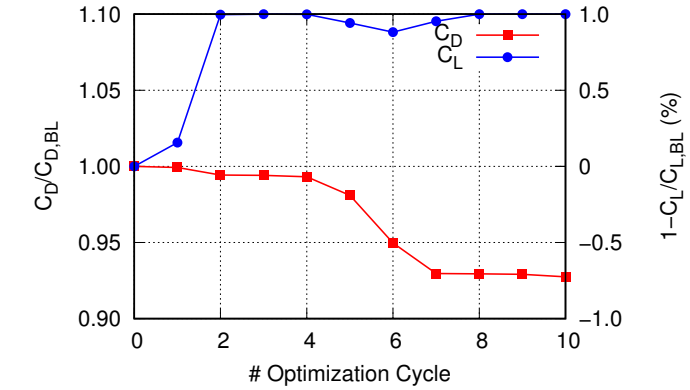
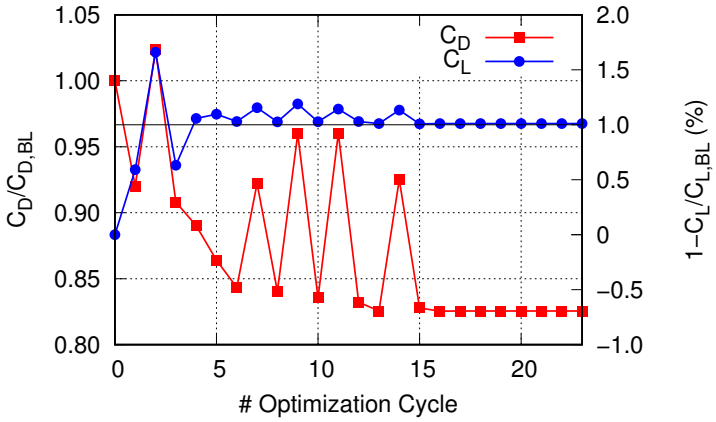


Figure 3.3: *Optimization of the NLF(1)-0416 Airfoil*: (a) Primal and adjoint for (b) C_D and (c) C_L intermittency fields using the SA-sLM2015 model.

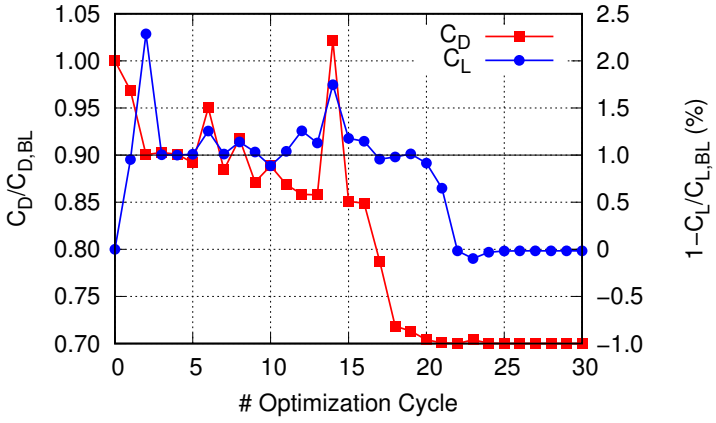
After verifying the SDs, the optimization of the NLF(1)–0416 airfoil for min. C_D under the double-sided inequality constraint that C_L remains close to that of the baseline airfoil (should not change by more than $\pm 1\%$), is carried out. An additional inequality constraint, requiring that the airfoil area should not drop below 90% of the baseline one is imposed. The optimization is performed three times, once for each transition model variant; the convergence histories are plotted in Fig. 3.4. All optimization runs are performed using the sequential least squares programming algorithm (SLSQP), [105]. The optimizations based on the SA-noft2-Gamma-Retheta, the SA-LM2015 and SA-sLM2015 models resulted in $\sim 7.5\%$, $\sim 17.4\%$ and $\sim 28.7\%$ reduction of the C_D value, respectively, maintaining the C_L close to the baseline one by $\pm 1\%$. The volumes (areas) of the optimized airfoils are decreased by 10%, reaching the minimum allowed value of the imposed constraint.



(a) SA-noft2-Gamma-Retheta



(b) SA-LM2015



(c) SA-sLM2015

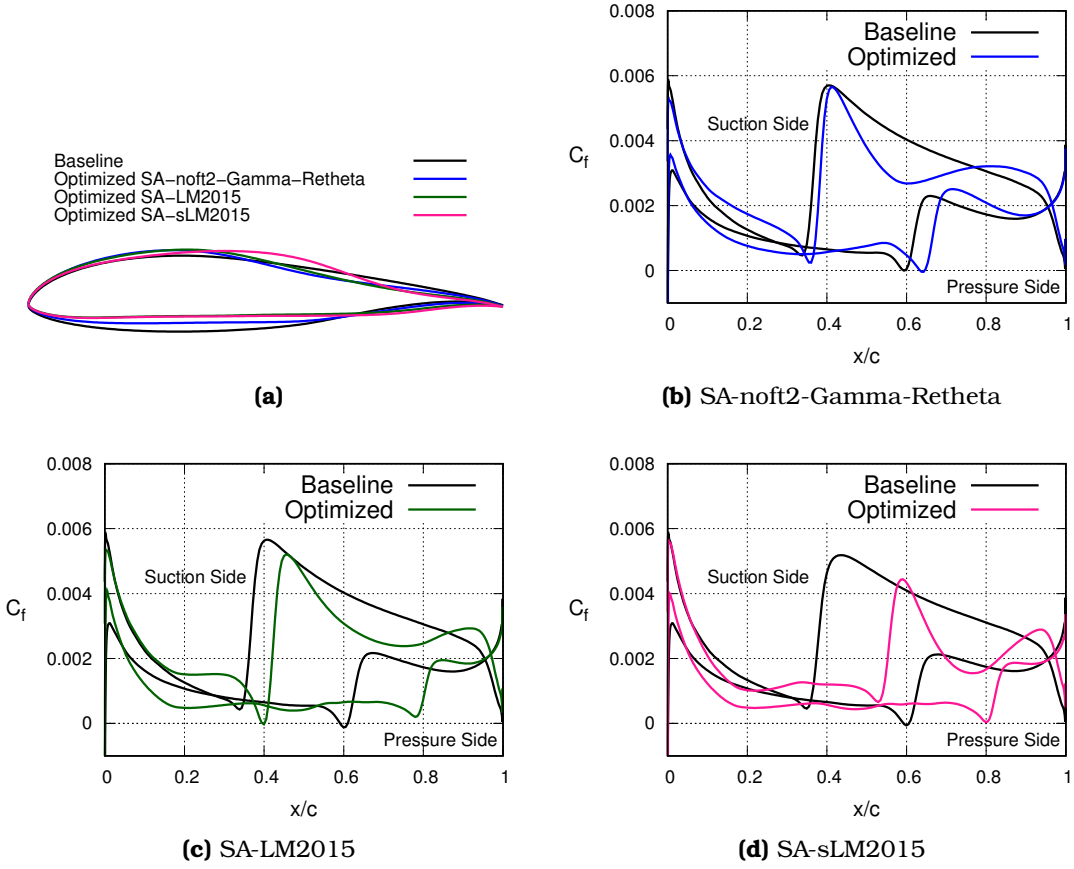
Figure 3.4: *Optimization of the NLF(1)-0416 Airfoil*: Evolution of the objective (C_D) and constraint (C_L) functions during the optimization based on (a) SA-noft2-Gamma-Retheta, (b) SA-LM2015 and (c) SA-sLM2015 transition model.

It is interesting to investigate the performance of each transition model variant at the optimized geometries obtained by the other variants. These are summarized in Table 3.1 where each row corresponds to an evaluation with a transition model variant and each column to the optimized airfoil using a transition model variant. The three models have similar performance (C_D and C_L values) for each optimized solution (the airfoil geometry resulted from each optimization procedure), i.e. they lead to similar C_D reduction, satisfying the C_L and volume constraints. It seems that the optimization runs based on the SA-noft2-Gamma-Retheta and the SA-LM2015 transition models have been trapped into local minima, since when the optimized airfoil based on the SA-sLM2015 model is evaluated on them their C_D is even lower.

Optimized on / **Re-Evaluated on**	SA-noft2-Gamma-Retheta	SA-LM2015	SA-sLM2015
SA-noft2-Gamma-Retheta	0.9254	0.8212	0.7062
SA-LM2015	0.9186	0.8257	0.7131
SA-sLM2015	0.9209	0.8282	0.7133

Table 3.1: *Optimization of the NLF(1)-0416 Airfoil*: $C_D/C_{D,BL}$ of the optimized airfoils. Each row corresponds to evaluation with a transition model and each column to the optimized airfoil using the corresponding transition model.

The geometry and the skin friction coefficient distribution for the baseline and the three optimized airfoils are presented in Fig. 3.5. All optimization runs resulted in an airfoil geometry with almost flat pressure side, Fig. 3.5a. Differences are located at the suction side, where the optimization relying on the SA-sLM2015 model increased and shifted the curvature to the trailing edge. These are reflected on the skin friction distribution of the optimized airfoils. Regarding the optimization on the SA-noft2-Gamma-Retheta model, there is a slight change in the C_f distribution, Fig. 3.5b. Regarding the optimizations based on the SA-LM2015 and SA-sLM2015 models, Figs. 3.5c and 3.5d, the transition location of the pressure side is more or less the same for the two airfoils, shifted by $\sim 20\%$ chord downstream. On the other hand, transition locations over the suction side differ; the optimized geometry based on the SA-sLM2015 model has the largest shift ($\sim 17\%$ chord) leading to a greater C_D reduction.



(a)

(b) SA-noft2-Gamma-Retheta

(c) SA-LM2015

(d) SA-sLM2015

Figure 3.5: *Optimization of the NLF(1)-0416 Airfoil*: (a) Baseline and optimized airfoil geometries. Skin friction coefficient distribution along the airfoil surface for the baseline and the optimized geometries for the optimizations based on the (b) SA-noft2-Gamma-Retheta, (c) SA-LM2015 and (d) SA-sLM2015 transition model.

3.9 Conclusions

In this chapter, the detailed derivation of the FI continuous adjoint method for transitional flows of compressible fluids was presented. The development of the continuous adjoint method for turbulent flows (with the Spalart–Allmaras turbulence model), as performed in [135], was extended to transitional flows by differentiating the SA-noft2-Gamma-Retheta, the SA-LM2015 and the SA-sLM2015 transition model. The mean flow field adjoint equations, the adjoint Spalart–Allmaras, the adjoint $\gamma - \tilde{Re}_{\theta t}$ and the adjoint Hamilton–Jacobi equations were derived. The adjoint boundary conditions for no-slip walls, inlet, outlet and far–field boundaries were developed. Finally, the sensitivity derivatives ($\frac{\delta J}{\delta b_i}$) were expressed.

The continuous adjoint method for transitional flows and the impact of the "frozen transition" assumption was investigated for the flow around an isolated

airfoil. The SDs of the C_D and C_L with or without the "frozen transition" assumption were verified against FDs. The continuous adjoint method reproduced the outcome the FDs with high accuracy. It was demonstrated that the "frozen transition" assumption can be harmful for the accuracy of the SDs, as some of them had even the wrong sign. Finally, the continuous adjoint to all SA-based transition model variants were used for the constrained optimization of the NLF(1)–0416 airfoil. The optimization runs improved the aerodynamic behavior of the airfoil, while meeting the required constraints. The optimization run based on the SA-sLM2015 transition model outperformed other variants; this is due to its numerical robustness and should be taken into account for the real-world optimization problems of Chapter 5.

Chapter 4

Consistent Discretization Schemes for the Continuous Adjoint

In this chapter, consistent discretization schemes for the continuous adjoint PDEs and their boundary conditions with a clear physical insight are proposed. The mathematical development of the continuous adjoint, as presented in Chapter 3, resulted in the Field Adjoint Equations and the Adjoint Boundary Conditions; these should be discretized and numerically solved. Finding appropriate discretization schemes for the adjoint PDEs is a challenge, since these can significantly affect the accuracy of the computed SDs if not consistent with the discretized flow problem. This is not the case in discrete adjoint, [38, 103, 43], in which the adjoint equations result in discrete form and are consistent with the discretized primal problem. As such, the adjoint solver inherits the convergence characteristics from the primal one. However, as already mentioned, the main disadvantages of discrete adjoint is the high memory footprint of the adjoint code and the lack of understanding of the derived adjoint equations.

Herein, discretization schemes for the continuous adjoint PDEs and their boundary conditions which are inspired by the discrete adjoint, and, thus, are consistent with the discretized flow problem are proposed. In other words, the advantages of continuous and discrete adjoint are combined into the new *Think-Discrete Do-Continuous* (*TDDC*) adjoint. This is achieved by developing discretization schemes for the differential operators (convection, diffusion, gradients, etc.) of the continuous adjoint PDEs and their boundary conditions, to replicate what discrete adjoint does, without its weaknesses though. By doing so, the new *TDDC* adjoint computes SDs with the same accuracy as discrete adjoint, without though an occasionally excessive memory footprint; in contrast, the *TDDC* adjoint retains the useful insight into the adjoint equations and boundary conditions.

The mathematical development of the discrete adjoint will be performed for the Spalart–Allmaras turbulence model coupled with the $\gamma-\tilde{Re}_{\theta t}$ transition model (the SA-sLM2015 variant). In discrete adjoint, the objective function J is augmented

by the inner product of the discretized residuals of the primal equations and the adjoint variables, resulting in the augmented objective function J_{aug},

$$J_{\text{aug}} = J + \sum_P \Psi_n^P R_n^{\text{MF},P} + \sum_P \tilde{\nu_a}^P R^{\tilde{\nu},P} + \sum_P \gamma_a^P R^{\gamma,P} + \sum_P \tilde{Re}_a^P R^{\tilde{Re}_{\theta t},P} + \sum_P \Delta_a^P R^{\Delta,P} \tag{4.1}$$

where $\Psi_n^P, (n = 1, ..., 5)$, $\tilde{\nu_a}^P$, γ_a^P, $\tilde{Re}_a^P$ and Δ_a^P are the mean flow adjoint variables, the adjoint Spalart–Allmaras variable, the adjoint intermittency, the adjoint transition momentum-thickness Reynolds number and the adjoint distance, respectively, all of them at node P (internal or boundary). $R_n^{\text{MF},P}, R^{\tilde{\nu},P}, R^{\gamma,P}, R^{\tilde{Re}_{\theta t},P}, R^{\Delta,P}$ stand for the discretized residuals at node P. Differentiating Eq. 4.1 w.r.t. the design variables b_i results in

$$\frac{\delta J_{\text{aug}}}{\delta b_i} = \frac{\delta J}{\delta b_i} + \underbrace{\sum_P \Psi_n^P \frac{\delta R_n^{\text{MF},P}}{\delta b_i}}_{I^{\text{MF},P}} + \underbrace{\sum_P \tilde{\nu_a}^P \frac{\delta R^{\tilde{\nu},P}}{\delta b_i}}_{I^{\text{SA},P}} + \underbrace{\sum_P \gamma_a^P \frac{\delta R^{\gamma,P}}{\delta b_i}}_{I^{\gamma,P}} + \underbrace{\sum_P \tilde{Re}_a^P \frac{\delta R^{\tilde{Re}_{\theta t},P}}{\delta b_i}}_{I^{\tilde{Re}_{\theta t},P}} + \underbrace{\sum_P \Delta_a^P \frac{\delta R^{\Delta,P}}{\delta b_i}}_{I^{\Delta,P}} \tag{4.2}$$

During the mathematical development of Eq. 4.2, summations containing derivatives of the flow variables ($\frac{\delta \mathcal{U}_n}{\delta b_i}$) arise; similar to the development of the continuous adjoint, these are collected together and their multipliers are set to zero. In what follows, the six terms on the r.h.s. on Eq. 4.2 are hand-differentiated.

4.1 Differentiation of the Mean Flow Equations

Term $\mathcal{I}^{\text{MF},P}$ in Eq. 4.2 is expanded as

$$\mathcal{I}^{\text{MF},P} = \underbrace{\sum_P \Psi_n^P \frac{\delta R_n^{\text{inv},P}}{\delta b_i}}_{\mathcal{I}^{\text{inv},P}} \underbrace{- \sum_P \Psi_n^P \frac{\delta R_n^{\text{vis},P}}{\delta b_i}}_{\mathcal{I}^{\text{vis},P}} \tag{4.3}$$

in order to separately handle the inviscid and viscous terms.

4.1.1 Differentiation of Inviscid Terms

In Eq. 4.3, the inviscid term $\mathcal{I}^{\text{inv},P}$ reads

$$\mathcal{I}^{\text{inv},P} = \sum_P \Psi_n^P \sum_{Q \in \mathcal{N}(P)} \frac{\delta \Phi_n^{\text{inv},PQ}}{\delta b_i} + \sum_P \Psi_n^P \sum_{f \in \mathcal{B}(P)} \frac{\delta \Phi_n^{\text{inv},f}}{\delta b_i} \tag{4.4}$$

Assuming that the Roe's scheme, Eq. 2.49, is used to discretize the inviscid fluxes, the first term of the r.h.s. in Eq. 4.4 which corresponds to the internal fluxes reads

$$\begin{aligned}
\sum_P \Psi_n^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi_n^{\text{inv},PQ}}{\delta b_i} &= \frac{1}{2}\sum_P \Psi_n^P \sum_Q A_{nmk}^P \mathsf{n}_k^{PQ} \frac{\delta U_m^P}{\delta b_i} + \frac{1}{2}\sum_P \Psi_n^P \sum_Q A_{nmk}^Q \mathsf{n}_k^{PQ} \frac{\delta U_m^Q}{\delta b_i} \\
&+ \frac{1}{2}\sum_P \Psi_n^P \sum_Q \left[\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^L}{\partial V_\ell^L} - \left(U_m^R - U_m^L\right) \frac{\partial \left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^L} \right] \frac{\delta V_\ell^L}{\delta b_i} \\
&- \frac{1}{2}\sum_P \Psi_n^P \sum_Q \left[\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^R}{\partial V_\ell^R} + \left(U_m^R - U_m^L\right) \frac{\partial \left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^R} \right] \frac{\delta V_\ell^R}{\delta b_i} \\
&+ \frac{1}{2}\sum_P \Psi_n^P \sum_Q \left[\left(A_{nmk}^P U_m^P + A_{nmk}^Q U_m^Q\right) - \left(U_m^R - U_m^L\right) \frac{\partial \left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|}{\partial \mathsf{n}_k^{PQ}} \right] \frac{\delta \mathsf{n}_k^{PQ}}{\delta b_i}
\end{aligned} \tag{4.5}$$

For the sake of simplicity, $\sum_{Q\in\mathcal{N}(P)}$ is hereafter written as $\sum_Q$. To easily demonstrate the idea behind the *TDDC* adjoint, a first-order accuracy is considered ($P \equiv L$, $Q \equiv R$), though, the second-order schemes are used in all studies. Then, Eq. 4.5 reads

$$\begin{aligned}
&\sum_P \Psi_n^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi_n^{\text{inv},PQ}}{\delta b_i} = \\
&= \frac{1}{2}\sum_P \Psi_n^P \sum_Q \left[A_{nmk}^P \mathsf{n}_k^{PQ} \frac{\partial U_m^P}{\partial V_\ell^P} + \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i} \\
&+ \frac{1}{2}\sum_P \Psi_n^P \sum_Q \left[A_{nmk}^Q \mathsf{n}_k^{PQ} \frac{\partial U_m^Q}{\partial V_\ell^Q} - \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^Q}{\partial V_\ell^Q} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^Q} \right] \frac{\delta V_\ell^Q}{\delta b_i} \\
&+ \frac{1}{2}\sum_P \Psi_n^P \sum_Q \left[\left(A_{nmk}^P U_m^P + A_{nmk}^Q U_m^Q\right) - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial \mathsf{n}_k^{PQ}} \right] \frac{\delta \mathsf{n}_k^{PQ}}{\delta b_i}
\end{aligned} \tag{4.6}$$

where the first/second term contributes to the flux of the adjoint mean flow equation at node P/Q. The same mathematical development must be carried out for all neighboring nodes Q, collecting contributions to node P. This procedure is equivalent (and can be omitted) to swapping nodes Q with P. By doing so, one

gets

$$
\begin{aligned}
&\frac{1}{2}\sum_P\sum_Q \Psi_n^P \left[A_{nmk}^P \mathsf{n}_k^{PQ} \frac{\partial U_m^P}{\partial V_\ell^P} + \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i} \\
&+\frac{1}{2}\sum_P\sum_Q \Psi_n^P \left[A_{nmk}^Q \mathsf{n}_k^{PQ} \frac{\partial U_m^Q}{\partial V_\ell^Q} - \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^Q}{\partial V_\ell^Q} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^Q} \right] \frac{\delta V_\ell^Q}{\delta b_i} \\
&\left(\text{Swap } P \rightleftarrows Q \text{ in terms including } \frac{\delta V_\ell^Q}{\delta b_i} \right) \\
&=\frac{1}{2}\sum_P\sum_Q \Psi_n^P \left[A_{nmk}^P \mathsf{n}_k^{PQ} \frac{\partial U_m^P}{\partial V_\ell^P} + \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i} \\
&+\frac{1}{2}\sum_P\sum_Q \Psi_n^Q \left[A_{nmk}^P \mathsf{n}_k^{QP} \frac{\partial U_m^P}{\partial V_\ell^P} - \left|\tilde{A}_{nmk}^{QP}\mathsf{n}_k^{QP}\right| \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^P - U_m^Q\right) \frac{\partial \left|\tilde{A}_{nmk}^{QP}\mathsf{n}_k^{QP}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i} \\
&\text{(Re-arrange)} \\
&=\frac{1}{2}\sum_P\sum_Q \Psi_n^P \left[A_{nmk}^P \mathsf{n}_k^{PQ} \frac{\partial U_m^P}{\partial V_\ell^P} + \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i} \\
&-\frac{1}{2}\sum_P\sum_Q \Psi_n^Q \left[A_{nmk}^P \mathsf{n}_k^{PQ} \frac{\partial U_m^P}{\partial V_\ell^P} - \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i} \\
&\text{(Get the final contribution)} \\
&=\frac{1}{2}\sum_P\sum_Q \left(\Psi_n^P - \Psi_n^Q\right) \left[\left(A_{nmk}^P \mathsf{n}_k^{PQ} + \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right| \right) \frac{\partial U_m^P}{\partial V_\ell^P} - \left(U_m^Q - U_m^P\right) \frac{\partial \left|\tilde{A}_{nmk}^{PQ}\mathsf{n}_k^{PQ}\right|}{\partial V_\ell^P} \right] \frac{\delta V_\ell^P}{\delta b_i}
\end{aligned}
\tag{4.7}
$$

Similarly, when second–order accuracy is used (without or with the van-Leer van-Albada limiter), terms $\frac{\delta U_m^P}{\delta b_i}$, $\frac{\delta U_m^Q}{\delta b_i}$ and $\frac{\delta U_m^\Lambda}{\delta b_i}$ appear contributing to the adjoint flux at node P, nodes $Q \in \mathcal{N}(P)$ and nodes $\Lambda \in \mathcal{N}(P)$. As already mentioned, Q are neighbors of P connected via a grid edge and Λ are all (excluding P) vertices of finite volumes having P as one of their vertices. The same mathematical development must be performed for all nodes Q and Λ giving contributions to node P. As described in Eq. 4.7, this procedure can be omitted by swapping nodes Q with P and Λ with P, when the contribution to the adjoint flux of node Q or Λ appears

in the mathematical expression. By doing so, Eq. 4.5 becomes

$$\begin{aligned}\sum_P \Psi_n^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi_n^{\mathrm{inv},PQ}}{\delta b_i} &= \underbrace{\frac{1}{2}\sum_P\sum_Q \left(\Psi_n^P-\Psi_n^Q\right) A_{nq}^P \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}} \\ &+ \underbrace{\frac{1}{2}\sum_P\sum_Q \left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right|\frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right)\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right]\frac{\delta V_\ell^L}{\delta b_i}}_{\mathcal{I}^L} \\ &+ \underbrace{\frac{1}{2}\sum_P\sum_Q \left(\Psi_n^P-\Psi_n^Q\right)\left(A_{nmk}^P U_m^P + U_m^L\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial \mathsf{n}_k^{PQ}}\right)\frac{\delta \mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}}\end{aligned} \tag{4.8}$$

The L state can be expressed either based on Eq. 2.50 (second–order accuracy; no limiter) or Eq. 2.51 (second–order accuracy; van Leer-van Albada limiter). For second–order accuracy without limiter, $\mathcal{I}^L$ becomes

$$\begin{aligned}\mathcal{I}^L &= \underbrace{\frac{1}{2}\sum_P\sum_Q \left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right|\frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right)\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right]\frac{\partial V_\ell^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}} \\ &+ \underbrace{\sum_P \mathcal{D}_{\ell r}^P D_r^P \frac{\partial V_\ell^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i} + \sum_P\sum_\Lambda \mathcal{D}_{\ell r}^\Lambda Z_r^{\Lambda P}\frac{\partial V_\ell^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}} \\ &+ \underbrace{\frac{1}{4}\sum_P\sum_Q \left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right|\frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right)\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right]\frac{\partial V_\ell^P}{\partial x_r}\frac{\delta \mathsf{t}_r^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}} \\ &+ \underbrace{\sum_P \mathcal{D}_{\ell r}^P V_\ell^P \frac{\delta D_r^P}{\delta b_i} + \sum_P\sum_\Lambda \mathcal{D}_{\ell r}^\Lambda V_\ell^P \frac{\delta Z_r^{\Lambda P}}{\delta b_i}}_{\rightarrow\text{SD}}\end{aligned} \tag{4.9}$$

If the van Leer-van Albada limiter (Eq. 2.51) is used, the gradient of the limiting

function $\mathscr{L}$ is expressed as

$$
\begin{aligned}
\left.\frac{\delta \mathscr{L}_\ell\left(\alpha_\ell, \beta_\ell\right)}{\delta b_i}\right|_P &= K_\ell^{\alpha_\ell^P, \beta_\ell}\left(\frac{\partial \mathscr{L}_\ell\left(\alpha_\ell^P, \beta_\ell\right)}{\partial \alpha_\ell^P} \frac{\delta \alpha_\ell^P}{\delta b_i}+\frac{\partial \mathscr{L}_\ell\left(\alpha_\ell^P, \beta_\ell\right)}{\partial \beta_\ell} \frac{\delta \beta_\ell}{\delta b_i}\right) \\
\frac{\delta \alpha_\ell^P}{\delta b_i} &= 2 \mathfrak{t}_r^{PQ} \frac{\delta}{\delta b_i}\left(\frac{\partial V_\ell^P}{\partial x_r}\right)+2 \frac{\partial V_\ell^P}{\partial x_r} \frac{\delta \mathfrak{t}_r^{PQ}}{\delta b_i}-\frac{\delta V_\ell^Q}{\delta b_i}+\frac{\delta V_\ell^P}{\delta b_i} \\
\frac{\delta \beta_\ell}{\delta b_i} &= \frac{\delta V_\ell^Q}{\delta b_i}-\frac{\delta V_\ell^P}{\delta b_i} \\
\frac{\partial \mathscr{L}(\alpha, \beta)}{\partial \alpha} &= \frac{\left(\beta^2+\epsilon\right)\left(-\alpha^2+2 \alpha \beta+\beta^2+2 \epsilon\right)}{\left(\alpha^2+\beta^2+2 \epsilon\right)^2} \\
\frac{\partial \mathscr{L}(\alpha, \beta)}{\partial \beta} &= \frac{\left(\alpha^2+\epsilon\right)\left(-\beta^2+2 \alpha \beta+\alpha^2+2 \epsilon\right)}{\left(\alpha^2+\beta^2+2 \epsilon\right)^2} \\
K_\ell^{\alpha, \beta} &= \begin{cases} 1 & , \text{if } \alpha \beta>0 \\ 0 & , \text{if } \alpha \beta \leq 0 \end{cases}
\end{aligned}
\tag{4.10}
$$

In what follows, $K_\ell^{\alpha_\ell^*, \beta_\ell} \frac{\partial \mathscr{L}_\ell\left(\alpha_\ell^*, \beta_\ell\right)}{\partial \alpha_\ell^*}$ and $K_\ell^{\alpha_\ell^*, \beta_\ell} \frac{\partial \mathscr{L}_\ell\left(\alpha_\ell^*, \beta_\ell\right)}{\partial \beta_\ell}$ will be denoted as $\frac{\partial \mathscr{L}_\ell^*}{\partial \alpha_\ell^*}$ and $\frac{\partial \mathscr{L}_\ell^*}{\partial \beta_\ell}$, respectively (the * superscript may refer either to node P or Q). Then $\mathcal{I}^L$ reads

$$
\begin{aligned}
\mathcal{I}^L = & \underbrace{\frac{1}{2} \sum_P \sum_Q\left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right| \frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right) \frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right] \frac{\partial V_\ell^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
& +\underbrace{\frac{1}{4} \sum_P \sum_Q\left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right| \frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right) \frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right]\left[\frac{\partial \mathscr{L}_\ell^P}{\partial \alpha_\ell^P}-\frac{\partial \mathscr{L}_\ell^P}{\partial \beta_\ell}\right] \frac{\partial V_\ell^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
& +\underbrace{\frac{1}{4} \sum_P \sum_Q\left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right| \frac{\partial U_m^R}{\partial V_\ell^R}+\left(U_m^R-U_m^L\right) \frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^R}\right]\left[\frac{\partial \mathscr{L}_\ell^Q}{\partial \alpha_\ell^Q}-\frac{\partial \mathscr{L}_\ell^Q}{\partial \beta_\ell}\right] \frac{\partial V_\ell^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
& +\underbrace{\sum_P \mathcal{D}_{\ell r}^P D_r^P \frac{\partial V_\ell^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}+\sum_P \sum_\Lambda \mathcal{D}_{\ell r}^\Lambda Z_r^{\Lambda P} \frac{\partial V_\ell^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}
\end{aligned}
$$

$$+\underbrace{\frac{1}{2}\sum_{P}\sum_{Q}\left(\Psi_n^P-\Psi_n^Q\right)\left[\left|\tilde{A}_{nm}^{LR}\right|\frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right)\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right]\frac{\partial\mathscr{L}_\ell^P}{\partial\alpha_\ell^P}\frac{\partial V_\ell^P}{\partial x_r}\frac{\delta \mathsf{t}_r^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}}$$

$$+\underbrace{\sum_{P}\mathcal{D}_{\ell r}^P V_\ell^P\frac{\delta D_r^P}{\delta b_i}+\sum_{P}\sum_{\Lambda}\mathcal{D}_{\ell r}^{\Lambda}V_\ell^P\frac{\delta Z_r^{\Lambda P}}{\delta b_i}}_{\rightarrow\text{SD}} \tag{4.11}$$

where

$$\mathcal{D}_{\ell r}^P=\begin{cases}\frac{1}{4}\sum\limits_{Q}\left(\Psi_n^P-\Psi_n^Q\right)\left(\left|\tilde{A}_{nm}^{LR}\right|\frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right)\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right)\mathsf{t}_r^{PQ} & \text{, no limiter}\\ \frac{1}{2}\sum\limits_{Q}\left(\Psi_n^P-\Psi_n^Q\right)\left(\left|\tilde{A}_{nm}^{LR}\right|\frac{\partial U_m^L}{\partial V_\ell^L}-\left(U_m^R-U_m^L\right)\frac{\partial\left|\tilde{A}_{nm}^{LR}\right|}{\partial V_\ell^L}\right)\mathsf{t}_r^{PQ}\frac{\partial\mathscr{L}_\ell^P}{\partial\alpha_\ell^P} & \text{, van-Leer van-Albada}\end{cases} \tag{4.12}$$

The detailed differentiation of the absolute Jacobian can be found in Appendix A.

The second term on the r.h.s. of Eq. 4.4 which includes the fluxes crossing boundary faces depends on the type of boundaries and the corresponding boundary conditions. For wall boundaries ($f\in\mathcal{B}^{\mathcal{W}}(P)$), $\Phi_{m+1}^{\text{inv},f}=p^P\mathsf{n}_m^f$ and

$$\sum_{P}\Psi_n^P\sum_{f\in\mathcal{B}^{\mathcal{W}}(P)}\frac{\delta\Phi_n^{\text{inv},f}}{\delta b_i}=\underbrace{\sum_{P}\sum_{f\in\mathcal{B}^{\mathcal{W}}(P)}\Psi_{m+1}^P\mathsf{n}_m^f\frac{\partial p^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}+\underbrace{\sum_{P}\sum_{f\in\mathcal{B}^{\mathcal{W}}(P)}\Psi_{m+1}^P p^P\frac{\delta\mathsf{n}_m^f}{\delta b_i}}_{\rightarrow\text{SD}} \tag{4.13}$$

For the subsonic inlet boundaries ($f\in\mathcal{B}^{\mathcal{I}}(P)$) in 3D, let $V_j^{\text{loc}},(j=1,\ldots,4)$ be the quantities for which a Dirichlet condition is imposed and V_5^{loc} the flow quantity value extrapolated from the interior of the domain. Then,

$$\sum_{P}\Psi_n^P\sum_{f\in\mathcal{B}^{\mathcal{I}}(P)}\frac{\delta\Phi_n^{\text{inv},f}}{\delta b_i}=\underbrace{\frac{1}{2}\sum_{P}\Psi_n^P\sum_{f\in\mathcal{B}^{\mathcal{I}}(P)}\left[\left|\tilde{A}_{nqk}^P\mathsf{n}_k^f\right|-\left(U_m^Q-U_m^P\right)\frac{\partial\left|\tilde{A}_{nmk}^P\mathsf{n}_k^f\right|}{\partial V_\ell^P}\frac{\partial V_\ell^P}{\partial U_q^P}\right]\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\sum_{P}\Psi_n^P\sum_{f\in\mathcal{B}^{\mathcal{I}}(P)}\left[A_{nmk}^Q\mathsf{n}_k^f-\frac{1}{2}\left|\tilde{A}_{nmk}^P\mathsf{n}_k^f\right|\right]\frac{\partial U_m^Q}{\partial V_5^{\text{loc}}}\frac{\partial V_5^{\text{loc}}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}$$

$$+\underbrace{\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{I}}(P)} \left[\tilde{A}^Q_{nmk}U^Q_m - \frac{1}{2}\left(U^Q_m - U^P_m\right)\frac{\partial\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|}{\partial \mathsf{n}^f_k}\right]\frac{\delta \mathsf{n}^f_k}{\delta b_i}}_{\rightarrow \text{SD}} \tag{4.14}$$

For the subsonic outlet boundaries, ($f \in \mathcal{B}^{\mathcal{O}}(P)$), let $V_j^{\text{loc}}, (j = 1, \ldots, 4)$ be the quantities extrapolated from the flow domain and V_5^{loc} the quantity for which a Dirichlet condition is imposed. Then,

$$\begin{aligned}
\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \frac{\delta\Phi_n^{\text{inv},f}}{\delta b_i} = &\underbrace{\frac{1}{2}\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \left[\left|\tilde{A}^P_{nqk}\mathsf{n}^f_k\right| - \left(U^Q_m - U^P_m\right)\frac{\partial\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|}{\partial V^P_\ell}\frac{\partial V^P_\ell}{\partial U^P_q}\right]\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \left[A^Q_{nmk}\mathsf{n}^f_k - \frac{1}{2}\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|\right]\frac{\partial U^Q_m}{\partial V^{\text{loc}}_1}\frac{\partial V^{\text{loc}}_1}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \left[A^Q_{nmk}\mathsf{n}^f_k - \frac{1}{2}\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|\right]\frac{\partial U^Q_m}{\partial V^{\text{loc}}_2}\frac{\partial V^{\text{loc}}_2}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \left[A^Q_{nmk}\mathsf{n}^f_k - \frac{1}{2}\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|\right]\frac{\partial U^Q_m}{\partial V^{\text{loc}}_3}\frac{\partial V^{\text{loc}}_3}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \left[A^Q_{nmk}\mathsf{n}^f_k - \frac{1}{2}\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|\right]\frac{\partial U^Q_m}{\partial V^{\text{loc}}_4}\frac{\partial V^{\text{loc}}_4}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\underbrace{+\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{O}}(P)} \left[\tilde{A}^Q_{nmk}U^Q_m - \frac{1}{2}\left(U^Q_m - U^P_m\right)\frac{\partial\left|\tilde{A}^P_{nmk}\mathsf{n}^f_k\right|}{\partial \mathsf{n}^f_k}\right]\frac{\delta \mathsf{n}^f_k}{\delta b_i}}_{\rightarrow \text{SD}}
\end{aligned} \tag{4.15}$$

For the far-field boundaries $f \in \mathcal{B}^{\mathcal{F}}(P)$, using Eq. 2.54,

$$\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{F}}(P)} \frac{\delta\Phi_n^{\text{inv},f}}{\delta b_i} = \sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{F}}(P)} \frac{\delta}{\delta b_i}\left(A^{+P}_{nmk}\mathsf{n}^f_k U^P_m + A^{-P}_{nmk}\mathsf{n}^f_k U^Q_m\right)$$

$$= \underbrace{\sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{F}}(P)} \left[A_{nqk}^{+P} \mathsf{n}_k^f + \frac{1}{2}\left(U_m^P - U_m^Q\right) \frac{\partial \left| A_{nmk}^P \mathsf{n}_k^f \right|}{\partial V_\ell^P} \frac{\partial V_\ell^P}{\partial U_q^P} \right] \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$+ \underbrace{\frac{1}{2} \sum_P \Psi_n^P \sum_{f\in\mathcal{B}^{\mathcal{F}}(P)} \left[A_{nmk}^P \left(U_m^P + U_m^Q\right) + \left(U_m^P - U_m^Q\right) \frac{\partial \left| A_{nmk}^P \right|}{\partial \mathsf{n}_k^f} \right] \frac{\delta \mathsf{n}_k^f}{\delta b_i}}_{\rightarrow \text{SD}} \tag{4.16}$$

4.1.2 Differentiation of Viscous Terms

The viscous term $\mathcal{I}^{\text{vis},P}$ in Eq. 4.3, discretized based on Eq. 2.58, becomes

$$\begin{aligned}
\mathcal{I}^{\text{vis},P} &= -\sum_P \Psi_n^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta \Phi_n^{\text{vis},PQ}}{\delta b_i} - \sum_P \Psi_n^P \sum_{f\in\mathcal{B}(P)} \frac{\delta \Phi_n^{\text{vis},f}}{\delta b_i} \\
&= -\sum_P \Psi_{m+1}^P \sum_Q \frac{\delta}{\delta b_i}\left[\tau_{km}^{PQ} \mathsf{n}_k^{PQ}\right] - \sum_P \Psi_5^P \sum_Q \frac{\delta}{\delta b_i}\left[v_\ell^{A,PQ} \tau_{k\ell}^{PQ} \mathsf{n}_k^{PQ} + q_k^{PQ} \mathsf{n}_k^{PQ}\right] \\
&\quad - \sum_P \Psi_{m+1}^P \sum_f \frac{\delta}{\delta b_i}\left[\tau_{km}^P \mathsf{n}_k^f\right] - \sum_P \Psi_5^P \sum_f \frac{\delta}{\delta b_i}\left[v_\ell^{A,P} \tau_{k\ell}^P \mathsf{n}_k^f + q_k^P \mathsf{n}_k^f\right] \\
&= \underbrace{-\sum_P \sum_Q \left[\Psi_{m+1}^P + \Psi_5^P v_m^{A,PQ}\right] \mathsf{n}_k^{PQ} \frac{\delta \tau_{km}^{PQ}}{\delta b_i} - \sum_P \sum_f \left(\Psi_{m+1}^P + \Psi_5^P v_m^{A,P}\right) \mathsf{n}_k^f \frac{\delta \tau_{km}^P}{\delta b_i}}_{\mathcal{I}^{\tau_{km}}} \\
&\quad \underbrace{-\sum_P \sum_Q \Psi_5^P \mathsf{n}_k^{PQ} \frac{\delta q_k^{PQ}}{\delta b_i} - \sum_P \sum_f \Psi_5^P \mathsf{n}_k^f \frac{\delta q_k^P}{\delta b_i}}_{\mathcal{I}^{q_k}} \\
&\quad \underbrace{-\sum_P \sum_Q \Psi_5^P \tau_{km}^{PQ} \mathsf{n}_k^{PQ} \frac{\delta v_m^{A,PQ}}{\delta b_i} - \sum_P \sum_f \Psi_5^P \tau_{km}^P \mathsf{n}_k^f \frac{\delta v_m^{A,P}}{\delta b_i}}_{\mathcal{I}^{v_m^A}} \\
&\quad \underbrace{-\sum_P \sum_Q \left[\Psi_{m+1}^P \tau_{km}^{PQ} + \Psi_5^P \left(v_\ell^{A,PQ} \tau_{k\ell}^{PQ} + q_k^{PQ}\right)\right] \frac{\delta \mathsf{n}_k^{PQ}}{\delta b_i}}_{\mathcal{I}^{\mathsf{n}_k}}
\end{aligned}$$

$$\underbrace{-\sum_f \left[\Psi^P_{m+1}\tau^P_{km}+\Psi^P_5\left(v^{A,P}_\ell\tau^P_{k\ell}+q^P_k\right)\right]\frac{\delta \mathsf{n}^f_k}{\delta b_i}}_{\mathcal{I}^{\mathsf{n}_k}} \tag{4.17}$$

Using Eqs. 2.3 and 2.58, $\mathcal{I}^{\tau_{km}}$ is expanded as

$$I^{\tau_{km}} = \underbrace{-\frac{1}{2}\sum_P\sum_Q\left[\left(\Psi^P_{m+1}-\Psi^Q_{m+1}\right)+\left(\Psi^P_5-\Psi^Q_5\right)v^{A,PQ}_m\right]\mathsf{n}^{PQ}_k\frac{\tau^{PQ}_{km}}{\mu^{PQ}+\mu^{PQ}_t}\frac{\partial\mu^P}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow Suth}$$
$$\underbrace{-\frac{1}{2}\sum_P\sum_Q\left[\left(\Psi^P_{m+1}-\Psi^Q_{m+1}\right)+\left(\Psi^P_5-\Psi^Q_5\right)v^{A,PQ}_m\right]\mathsf{n}^{PQ}_k\frac{\tau^{PQ}_{km}}{\mu^{PQ}+\mu^{PQ}_t}\frac{\delta\mu^P_t}{\delta b_i}}_{\mathcal{I}^{\mu_t}}$$
$$\underbrace{-\sum_P\sum_Q\tau^{P,PQ}_{\text{adj},km}\frac{\delta}{\delta b_i}\left(\left.\frac{\partial v^A_m}{\partial x_k}\right|_{PQ}\right)+\sum_P\sum_f\tau^{P,f}_{\text{adj},km}\frac{\delta}{\delta b_i}\left(\frac{\partial v^{A,P}_m}{\partial x_k}\right)}_{\mathcal{I}^{\tau_{km},1}}$$

The adjoint stress $\tau^{P,PQ}_{\text{adj},km}$ is defined as

$$\tau^{*,\alpha\beta}_{\text{adj},km}=\frac{\mu^{\alpha\beta}+\mu^{\alpha\beta}_t}{\text{Re}_0}\left[\left(\Psi^*_{m+1}+\Psi^*_5 v^{A,\alpha\beta}_m\right)\mathsf{n}^{\alpha\beta}_k+\left(\Psi^*_{k+1}+\Psi^*_5 v^{A,\alpha\beta}_k\right)\mathsf{n}^{\alpha\beta}_m-\frac{2}{3}\delta_{km}\left(\Psi^*_{\ell+1}+\Psi^*_5 v^{A,\alpha\beta}_\ell\right)\mathsf{n}^{\alpha\beta}_\ell\right]$$

where $\alpha\beta$ denotes the edge for which $\tau_{\text{adj},km}$ is computed and $*$ stands for either α or β, i.e. the node where Ψ is computed.

Then, $\mathcal{I}^{\tau_{km},1}$ is developed as

$$\mathcal{I}^{\tau_{km},1} = -\sum_P\sum_Q\tau^{P,PQ}_{\text{adj},km}\frac{\delta}{\delta b_i}\left[\left.\overline{\frac{\partial v^A_m}{\partial x_k}}\right|_{PQ}-\left.\overline{\frac{\partial v^A_m}{\partial x_\ell}}\right|_{PQ}\hat{\mathsf{t}}^{PQ}_\ell\hat{\mathsf{t}}^{PQ}_k+\frac{v^{A,Q}_m-v^{A,P}_m}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}}\hat{\mathsf{t}}^{PQ}_k\right]$$
$$= -\sum_P\sum_Q\left(\tau^{P,PQ}_{\text{adj},km}-\tau^{Q,PQ}_{\text{adj},km}\right)\frac{\delta}{\delta b_i}\left[\frac{1}{2}\left(\delta_{k\ell}-\hat{\mathsf{t}}^{PQ}_\ell\hat{\mathsf{t}}^{PQ}_k\right)\frac{\partial v^{A,P}_m}{\partial x_\ell}-\frac{v^{A,P}_m}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}}\hat{\mathsf{t}}^{PQ}_k\right]$$
$$= \underbrace{\sum_P\sum_Q\left(\tau^{P,PQ}_{\text{adj},km}-\tau^{Q,PQ}_{\text{adj},km}\right)\frac{\hat{\mathsf{t}}^{PQ}_k}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}}\frac{\partial v^{A,P}_m}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{-\sum_P\left[\mathcal{T}^P_{\ell m}D^P_\ell+\sum_\Lambda\mathcal{T}^\Lambda_{\ell m}Z^{\Lambda P}_\ell\right]\frac{\partial v^{A,P}_m}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\to\text{FAE\_MF}}\underbrace{-\sum_P\mathcal{T}^P_{\ell m}v^{A,P}_m\frac{\delta D^P_\ell}{\delta b_i}}_{\to\text{SD}}$$

$$\underbrace{+\frac{1}{2}\sum_P\sum_Q\left(\tau^{P,PQ}_{\text{adj},km}-\tau^{Q,PQ}_{\text{adj},km}\right)\frac{\partial v^{A,P}_m}{\partial x_\ell}\left(\hat{\mathsf{t}}^{PQ}_\ell\frac{\delta\hat{\mathsf{t}}^{PQ}_k}{\delta b_i}+\hat{\mathsf{t}}^{PQ}_k\frac{\delta\hat{\mathsf{t}}^{PQ}_\ell}{\delta b_i}\right)-\sum_P\sum_\Lambda\mathcal{T}^\Lambda_{\ell m}v^{A,P}_m\frac{\delta Z^{\Lambda P}_\ell}{\delta b_i}}_{\to\text{SD}}$$

$$\underbrace{-\sum_P\sum_Q\left(\tau^{P,PQ}_{\text{adj},km}-\tau^{Q,PQ}_{\text{adj},km}\right)v^{A,P}_m\left(-\frac{1}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}}\frac{\delta\hat{\mathsf{t}}^{PQ}_k}{\delta b_i}+\hat{\mathsf{t}}^{PQ}_k\frac{x^{PQ}_\ell}{\sqrt[3]{x^{PQ}_\ell x^{PQ}_\ell}}\frac{\delta x^{PQ}_\ell}{\delta b_i}\right)}_{\to\text{SD}}$$

$\mathcal{I}^{q_k}$ reads

$$\mathcal{I}^{q_k}=\underbrace{-\frac{1}{2}\frac{\mathcal{C}_p}{\text{Re}_0}\sum_P\sum_Q\left(\Psi^P_5-\Psi^Q_5\right)\frac{1}{\text{Pr}}\left.\frac{\partial T}{\partial x_k}\right|_{PQ}\mathsf{n}^{PQ}_k\frac{\partial\mu^P}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\to Suth}$$

$$\underbrace{-\frac{1}{2}\frac{\mathcal{C}_p}{\text{Re}_0}\sum_P\sum_Q\left(\Psi^P_5-\Psi^Q_5\right)\frac{1}{\text{Pr}_\text{t}}\left.\frac{\partial T}{\partial x_k}\right|_{PQ}\mathsf{n}^{PQ}_k\frac{\delta\mu^P_t}{\delta b_i}}_{I^{\mu t}}$$

$$\underbrace{-\sum_P\sum_Q q^{P,PQ}_{\text{adj},k}\frac{\delta}{\delta b_i}\left(\left.\frac{\partial T}{\partial x_k}\right|_{PQ}\right)-\sum_P\sum_f q^{P,f}_{\text{adj},k}\frac{\delta}{\delta b_i}\left(\frac{\partial T^P}{\partial x_k}\right)}_{\mathcal{I}^{q_k,1}}$$

where $q^{P,PQ}_{\text{adj},k}$ is defined as

$$q^{*,\alpha\beta}_{\text{adj},k}=\frac{\mathcal{C}_p}{\text{Re}_0}\left(\frac{\mu^{\alpha\beta}}{\text{Pr}}+\frac{\mu^{\alpha\beta}_t}{\text{Pr}_\text{t}}\right)\Psi^*_5\mathsf{n}^{\alpha\beta}_k$$

Then,

$$\mathcal{I}^{q_k,1}=\underbrace{\sum_P\sum_Q\left(q^{P,PQ}_{\text{adj},k}-q^{Q,PQ}_{\text{adj},k}\right)\frac{\hat{\mathsf{t}}^{PQ}_k}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}}\frac{\partial T^P}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}-\sum_P\left[\mathcal{Q}^P_\ell D^P_\ell+\sum_\Lambda\mathcal{Q}^\Lambda_\ell Z^{\Lambda P}_\ell\right]\frac{\partial T^P}{\partial U^P_q}\frac{\delta U^P_q}{\delta b_i}}_{\to\text{FAE\_MF}}$$

$$\underbrace{+\frac{1}{2}\sum_P\sum_Q\left(q_{\text{adj},k}^{P,PQ}-q_{\text{adj},k}^{Q,PQ}\right)\frac{\partial T^P}{\partial x_\ell}\left(\hat{\mathsf{t}}_k^{PQ}\frac{\delta\hat{\mathsf{t}}_\ell^{PQ}}{\delta b_i}+\hat{\mathsf{t}}_\ell^{PQ}\frac{\delta\hat{\mathsf{t}}_k^{PQ}}{\delta b_i}\right)-\sum_P\sum_\Lambda\mathcal{Q}_\ell^\Lambda T^P\frac{\delta Z_\ell^{\Lambda P}}{\delta b_i}}_{\rightarrow\text{SD}}$$

$$\underbrace{-\sum_P\sum_Q\left(q_{\text{adj},k}^{P,PQ}-q_{\text{adj},k}^{Q,PQ}\right)T^P\left(-\frac{1}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}\frac{\delta\hat{\mathsf{t}}_k^{PQ}}{\delta b_i}+\hat{\mathsf{t}}_k^{PQ}\frac{x_\ell^{PQ}}{\sqrt[3]{x_\ell^{PQ}x_\ell^{PQ}}}\frac{\delta x_\ell^{PQ}}{\delta b_i}\right)-\sum_P\mathcal{Q}_\ell^P T^P\frac{\delta D_\ell^P}{\delta b_i}}_{\rightarrow\text{SD}}$$

where

$$\mathcal{T}_{\ell m}^P=\frac{1}{2}\sum_Q\left(\tau_{\text{adj},km}^{P,PQ}-\tau_{\text{adj},km}^{Q,PQ}\right)\left(\delta_{k\ell}-\hat{\mathsf{t}}_\ell^{PQ}\hat{\mathsf{t}}_k^{PQ}\right)+\sum_f\tau_{\text{adj},\ell m}^{P,f}$$

$$\mathcal{Q}_\ell^P=\frac{1}{2}\sum_Q\left(q_{\text{adj},k}^{P,PQ}-q_{\text{adj},k}^{Q,PQ}\right)\left(\delta_{k\ell}-\hat{\mathsf{t}}_\ell^{PQ}\hat{\mathsf{t}}_k^{PQ}\right)+\sum_f q_{\text{adj},\ell}^{P,f}$$

$\mathcal{I}^{v_m^A}$ and $\mathcal{I}^{\mathsf{n}_k}$ are analyzed as

$$\mathcal{I}^{v_m^A}=\underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi_5^P-\Psi_5^Q\right)\tau_{km}^{PQ}\mathsf{n}_k^{PQ}\frac{\partial v_m^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}-\sum_P\sum_f\Psi_5^P\tau_{km}^P\mathsf{n}_k^f\frac{\partial v_m^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}$$

$$\mathcal{I}^{\mathsf{n}_k}=\underbrace{-\sum_P\sum_Q\left(\tau_{\text{adj},km}^{P,PQ}-\tau_{\text{adj},km}^{Q,PQ}\right)\left[\frac{1}{2}\left(\delta_{k\ell}-\hat{\mathsf{t}}_\ell^{PQ}\hat{\mathsf{t}}_k^{PQ}\right)\frac{\partial T^P}{\partial x_\ell}-\frac{T^P}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}\hat{\mathsf{t}}_k^{PQ}\right]\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}}$$

$$\underbrace{-\sum_P\sum_Q\left(q_{\text{adj},k}^{P,PQ}-q_{\text{adj},k}^{Q,PQ}\right)\left[\frac{1}{2}\left(\delta_{k\ell}-\hat{\mathsf{t}}_\ell^{PQ}\hat{\mathsf{t}}_k^{PQ}\right)\frac{\partial T^P}{\partial x_\ell}-\frac{T^P}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}\hat{\mathsf{t}}_k^{PQ}\right]\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}}$$

$$\underbrace{-\sum_P\sum_Q\left(\Psi_5^P-\Psi_5^Q\right)v_m^{A,P}\tau_{km}^{PQ}\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}+\sum_P\sum_f\left[\Psi_{m+1}^P\tau_{km}^P+\Psi_5^P\left(v_\ell^{A,P}\tau_{\ell k}^P+q_k^P\right)\right]\frac{\delta\mathsf{n}_k^f}{\delta b_i}}_{\rightarrow\text{SD}}$$

$\mathcal{I}^{\mu_t}$ is further analyzed as

$$\mathcal{I}^{\mu_t} = \underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi^P_{m+1}-\Psi^Q_{m+1}\right)\mathsf{n}^{PQ}_k\frac{\tau^{PQ}_{km}}{(\mu+\mu_t)^{PQ}}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]^P\frac{\delta\tilde{\mu}}{\delta b_i}}_{\rightarrow\text{FAE\_SA}}$$
$$\underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi^P_{5}-\Psi^Q_{5}\right)v^{A,PQ}_m\mathsf{n}^{PQ}_k\frac{\tau^{PQ}_{km}}{(\mu+\mu_t)^{PQ}}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]^P\frac{\delta\tilde{\mu}}{\delta b_i}}_{\rightarrow\text{FAE\_SA}}$$
$$\underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi^P_{5}-\Psi^Q_{5}\right)\mathsf{n}^{PQ}_k\frac{\mathcal{C}_p}{\mathrm{Pr_t}}\left.\frac{\partial T}{\partial x_k}\right|_{PQ}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]^P\frac{\delta\tilde{\mu}}{\delta b_i}}_{\rightarrow\text{FAE\_SA}}$$
$$\underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi^P_{m+1}-\Psi^Q_{m+1}\right)\mathsf{n}^{PQ}_k\frac{\tau^{PQ}_{km}}{(\mu+\mu_t)^{PQ}}\left[\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U^P_q}\right]^P\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow Suth}$$
$$\underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi^P_{5}-\Psi^Q_{5}\right)v^{A,PQ}_m\mathsf{n}^{PQ}_k\frac{\tau^{PQ}_{km}}{(\mu+\mu_t)^{PQ}}\left[\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U^P_q}\right]^P\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow Suth}$$
$$\underbrace{-\frac{1}{2}\sum_P\sum_Q\left(\Psi^P_{5}-\Psi^Q_{5}\right)\mathsf{n}^{PQ}_k\frac{\mathcal{C}_p}{\mathrm{Pr_t}}\left.\frac{\partial T}{\partial x_k}\right|_{PQ}\left[\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\frac{\partial\mu}{\partial U^P_q}\right]^P\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow Suth}$$

4.2 Differentiation of the Spalart–Allmaras Equation

In Eq. 4.2, $\mathcal{I}^{\mathrm{SA},P}$ is decomposed in three terms arising from the differentiation of the convection, diffusion and source terms in the Spalart–Allmaras PDE,

$$\mathcal{I}^{\mathrm{SA},P} = \underbrace{\sum_P \tilde{\nu_a}^P\frac{\delta\mathrm{SA}^{c,P}}{\delta b_i}}_{I^{\mathrm{SA},P}_{\mathrm{conv}}} + \underbrace{\sum_P \tilde{\nu_a}^P\frac{\delta\mathrm{SA}^{d,P}}{\delta b_i}}_{I^{\mathrm{SA},P}_{\mathrm{diff}}} + \underbrace{\sum_P \tilde{\nu_a}^P\frac{\delta\mathrm{SA}^{s,P}}{\delta b_i}}_{I^{\mathrm{SA},P}_{\mathrm{src}}}$$

The convection term $\mathcal{I}_{\text{conv}}^{\text{SA},P}$ is developed as

$$\mathcal{I}_{\text{conv}}^{\text{SA},P} = \sum_P \tilde{\nu}_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi_{\text{conv}}^{\tilde{\nu},PQ}}{\delta b_i} + \sum_P \tilde{\nu}_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta\left(\left(\rho v_k^A \tilde{\nu}\right)^f \mathsf{n}_k^f\right)}{\delta b_i} \tag{4.18}$$

The discretization of the first term in Eq. 4.18 can be based on either the FVS or the Roe's scheme, see Eqs. 2.65 and 2.66. Using the FVS scheme

$$\begin{aligned}\sum_P \tilde{\nu}_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi_{\text{conv}}^{\tilde{\nu},PQ}}{\delta b_i}\bigg|_{\text{FVS}} &= \underbrace{\frac{1}{2}\sum_P\sum_Q \left(\tilde{\nu}_a^P - \tilde{\nu}_a^Q\right)\tilde{\mu}^{PQ}\mathsf{n}_k^{PQ}\frac{\partial v_k^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}} \\ &\underbrace{+\sum_P\sum_Q \left(\tilde{\nu}_a^P - \tilde{\nu}_a^Q\right)\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\frac{\partial\tilde{\mu}^{PQ}}{\partial\tilde{\mu}^P}\frac{\delta\tilde{\mu}^P}{\delta b_i}}_{\rightarrow\text{FAE\_SA}} \underbrace{+\frac{1}{2}\sum_P\sum_Q \left(\tilde{\nu}_a^P - \tilde{\nu}_a^Q\right)\tilde{\mu}^{PQ}v_k^{A,P}\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}}\end{aligned} \tag{4.19}$$

while for the Roe scheme

$$\begin{aligned}\sum_P \tilde{\nu}_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi_{\text{conv}}^{\tilde{\nu},PQ}}{\delta b_i}\bigg|_{\text{Roe}} &= \underbrace{\frac{1}{2}\left(\tilde{\nu}_a^P - \tilde{\nu}_a^Q\right)\left[\tilde{\mu}^P\mathsf{n}_k^{PQ} - \frac{1}{2}sign\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\left(\tilde{\mu}^Q - \tilde{\mu}^P\right)\mathsf{n}_k^{PQ}\right]\frac{\partial v_k^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}} \\ &\underbrace{+\frac{1}{2}\left(\tilde{\nu}_a^P - \tilde{\nu}_a^Q\right)\left(v_k^{A,P}\mathsf{n}_k^{PQ} + \left|v_k^{A,PQ}\mathsf{n}_k^{PQ}\right|\right)\frac{\delta\tilde{\mu}^P}{\delta b_i}}_{\rightarrow\text{FAE\_SA}} \\ &\underbrace{+\frac{1}{2}\left(\tilde{\nu}_a^P - \tilde{\nu}_a^Q\right)\left[v_k^{A,P}\tilde{\mu}^P - \frac{1}{2}sign\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\left(\tilde{\mu}^Q - \tilde{\mu}^P\right)v_k^{A,P}\right]\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}}\end{aligned} \tag{4.20}$$

The last term in Eq. 4.18 is analyzed as

$$\sum_P \tilde{\nu}_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta\left(\left(\rho v_k^A\tilde{\nu}\right)^f \mathsf{n}_k^f\right)}{\delta b_i} = \underbrace{\sum_P \tilde{\nu}_a^P \sum_f \frac{\partial\left(v_k^A\tilde{\mu}\right)^f}{\partial U_q^P}\mathsf{n}_k^f\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}$$

$$+\underbrace{\sum_P \tilde{\nu}_a^P \sum_f \frac{\partial\left(v_k^A \tilde{\mu}\right)^f}{\partial \tilde{\mu}^P} \mathsf{n}_k^f \frac{\delta \tilde{\mu}^P}{\delta b_i}}_{\rightarrow \text{FAE\_SA}} + \underbrace{\sum_P \tilde{\nu}_a^P \sum_f \left(v_k^A \tilde{\mu}\right)^f \frac{\delta \mathsf{n}_k^f}{\delta b_i}}_{\rightarrow \text{SD}} \tag{4.21}$$

The diffusion term $\mathcal{I}_{\text{diff}}^{\text{SA},P}$ is treated as

$$\begin{aligned}
\mathcal{I}_{\text{diff}}^{\text{SA},P} &= \sum_P \tilde{\nu}_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta \Phi_{\text{diff}}^{\tilde{\nu},PQ}}{\delta b_i} + \sum_P \tilde{\nu}_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta \Phi_{\text{diff}}^{\tilde{\nu},f}}{\delta b_i} \\
&= \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P \sum_Q \tilde{\nu}_a^P \left[\nu^{PQ} + (1+c_{b_2})\,\tilde{\nu}^{PQ} - c_{b_2}\tilde{\nu}^P\right] \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{PQ} \mathsf{n}_k^{PQ} \frac{\partial \rho^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\quad \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P \tilde{\nu}_a^P \sum_f \left[\nu^f + (1+c_{b_2})\,\tilde{\nu}^f - c_{b_2}\tilde{\nu}^P\right] \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{f} \mathsf{n}_k^f \frac{\partial \rho^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\quad \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P \sum_Q \left[\frac{(\rho\tilde{\nu}_a)^P - (\rho\tilde{\nu}_a)^Q}{2}(1+c_{b_2}) - c_{b_2}(\rho\tilde{\nu}_a)^P\right] \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{PQ} \mathsf{n}_k^{PQ} \frac{\partial \tilde{\nu}^P}{\partial \rho^P} \frac{\partial \rho^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\quad \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P (\rho\tilde{\nu}_a)^P \sum_f \left((1+c_{b_2})\frac{\partial \tilde{\nu}^f}{\partial \tilde{\nu}^P} - c_{b_2}\right) \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{f} \mathsf{n}_k^f \frac{\partial \tilde{\nu}^P}{\partial \rho^P} \frac{\partial \rho^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\quad \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P \left[\sum_Q \frac{(\rho\tilde{\nu}_a)^P - (\rho\tilde{\nu}_a)^Q}{2} \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{PQ} \mathsf{n}_k^{PQ} + \tilde{\nu}_a^P \sum_f \rho^P \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{f} \mathsf{n}_k^f \frac{\partial \nu^f}{\partial \nu^P}\right] \frac{\partial \nu^P}{\partial \rho^P} \frac{\partial \rho^P}{\partial U_q^P} \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
&\quad \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P \sum_Q \left[\frac{(\rho\tilde{\nu}_a)^P - (\rho\tilde{\nu}_a)^Q}{2}(1+c_{b_2}) - c_{b_2}(\rho\tilde{\nu}_a)^P\right] \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{PQ} \mathsf{n}_k^{PQ} \frac{\partial \tilde{\nu}^P}{\partial \tilde{\mu}^P} \frac{\delta \tilde{\mu}^P}{\delta b_i}}_{\rightarrow \text{FAE_SA}} \\
&\quad \underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma} \sum_P (\rho\tilde{\nu}_a)^P \sum_f \left((1+c_{b_2})\frac{\partial \tilde{\nu}^f}{\partial \tilde{\nu}^P} - c_{b_2}\right) \left.\frac{\partial \tilde{\nu}}{\partial x_k}\right|_{f} \mathsf{n}_k^f \frac{\partial \tilde{\nu}^P}{\partial \tilde{\mu}^P} \frac{\delta \tilde{\mu}^P}{\delta b_i}}_{\rightarrow \text{FAE_SA}}
\end{aligned}$$

$$
\begin{aligned}
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\sum_P\left[\sum_Q\frac{(\rho\tilde{\nu_a})^P-(\tilde{\nu_a}\rho)^Q}{2}\frac{\partial\tilde{\nu}}{\partial x_k}\bigg|_{PQ}\mathsf{n}_k^{PQ}+(\rho\tilde{\nu_a})^P\sum_f\frac{\partial\tilde{\nu}}{\partial x_k}\bigg|_f\mathsf{n}_k^f\frac{\partial\nu^f}{\partial\nu^P}\right]\frac{\partial\nu^P}{\partial\mu^P}\frac{\partial\mu^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow Suth}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\sum_P\sum_Q\left[\frac{(\rho\tilde{\nu_a})^P-(\rho\tilde{\nu_a})^Q}{2}\left[\nu^P+(1+c_{b_2})\,\tilde{\nu}^P\right]-c_{b_2}\,(\rho\tilde{\nu_a}\tilde{\nu})^P\right]\frac{\partial\tilde{\nu}}{\partial x_k}\bigg|_{PQ}\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow \mathrm{SD}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\sum_P(\rho\tilde{\nu_a})^P\sum_f\left[\nu^f+(1+c_{b_2})\,\tilde{\nu}^f-c_{b_2}\tilde{\nu}^P\right]\frac{\partial\tilde{\nu}}{\partial x_k}\bigg|_f\frac{\delta\mathsf{n}_k^f}{\delta b_i}}_{\rightarrow \mathrm{SD}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\sum_P(\rho\tilde{\nu_a})^P\sum_Q\left[\nu^{PQ}+(1+c_{b_2})\,\tilde{\nu}^{PQ}-c_{b_2}\tilde{\nu}^P\right]\mathsf{n}_k^{PQ}\frac{\delta}{\delta b_i}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\bigg|_{PQ}\right)}_{\mathcal{I}_{\mathrm{diff}}^{\mathrm{SA},1}}\\
&\underbrace{-\frac{1}{\mathrm{Re}_0\,\sigma}\sum_P(\rho\tilde{\nu_a})^P\sum_f\left[\nu^f+(1+c_{b_2})\,\tilde{\nu}^f-c_{b_2}\tilde{\nu}^P\right]\mathsf{n}_k^f\frac{\delta}{\delta b_i}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\bigg|^P\right)}_{\mathcal{I}_{\mathrm{diff}}^{\mathrm{SA},1}}
\end{aligned}
\tag{4.22}
$$

Then, $\mathcal{I}_{\mathrm{diff}}^{\mathrm{SA},1}$ is expressed as

$$
\begin{aligned}
\mathcal{I}_{\mathrm{diff}}^{\mathrm{SA},1}=&\underbrace{-\sum_P\left[\sum_Q\left(\mathcal{C}_{\mathrm{SA},k}^{P,PQ}-\mathcal{C}_{\mathrm{SA},k}^{Q,PQ}\right)\frac{\hat{\mathsf{t}}_k^{PQ}}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}-\left(\mathscr{C}_{\mathrm{SA},\ell}^P D_\ell^P+\sum_\Lambda\mathscr{C}_{\mathrm{SA},\ell}^\Lambda Z_\ell^{\Lambda,P}\right)\right]\frac{\tilde{\nu}^P}{\rho^P}\frac{\partial\rho^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \mathrm{FAE\_MF}}\\
&\underbrace{+\sum_P\left[\sum_Q\left(\mathcal{C}_{\mathrm{SA},k}^{P,PQ}-\mathcal{C}_{\mathrm{SA},k}^{Q,PQ}\right)\frac{\hat{\mathsf{t}}_k^{PQ}}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}-\left(\mathscr{C}_{\mathrm{SA},\ell}^P D_\ell^P+\sum_\Lambda\mathscr{C}_{\mathrm{SA},\ell}^\Lambda Z_\ell^{\Lambda,P}\right)\right]\frac{1}{\rho^P}\frac{\delta\tilde{\mu}^P}{\delta b_i}}_{\rightarrow \mathrm{FAE\_SA}}\\
&\underbrace{+\frac{1}{2}\sum_P\sum_Q\left(\mathcal{C}_{\mathrm{SA},k}^{P,PQ}-\mathcal{C}_{\mathrm{SA},k}^{Q,PQ}\right)\frac{\partial\tilde{\nu}^P}{\partial x_\ell}\left(\hat{\mathsf{t}}_k^{PQ}\frac{\delta\hat{\mathsf{t}}_\ell^{PQ}}{\delta b_i}+\hat{\mathsf{t}}_\ell^{PQ}\frac{\delta\hat{\mathsf{t}}_k^{PQ}}{\delta b_i}\right)-\sum_P\sum_\Lambda\mathscr{C}_{\mathrm{SA},\ell}^\Lambda\tilde{\nu}^P\frac{\delta Z_\ell^{\Lambda P}}{\delta b_i}}_{\mathrm{SD}}\\
&\underbrace{-\sum_P\sum_Q\left(\mathcal{C}_{\mathrm{SA},k}^{P,PQ}-\mathcal{C}_{\mathrm{SA},k}^{Q,PQ}\right)\tilde{\nu}^P\left(-\frac{1}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}\frac{\delta\hat{\mathsf{t}}_k^{PQ}}{\delta b_i}+\hat{\mathsf{t}}_k^{PQ}\frac{x_\ell^{PQ}}{\sqrt[3]{x_\ell^{PQ}x_\ell^{PQ}}}\frac{\delta x_\ell^{PQ}}{\delta b_i}\right)-\sum_P\mathscr{C}_{\mathrm{SA},\ell}^P\tilde{\nu}^P\frac{\delta D_\ell^P}{\delta b_i}}_{\mathrm{SD}}
\end{aligned}
\tag{4.23}
$$

where

$$\mathscr{C}^{P,PQ}_{\mathrm{SA},\ell}=\frac{1}{2}\sum_{Q}\left(\mathcal{C}^{P,PQ}_{\mathrm{SA},k}-\mathcal{C}^{Q,PQ}_{\mathrm{SA},k}\right)\left(\delta_{k\ell}-\hat{\mathsf{t}}^{PQ}_{\ell}\hat{\mathsf{t}}^{PQ}_{k}\right)+\sum_{f}\mathcal{C}^{P,f}_{\mathrm{SA},\ell}$$

$$\mathcal{C}^{*,\alpha\beta}_{\mathrm{SA},k}=\frac{1}{\mathrm{Re}_0\,\sigma}\left[(\rho\tilde{\nu}_a)^*\left[\nu^{\alpha\beta}+(1+c_{b_2})\,\tilde{\nu}^{\alpha\beta}\right]-c_{b_2}\,(\rho\tilde{\nu}_a\tilde{\nu})^*\right]\mathsf{n}^{\alpha\beta}_k$$

The source term $\mathcal{I}^{\mathrm{SA},P}_{\mathrm{src}}$ is expanded as

$$\begin{aligned}\mathcal{I}^{\mathrm{SA},P}_{src}=&\underbrace{\sum_P\frac{\tilde{\nu}^P_a}{\mathrm{Re}_0}\left[-\left(c_{w_1}f_w-\frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\left(\frac{\tilde{\nu}}{\Delta}\right)^2+\mathcal{C}_r\mathcal{P}\,(r,\rho)\right]^P\frac{\partial\rho^P}{\partial U^P_q}\Omega_P\frac{\delta U^P_q}{\delta b_i}}_{\rightarrow\text{FAE_MF}}\\
&\underbrace{+\sum_P\tilde{\nu}^P_a\left[-\gamma_{eff}c_{b_1}\,(1-f_{t_2})\,\tilde{S}+\frac{2}{\mathrm{Re}_0}\left(c_{w_1}f_w-\frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\frac{\tilde{\nu}}{\Delta^2}+\mathcal{C}_r\mathcal{P}\,(r,\tilde{\mu})\right]^P\Omega_P\frac{\delta\tilde{\mu}^P}{\delta b_i}}_{\rightarrow\text{FAE_SA}}\\
&\underbrace{+\sum_P\tilde{\nu}^P_a\mathcal{C}^P_{\tilde{S}}\Omega_P\frac{\delta\tilde{S}^P}{\delta b_i}}_{\mathcal{I}^{\tilde{S}}}+\underbrace{\sum_P\tilde{\nu}^P_a\mathcal{C}^P_{f_{t_2}}\Omega_P\frac{\delta f^P_{t_2}}{\delta b_i}}_{\mathcal{I}^{f_{t_2}}}+\underbrace{\sum_P\tilde{\nu}^P_a\mathcal{C}^P_{\gamma_{eff}}\Omega_P\frac{\delta\gamma^P_{eff}}{\delta b_i}}_{\mathcal{I}^{\gamma_{eff}}}\\
&\underbrace{+\sum_P\tilde{\nu}^P_a\left[\mathcal{C}_r\mathcal{P}\,(r,\Delta)-2\rho\left(c_{w_1}f_w-\frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\frac{\tilde{\nu}^2}{\Delta^3}\right]^P\Omega_P\frac{\delta\Delta^P}{\delta b_i}}_{\rightarrow\text{FAE_}\Delta}+\underbrace{\sum_P\tilde{\nu}^P_a S^P_{\tilde{\nu}}\frac{\delta\Omega_P}{\delta b_i}}_{\rightarrow\text{SD}}\end{aligned}\tag{4.24}$$

where $\mathcal{C}_r,\mathcal{C}_{\tilde{S}},\mathcal{C}_{f_{t_2}},\mathcal{C}_{\gamma_{eff}}$ are defined in Sec. 3.2. Term $\mathcal{I}^{\tilde{S}}$ is developed as

$$\begin{aligned}\mathcal{I}^{\tilde{S}}=&\underbrace{\sum_P\left[\tilde{\nu}_a\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)\frac{\partial\rho}{\partial U_\ell}\right]^P\Omega_P\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow\text{FAE_MF}}+\underbrace{\sum_P\left[\frac{\tilde{\nu}_a}{\rho}\mathcal{C}_{\tilde{S}}\left[\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)+\mathcal{C}_\chi\mathcal{P}\,(\chi,\tilde{\mu})\right]\right]^P\Omega_P\frac{\delta\tilde{\mu}^P}{\delta b_i}}_{\rightarrow\text{FAE_SA}}\\
&\underbrace{+\sum_P\left[\tilde{\nu}_a\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\zeta\right)\right]^P\Omega_P\frac{\delta\zeta^P}{\delta b_i}}_{\mathcal{I}^{\zeta}}+\underbrace{\sum_P\left[\tilde{\nu}_a\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},S\right)\right]^P\Omega_P\frac{\delta S^P}{\delta b_i}}_{\mathcal{I}^{S}}\\
&\underbrace{+\sum_P\left[\tilde{\nu}_a\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\Delta\right)\right]^P\Omega_P\frac{\delta\Delta^P}{\delta b_i}}_{\rightarrow\text{FAE_}\Delta}+\underbrace{\sum_P\left[\tilde{\nu}_a\mathcal{C}_{\tilde{S}}\mathcal{P}\,(\chi,\mu)\frac{\partial\mu}{\partial U_\ell}\right]^P\Omega_P\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow\textit{Suth}}\end{aligned}\tag{4.25}$$

Integrals $\mathcal{I}^{f_{t_2}}$ and $I^{\gamma_{eff}}$ are developed as follows: terms $\mathcal{C}_{\gamma_{eff}}, \mathcal{C}_{\zeta}^{\text{SA}}, \mathcal{C}_{S}^{\text{SA}}$ are defined in Sec. 3.2. For the Spalart–Allmaras turbulence model (no transition model)

$$\mathcal{I}_{\text{SA}}^{f_{t_2}} = \underbrace{\sum_P \left[\tilde{\nu}_a \mathcal{C}_{f_{t_2}} \mathcal{P}\left(f_{t_2}, \chi\right) \mathcal{P}\left(\chi, \tilde{\mu}\right)\right]^P \Omega_P \frac{\delta \tilde{\mu}^P}{\delta b_i}}_{\rightarrow \text{FAE\_SA}} + \underbrace{\sum_P \left[\tilde{\nu}_a \mathcal{C}_{f_{t_2}} \mathcal{P}\left(f_{t_2}, \chi\right) \mathcal{P}\left(\chi, \mu\right)\right]^P \Omega_P \frac{\partial \mu^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \textit{Suth}}$$

$$\mathcal{I}_{\text{SA}}^{\gamma_{eff}} = 0 \tag{4.26}$$

The development of the *TDDC* adjoint is shown for the SA-sLM2015 transition model variant. Its extension to the rest variants is straightforward. It is

$$\begin{aligned}
\mathcal{I}_{\text{SA-sLM}}^{f_{t_2}} &= 0 \\
\mathcal{I}_{\text{SA-sLM}}^{\gamma_{eff}} &= \underbrace{-\sum_P \left[\tilde{\nu}_a \frac{\gamma}{\rho} \mathcal{C}_{\gamma_{eff}}\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} + \underbrace{\sum_P \left[\tilde{\nu}_a \frac{1}{\rho} \mathcal{C}_{\gamma_{eff}}\right]^P \Omega_P \frac{\delta \left(\rho\gamma\right)^P}{\delta b_i}}_{\rightarrow \text{FAE_}\gamma}
\end{aligned} \tag{4.27}$$

Finally, terms $\mathcal{I}^{\zeta}$ and $\mathcal{I}^{S}$ are written as

$$\begin{aligned}
\mathcal{I}^{\zeta} = & \underbrace{\sum_P \left[\tilde{\nu}_a \frac{\mathcal{C}_{\zeta}^{\text{SA}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{knr} \frac{\partial v_r^A}{\partial x_n}\right]^P \Omega_P D_\ell^P \frac{\partial v_m^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
& + \underbrace{\sum_P \sum_\Lambda \left[\tilde{\nu}_a \frac{\mathcal{C}_{\zeta}^{\text{SA}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{knr} \frac{\partial v_r^A}{\partial x_n}\right]^\Lambda \Omega_\Lambda Z_\ell^{\Lambda P} \frac{\partial v_m^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
& + \underbrace{\sum_P \left[\tilde{\nu}_a \frac{\mathcal{C}_{\zeta}^{\text{SA}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{knr} \frac{\partial v_r^A}{\partial x_n} v_m^A\right]^P \Omega_P \frac{\delta D_\ell^P}{\delta b_i} + \sum_P \sum_\Lambda \left[\tilde{\nu}_a \frac{\mathcal{C}_{\zeta}^{\text{SA}}}{\zeta} \varepsilon_{k\ell m} \varepsilon_{knr} \frac{\partial v_r^A}{\partial x_n}\right]^\Lambda \Omega_\Lambda v_m^{A,P} \frac{\delta Z_\ell^{\Lambda P}}{\delta b_i}}_{\rightarrow \text{SD}} \\
\mathcal{I}^{S} = & \underbrace{\sum_P \left[\mathcal{C}_S^{\text{SA},P} \frac{S_{km}^P}{S^P} D_m^P + \sum_\Lambda \mathcal{C}_S^{\text{SA},\Lambda} \frac{S_{km}^\Lambda}{S^\Lambda} Z_m^{\Lambda P}\right] \frac{\partial v_k^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE_MF}} \\
& + \underbrace{\sum_P \left[\mathcal{C}_S^{\text{SA},P} \frac{S_{km}^P}{S^P} v_k^P \frac{\delta D_m^P}{\delta b_i} + \sum_\Lambda \mathcal{C}_S^{\text{SA},\Lambda} \frac{S_{km}^\Lambda}{S^\Lambda} v_k^{A,P} \frac{\delta Z_m^\Lambda}{\delta b_i}\right]}_{\rightarrow \text{SD}}
\end{aligned} \tag{4.28}$$

4.3 Differentiation of the $\gamma-\tilde{Re}_{\theta t}$ Transition Model Equations

Terms $\mathcal{I}^{\gamma,P}, \mathcal{I}^{\tilde{Re}_{\theta t},P}$ in Eq. 4.2 split in three terms each, arising from the differentiation of the convection, diffusion and source terms of the γ and $\tilde{Re}_{\theta t}$ PDEs,

$$\mathcal{I}^{\gamma,P} = \underbrace{\sum_P \gamma_a^P \frac{\delta\gamma^{c,P}}{\delta b_i}}_{\mathcal{I}^{\gamma,P}_{conv}} + \underbrace{\sum_P \gamma_a^P \frac{\delta\gamma^{d,P}}{\delta b_i}}_{\mathcal{I}^{\gamma,P}_{diff}} + \underbrace{\sum_P \gamma_a^P \frac{\delta\gamma^{s,P}}{\delta b_i}}_{\mathcal{I}^{\gamma,P}_{src}}$$

$$\mathcal{I}^{\tilde{Re}_{\theta t}} = \underbrace{\sum_P \tilde{Re}_a^P \frac{\delta\tilde{Re}_{\theta t}^{c,P}}{\delta b_i}}_{\mathcal{I}^{\tilde{Re}_{\theta t},P}_{conv}} + \underbrace{\sum_P \tilde{Re}_a^P \frac{\delta\tilde{Re}_{\theta t}^{d,P}}{\delta b_i}}_{\mathcal{I}^{\tilde{Re}_{\theta t},P}_{diff}} + \underbrace{\sum_P \tilde{Re}_a^P \frac{\delta\tilde{Re}_{\theta t}^{s,P}}{\delta b_i}}_{\mathcal{I}^{\tilde{Re}_{\theta t},P}_{src}}$$

The convection terms $\mathcal{I}^{\gamma,P}_{\text{conv}}$ and $\mathcal{I}^{\tilde{Re}_{\theta t},P}_{\text{conv}}$ are developed as

$$\mathcal{I}^{\gamma,P}_{\text{conv}} = \sum_P \gamma_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi^{\gamma_a,PQ}_{\text{conv}}}{\delta b_i} + \sum_P \gamma_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta\left(\left(\rho v_k^A \gamma\right)^f \mathsf{n}_k^f\right)}{\delta b_i} \tag{4.29}$$

$$\mathcal{I}^{\tilde{Re}_{\theta t},P}_{\text{conv}} = \sum_P \tilde{Re}_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi^{\gamma_a,PQ}_{\text{conv}}}{\delta b_i} + \sum_P \tilde{Re}_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta\left(\left(\rho v_k^A \tilde{Re}_{\theta t}\right)^f \mathsf{n}_k^f\right)}{\delta b_i} \tag{4.30}$$

The mathematical development of the convection terms is similar to that presented for the Spalart–Allmaras turbulence model. The discretization of the first terms in Eq. 4.29, only for the internal fluxes is expressed based either on the Roe's or the FVS scheme,

$$\sum_P \varphi_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta\Phi^{\varphi,PQ}_{\text{conv}}}{\delta b_i}\bigg|_{\text{FVS}} = \underbrace{\frac{1}{2}\sum_P\sum_Q \left(\varphi_a^P - \varphi_a^Q\right)(\rho\varphi)^{PQ}\,\mathsf{n}_k^{PQ}\frac{\partial v_k^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}$$

$$\underbrace{+\sum_P\sum_Q \left(\varphi_a^P - \varphi_a^Q\right)\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\frac{\partial(\rho\varphi)^{PQ}}{\partial(\rho\varphi)^P}\frac{\delta(\rho\varphi)^P}{\delta b_i}}_{\rightarrow\text{FAE\_}\varphi} \underbrace{+\frac{1}{2}\sum_P\sum_Q \left(\varphi_a^P - \varphi_a^Q\right)(\rho\varphi)^{PQ}\,v_k^{A,P}\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow\text{SD}} \tag{4.31}$$

$$\sum_P \varphi_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta \Phi_{\text{conv}}^{\varphi,PQ}}{\delta b_i}\bigg|_{\text{Roe}} =$$

$$\underbrace{\frac{1}{2}\left(\varphi_a^P - \varphi_a^Q\right)\left[(\rho\varphi)^P \mathsf{n}_k^{PQ} - \frac{1}{2} sign\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\left((\rho\varphi)^Q - (\rho\varphi)^P\right)\mathsf{n}_k^{PQ}\right]\frac{\partial v_k^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{+\frac{1}{2}\left(\varphi_a^P - \varphi_a^Q\right)\left(v_k^{A,P}\mathsf{n}_k^{PQ} + \left|v_k^{A,PQ}\mathsf{n}_k^{PQ}\right|\right)\frac{\delta\left(\rho\varphi\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\varphi}$$

$$\underbrace{+\frac{1}{2}\left(\varphi_a^P - \varphi_a^Q\right)\left[v_k^{A,P}(\rho\varphi)^P - \frac{1}{2} sign\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\left((\rho\varphi)^Q - (\rho\varphi)^P\right)v_k^{A,P}\right]\frac{\delta \mathsf{n}_k^{PQ}}{\delta b_i}}_{\rightarrow \text{SD}} \tag{4.32}$$

where φ and φ_a stand for $\gamma, \tilde{Re}_{\theta t}$ and $\gamma_a, \tilde{Re}_a$, respectively. The last terms in Eq. 4.29, which is related to the boundary fluxes, read

$$\sum_P \varphi_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta\left(\left(\rho v_k^A \varphi\right)^f \mathsf{n}_k^f\right)}{\delta b_i} = \underbrace{\sum_P \varphi_a^P \sum_f \frac{\partial\left(\rho v_k^A \varphi\right)^f}{\partial U_q^P}\mathsf{n}_k^f \frac{\delta U_q^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{+\sum_P \varphi_a^P \sum_f \frac{\partial\left(\rho v_k^A \varphi\right)^f}{\partial\left(\rho\varphi\right)^P}\mathsf{n}_k^f \frac{\delta\left(\rho\varphi\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\varphi} + \underbrace{\sum_P \varphi_a^P \sum_f \left(\rho v_k^A \varphi\right)^f \frac{\delta \mathsf{n}_k^f}{\delta b_i}}_{\rightarrow \text{SD}} \tag{4.33}$$

The diffusion terms $\mathcal{I}_{\text{diff}}^{\gamma,P}$ and $\mathcal{I}_{\text{diff}}^{\tilde{Re}_{\theta t},P}$ are expanded as

$$\mathcal{I}_{\text{diff}}^{\gamma,P} = -\sum_P \gamma_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta \Phi_{\text{diff}}^{\gamma,PQ}}{\delta b_i} - \sum_P \gamma_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta \Phi_{\text{diff}}^{\gamma,f}}{\delta b_i} \tag{4.34}$$

$$\mathcal{I}_{\text{diff}}^{\tilde{Re}_{\theta t},P} = -\sum_P \tilde{Re}_a^P \sum_{Q\in\mathcal{N}(P)} \frac{\delta \Phi_{\text{diff}}^{\tilde{Re}_{\theta t},PQ}}{\delta b_i} - \sum_P \tilde{Re}_a^P \sum_{f\in\mathcal{B}(P)} \frac{\delta \Phi_{\text{diff}}^{\tilde{Re}_{\theta t},f}}{\delta b_i} \tag{4.35}$$

where

$$\mathcal{I}_{\text{diff}}^{\gamma,P} = \underbrace{-\frac{1}{\text{Re}_0}\sum_P\left[\frac{1}{2}\sum_Q\left(\gamma_a^P-\gamma_a^Q\right)\frac{1}{\sigma_f}\frac{\partial\gamma}{\partial x_k}\bigg|_{PQ}\mathsf{n}_k^{PQ}+\gamma_a^P\sum_f\frac{1}{\sigma_f}\frac{\partial\gamma}{\partial x_k}\bigg|_f\mathsf{n}_k^f\frac{\partial\mu_t^f}{\partial\mu_t^P}\right]\frac{\delta\mu_t^P}{\delta b_i}}_{\mathcal{I}^{\mu_t}}$$
$$\underbrace{-\frac{1}{\text{Re}_0}\sum_P\left[\frac{1}{2}\sum_Q\left(\gamma_a^P-\gamma_a^Q\right)\frac{\partial\gamma}{\partial x_k}\bigg|_{PQ}\mathsf{n}_k^{PQ}+\gamma_a^P\sum_f\frac{\partial\gamma}{\partial x_k}\bigg|_f\mathsf{n}_k^f\frac{\partial\mu^f}{\partial\mu^P}\right]\frac{\partial\mu^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow Suth}$$
$$\underbrace{-\frac{1}{\text{Re}_0}\sum_P\left[\frac{1}{2}\sum_Q\left(\gamma_a^P-\gamma_a^Q\right)\left(\mu+\frac{\mu_t}{\sigma_f}\right)^P\frac{\partial\gamma}{\partial x_k}\bigg|_{PQ}\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}+\gamma_a^P\sum_f\left(\mu+\frac{\mu_t}{\sigma_f}\right)^f\frac{\partial\gamma}{\partial x_k}\bigg|_f\frac{\delta\mathsf{n}_k^f}{\delta b_i}\right]}_{\rightarrow SD}$$
$$\underbrace{-\frac{1}{\text{Re}_0}\sum_P\gamma_a^P\left[\sum_Q\left(\mu+\frac{\mu_t}{\sigma_f}\right)^{PQ}\mathsf{n}_k^{PQ}\frac{\delta}{\delta b_i}\left(\frac{\partial\gamma}{\partial x_k}\bigg|_{PQ}\right)+\sum_f\left(\mu+\frac{\mu_t}{\sigma_f}\right)^f\mathsf{n}_k^f\frac{\delta}{\delta b_i}\left(\frac{\partial\gamma}{\partial x_k}\bigg|^P\right)\right]}_{I_{\text{diff}}^{\gamma,1}} \tag{4.36}$$

$$\mathcal{I}_{\text{diff}}^{\tilde{R}e_{\theta t},P} = \underbrace{-\frac{1}{\text{Re}_0}\sum_P\left[\frac{1}{2}\sum_Q\left(\tilde{R}e_a^P-\tilde{R}e_a^Q\right)\sigma_{\theta,t}\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_{PQ}\mathsf{n}_k^{PQ}+\tilde{R}e_a^P\sum_f\sigma_{\theta,t}\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_f\mathsf{n}_k^f\frac{\partial\mu_t^f}{\partial\mu_t^P}\right]\frac{\delta\mu_t^P}{\delta b_i}}_{\mathcal{I}^{\mu_t}}$$
$$\underbrace{-\frac{1}{\text{Re}_0}\sum_P\left[\frac{1}{2}\sum_Q\left(\tilde{R}e_a^P-\tilde{R}e_a^Q\right)\sigma_{\theta,t}\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_{PQ}\mathsf{n}_k^{PQ}+\tilde{R}e_a^P\sum_f\sigma_{\theta,t}\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_f\mathsf{n}_k^f\frac{\partial\mu^f}{\partial\mu^P}\right]\frac{\partial\mu^P}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow Suth}$$
$$\underbrace{-\frac{1}{\text{Re}_0}\sum_P\left[\frac{1}{2}\sum_Q\left(\tilde{R}e_a^P-\tilde{R}e_a^Q\right)\sigma_{\theta,t}(\mu+\mu_t)^P\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_{PQ}\frac{\delta\mathsf{n}_k^{PQ}}{\delta b_i}+\tilde{R}e_a^P\sum_f\sigma_{\theta,t}(\mu+\mu_t)^f\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_f\frac{\delta\mathsf{n}_k^f}{\delta b_i}\right]}_{\rightarrow SD}$$
$$\underbrace{-\frac{1}{\text{Re}_0}\sum_P\tilde{R}e_a^P\left[\sum_Q\sigma_{\theta,t}(\mu+\mu_t)^{PQ}\mathsf{n}_k^{PQ}\frac{\delta}{\delta b_i}\left(\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|_{PQ}\right)+\sum_f\sigma_{\theta,t}\,(\mu+\mu_t)^f\mathsf{n}_k^f\frac{\delta}{\delta b_i}\left(\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\bigg|^P\right)\right]}_{I_{\text{diff}}^{\tilde{R}e_{\theta t},1}} \tag{4.37}$$

where $\mathcal{I}^{\gamma,1}_{\text{diff}}$ and $\mathcal{I}^{\tilde{Re}_{\theta t},1}_{\text{diff}}$ read

$$\mathcal{I}^{\varphi,1}_{\text{diff}} = \underbrace{-\sum_P \left[\sum_Q \left(\mathcal{C}^{P,PQ}_{\varphi,k} - \mathcal{C}^{Q,PQ}_{\varphi,k} \right) \frac{\hat{\mathsf{t}}^{PQ}_k}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}} - \left(\mathscr{C}^P_{\varphi,\ell} D^P_\ell + \sum_\Lambda \mathscr{C}^\Lambda_{\varphi,\ell} Z^{\Lambda,P}_\ell \right) \right] \frac{\varphi^P}{\rho^P} \frac{\partial \rho^P}{\partial U^P_q} \frac{\delta U^P_q}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{+\sum_P \left[\sum_Q \left(\mathcal{C}^{P,PQ}_{\varphi,k} - \mathcal{C}^{Q,PQ}_{\varphi,k} \right) \frac{\hat{\mathsf{t}}^{PQ}_k}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}} - \left(\mathscr{C}^P_{\varphi,\ell} D^P_\ell + \sum_\Lambda \mathscr{C}^\Lambda_{\varphi,\ell} Z^{\Lambda,P}_\ell \right) \right] \frac{1}{\rho^P} \frac{\delta \left(\rho \varphi \right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\varphi}$$

$$\underbrace{+\frac{1}{2} \sum_P \sum_Q \left(\mathcal{C}^{P,PQ}_{\varphi,k} - \mathcal{C}^{Q,PQ}_{\varphi,k} \right) \frac{\partial \varphi^P}{\partial x_\ell} \left(\hat{\mathsf{t}}^{PQ}_k \frac{\delta \hat{\mathsf{t}}^{PQ}_\ell}{\delta b_i} + \hat{\mathsf{t}}^{PQ}_\ell \frac{\delta \hat{\mathsf{t}}^{PQ}_k}{\delta b_i} \right) - \sum_P \sum_\Lambda \mathscr{C}^\Lambda_{\varphi,\ell} \varphi^P \frac{\delta Z^{\Lambda P}_\ell}{\delta b_i}}_{\text{SD}}$$

$$\underbrace{-\sum_P \sum_Q \left(\mathcal{C}^{P,PQ}_{\varphi,k} - \mathcal{C}^{Q,PQ}_{\varphi,k} \right) \varphi^P \left(-\frac{1}{\sqrt{x^{PQ}_\ell x^{PQ}_\ell}} \frac{\delta \hat{\mathsf{t}}^{PQ}_k}{\delta b_i} + \hat{\mathsf{t}}^{PQ}_k \frac{x^{PQ}_\ell}{\sqrt[3]{x^{PQ}_\ell x^{PQ}_\ell}} \frac{\delta x^{PQ}_\ell}{\delta b_i} \right) - \sum_P \mathscr{C}^P_{\varphi,\not{\varphi}} \varphi^P \frac{\delta D^P_\ell}{\delta b_i}}_{\text{SD}} \tag{4.38}$$

Also, φ stands for either γ or $\tilde{Re}_{\theta t}$ and

$$\mathscr{C}^{P,PQ}_{\gamma,\ell} = \frac{1}{2} \sum_Q \left(\mathcal{C}^{P,PQ}_{\gamma,k} - \mathcal{C}^{Q,PQ}_{\gamma,k} \right) \left(\delta_{k\ell} - \hat{\mathsf{t}}^{PQ}_\ell \hat{\mathsf{t}}^{PQ}_k \right) + \sum_f \mathcal{C}^{P,f}_{\gamma,\ell}, \; \mathcal{C}^{*,\alpha\beta}_{\gamma,k} = \frac{1}{\text{Re}_0} \left(\mu + \frac{\mu_t}{\sigma_f} \right)^{\alpha\beta} \gamma^*_a \mathsf{n}^{\alpha\beta}_k$$

$$\mathscr{C}^{P,PQ}_{\tilde{Re}_{\theta t},\ell} = \frac{1}{2} \sum_Q \left(\mathcal{C}^{P,PQ}_{\tilde{Re}_{\theta t},k} - \mathcal{C}^{Q,PQ}_{\tilde{Re}_{\theta t},k} \right) \left(\delta_{k\ell} - \hat{\mathsf{t}}^{PQ}_\ell \hat{\mathsf{t}}^{PQ}_k \right) + \sum_f \mathcal{C}^{P,f}_{\tilde{Re}_{\theta t},\ell}, \; \mathcal{C}^{*,\alpha\beta}_{\tilde{Re}_{\theta t},k} = \frac{1}{\text{Re}_0} \sigma_{\theta,t} (\mu + \mu_t)^{\alpha\beta} \tilde{Re}^*_a \mathsf{n}^{\alpha\beta}_k$$

Source terms $\mathcal{I}^{\gamma,P}_{\text{src}}$ and $\mathcal{I}^{\tilde{Re}_{\theta t},P}_{\text{src}}$ are written as

$$\mathcal{I}^{\gamma,P}_{\text{src}} = -\sum_P \gamma^P_a \frac{\delta \left(P^P_\gamma \Omega_P \right)}{\delta b_i} + \sum_P \gamma^P_a \frac{\delta \left(E^P_\gamma \Omega_P \right)}{\delta b_i} \tag{4.39}$$

$$\mathcal{I}^{\tilde{Re}_{\theta t},P}_{\text{src}} = -\sum_P \tilde{Re}^P_a \frac{\delta \left(P^P_{\theta,t} \Omega_P \right)}{\delta b_i} - \sum_P \tilde{Re}^P_a \frac{\delta \left(D^P_{SCF} \Omega_P \right)}{\delta b_i} \tag{4.40}$$

For the sources in the γ equation,

$$
\begin{aligned}
-\sum_P \gamma_a^P \frac{\delta\left(P_\gamma^P \Omega_P\right)}{\delta b_i} = & \underbrace{-\sum_P \left[\gamma_a c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{+\frac{1}{2}\sum_P \left[\gamma\gamma_a c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1/\gamma - 3c_{\epsilon_1}\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P}\frac{\delta U_\ell}{\delta b_i}^P}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{-\sum_P \left[\gamma_a \frac{\tilde{R}e_{\theta t}}{\rho} \mathcal{C}_{F_{length}} \mathcal{P}\left(F_{length}, F_{length1}\right) \mathcal{P}\left(F_{length1}, \tilde{R}e_{\theta t}\right)\right]^p \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
& \underbrace{-\frac{1}{2}\sum_P \left[\gamma_a c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1/\gamma - 3c_{\epsilon_1}\right)\right]^P \Omega_P \frac{\delta\left(\rho\gamma\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\gamma} \\
& \underbrace{+\sum_P \left[\gamma_a \frac{1}{\rho} \mathcal{C}_{F_{length}} \mathcal{P}\left(F_{length}, F_{length1}\right) \mathcal{P}\left(F_{length1}, \tilde{R}e_{\theta t}\right)\right]^P \Omega_P \frac{\delta\left(\rho\tilde{R}e_{\theta t}\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}} \\
& \underbrace{-\sum_P \left[\gamma_a \rho c_{\alpha_1} F_{length} F_{onset} \mathcal{P}\left(\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right), \zeta\right)\sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right)\right]^P \Omega_P \frac{\delta\zeta}{\delta b_i}^P}_{\mathcal{I}^\zeta} \\
& \underbrace{-\sum_P \left[\rho c_{\alpha_1} F_{length} F_{onset} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right)\right]^P \frac{\delta\Omega_P}{\delta b_i}}_{\rightarrow \text{SD}} \\
& \underbrace{-\sum_P \left[\gamma_a \rho c_{\alpha_1} F_{length} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1 - c_{\epsilon_1}\gamma\right)\right]^P \Omega_P \frac{\delta F_{onset}^P}{\delta b_i}}_{\mathcal{I}^{F_{onset}}}
\end{aligned}
\tag{4.41}
$$

$$
\sum_P \gamma_a^P \frac{\delta\left(E_\gamma^P \Omega_P\right)}{\delta b_i} = \underbrace{\sum_P \left[\gamma_a c_{\alpha_2} \phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\gamma F_{turb}\left(c_{\epsilon_2}\gamma - 1\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}
$$

$$
\begin{aligned}
&\underbrace{-\sum_P\left[\gamma\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\right]^P\Omega_P\frac{\partial\rho^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\sum_P\left[\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right)\right]^P\Omega_P\frac{\delta\left(\rho\gamma\right)^P}{\delta b_i}}_{\rightarrow\text{FAE\_}\gamma}\\
&\underbrace{+\sum_P\left[\gamma_a\rho c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},F_{onset}\right)\right]^P\Omega_P\frac{\delta F_{onset}^P}{\delta b_i}}_{\mathcal{I}^{F_{onset}}}\\
&\underbrace{+\sum_P\left[\gamma_a\rho c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu_t\right)\right]^P\Omega_P\frac{\delta\mu_t{}^P}{\delta b_i}}_{\mathcal{I}^{\mu_t}}\\
&\underbrace{+\sum_P\left[\gamma_a\rho c_{\alpha_2}\gamma F_{turb}\mathcal{P}\left(\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right),\zeta\right)\left(c_{\epsilon_2}\gamma-1\right)\right]^P\Omega_P\frac{\delta\zeta^P}{\delta b_i}}_{\mathcal{I}^{\zeta}}\\
&\underbrace{+\sum_P\left[\gamma_a\rho c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\mathcal{P}\left(F_{turb},\mu\right)\right]^P\Omega_P\frac{\partial\mu^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow Suth}\\
&\underbrace{+\sum_P\left[\gamma_a\rho c_{\alpha_2}F_{turb}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma\left(c_{\epsilon_2}\gamma-1\right)\right]^P\frac{\delta\Omega_P}{\delta b_i}}_{\rightarrow\text{SD}}
\end{aligned}
\tag{4.42}
$$

and

$$
\begin{aligned}
\mathcal{I}^{F_{onset}}=&\underbrace{-\sum_P\left[\gamma_a\frac{\tilde{R}e_{\theta t}}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{R}e_{\theta t}\right)\right]^P\Omega_P\frac{\partial\rho^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\sum_P\left[\gamma_a\frac{1}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{R}e_{\theta t}\right)\right]^P\Omega_P\frac{\delta\left(\rho\tilde{R}e_{\theta t}\right)^P}{\delta b_i}}_{\rightarrow\text{FAE\_}\tilde{R}e_{\theta t}}
\end{aligned}
$$

$$\underbrace{+\sum_P \left[\gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, S\right)\right]^P \Omega_P \frac{\delta S^P}{\delta b_i}}_{\mathcal{I}^S}$$

$$\underbrace{+\sum_P \left[\gamma_a \mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, \Delta\right)\right]^P \Omega_P \frac{\delta \Delta^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\Delta}$$

$$\underbrace{+\sum_P \left[\gamma_a \left[\mathcal{C}_{F_{onset1}} \mathcal{P}\left(F_{onset1}, Re_S\right) \mathcal{P}\left(Re_S, \mu\right) + \mathcal{C}_{F_{onset3}} \mathcal{P}\left(F_{onset3}, \mu\right)\right]\right]^P \Omega_P \frac{\partial \mu^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow Suth}$$

For the $\tilde{R}e_{\theta t}$ equation sources,

$$-\sum_P \tilde{R}e_a^P \frac{\delta\left(P_{\theta,t}^P \Omega_P\right)}{\delta b_i} = \underbrace{-\sum_P \left[\tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}} \left(Re_{\theta,t}^{eq} - \tilde{R}e_{\theta t}\right)(1 - F_{\theta t})\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{+\sum_P \left[\rho \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}^2} \left(Re_{\theta,t}^{eq} - \tilde{R}e_{\theta t}\right)(1 - F_{\theta t}) \mathcal{P}\left(\mathcal{T}, \rho\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{+\sum_P \left[\rho \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}^2} \left(Re_{\theta,t}^{eq} - \tilde{R}e_{\theta t}\right)(1 - F_{\theta t}) \mathcal{P}\left(\mathcal{T}, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A}\right]^P \Omega_P \frac{\partial v_k^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}$$

$$\underbrace{-\sum_P \left[\tilde{R}e_a \tilde{R}e_{\theta t} \frac{c_{\theta,t}}{\mathcal{T}} (1 - F_{\theta t})\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \underbrace{+\sum_P \left[\tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}} (1 - F_{\theta t})\right]^P \Omega_P \frac{\delta\left(\rho \tilde{R}e_{\theta t}\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}}$$

$$\underbrace{-\sum_P \left[\tilde{R}e_a \rho \frac{c_{\theta,t}}{\mathcal{T}} (1 - F_{\theta t})\right]^P \Omega_P \frac{\delta R e_{\theta,t}^{eq\ P}}{\delta b_i}}_{\mathcal{I}^{Re_{\theta,t}^{eq}}} \underbrace{+\sum_P \left[\tilde{R}e_a \rho \frac{c_{\theta,t}}{\mathcal{T}} \left(Re_{\theta,t}^{eq} - \tilde{R}e_{\theta t}\right)\right]^P \Omega_P \frac{\delta F_{\theta t}^P}{\delta b_i}}_{\mathcal{I}^{F_{\theta t}}}$$

$$\underbrace{+\sum_P \left[\rho \tilde{R}e_a \frac{c_{\theta,t}}{\mathcal{T}^2} \left(Re_{\theta,t}^{eq} - \tilde{R}e_{\theta t}\right)(1 - F_{\theta t}) \mathcal{P}\left(\mathcal{T}, \mu\right)\right]^P \Omega_P \frac{\partial \mu^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow Suth} \tag{4.43}$$

and

$$
\begin{aligned}
&-\sum_P \tilde{Re}_a^P \frac{\delta\left(D_{SCF}^P \Omega_P\right)}{\delta b_i} = \\
&\underbrace{-\sum_P \left[\tilde{Re}_a \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right) F_{\theta t}\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\sum_P \left[\tilde{Re}_a \rho \frac{c_{\theta,t}}{\mathcal{T}^2} c_{crossflow} \phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right) F_{\theta t} \mathcal{P}\left(\mathcal{T}, \rho\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\sum_P \left[\tilde{Re}_a \rho \frac{c_{\theta,t}}{\mathcal{T}^2} c_{crossflow} \phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right) F_{\theta t} \mathcal{P}\left(\mathcal{T}, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A}\right]^P \Omega_P \frac{\partial v_k^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\sum_P \left[\tilde{Re}_{\theta t} \tilde{Re}_a \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \mathcal{P}\left(\phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right), \tilde{Re}_{\theta t}\right) F_{\theta t}\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{-\sum_P \left[\tilde{Re}_a \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \mathcal{P}\left(\phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right), \tilde{Re}_{\theta t}\right) F_{\theta t}\right]^P \Omega_P \frac{\delta\left(\rho \tilde{Re}_{\theta t}\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\tilde{Re}_{\theta t}} \\
&\underbrace{-\sum_P \left[\tilde{Re}_a \rho \frac{c_{\theta,t}}{\mathcal{T}} c_{crossflow} \phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right)\right]^P \Omega_P \frac{\delta F_{\theta t}^P}{\delta b_i}}_{\mathcal{I}^{F_{\theta t}}} \underbrace{+\sum_P \left[\tilde{Re}_a \mathcal{C}_{Re_{SCF}}\right]^P \Omega_P \frac{\delta Re_{SCF}^P}{\delta b_i}}_{\mathcal{I}^{Re_{SCF}}} \\
&\underbrace{+\sum_P \left[\tilde{Re}_a \rho \frac{c_{\theta,t}}{\mathcal{T}^2} c_{crossflow} \phi_{-300}\left(Re_{SCF} - \tilde{Re}_{\theta t}, 0\right) F_{\theta t} \mathcal{P}\left(\mathcal{T}, \mu\right)\right]^P \Omega_P \frac{\partial \mu^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i} \mathrm{d}\Omega}_{\rightarrow \mathit{Suth}} \quad (4.44)
\end{aligned}
$$

Term $\mathcal{I}^{F_{\theta t}}$ becomes

$$
\begin{aligned}
\mathcal{I}^{F_{\theta t}} = &\underbrace{\sum_P \left[\tilde{Re}_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, \rho\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\sum_P \left[\tilde{Re}_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \rho\right)\right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}}
\end{aligned}
$$

$$
\begin{aligned}
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \right]^P \Omega_P \frac{\partial v_k^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{-\sum_P \left[\tilde{R}e_a \frac{\tilde{R}e_{\theta t}}{\rho} \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \tilde{R}e_{\theta t}\right) \right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\sum_P \left[\frac{\tilde{R}e_a}{\rho} \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \tilde{R}e_{\theta t}\right) \right]^P \Omega_P \frac{\delta \left(\rho \tilde{R}e_{\theta t}\right)^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\tilde{R}e_{\theta t}} \\
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \zeta\right) \right]^P \Omega_P \frac{\delta \zeta}{\delta b_i} \mathrm{d}\Omega}_{\mathcal{I}^\zeta} \\
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, S\right) \right]^P \Omega_P \frac{\delta S^P}{\delta b_i}}_{\mathcal{I}^S} \\
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, \Delta\right) \right]^P \Omega_P \frac{\delta \Delta^P}{\delta b_i}}_{\rightarrow \text{FAE\_}\Delta} \\
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{F_{\theta t}} \mathcal{P}\left(F_{\theta t}, F_{wake}\right) \mathcal{P}\left(F_{wake}, Re_S\right) \mathcal{P}\left(Re_S, \mu\right) \right]^P \Omega_P \frac{\partial \mu^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow Suth} \\
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{F_{\theta t}} F_{wake} \mathcal{P}\left(F_{\theta t}, \delta\right) \mathcal{P}\left(\delta, \mu\right) \right]^P \Omega_P \frac{\partial \mu^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow Suth}
\end{aligned}
\tag{4.45}
$$

$\mathcal{I}^{Re_{\theta,t}^{eq}}$ is expressed as

$$
\begin{aligned}
\mathcal{I}^{Re_{\theta,t}^{eq}} = &\underbrace{\sum_P \left[\tilde{R}e_a \mathcal{C}_{Re_{\theta,t}^{eq}} \left[\mathcal{P}\left(\lambda_\theta, \rho\right) + \mathcal{P}\left(\lambda_\theta, \theta\right) \mathcal{P}\left(\theta, \rho\right) \right] \right]^P \Omega_P \frac{\partial \rho^P}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \\
&\underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{Re_{\theta,t}^{eq}} \mathcal{P}\left(\lambda_\theta, \theta\right) \mathcal{P}\left(\theta, |\mathbf{U}|\right) \frac{\partial |\mathbf{U}|}{\partial v_k^A} \right]^P \Omega_P \frac{\partial v_k^{A,P}}{\partial U_\ell^P} \frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow \text{FAE\_MF}} \underbrace{+\sum_P \left[\tilde{R}e_a \mathcal{C}_{d|\mathbf{U}|ds} \right]^P \Omega_P \frac{\delta}{\delta b_i} \left(\frac{d|\mathbf{U}|}{ds} \right)^P}_{\mathcal{I}^{d|\mathbf{U}|ds}}
\end{aligned}
$$

$$\underbrace{+\sum_P\left[\tilde{R}e_a\mathcal{C}_{Re^{eq}_{\theta,t}}\left[\mathcal{P}\left(\lambda_\theta,\mu\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\mu\right)\right]\right]^P\Omega_P\frac{\partial\mu^P}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow Suth} \tag{4.46}$$

and

$$\begin{aligned}\mathcal{I}^{d|\mathbf{U}|ds}=&\underbrace{\sum_P\left[\tilde{R}e_a\mathcal{C}_{d|\mathbf{U}|ds}\left[-\frac{2}{|\mathbf{U}|}\frac{d\,|\mathbf{U}|}{ds}\frac{\partial\,|\mathbf{U}|}{\partial v^A_k}+\frac{v^A_m}{|\mathbf{U}|^2}\left(\frac{\partial v^A_k}{\partial x_m}+\frac{\partial v^A_m}{\partial x_k}\right)\right]\right]^P\Omega_P\frac{\partial v^{A,P}_k}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow \text{FAE_MF}}\\
&\underbrace{+\sum_P\left[\mathscr{C}^{d|\mathbf{U}|ds,P}_{km}D^P_m+\sum_\Lambda\mathscr{C}^{d|\mathbf{U}|ds,\Lambda}_{km}Z^{\Lambda P}_m\right]\frac{\partial v^{A,P}_k}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow \text{FAE_MF}}\\
&\underbrace{+\sum_P\mathscr{C}^{d|\mathbf{U}|ds,P}_{km}v^{A,P}_k\frac{\delta D^P_m}{\delta b_i}+\sum_P\sum_\Lambda\mathscr{C}^{d|\mathbf{U}|ds,\Lambda}_{km}v^{A,P}_k\frac{\delta Z^{\Lambda P}_m}{\delta b_i}}_{\rightarrow \text{SD}}\end{aligned} \tag{4.47}$$

where

$$\mathscr{C}^{d|\mathbf{U}|ds,P}_{km}=\left[\tilde{R}e_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v^A_kv^A_m}{|\mathbf{U}|^2}\right]^P\Omega_P$$

$\mathcal{I}^{Re_{SCF}}$ is expressed as

$$\begin{aligned}\mathcal{I}^{Re_{SCF}}=&\underbrace{-\sum_P\left[\tilde{R}e_a\mathcal{C}_{Re_{SCF}}\frac{0.82\mu}{\rho^2\,|\mathbf{U}|}\right]^P\Omega_P\frac{\partial\rho^P}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow \text{FAE_MF}}\underbrace{-\sum_P\left[\tilde{R}e_a\mathcal{C}_{Re_{SCF}}\frac{0.82\mu}{\rho\,|\mathbf{U}|^2}\frac{\partial\,|\mathbf{U}|}{\partial v^A_k}\right]^P\Omega_P\frac{\partial v^{A,P}_k}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow \text{FAE_MF}}\\
&\underbrace{+\sum_P\left[\tilde{R}e_a\mathcal{C}_{\mathcal{D}H_{CF}}\right]^P\Omega_P\frac{\delta\mathcal{D}H^P_{CF}}{\delta b_i}}_{\mathcal{I}^{\mathcal{D}H_{CF}}}\underbrace{+\sum_P\left[\tilde{R}e_a\mathcal{C}_{Re_{SCF}}\frac{0.82Re_{SCF}}{\rho\,|\mathbf{U}|}\right]^P\Omega_P\frac{\partial\mu^P}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\rightarrow Suth}\end{aligned} \tag{4.48}$$

and

$$\mathcal{I}^{\mathcal{D}H_{CF}}=\underbrace{\sum_P\left[\tilde{R}e_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},H_{CF}\right)\mathcal{P}\left(H_{CF},|\mathbf{U}|\right)\frac{\partial\,|\mathbf{U}|}{\partial v^A_k}\right]^P\Omega_P\frac{\partial v^{A,P}_k}{\partial U^P_\ell}\frac{\delta U^P_\ell}{\delta b_i}}_{\text{FAE\_MF}}$$

$$\begin{aligned}
&\underbrace{+\sum_P\left[\tilde{Re}_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},\mu_t\right)\right]^P\Omega_P\frac{\delta\mu_t^P}{\delta b_i}}_{\mathcal{I}^{\mu_t}}\underbrace{+\sum_P\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\right]^P\Omega_P\frac{\delta\zeta_{streamwise}^P}{\delta b_i}}_{\mathcal{I}^{\zeta_{streamwise}}}\\
&\underbrace{+\sum_P\left[\tilde{Re}_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},H_{CF}\right)\mathcal{P}\left(H_{CF},\Delta\right)\right]^P\Omega_P\frac{\delta\Delta^P}{\delta b_i}}_{\rightarrow\text{FAE_}\Delta}\\
&\underbrace{+\sum_P\left[\tilde{Re}_a\mathcal{C}_{\Delta H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},\mu\right)\right]^P\Omega_P\frac{\partial\mu^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow Suth}
\end{aligned}\tag{4.49}$$

$$\begin{aligned}
\mathcal{I}^{\zeta_{streamwise}}=&\underbrace{\sum_P\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{1}{|\mathbf{U}|}\frac{\partial v_m^A}{\partial x_q}\left[\varepsilon_{kqm}+\frac{u_r}{|\mathbf{U}|}\varepsilon_{rqm}\frac{\partial|\mathbf{U}|}{\partial v_k^A}\right]\right]^P\Omega_P\frac{\partial v_k^{A,P}}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\text{FAE_MF}}\\
&\underbrace{+\sum_P\left[\mathscr{C}_{qm}^{\zeta_{streamwise},P}D_q^P+\sum_\Lambda\mathscr{C}_{qm}^{\zeta_{streamwise},\Lambda}Z_q^{\Lambda P}\right]\frac{\partial v_m^{A,P}}{\partial U_q^P}\frac{\delta U_q^P}{\delta b_i}}_{\rightarrow\text{FAE_MF}}\\
&\underbrace{+\sum_P\mathscr{C}_{qm}^{\zeta_{streamwise},P}v_m^{A,P}\frac{\delta D_q^P}{\delta b_i}+\sum_P\sum_\Lambda\mathscr{C}_{qm}^{\zeta_{streamwise},\Lambda}v_m^{A,P}\frac{\delta Z_q^{\Lambda P}}{\delta b_i}}_{\rightarrow\text{SD}}
\end{aligned}\tag{4.50}$$

$$\mathscr{C}_{qm}^{\zeta_{streamwise},P}=\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_k^A}{|\mathbf{U}|}\varepsilon_{kqm}\right]^P\Omega_P$$

Terms $\mathcal{I}^\zeta$ and $\mathcal{I}^S$ are expressed as

$$\begin{aligned}
\mathcal{I}^\zeta=&\underbrace{\sum_P\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{knr}\frac{\partial v_r^A}{\partial x_n}\right]^P\Omega_P D_\ell^P\frac{\partial v_m^{A,P}}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\text{FAE_MF}}\\
&\underbrace{+\sum_P\sum_\Lambda\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{knr}\frac{\partial v_r^A}{\partial x_n}\right]^\Lambda\Omega_\Lambda Z_\ell^{\Lambda P}\frac{\partial v_m^{A,P}}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\text{FAE_MF}}
\end{aligned}$$

$$
\begin{aligned}
&\underbrace{+\sum_P\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\gamma-\tilde{R}e_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{knr}\frac{\partial v_r^A}{\partial x_n}v_m^A\right]^P\Omega_P\frac{\delta D_\ell^P}{\delta b_i}+\sum_P\sum_\Lambda\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\gamma-\tilde{R}e_{\theta t}}}{\zeta}\varepsilon_{k\ell m}\varepsilon_{knr}\frac{\partial v_r^A}{\partial x_n}\right]^\Lambda\Omega_\Lambda v_m^{A,P}\frac{\delta Z_\ell^{\Lambda P}}{\delta b_i}}_{\rightarrow\text{SD}}\\
&\mathcal{I}^S=\underbrace{\sum_P\left[\mathcal{C}_S^{\gamma-\tilde{R}e_{\theta t},P}\frac{S_{km}^P}{S^P}D_m^P+\sum_\Lambda\mathcal{C}_S^{\gamma-\tilde{R}e_{\theta t},\Lambda}\frac{S_{km}^\Lambda}{S^\Lambda}Z_m^{\Lambda P}\right]\frac{\partial v_k^{A,P}}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\text{FAE\_MF}}\\
&\underbrace{+\sum_P\left[\mathcal{C}_S^{\gamma-\tilde{R}e_{\theta t},P}\frac{S_{km}^P}{S^P}v_k^P\frac{\delta D_m^P}{\delta b_i}+\sum_\Lambda\mathcal{C}_S^{\gamma-\tilde{R}e_{\theta t},\Lambda}\frac{S_{km}^\Lambda}{S^\Lambda}v_k^{A,P}\frac{\delta Z_m^\Lambda}{\delta b_i}\right]}_{\rightarrow\text{SD}}
\end{aligned}
\tag{4.51}
$$

Finally, $\mathcal{I}^{\mu_t}$ becomes

$$
\begin{aligned}
\mathcal{I}^{\mu_t}=&\underbrace{\sum_P\left[\mathcal{C}_{\mu_t}^{\gamma-\tilde{R}e_{\theta t}}\left[\mathcal{P}\left(\mu_t,\tilde{\mu}\right)+\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\tilde{\mu}\right)\right]\right]^P\Omega_P\frac{\delta\tilde{\mu}^P}{\delta b_i}}_{\rightarrow\text{FAE\_SA}}\\
&\underbrace{+\sum_P\left[\mathcal{C}_{\mu_t}^{\gamma-\tilde{R}e_{\theta t}}\mathcal{P}\left(\mu_t,f_{v_1}\right)\mathcal{P}\left(f_{v_1},\chi\right)\mathcal{P}\left(\chi,\mu\right)\right]^P\Omega_P\frac{\partial\mu^P}{\partial U_\ell^P}\frac{\delta U_\ell^P}{\delta b_i}}_{\rightarrow\textit{Suth}}
\end{aligned}
\tag{4.52}
$$

4.4 Discretization of the Adjoint Equations

Eliminating all terms denoted as FAE_MF, FAE_SA, FAE_γ, FAE_$\tilde{R}e_{\theta t}$ and FAE_Δ leads to the discrete (through hand-differentiation) adjoint equations to the mean flow, the Spalart–Allmaras model, the $\gamma-\tilde{R}e_{\theta t}$ model variables and the Hamilton–Jacobi equation. Terms denoted as *Suth* contribute to the mean flow field adjoint equations only if the Sutherland's law is used to compute the dynamic viscosity in terms of temperature. Based on the discrete adjoint equations, consistent (with the discretized flow problem) discretization schemes for the continuous adjoint PDEs, Eq. 3.67, are sought. The fact that the continuous adjoint PDEs are in non-conservative form should also be kept in mind. The proposed discretization schemes refer to the fluxes crossing the boundaries of a finite volume defined around any node P, Fig. 2.1.

4.4.1 Discretization of the Adjoint Inviscid Terms

Integrating the inviscid term of Eq. 3.67a over Ω_P, in the standard finite volume notation, and making the necessary approximations results in the balance of

inviscid adjoint fluxes (notations as in Fig. 2.1), written in the form

$$-\int_{\Omega_P} A_{nmk}\frac{\partial \Psi_n}{\partial x_k}d\Omega \simeq \sum_{Q\in\mathcal{N}(P)} \Phi_m^{\text{inv,adj},PQ} + \sum_{f\in\mathcal{B}(P)} \Phi_m^{\text{inv,adj},f} \tag{4.53}$$

Internal Adjoint Fluxes

The internal flux $\Phi_m^{\text{inv,adj},PQ}$ is often discretized using a non-conservative scheme, [53, 111, 16, 106, 49, 80], which mimics that used to discretize the primal problem, Eq. 2.49,

$$\Phi_{m,std}^{\text{inv,adj},PQ} = -\frac{1}{2}\left(\Psi_n^P + \Psi_n^Q\right) A_{nmk}^P \mathsf{n}_k^{PQ} - \frac{1}{2}\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|\left(\Psi_n^R - \Psi_n^L\right) \tag{4.54}$$

The adjoint solver that makes use of the above discretization scheme will be referred as "standard" continuous adjoint. Subscript "std" refers to this discretization.

Inspired by the hand-differentiated discrete adjoint equations, the following expression for the adjoint fluxes

$$\Phi_m^{\text{inv,adj},PQ} = -\frac{1}{2}\left(\Psi_n^P + \Psi_n^Q\right) A_{nmk}^P \mathsf{n}_k^{PQ} - \frac{1}{2}\left[\left(\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\Psi_n\right)^{R,\text{adj}} - \left(\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\Psi_n\right)^{L,\text{adj}}\right]\frac{\partial V_\ell^P}{\partial U_m^P} \tag{4.55}$$

is proposed instead. In Eq. 4.55, for second-order accuracy without limiter, the "L"eft and "R"ight adjoint states are given by

$$\phi^{L,\text{adj}} = \phi^P + \frac{1}{2}\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\bigg|^{P,\text{adj}}, \frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\bigg|^{P,\text{adj}} = \mathsf{t}_r^{PQ} D_r^P \phi^P + \sum_{\Lambda\in\mathcal{N}(P)} Z_r^{\Lambda P} \sum_{M\in\mathcal{N}(\Lambda)} \mathsf{t}_r^{\Lambda M}\phi^\Lambda \tag{4.56a}$$

$$\phi^{R,\text{adj}} = \phi^Q + \frac{1}{2}\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\bigg|^{Q,\text{adj}}, \frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\bigg|^{Q,\text{adj}} = \mathsf{t}_r^{PQ} D_r^P \phi^Q + \sum_{\Lambda\in\mathcal{N}(P)} Z_r^{\Lambda P} \sum_{M\in\mathcal{N}(\Lambda)} \mathsf{t}_r^{\Lambda M}\phi^M \tag{4.56b}$$

and M is any node connected to $\Lambda\in\mathcal{N}(P)$ by an edge. The modified absolute Jacobian $\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|$ is introduced and defined as

$$\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right| = \left|\tilde{A}_{nqk}^{LR}\mathsf{n}_k\right|\frac{\partial U_q^L}{\partial V_\ell^L} - \left(U_q^R - U_q^L\right)\frac{\partial\left|\tilde{A}_{nqk}^{LR}\mathsf{n}_k\right|}{\partial V_\ell^L} \tag{4.57}$$

This expression includes the derivatives of the absolute Jacobian w.r.t. the primitive flow variables and corresponds to an "adjoint" Taylor expansion of the absolute Jacobian in the $\mathbf{V}$ space. It should be noted that, if the primal discretization was first–order accurate, an expansion in the $\mathbf{U}$ space would have been used instead.

Apart from the use of the modified absolute Jacobian, instead of the standard one, the adjoint L and R states ("L, adj", "R, adj") for an adjoint variable are defined differently than in the primal problem. In the primal discretization, Eq. 2.50, the L state of any variable ϕ, along edge PQ, results from a first–order Taylor expansion and is the sum of ϕ^P and the inner product of the half of $\overrightarrow{PQ}$ and the spatial gradient of ϕ at P. The latter depends on the ϕ values at P and its neighbors Λ_i (Fig. 2.1). In Eq. 4.56, the adjoint L ("L, adj") state of ϕ at edge PQ is the sum of the value of ϕ at P and half of the divergence of $\overrightarrow{PQ}$ scaled by ϕ, viz. $\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\Big|^{P,\text{adj}}$. The same holds for R and "R, adj".

For second–order accuracy using the van Leer-van Albada limiter, the "L"eft and "R"ight adjoint states are given by

$$\phi^{L,\text{adj}} = \phi^P + \frac{1}{2}\frac{\partial}{\partial x_r}\left[\mathsf{t}_r\phi\frac{\partial\mathscr{L}_j}{\partial\alpha_j}\right]^{P,\text{adj}} + \frac{1}{2}\mathscr{L}^{P,\text{adj}}(\phi) \tag{4.58a}$$

$$\phi^{R,\text{adj}} = \phi^Q + \frac{1}{2}\frac{\partial}{\partial x_r}\left[\mathsf{t}_r\phi\frac{\partial\mathscr{L}_j}{\partial\alpha_j}\right]^{Q,\text{adj}} + \frac{1}{2}\mathscr{L}^{Q,\text{adj}}(\phi) \tag{4.58b}$$

where $\mathscr{L}^{\text{adj}}$ is defined as

$$\mathscr{L}^{P,\text{adj}}\left(\Psi_n^P\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\right) = \Psi_n^P\left[\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\left(\frac{\partial\mathscr{L}_j^P}{\partial\alpha_j^P} - \frac{\partial\mathscr{L}_j^P}{\partial\beta_j}\right) + \left|\tilde{\mathcal{A}}_{n\ell k}^{RL}\mathsf{n}_k\right|\left(\frac{\partial\mathscr{L}_j^Q}{\partial\alpha_j^Q} - \frac{\partial\mathscr{L}_j^Q}{\partial\beta_j}\right)\right] \tag{4.59a}$$

$$\mathscr{L}^{Q,\text{adj}}\left(\Psi_n^Q\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\right) = \Psi_n^Q\left[\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\left(\frac{\partial\mathscr{L}_j^P}{\partial\alpha_j^P} - \frac{\partial\mathscr{L}_j^P}{\partial\beta_j}\right) + \left|\tilde{\mathcal{A}}_{n\ell k}^{RL}\mathsf{n}_k\right|\left(\frac{\partial\mathscr{L}_j^Q}{\partial\alpha_j^Q} - \frac{\partial\mathscr{L}_j^Q}{\partial\beta_j}\right)\right] \tag{4.59b}$$

In Eq. 4.59, the non-conservative expression of the modified absolute Jacobian $\left|\tilde{\mathcal{A}}_{n\ell k}^{RL}\mathsf{n}_k\right|$ should be noted. Moreover, information regarding the gradient of the $\mathscr{L}$ w.r.t. the primitive flow variables is also taken into account.

Comparing the "standard" discretization scheme for the adjoint equations, Eq. 4.54, and that of the *TDDC* adjoint, Eq. 4.55, the discretization of the central part is common in both schemes; what differs (which is, practically, the novelty of the proposed *TDDC* adjoint) is the discretization of the dissipation part with the introduction of the modified absolute Jacobian and the extrapolation to "L, adj"

and "R, adj", Eqs. 4.56 and 4.58.

Boundary Adjoint Fluxes

The discretization of the boundary fluxes $\Phi_m^{\text{inv,adj},f}$ in Eq. 4.53 for the wall, inlet/outlet and far–field boundaries follows. Recall that, in the internal *TDDC* adjoint flux $\Phi_m^{\text{inv,adj},PQ}$, Eq. 4.8, term $\frac{1}{2}\sum_Q \left(\Psi_n^P - \Psi_n^Q\right) A_{nmk}^P \mathsf{n}_k^{PQ}$ is re-written as $-\frac{1}{2}\sum_Q \left(\Psi_n^P + \Psi_n^Q\right) A_{nmk}^P \mathsf{n}_k^{PQ} + \Psi_n^P A_{nmk}^P \sum_Q \mathsf{n}_k^{PQ}$ and the last term is eliminated since the sum of n_k^{PQ} is zero. This is not, though, the case for finite volumes formed around boundary vertices. For such

$$\Psi_n^P A_{nmk}^P \sum_Q \mathsf{n}_k^{PQ} = -\Psi_n^P A_{nmk}^P \sum_f \mathsf{n}_k^f \tag{4.60}$$

which means that term $-\Psi_n^P A_{nmk}^P \mathsf{n}_k^f$ must be included in all adjoint fluxes crossing boundary faces, irrespective of the type of boundary conditions applied there.

Wall Boundaries: For a wall face $f \in \mathcal{B}^{\mathcal{W}}(P)$ the adjoint flux in Eq. 4.53 is discretized as $\Phi_m^{\text{inv,adj},f} = -\Psi_n^Q A_{nmk}^P \mathsf{n}_k^f$, where Q refers to a halo node and $\Psi_n^Q = \Psi_n^P$. This can also be written as

$$\Phi_m^{\text{inv,adj},f} = \begin{bmatrix} v_k^{A,P}\mathsf{n}_k^f \left[\Psi_{q+1} v_q^A + \Psi_5 h_t\right]^P - \frac{\gamma-1}{2} v_k^{A,P} v_k^{A,P} \left[\Psi_{q+1}^P \mathsf{n}_q^f + \Psi_5^P v_k^{A,P} \mathsf{n}_k^f\right] \\ -\left[\Psi_1 + \Psi_{q+1} v_q^A + \Psi_5 h_t\right]^P \mathsf{n}_1^f + (\gamma-1)\left[\Psi_{q+1}^P \mathsf{n}_q^f + \Psi_5^P v_q^{A,P} \mathsf{n}_q^f\right] v_k^{A,P} - \Psi_2^P v_q^{A,P} \mathsf{n}_q^f \\ -\left[\Psi_1 + \Psi_{q+1} v_q^A + \Psi_5 h_t\right]^P \mathsf{n}_2^f + (\gamma-1)\left[\Psi_{q+1}^P \mathsf{n}_q^f + \Psi_5^P v_q^{A,P} \mathsf{n}_q^f\right] v_k^{A,P} - \Psi_3^P v_q^{A,P} \mathsf{n}_q^f \\ -\left[\Psi_1 + \Psi_{q+1} v_q^A + \Psi_5 h_t\right]^P \mathsf{n}_3^f + (\gamma-1)\left[\Psi_{q+1}^P \mathsf{n}_q^f + \Psi_5^P v_q^{A,P} \mathsf{n}_q^f\right] v_k^{A,P} - \Psi_4^P v_q^{A,P} \mathsf{n}_q^f \\ -(\gamma-1)\left[\Psi_{q+1}^P \mathsf{n}_q^f + \Psi_5^P v_k^{A,P} \mathsf{n}_k^f\right] - \Psi_5^P v_k^{A,P} \mathsf{n}_k^f \end{bmatrix} \tag{4.61}$$

This expression is further processed by considering the ABC for wall boundaries, Eq. 3.99 ($\Psi_{q+1}\mathsf{n}_q + \frac{\partial J}{\partial p} = 0$), and term $\Psi_{q+1}^P \mathsf{n}_q^f$ is thus replaced by $-\left.\frac{\partial J}{\partial p}\right|^P$. Therefore,

Eq. 4.61 takes the form

$$\Phi_m^{\text{inv,adj},f} = \begin{bmatrix} v_k^{A,P}\mathsf{n}_k^f\left[\Psi_{q+1}v_q^A - \Psi_5\left(-h_t + \frac{\gamma-1}{2}v_k^A v_k^A\right)\right]^P \\ -\left[\Psi_1 + \Psi_{q+1}v_q^A + \Psi_5 h_t\right]^P \mathsf{n}_1^f + v_k^{A,P}\mathsf{n}_k^f\left[-\Psi_2 + (\gamma-1)\Psi_5 v_1^A\right]^P \\ -\left[\Psi_1 + \Psi_{q+1}v_q^A + \Psi_5 h_t\right]^P \mathsf{n}_2^f + v_k^{A,P}\mathsf{n}_k^f\left[-\Psi_3 + (\gamma-1)\Psi_5 v_2^A\right]^P \\ -\left[\Psi_1 + \Psi_{q+1}v_q^A + \Psi_5 h_t\right]^P \mathsf{n}_3^f + v_k^{A,P}\mathsf{n}_k^f\left[-\Psi_4 + (\gamma-1)\Psi_5 v_3^A\right]^P \\ -\gamma\Psi_5^P v_k^{A,P}\mathsf{n}_k^f \end{bmatrix} + \frac{\partial J}{\partial p}\bigg|^P \frac{\partial p}{\partial U_m}\bigg|^P \tag{4.62}$$

for use in *TDDC* adjoint. Eq. 4.62 is identical to the expression of Eq. 4.13 (term marked with → FAE_MF) plus the aforementioned missing term of Eq. 4.60.

Inlet/Outlet Boundaries: At the inlet and outlet boundaries, $f \in \mathcal{B}^{\mathcal{I}}(P)$ and $f \in \mathcal{B}^{\mathcal{O}}(P)$ respectively, let V_j^{loc} be the quantities for which a Dirichlet condition is imposed and V_λ^{loc} the flow quantity values extrapolated from the interior of the domain (zero Neumann condition). For the inlet boundaries, $j = 1,\ldots,4$ and $\lambda = 5$, while for outlet $j = 5$ and $\lambda = 1,\ldots,4$. For these boundaries, the adjoint flux on Eq. 4.53 is discretized as (with the above meaning for repeated indices λ and j)

$$\begin{aligned} \Phi_m^{\text{inv,adj},f} &= -\Psi_n^Q A_{nmk}^P \mathsf{n}_k^f = -\Psi_n^Q A_{nqk}^P \mathsf{n}_k^f \left[\frac{\partial U_q}{\partial V_j^{\text{loc}}}\frac{\partial V_j^{\text{loc}}}{\partial U_m} + \frac{\partial U_q}{\partial V_\lambda^{\text{loc}}}\frac{\partial V_\lambda^{\text{loc}}}{\partial U_m}\right] \\ &= V_j^{\text{adj,loc},Q}\frac{\partial V_j^{\text{loc}}}{\partial U_m} + V_\lambda^{\text{adj,loc},Q}\frac{\partial V_\lambda^{\text{loc}}}{\partial U_m} \end{aligned} \tag{4.63}$$

where

$$V_\kappa^{\text{adj,loc},*} = -\Psi_n^* A_{nqk}^P \mathsf{n}_k^f \frac{\partial U_q^Q}{\partial V_\kappa^{\text{loc}}} \tag{4.64}$$

For the quantities V_j^{loc}, for which a Dirichlet condition is imposed in the primal problem, $V_j^{\text{adj,loc}}$ are extrapolated from the interior domain, i.e. $V_j^{\text{adj,loc},Q} = V_j^{\text{adj,loc},P}$. In contrast, in the primal problem, $V_\lambda{}^{\text{loc}}$ are extrapolated from the interior domain while a Dirichlet condition is imposed, based on the ABC (Eq. 3.109), to $V_\lambda^{\text{adj,loc},Q} = \frac{\partial J}{\partial V_\lambda}$. By doing so, in the *TDDC* adjoint, Eq. 4.63 becomes

$$\Phi_m^{\text{inv,adj},f} = V_j^{\text{adj,loc},P}\frac{\partial V_j^{\text{loc}}}{\partial U_m^P} + \frac{\partial J}{\partial V_\lambda^{\text{loc}}}\frac{\partial V_\lambda^{\text{loc}}}{\partial U_m^P} \tag{4.65}$$

Eq. 4.65, after the missing term of Eq. 4.60, becomes identical to Eqs. 4.14 and 4.15 (terms marked with → FAE_MF).

Far–Field Boundaries: At the far–field boundaries, $f \in \mathcal{B}^{\mathcal{F}}(P)$, the adjoint flux of Eq. 4.53 is discretized according to the *TDDC* adjoint as

$$\Phi_m^{\text{inv,adj},f} = -\Psi_n^P \mathcal{A}_{nmk}^{-P} \mathsf{n}_k^f - \Psi_n^Q \mathcal{A}_{nmk}^{-P} \mathsf{n}_k^f \tag{4.66}$$

where, at the halo node Q, $\Psi_n^Q = 0$ and

$$\begin{aligned} \mathcal{A}_{nmk}^{-P} \mathsf{n}_k^f &= \frac{1}{2}\left(A_{nmk}^P \mathsf{n}_k^f - \left|\mathcal{A}_{nmk}^P \mathsf{n}_k^f\right|\right) \\ \left|\mathcal{A}_{nmk}^P \mathsf{n}_k^f\right| &= \left|A_{nmk}^P \mathsf{n}_k^f\right| - \left(U_q^Q - U_q^P\right) \frac{\partial \left|A_{nqk}^P \mathsf{n}_k^f\right|}{\partial U_m^P} \end{aligned} \tag{4.67}$$

$\left|\mathcal{A}_{nmk}^P \mathsf{n}_k^f\right|$ is the modified absolute Jacobian. Eq. 4.66 is the newly proposed, consistent non-conservative discretization scheme derived in this PhD Thesis by the *TDDC* adjoint.

4.4.2 Discretization of the Adjoint Viscous Terms

Integrating the viscous term of Eq. 3.67a over Ω_P, in the standard finite volume notation, and making the necessary approximations result in the balance of viscous adjoint fluxes (notations as in Fig. 2.1), written in the form

$$-\int_{\Omega_P}\left[\left(\frac{\partial \tau_{kq}^{\text{adj}}}{\partial x_k} - \tau_{kq}\frac{\partial \Psi_5}{\partial x_k}\right)\frac{\partial v_q^A}{\partial U_m} + \frac{\partial q_k^{\text{adj}}}{\partial x_k}\frac{\partial T}{\partial U_m}\right]\mathrm{d}\Omega \simeq -\sum_{Q\in\mathcal{N}(P)} \Phi_m^{\text{vis,adj},PQ} - \sum_{f\in\mathcal{B}(P)} \Phi_m^{\text{vis,adj},f} \tag{4.68}$$

Internal Adjoint Fluxes

The internal flux $\Phi_m^{\text{vis,adj},PQ}$ in Eq. 4.68 splits in three terms, namely

$$-\sum_{Q\in\mathcal{N}(P)}\left(\Phi_{q,\tau_{km}}^{\text{vis,adj},PQ}\frac{\partial v_q^{A,P}}{\partial U_m^P} - \Phi_{q,v_m^A}^{\text{vis,adj},PQ}\frac{\partial v_q^{A,P}}{\partial U_m^P} + \Phi_{q_k}^{\text{vis,adj},PQ}\frac{\partial T^P}{\partial U_m^P}\right)$$

Based on the hand–differentiated discrete adjoint equations, Sec. 4.1.2, the proposed consistent discretization scheme for the adjoint viscous fluxes is

$$\Phi^{\text{vis,adj},PQ}_{q,\tau_{km}} = \left[\frac{\mu+\mu_t}{\text{Re}_0}\mathsf{n}_k\left(\frac{\partial\Psi_{q+1}}{\partial x_k}+\frac{\partial\Psi_5}{\partial x_k}v_q^A\right)\right]^{\text{adj}}_{PQ} + \left[\frac{\mu+\mu_t}{\text{Re}_0}\mathsf{n}_q\left(\frac{\partial\Psi_{k+1}}{\partial x_k}+\frac{\partial\Psi_5}{\partial x_k}v_k^A\right)\right]^{\text{adj}}_{PQ} - \left[\frac{2}{3}\delta_{kq}\frac{\mu+\mu_t}{\text{Re}_0}\mathsf{n}_\ell\left(\frac{\partial\Psi_{\ell+1}}{\partial x_k}+\frac{\partial\Psi_5}{\partial x_k}v_\ell^A\right)\right]^{\text{adj}}_{PQ} \tag{4.69}$$

$$\Phi^{\text{vis,adj},PQ}_{q,v_m^A} = \frac{1}{2}\left(\Psi_5^Q-\Psi_5^P\right)\tau_{kq}^{PQ}\mathsf{n}_k^{PQ} \tag{4.70}$$

$$\Phi^{\text{vis,adj},PQ}_{q_k} = \left[\frac{\mathcal{C}_p}{\text{Re}_0}\left(\frac{\mu}{\text{Pr}}+\frac{\mu_t}{\text{Pr}_\text{t}}\right)\mathsf{n}_k\frac{\partial\Psi_5}{\partial x_k}\right]^{\text{adj}}_{PQ} \tag{4.71}$$

where the adjoint spatial gradient of an adjoint quantity ϕ, multiplied by a primal quantity B, at the interface between P and Q is defined as

$$\left[B\frac{\partial\phi}{\partial x_m}\right]^{\text{adj}}_{PQ} = \frac{1}{2}\left[\left[B\left(\delta_{m\ell}-\hat{\mathsf{t}}_\ell\hat{\mathsf{t}}_m\right)\frac{\partial\phi}{\partial x_\ell}\right]^{P,\text{adj}} - \left[B\left(\delta_{m\ell}-\hat{\mathsf{t}}_\ell\hat{\mathsf{t}}_m\right)\frac{\partial\phi}{\partial x_\ell}\right]^{Q,\text{adj}}\right] + B^{PQ}\frac{\phi^Q-\phi^P}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}\hat{\mathsf{t}}_m^{PQ} \tag{4.72}$$

In a "standard" discretization scheme, Eq. 2.58 should be used instead of the proposed Eq. 4.72 for the discretization of the adjoint viscous fluxes; the former is re-written for the purpose of comparison as

$$\left.\frac{\partial\phi}{\partial x_m}\right|_{PQ} = \frac{1}{2}\left(\delta_{m\ell}-\hat{\mathsf{t}}_\ell^{PQ}\hat{\mathsf{t}}_m^{PQ}\right)\left(\left.\frac{\partial\phi}{\partial x_\ell}\right|^P+\left.\frac{\partial\phi}{\partial x_\ell}\right|^Q\right)+\frac{\phi^Q-\phi^P}{\sqrt{x_\ell^{PQ}x_\ell^{PQ}}}\hat{\mathsf{t}}_m^{PQ} \tag{4.73}$$

An additional difference between "standard" and *TDDC* adjoint is the edge which the primal quantity B is computed on. In the "standard" discretization scheme, the primal quantity B is computed based on the edge PQ and is, then, multiplied by the spatial gradient of the adjoint gradient at the interface PQ, Eq. 2.58. In the *TDDC* adjoint, the computation of the primal quantity B is performed at the edge that appears at the computation of the adjoint spatial gradient.

Boundary Adjoint Fluxes

The discretization of the boundary flux $\Phi_m^{\text{vis,adj},f}$ for the wall, inlet, outlet and far–field boundaries follows.

Wall Boundaries: For the wall boundaries, $f \in \mathcal{B}^{\mathcal{W}}(P)$, the boundary flux $\Phi_m^{\text{vis,adj},f}$ in Eq. 4.68, taking also into account J's contribution, can be re-written as

$$\Phi_m^{\text{vis,adj},f} = \frac{\partial J}{\partial(\tau_{\ell k}\mathsf{n}_k\mathsf{n}_\ell)}\bigg|^f + \frac{\partial J}{\partial(\tau_{\ell k}\mathsf{n}_k\mathsf{t}_\ell)}\bigg|^f + \frac{\partial J}{\partial(q_k\mathsf{n}_k)}\bigg|^f + \frac{\partial J}{\partial T}\bigg|^f - \left(\Phi_{q,\tau_{km}}^{\text{vis,adj},f}\frac{\partial v_q^{A,P}}{\partial U_m^P} - \Phi_{q,v_m^A}^{\text{vis,adj},f}\frac{\partial v_q^{A,P}}{\partial U_m^P} + \Phi_{q_k}^{\text{vis,adj},f}\frac{\partial T^P}{\partial U_m^P}\right)$$

where the third and fourth term vanishes in case of adiabatic/constant heat flux or constant tempretature wall, respectively. Based on the *TDDC* adjoint, the proposed discretization schemes for the boundary fluxes are

$$\Phi_{q,\tau_{km}}^{\text{vis,adj},f} = \frac{\partial \tau_{\text{adj},q\ell}}{\partial x_\ell}\bigg|^{f,\text{adj}} \tag{4.74}$$

$$\Phi_{q,v_m^A}^{\text{vis,adj},f} = -\Psi_5^P \tau_{kq}^P \mathsf{n}_k^f \tag{4.75}$$

$$\Phi_{q_k}^{\text{vis,adj},f} = \frac{\partial q_{\text{adj},\ell}}{\partial x_\ell}\bigg|^{f,\text{adj}} \tag{4.76}$$

where the expression of Eq. 4.56 is used for the computation of $\frac{\partial \tau_{\text{adj}q\ell}}{\partial x_\ell}\Big|^{f,\text{adj}}$ and $\frac{\partial q_{\text{adj},\ell}}{\partial x_\ell}\Big|^{f,\text{adj}}$.

Inlet/Outlet/Far–Field Boundaries: Regarding the boundary adjoint fluxes, in the continuous adjoint, the total variation in the viscous stresses and heat flux is assumed to be zero along the inlet, outlet and far–field boundaries, $f \in \mathcal{B}^{\mathcal{I}}(P)$, $f \in \mathcal{B}^{\mathcal{O}}(P)$ and $f \in \mathcal{B}^{\mathcal{F}}(P)$, respectively. On the other hand, in discrete adjoint there are, inevitably, contributions of the viscous stresses and heat fluxes along the inlet, outlet and far–field boundaries. This contribution is negligible, and if omitted, it is expected not to affect the accuracy of the computed sensitivity derivatives. As a result, the continuous (even the *TDDC*) adjoint ignores any contribution from the viscous terms.

4.4.3 Discretization of the Adjoint Source Terms

The source terms of the adjoint mean flow equations of Eq. 3.67a ($\mathcal{K}_m^{\text{SA}}$, $\mathcal{K}_m^{\gamma-\tilde{R}e_{\theta t}}$) include terms that arise from the differentiation of the convection, diffusion and sources of the Spalart–Allmaras and the $\gamma-\tilde{R}e_{\theta t}$ models. These terms are treated

separately. Terms coming from the differentiation of the convection terms of the Spalart–Allmaras and $\gamma - \tilde{Re}_{\theta t}$ equations are written as

$$-\int_{\Omega_P} \rho\varphi \frac{\partial \varphi_a}{\partial x_k}\frac{\partial v_k^A}{\partial U_m}\mathrm{d}\Omega \simeq \sum_{Q\in\mathcal{N}(P)} \Phi_{k,\mathrm{conv}}^{\mathrm{src,adj},PQ}\frac{\partial v_k^{A,P}}{\partial U_m^P} + \sum_{f\in\mathcal{B}(P)} \Phi_{k,\mathrm{conv}}^{\mathrm{src,adj},f}\frac{\partial v_k^{A,P}}{\partial U_m^P} \tag{4.77}$$

Based on the *TDDC* adjoint, fluxes $\Phi_{k,\mathrm{conv}}^{\mathrm{src,adj},PQ}$ and $\Phi_{k,\mathrm{conv}}^{\mathrm{src,adj},f}$ are discretized as

$$\Phi_{k,\mathrm{conv}}^{\mathrm{src,adj},PQ} = -\frac{1}{2}\Big[(\rho\varphi)^P\left(\varphi_a^P+\varphi_a^Q\right)\mathsf{n}_k^{PQ} + \frac{1}{2}\left(\varphi_a^P-\varphi_a^Q\right)\left((\rho\varphi)^Q-(\rho\varphi)^P\right) sign\left(v_k^{A,PQ}\mathsf{n}_k^{PQ}\right)\mathsf{n}_k^{PQ}\Big] \tag{4.78}$$

$$\Phi_{k,\mathrm{conv}}^{\mathrm{src,adj},f} = 0 \tag{4.79}$$

where φ and φ_a stand for $\tilde{\nu}, \gamma, \tilde{Re}_{\theta t}$ and $\tilde{\nu}_a, \gamma_a, \tilde{Re}_a$, respectively. Similar to the discretization of the inviscid fluxes, Sec. 4.4.1, terms $\frac{1}{2}\sum_Q (\rho\varphi)^P\left(\varphi_a^P-\varphi_a^Q\right)\mathsf{n}_k^{PQ}$ of Eqs. 4.20 and 4.32 are re-written as $-\frac{1}{2}\sum_Q (\rho\varphi)^P\left(\varphi_a^P+\varphi_a^Q\right)\mathsf{n}_k^{PQ} - (\rho\varphi)^P\varphi_a^P\sum_f \mathsf{n}_k^f$.

Terms coming from the differentiation of the Spalart–Allmaras diffusion become

$$\begin{aligned}
&-\int_{\Omega_P}\frac{\tilde{\nu}_a}{\mathrm{Re}_0\,\sigma}\left[\frac{\partial}{\partial x_k}\left[\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial\tilde{\nu}}{\partial x_k}\right]-c_{b_2}\tilde{\nu}\frac{\partial\tilde{\nu}}{\partial x_k}\right]\mathrm{d}\Omega - \int_{\Omega_P}\frac{1}{\mathrm{Re}_0\,\sigma}\frac{\partial\left(\rho\tilde{\nu}_a\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\frac{\nu}{\rho}\mathrm{d}\Omega\\
&-\int_{\Omega_P}\frac{1}{\mathrm{Re}_0\,\sigma}\left[(1+c_{b_2})\frac{\partial\left(\rho\tilde{\nu}_a\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}+c_{b_2}\rho\tilde{\nu}_a\frac{\partial}{\partial x_k}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\right)\right]\frac{\tilde{\nu}}{\rho}\mathrm{d}\Omega\\
&+\int_{\Omega_P}\left[\frac{\partial}{\partial x_k}\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial\left(\rho\tilde{\nu}_a\right)}{\partial x_k}-c_{b_2}\frac{\partial}{\partial x_k}\left(\frac{\partial\left(\rho\tilde{\nu}_a\tilde{\nu}\right)}{\partial x_k}\right)\right]\frac{\tilde{\nu}}{\rho}\mathrm{d}\Omega\\
&\simeq \sum_{Q\in\mathcal{N}(P)}\Phi_{\mathrm{diff,SA},\rho}^{\mathrm{src,adj},PQ}+\sum_{Q\in\mathcal{N}(P)}\Phi_{\mathrm{diff,SA},\nu}^{\mathrm{src,adj},PQ}\frac{\partial\nu^P}{\partial\rho^P}+\sum_{Q\in\mathcal{N}(P)}\Phi_{\mathrm{diff,SA},\tilde{\nu}}^{\mathrm{src,adj},PQ}\frac{\partial\tilde{\nu}^P}{\partial\rho^P}\\
&+\sum_{f\in\mathcal{B}(P)}\Phi_{\mathrm{diff,SA},\rho}^{\mathrm{src,adj},f}+\sum_{f\in\mathcal{B}(P)}\Phi_{\mathrm{diff,SA},\nu}^{\mathrm{src,adj},f}\frac{\partial\nu^P}{\partial\rho^P}+\sum_{f\in\mathcal{B}(P)}\Phi_{\mathrm{diff,SA},\tilde{\nu}}^{\mathrm{src,adj},f}\frac{\partial\tilde{\nu}^P}{\partial\rho^P}
\end{aligned} \tag{4.80}$$

and, the *TDDC* adjoint proposes the following discretization schemes

$$\Phi_{\mathrm{diff,SA},\rho}^{\mathrm{src,adj},PQ} = -\frac{1}{\mathrm{Re}_0\,\sigma}\tilde{\nu}_a^P\left[\nu^{PQ}+(1+c_{b_2})\,\tilde{\nu}^{PQ}-c_{b_2}\tilde{\nu}^P\right]\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ} \tag{4.81}$$

$$\Phi^{\text{src,adj},PQ}_{\text{diff,SA},\nu} = \frac{1}{\text{Re}_0\,\sigma}\left[(\rho\tilde{\nu_a})^{PQ} - (\rho\tilde{\nu_a})^{P}\right]\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ} \tag{4.82}$$

$$\begin{aligned}\Phi^{\text{src,adj},PQ}_{\text{diff,SA},\tilde{\nu}} = &-\frac{1}{\text{Re}_0\,\sigma}\left[(1+c_{b_2})\,(\rho\tilde{\nu_a})^{PQ} - (\rho\tilde{\nu_a})^{P}\right]\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ}\\ &+\frac{1}{\text{Re}_0\,\sigma}\left[\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right]\frac{\partial\,(\rho\tilde{\nu_a})}{\partial x_k}\mathsf{n}_k - c_{b_2}\frac{\partial\,(\rho\tilde{\nu_a}\tilde{\nu})}{\partial x_k}\mathsf{n}_k\right]^{\text{adj}}_{PQ}\end{aligned} \tag{4.83}$$

$$\Phi^{\text{src,adj},f}_{\text{diff,SA},\rho} = -\frac{1}{\text{Re}_0\,\sigma}\tilde{\nu_a}^{P}\left[\nu^f + (1+c_{b_2})\,\tilde{\nu}^f + c_{b_2}\tilde{\nu}^P\right]\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_{f}\mathsf{n}_k^{f} \tag{4.84}$$

$$\Phi^{\text{src,adj},f}_{\text{diff,SA},\nu} = -\frac{1}{\text{Re}_0\,\sigma}\,(\rho\tilde{\nu_a})^{P}\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_{f}\mathsf{n}_k^{f} \tag{4.85}$$

$$\Phi^{\text{src,adj},f}_{\text{diff,SA},\tilde{\nu}} = \frac{1}{\text{Re}_0\,\sigma}\frac{\partial}{\partial x_k}\left[\left[(\rho\tilde{\nu_a})\left[\nu+(1+c_{b_2})\,\tilde{\nu}\right] - c_{b_2}\,(\rho\tilde{\nu_a}\tilde{\nu})\right]\mathsf{n}_k\right]^{f,\text{adj}} \tag{4.86}$$

Terms coming from the differentiation of the γ and $\tilde{Re}_{\theta t}$ diffusion read

$$\begin{aligned}-\int_{\Omega_P}\frac{1}{\text{Re}_0}\frac{\gamma}{\rho}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\right]\text{d}\Omega &\simeq -\sum_{Q\in\mathcal{N}(P)}\Phi^{src,\text{adj},PQ}_{\text{diff},\gamma}\frac{\gamma^P}{\rho^P} - \sum_{f\in\mathcal{B}(P)}\Phi^{src,\text{adj},f}_{\text{diff},\gamma}\frac{\gamma^P}{\rho^P}\\ -\int_{\Omega_P}\frac{1}{\text{Re}_0}\frac{\tilde{Re}_{\theta t}}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\,(\mu+\mu_t)\,\frac{\partial\tilde{Re}_a}{\partial x_k}\right]\text{d}\Omega &\simeq -\sum_{Q\in\mathcal{N}(P)}\Phi^{src,\text{adj},PQ}_{\text{diff},\tilde{Re}_{\theta t}}\frac{\tilde{Re}^P_{\theta t}}{\rho^P} - \sum_{f\in\mathcal{B}(P)}\Phi^{src,\text{adj},f}_{\text{diff},\tilde{Re}_{\theta t}}\frac{\tilde{Re}^P_{\theta t}}{\rho^P}\end{aligned} \tag{4.87}$$

In the *TDDC* adjoint and using Eq. 4.72, fluxes $\Phi^{src,\text{adj},PQ}_{\text{diff},\gamma}$, $\Phi^{src,\text{adj},PQ}_{\text{diff},\tilde{Re}_{\theta t}}$, $\Phi^{src,\text{adj},f}_{\text{diff},\gamma}$, $\Phi^{src,\text{adj},f}_{\text{diff},\tilde{Re}_{\theta t}}$ are discretized as

$$\Phi^{src,\text{adj},PQ}_{\text{diff},\gamma} = \left[\frac{1}{\text{Re}_0}\left(\mu+\frac{\mu_t}{\sigma_f}\right)\mathsf{n}_k\frac{\partial\gamma_a}{\partial x_k}\right]^{\text{adj}}_{PQ}, \qquad \Phi^{src,\text{adj},f}_{\text{diff},\gamma} = \frac{\partial}{\partial x_k}\left[\frac{1}{\text{Re}_0}\left(\mu+\frac{\mu_t}{\sigma_f}\right)\mathsf{n}_k\frac{\partial\gamma_a}{\partial x_k}\right]^{f,\text{adj}}$$

$$\Phi^{src,\text{adj},PQ}_{\text{diff},\tilde{Re}_{\theta t}} = \left[\frac{1}{\text{Re}_0}\sigma_{\theta,t}(\mu+\mu_t)\mathsf{n}_k\frac{\partial\tilde{Re}_a}{\partial x_k}\right]^{\text{adj}}_{PQ}, \qquad \Phi^{src,\text{adj},f}_{\text{diff},\tilde{Re}_{\theta t}} = \frac{\partial}{\partial x_k}\left[\frac{1}{\text{Re}_0}\sigma_{\theta,t}(\mu+\mu_t)\mathsf{n}_k\frac{\partial\tilde{Re}_a}{\partial x_k}\right]^{f,\text{adj}}$$

For the discretization of the terms that arise from the differentiation of the Spalart–Allmaras and $\gamma-\tilde{Re}_{\theta t}$ diffusion, Eqs. 4.72 and 4.56 are used for the internal and boundary fluxes, respectively.

The remaining terms come from the differentiation of ζ, S, $d|\mathbf{U}|ds$ and $\zeta_{streamwise}$ that are present in the Spalart–Allmaras and $\gamma-\tilde{Re}_{\theta t}$ sources. Recall that all these terms include the gradient of the velocity components. Consistent discretization

schemes for them, based on the *TDDC* adjoint, are proposed using Eq. 4.56, and these are

$$\int_{\Omega_P} \frac{\partial}{\partial x_\ell}\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\text{SA}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\right]\mathrm{d}\Omega \simeq \frac{\partial}{\partial x_\ell}\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\text{SA}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\Omega\right]^{P,\text{adj}} \tag{4.88}$$

$$\int_{\Omega_P} \frac{\partial}{\partial x_m}\left[\tilde{\nu_a}\frac{\mathcal{C}_S^{\text{SA}}}{S}2S_{km}\right]\mathrm{d}\Omega \simeq \frac{\partial}{\partial x_m}\left[\tilde{\nu_a}\frac{\mathcal{C}_S^{\text{SA}}}{S}2S_{km}\Omega\right]^{P,\text{adj}} \tag{4.89}$$

$$\int_{\Omega_P} \frac{\partial}{\partial x_m}\left[\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v_k^A v_m^A}{|\mathbf{U}|^2}\right]\mathrm{d}\Omega \simeq \frac{\partial}{\partial x_m}\left[\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v_k^A v_m^A}{|\mathbf{U}|^2}\Omega\right]^{P,\text{adj}} \tag{4.90}$$

$$\int_{\Omega_P} \frac{\partial}{\partial x_q}\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_m^A}{|\mathbf{U}|}\varepsilon_{mqk}\right]\mathrm{d}\Omega \simeq \frac{\partial}{\partial x_q}\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_m^A}{|\mathbf{U}|}\varepsilon_{mqk}\Omega\right]^{P,\text{adj}} \tag{4.91}$$

$$\int_{\Omega_P} \frac{\partial}{\partial x_\ell}\left[\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\right]\mathrm{d}\Omega \simeq \frac{\partial}{\partial x_\ell}\left[\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\Omega\right]^{P,\text{adj}} \tag{4.92}$$

$$\int_{\Omega_P} \frac{\partial}{\partial x_m}\left[\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\right]\mathrm{d}\Omega \simeq \frac{\partial}{\partial x_m}\left[\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\Omega\right]^{P,\text{adj}} \tag{4.93}$$

4.4.4 Discretization of the Adjoint Turbulence Model

Consistent *TDDC* discretization schemes for the convection, diffusion and source terms of the adjoint Spalart–Allmaras equation (Eq. 3.67b) are proposed.

Convection Integrating the convection terms of Eq. 3.67b over a finite volume Ω_P and making the necessary approximations, one gets

$$-\int_{\Omega_P} v_k^A\frac{\partial\tilde{\nu_a}}{\partial x_k}\mathrm{d}\Omega \simeq \sum_{Q\in\mathcal{N}(P)}\Phi_{\text{conv}}^{\tilde{\nu_a},PQ} + \sum_{f\in\mathcal{B}(P)}\Phi_{\text{conv}}^{\tilde{\nu_a},f} \tag{4.94}$$

The discretization scheme of the adjoint flux $\Phi_{\text{conv}}^{\tilde{\nu_a},PQ}$ depends on whether the FVS, [129], or the Roe's, [117], upwind scheme is used for the discretization of the primal fluxes. When the FVS scheme is used, the consistent discretization

scheme of the *TDDC* adjoint is

$$\Phi_{\text{conv}}^{\tilde{\nu_a},PQ} = -\left(\tilde{\nu_a}\right)^{PQ}\left(v_k^A \mathsf{n}_k\right)^{PQ} \quad \text{with} \quad \left(\varphi_a\right)^{PQ} = \begin{cases} \tilde{\nu_a}^P \ , & \text{for} \ -\left(v_k^A \mathsf{n}_k\right)^{PQ} > 0 \\ \tilde{\nu_a}^Q \ , & \text{for} \ -\left(v_k^A \mathsf{n}_k\right)^{PQ} < 0 \end{cases} \tag{4.95}$$

$$\Phi_{\text{conv}}^{\tilde{\nu_a},f} = 0 \tag{4.96}$$

whereas, for the Roe's scheme, the consistent discretization scheme proposed by the *TDDC* adjoint is

$$\Phi_{\text{conv}}^{\tilde{\nu_a},PQ} = -\frac{1}{2}\left(\tilde{\nu_a}^P + \tilde{\nu_a}^Q\right) v_k^{A,P} \mathsf{n}_k^{PQ} - \frac{1}{2}\left(\tilde{\nu_a}^Q - \tilde{\nu_a}^P\right)\left|v_k^{A,PQ} \mathsf{n}_k^{PQ}\right| \tag{4.97}$$

$$\Phi_{\text{conv}}^{\tilde{\nu_a},f} = 0 \tag{4.98}$$

In both schemes, the *TDDC* adjoint proposes a downwind discretization scheme for the internal fluxes, in contrast to the upwind one used in the primal problem. Moreover, the *TDDC* adjoint results in zero flux for all boundaries.

Diffusion Integrating the diffusion term of Eq. 3.67b over a finite volume Ω_P yields

$$\begin{aligned} -\int_{\Omega_P} \mathcal{G}^{\text{SA},diff} \mathrm{d}\Omega = & \int_{\Omega_P} \frac{1}{\mathrm{Re}_0\,\sigma}\left[\left(1+c_{b_2}\right)\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k} + c_{b_2}\rho\tilde{\nu_a}\frac{\partial}{\partial x_k}\left(\frac{\partial\tilde{\nu}}{\partial x_k}\right)\right]\frac{1}{\rho}\mathrm{d}\Omega \\ & - \int_{\Omega_P} \frac{1}{\mathrm{Re}_0\,\sigma}\left[\frac{\partial}{\partial x_k}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\right] - c_{b_2}\frac{\partial}{\partial x_k}\left(\frac{\partial\left(\rho\tilde{\nu_a}\tilde{\nu}\right)}{\partial x_k}\right)\right]\frac{1}{\rho}\mathrm{d}\Omega \\ = & \simeq \sum_{Q\in\mathcal{N}(P)} \Phi_{\text{diff}}^{\tilde{\nu_a},PQ}\frac{1}{\rho^P} + \sum_{f\in\mathcal{B}(P)} \Phi_{\text{diff}}^{\tilde{\nu_a},f}\frac{1}{\rho^P} \end{aligned} \tag{4.99}$$

In the *TDDC* way, the adjoint diffusion fluxes should be discretized as

$$\begin{aligned} \Phi_{\text{diff}}^{\tilde{\nu_a},PQ} = & \frac{1}{\mathrm{Re}_0\,\sigma}\left[\left(1+c_{b_2}\right)\left(\rho\tilde{\nu_a}\right)^{PQ} - \left(\rho\tilde{\nu_a}\right)^P\right]\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ} \\ & - \frac{1}{\mathrm{Re}_0\,\sigma}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\mathsf{n}_k - c_{b_2}\frac{\partial\left(\rho\tilde{\nu_a}\tilde{\nu}\right)}{\partial x_k}\mathsf{n}_k\right]_{PQ}^{\text{adj}} \end{aligned} \tag{4.100}$$

$$\Phi_{\text{diff}}^{\tilde{\nu_a},f} = -\frac{1}{\mathrm{Re}_0\,\sigma}\left(\rho\tilde{\nu_a}\right)^P\left.\frac{\partial\tilde{\nu}}{\partial x_k}\right|_f - \frac{1}{\mathrm{Re}_0\,\sigma}\frac{\partial}{\partial x_k}\left[\left[\left(\rho\tilde{\nu_a}\right)\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right] - c_{b_2}\left(\rho\tilde{\nu_a}\tilde{\nu}\right)\right]\mathsf{n}_k\right]^{f,\text{adj}} \tag{4.101}$$

The *TDDC* adjoint proposes the use of Eqs. 4.72 and 4.56 for the computation of the spatial gradients instead of Eqs. 2.58 and 2.50 that are used for the discretization of the primal problem and, probably, of the "standard" continuous adjoint.

Sources Integrating over Ω_P the adjoint source terms that are coming from the differentiation of the mean flow equations, $\mathcal{G}^{\mu_t}$, and after making the appropriate approximations

$$\begin{aligned}\int_{\Omega_P}\mathcal{G}^{\mu_t}\frac{\partial\mu_t}{\partial\tilde{\mu}}\mathrm{d}\Omega&=\int_{\Omega_P}\left[\frac{\tau_{km}}{\mu+\mu_t}\left(\frac{\partial\Psi_{m+1}}{\partial x_k}+\frac{\partial\Psi_5}{\partial x_k}v_m^A\right)+\frac{\mathcal{C}_p}{\mathrm{Re}_0}\frac{\partial\Psi_5}{\partial x_k}\frac{\partial T}{\partial x_k}\frac{1}{\mathrm{Pr}_\mathrm{t}}\right]\frac{\partial\mu_t}{\partial\tilde{\mu}}\mathrm{d}\Omega\\&\simeq\sum_{Q\in\mathcal{N}(P)}\Phi^{PQ}_{\mathrm{adj},\mu_t}\frac{\partial\mu_t^P}{\partial\tilde{\mu}^P}+\sum_{f\in\mathcal{B}(P)}\Phi^{f}_{\mathrm{adj},\mu_t}\frac{\partial\mu_t^P}{\partial\tilde{\mu}^P}\end{aligned}\tag{4.102}$$

The proposed discretization schemes based on the *TDDC* adjoint are

$$\begin{aligned}\Phi^{PQ}_{\mathrm{adj},\mu_t}&=\left[\left(\Psi^{PQ}_{m+1}-\Psi^{P}_{m+1}\right)+\left(\Psi^{PQ}_{5}-\Psi^{P}_{5}\right)v_m^{A,PQ}\right]\mathsf{n}_k^{PQ}\frac{\tau^{PQ}_{km}}{\mu^{PQ}+\mu_t^{PQ}}+\frac{\mathcal{C}_p}{\mathrm{Re}_0}\left(\Psi^{PQ}_{5}-\Psi^{P}_{5}\right)\left.\frac{\partial T}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ}\\\Phi^{f}_{\mathrm{adj},\mu_t}&=-\left[\Psi^{P}_{m+1}+\Psi^{P}_{5}v_m^{A,P}\right]\mathsf{n}_k^{f}\frac{\tau^{P}_{km}}{\mu^{P}+\mu_t^{P}}-\frac{\mathcal{C}_p}{\mathrm{Re}_0}\Psi^{P}_{5}\left.\frac{\partial T}{\partial x_k}\right|_{f}\mathsf{n}_k^{f}\end{aligned}\tag{4.103}$$

Regarding the adjoint source terms that are coming from the differentiation of the Spalart–Allmaras model, $\mathcal{G}^{\mathrm{SA},src}$, a nodal discretization is proposed

$$\int_{\Omega_P}\mathcal{G}^{\mathrm{SA},src}\mathrm{d}\Omega\simeq\mathcal{G}^{\mathrm{SA},src,P}\Omega_P\tag{4.104}$$

Finally, integrating the adjoint source terms coming from the differentiation of the SA-sLM2015 model, $\mathcal{G}^{\mu_t,\gamma-\tilde{R}e_{\theta t}}$, over Ω_P, one gets

$$\begin{aligned}\int_{\Omega}\mathcal{G}^{\mu_t,\gamma-\tilde{R}e_{\theta t}}\frac{\partial\mu_t}{\partial\tilde{\mu}}\mathrm{d}\Omega&=\int_{\Omega}\frac{1}{\mathrm{Re}_0}\left(\frac{\partial\gamma_a}{\partial x_k}\frac{\partial\gamma}{\partial x_k}\frac{1}{\sigma_f}+\frac{\partial\tilde{R}e_a}{\partial x_k}\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\sigma_{\theta,t}\right)\frac{\partial\mu_t}{\partial\tilde{\mu}}\mathrm{d}\Omega+\int_{\Omega}\mathcal{C}^{\gamma-\tilde{R}e_{\theta t}}_{\mu_t}\frac{\partial\mu_t}{\partial\tilde{\mu}}\mathrm{d}\Omega\\&\simeq\sum_{Q\in\mathcal{N}(P)}\Phi^{\gamma-\tilde{R}e_{\theta t},PQ}_{\mathrm{adj},\mu_t}\frac{\partial\mu_t^P}{\partial\tilde{\mu}^P}+\sum_{f\in\mathcal{B}(P)}\Phi^{\gamma-\tilde{R}e_{\theta t},f}_{\mu_t}\frac{\partial\mu_t^P}{\partial\tilde{\mu}^P}+\left[\mathcal{C}^{\gamma-\tilde{R}e_{\theta t}}_{\mathrm{adj},\mu_t}\frac{\partial\mu_t}{\partial\tilde{\mu}}\right]^P\Omega_P\end{aligned}\tag{4.105}$$

In the *TDDC* adjoint, fluxes $\Phi_{\text{adj},\mu_t}^{\gamma-\tilde{R}e_{\theta t},PQ}$ and $\Phi_{\text{adj},\mu_t}^{\gamma-\tilde{R}e_{\theta t},f}$ should be discretized as

$$\Phi_{\text{adj},\mu_t}^{\gamma-\tilde{R}e_{\theta t},PQ} = \frac{1}{\text{Re}_0}\left(\left(\gamma_a^{PQ}-\gamma_a^{P}\right)\left.\frac{\partial\gamma}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ}\frac{1}{\sigma_f}+\left(\tilde{R}e_a^{PQ}-\tilde{R}e_a^{P}\right)\left.\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\right|_{PQ}\mathsf{n}_k^{PQ}\sigma_{\theta,t}\right)$$

$$\Phi_{\text{adj},\mu_t}^{\gamma-\tilde{R}e_{\theta t},f} = -\frac{1}{\text{Re}_0}\left(\gamma_a^{P}\left.\frac{\partial\gamma}{\partial x_k}\right|_{f}\mathsf{n}_k^{f}\frac{1}{\sigma_f}+\tilde{R}e_a^{P}\left.\frac{\partial\tilde{R}e_{\theta t}}{\partial x_k}\right|_{f}\mathsf{n}_k^{f}\sigma_{\theta,t}\right) \tag{4.106}$$

4.4.5 Discretization of the Adjoint Transition Model

Consistent *TDDC* discretization schemes for the convection, diffusion and source terms of the adjoint γ and adjoint $\tilde{R}e_{\theta t}$ equations (Eqs. 3.67c and 3.67d), are proposed.

Convection Integrating the convection terms of Eqs. 3.67c and 3.67d over a finite volume Ω_P and making the necessary approximations, one gets

$$-\int_{\Omega_P} v_k^A\frac{\partial\gamma_a}{\partial x_k}\text{d}\Omega \simeq \sum_{Q\in\mathcal{N}(P)}\Phi_{\text{conv}}^{\gamma_a,PQ}+\sum_{f\in\mathcal{B}(P)}\Phi_{\text{conv}}^{\gamma_a,f} \tag{4.107}$$

$$-\int_{\Omega_P} v_k^A\frac{\partial\tilde{R}e_a}{\partial x_k}\text{d}\Omega \simeq \sum_{Q\in\mathcal{N}(P)}\Phi_{\text{conv}}^{\tilde{R}e_a,PQ}+\sum_{f\in\mathcal{B}(P)}\Phi_{\text{conv}}^{\tilde{R}e_a,f} \tag{4.108}$$

The discretization scheme of the adjoint fluxes $\Phi_{\text{conv}}^{\gamma_a,PQ}$ and $\Phi_{\text{conv}}^{\tilde{R}e_a,PQ}$ depends on whether the FVS, [129], or the Roe's, [117], upwind scheme is used for the discretization of the primal fluxes. When the FVS scheme is used, the consistent discretization scheme of the *TDDC* adjoint is

$$\Phi_{\text{conv}}^{\varphi_a,PQ} = -\left(\varphi_a\right)^{PQ}\left(v_k^A\mathsf{n}_k\right)^{PQ} \quad \text{with} \quad \left(\varphi_a\right)^{PQ} = \begin{cases}\varphi_a^P\ , & \text{for } -\left(v_k^A\mathsf{n}_k\right)^{PQ} > 0\\ \varphi_a^Q\ , & \text{for } -\left(v_k^A\mathsf{n}_k\right)^{PQ} < 0\end{cases} \tag{4.109}$$

$$\Phi_{\text{conv}}^{\varphi_a,f} = 0 \tag{4.110}$$

whereas, for the Roe's scheme, this becomes

$$\Phi_{\text{conv}}^{\varphi_a,PQ} = -\frac{1}{2}\left(\varphi_a^P+\varphi_a^Q\right)v_k^{A,P}\mathsf{n}_k^{PQ}-\frac{1}{2}\left(\varphi_a^Q-\varphi_a^P\right)\left|v_k^{A,PQ}\mathsf{n}_k^{PQ}\right| \tag{4.111}$$

$$\Phi_{\text{conv}}^{\varphi_a,f} = 0 \tag{4.112}$$

where φ_a stands for either γ_a or $\tilde{Re}_a$. In both schemes, the *TDDC* adjoint proposes a downwind discretization scheme for the internal fluxes, in contrast to the upwind one used in the primal problem. Moreover, the *TDDC* results in zero flux for all boundaries.

Diffusion Integrating the diffusion terms of Eqs. 3.67c and 3.67d ($\mathcal{H}_{\gamma}^{\gamma-\tilde{Re}_{\theta t},diff}$ and $\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},diff}$) over a finite volume P and making appropriate approximations one obtains

$$
\begin{aligned}
-\int_{\Omega_P}\frac{1}{\mathrm{Re}_0}\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\right]\mathrm{d}\Omega &\simeq -\sum_{Q\in\mathcal{N}(P)}\Phi_{\mathrm{diff}}^{\gamma_a,PQ}\frac{1}{\rho^P}-\sum_{f\in\mathcal{B}(P)}\Phi_{\mathrm{diff}}^{\gamma_a,f}\frac{1}{\rho^P}\\
-\int_{\Omega_P}\frac{1}{\mathrm{Re}_0}\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\right]\mathrm{d}\Omega &\simeq -\sum_{Q\in\mathcal{N}(P)}\Phi_{\mathrm{diff}}^{\tilde{Re}_a,PQ}\frac{1}{\rho^P}-\sum_{f\in\mathcal{B}(P)}\Phi_{\mathrm{diff}}^{\tilde{Re}_a,f}\frac{1}{\rho^P}
\end{aligned}
\tag{4.113}
$$

In the *TDDC* adjoint, using Eqs. 4.55 and 4.72, the adjoint diffusion fluxes $\Phi_{\mathrm{diff}}^{\gamma_a,PQ}$, $\Phi_{\mathrm{diff}}^{\gamma_a,f}$, $\Phi_{\mathrm{diff}}^{\tilde{Re}_a,PQ}$, $\Phi_{\mathrm{diff}}^{\tilde{Re}_a,f}$ are discretized as

$$
\begin{aligned}
\Phi_{\mathrm{diff}}^{\gamma_a,PQ} &= \left[\frac{1}{\mathrm{Re}_0}\left(\mu+\frac{\mu_t}{\sigma_f}\right)\mathsf{n}_k\frac{\partial\gamma_a}{\partial x_k}\right]_{PQ}^{\mathrm{adj}}, \qquad \Phi_{\mathrm{diff}}^{\gamma_a,f} = \frac{\partial}{\partial x_k}\left[\frac{1}{\mathrm{Re}_0}\left(\mu+\frac{\mu_t}{\sigma_f}\right)\mathsf{n}_k\frac{\partial\gamma_a}{\partial x_k}\right]^{f,\mathrm{adj}}\\
\Phi_{\mathrm{diff}}^{\tilde{Re}_a,PQ} &= \left[\frac{1}{\mathrm{Re}_0}\sigma_{\theta,t}\left(\mu+\mu_t\right)\mathsf{n}_k\frac{\partial\tilde{Re}_a}{\partial x_k}\right]_{PQ}^{\mathrm{adj}}, \quad \Phi_{\mathrm{diff}}^{\tilde{Re}_a,f} = \frac{\partial}{\partial x_k}\left[\frac{1}{\mathrm{Re}_0}\sigma_{\theta,t}\left(\mu+\mu_t\right)\mathsf{n}_k\frac{\partial\tilde{Re}_a}{\partial x_k}\right]^{f,\mathrm{adj}}
\end{aligned}
$$

Following the discretization of the diffusion of the adjoint mean flow equations and of the adjoint Spalart–Allmaras equation, the use of Eqs. 4.72 and 4.56 is proposed by the *TDDC* adjoint for the computation of the spatial gradients; in contrast, a "standard" discretization scheme would make use of Eqs. 2.58 and 2.50.

Sources The source terms $\mathcal{H}_{\gamma}^{\mathrm{SA},src}$, $\mathcal{H}_{\tilde{Re}_{\theta t}}^{\mathrm{SA},src}$ $\mathcal{H}_{\gamma}^{\gamma-\tilde{Re}_{\theta t},src}$ and $\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},src}$ of Eqs. 3.67c and 3.67d at node P are assumed to remain constant within Ω_P, yielding

$$
\int_{\Omega_P} S_{\mathrm{src}}^{\gamma-\tilde{Re}_{\theta t},\mathrm{adj}}\mathrm{d}\Omega \simeq S_{\mathrm{src}}^{\gamma-\tilde{Re}_{\theta t},\mathrm{adj},P}\Omega_P \tag{4.114}
$$

4.5 Verification Cases

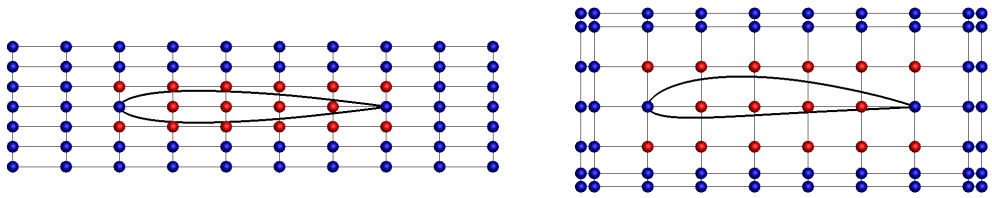
A series of 2D and 3D verification cases is presented to demonstrate the accuracy of SDs computed by the *TDDC* adjoint. The *TDDC* adjoint is compared with FDs and with the "standard" continuous adjoint (e.g. the continuous adjoint discretized by schemes widely used in the literature [53, 106]. The implementation of the *TDDC* adjoint in shape optimization is shown too. The test cases include inviscid and turbulent flows in internal and external aerodynamics. All optimization runs are performed using the sequential least squares programming algorithm (SLSQP), [105].

4.5.1 Shape Optimization of Isolated Airfoils

The first applications are dealing with the shape optimization of airfoils at inviscid flows; the NACA0012 and NACA4415 airfoils are the two background geometries. The flow conditions are summarized in Table 4.1; at these conditions, the flow around the baseline airfoils is transonic and the shock waves contribute to the computed drag. The accuracy of the SDs computed using the "standard" and the *TDDC* adjoint on grids of different qualities (from coarse to fine) is investigated. The impact of the SD accuracy on the convergence of the optimization algorithm is also studied. The airfoils are parameterized using the 10×7 volumetric NURBS lattices of Fig. 4.1.

Quantity	Symbol	NACA0012	NACA4415
Mach Number	M_∞	0.75	0.70
Anlge of Attack $(^\circ)$	AoA	2.0	2.0

Table 4.1: *Optimization of Isolated Airfoils*: Freestream flow Conditions.

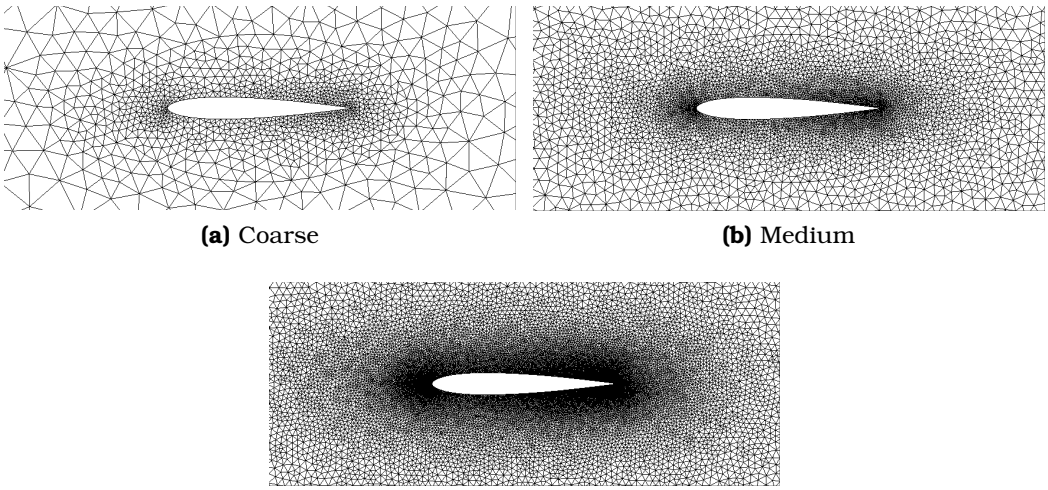


(a) NACA0012 **(b)** NACA4415

Figure 4.1: *Optimization of Isolated Airfoils*: Parameterization of the (a) NACA0012 and (b) NACA4415 isolated airfoils. Control points in blue are fixed while red ones are allowed to move in the normal-to-the-chord (vertical) directions, resulting in 16 design variables in total.

4.5.1.1 Optimization of the NACA0012 Isolated Airfoil

The shape optimization of the isolated NACA0012 airfoil for min. drag coefficient (C_D), with the constraint that the lift coefficient (C_L) remains close to the reference value ($\pm 1‰$) is carried out. An additional inequality constraint, requiring the airfoil area not to drop below 85% of the baseline one, is imposed.

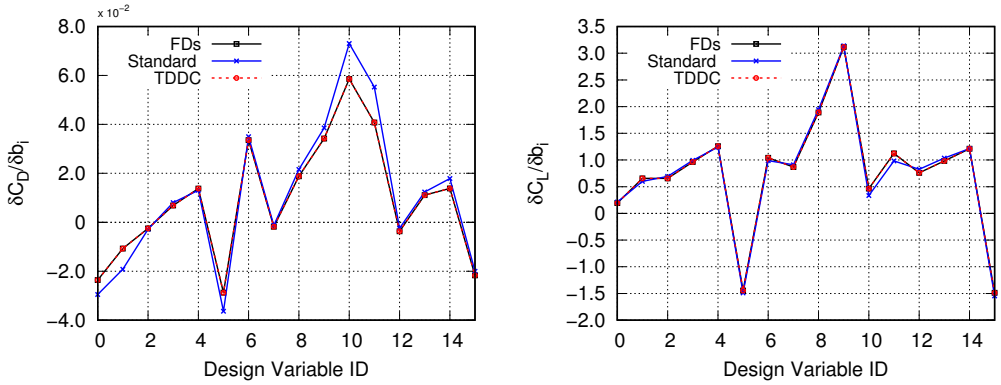


(a) Coarse **(b)** Medium

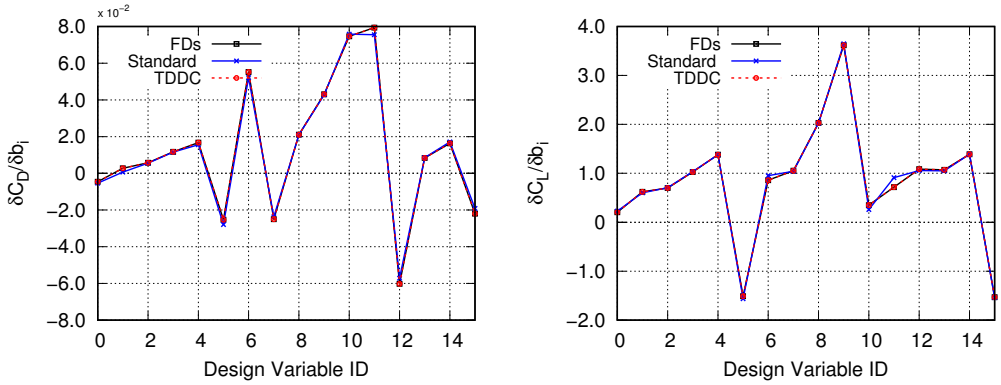
(c) Fine

Figure 4.2: *Optimization of the NACA0012 Airfoil*: Unstructured grids of different fineness. (a) Coarse ~ 1500, (b) Medium ~ 7800, (c) Fine ~ 26400 nodes.

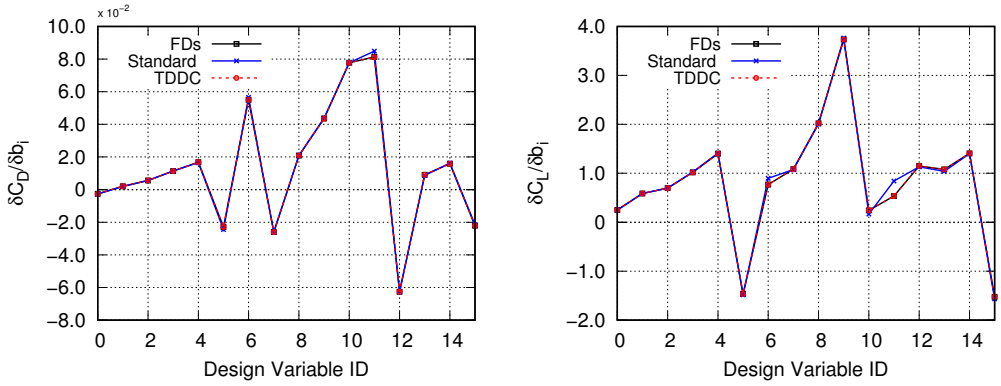
A parametric study on the accuracy of the SDs as computed by "standard" and *TDDC* adjoint with respect to the grid quality is performed; to do so, three grids of ~ 1500, ~ 7800 and ~ 26400 nodes are generated around the airfoil, Fig. 4.2, a flow analysis takes $\sim 6, 14$ and 24 s on a single NVIDIA V100 GPU, respectively. The computed SDs for C_D and C_L are illustrated in Fig. 4.3. In all grids, the *TDDC* adjoint perfectly matches the SDs computed by FDs (an accuracy of at least six significant digits is obtained), even on the coarse, non-adapted to the shock wave, grid. On the other hand, small differences (mainly on C_D) can be seen vis-à-vis to the "standard" continuous adjoint and, as expected, are reduced as grid becomes finer. Differences in the accuracy of the SDs affect, in fact, the optimization. On the coarse and medium grids, the *TDDC* adjoint results in an optimized solution of better quality at the same number of optimization cycles compared to the "standard" one (Figs. 4.4a $\sim 27.6\%$ instead of $\sim 27.3\%$, 4.4b $\sim 27\%$ instead of $\sim 21\%$ reduction), while a solution of same quality (i.e. $\sim 30\%$ reduction in the C_D value) is achieved in less cycles on the fine grid, Fig. 4.4c. The Mach number field around the baseline and optimized airfoils on the three grids (with the "standard" and the *TDDC* adjoint) is presented in Fig. 4.5.



(a) Coarse

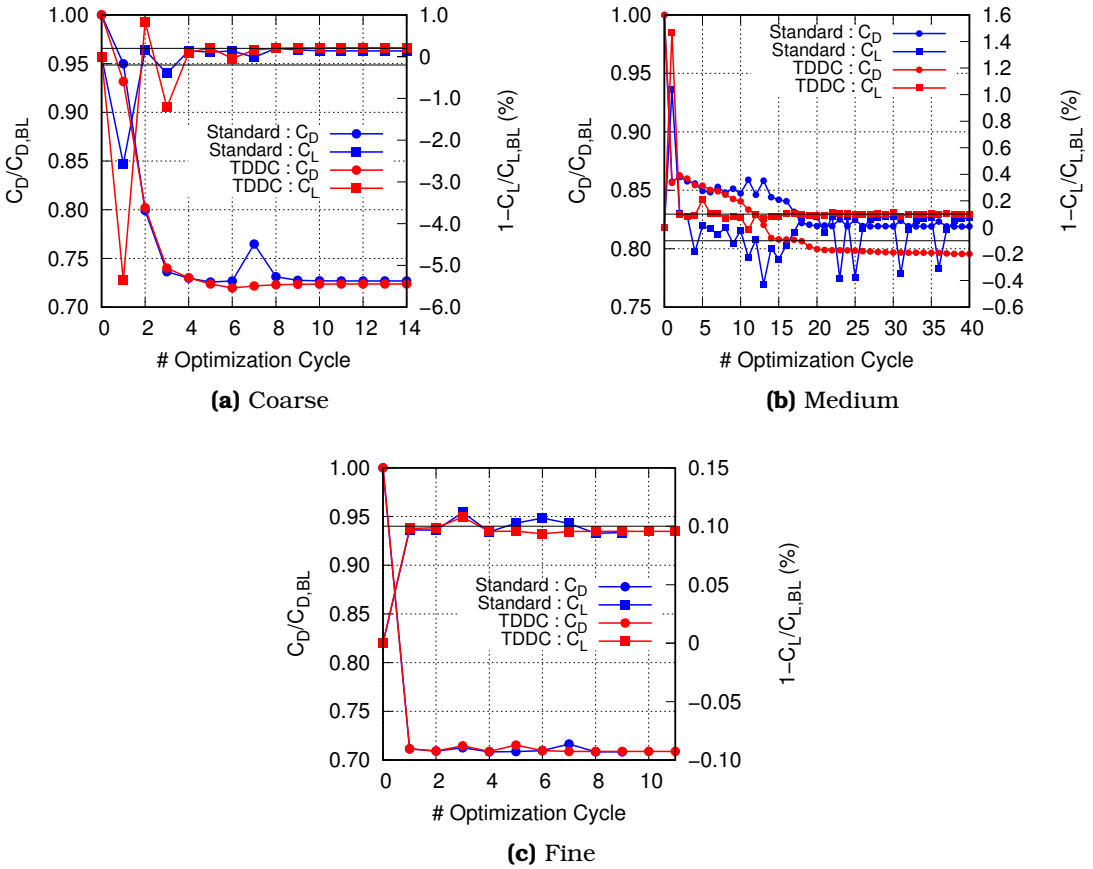


(b) Medium



(c) Fine

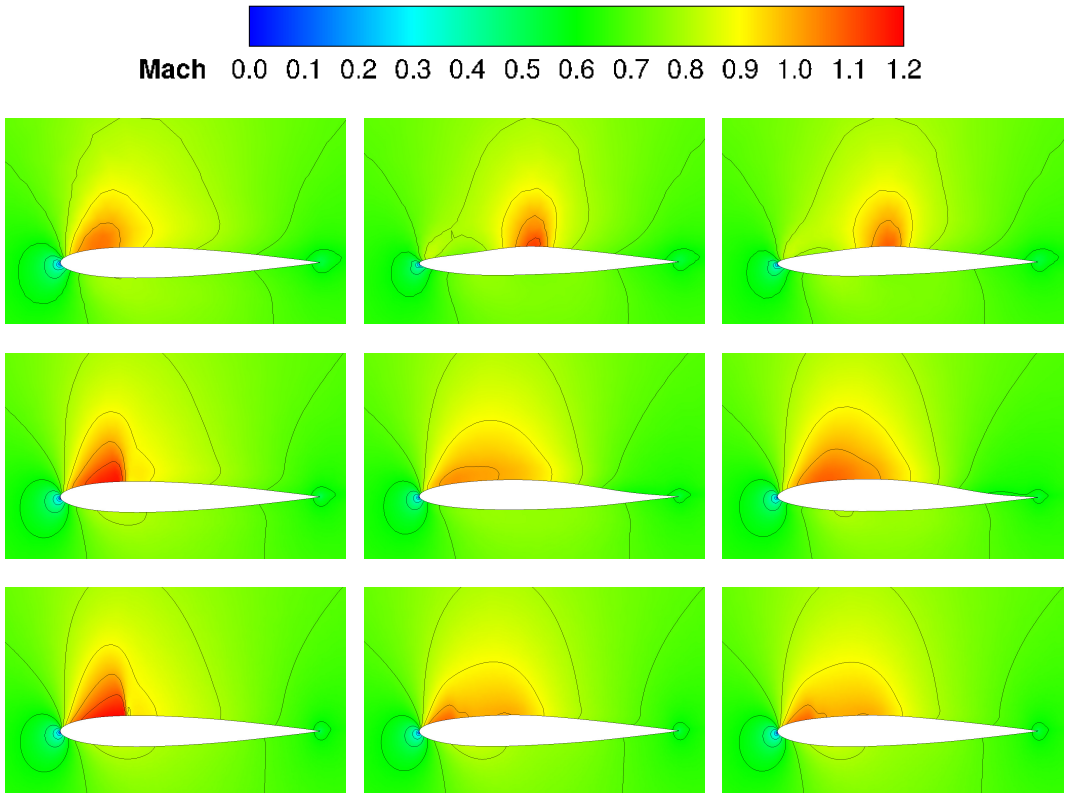
Figure 4.3: *Optimization of the NACA0012 Airfoil*: SDs of C_D (left) and C_L (right) computed on the (a) coarse, (b) medium and (c) fine grids of Fig. 4.2 using FDs (black), "standard" continuous (blue) and *TDDC* adjoint (red).



(a) Coarse **(b)** Medium

(c) Fine

Figure 4.4: *Optimization of the NACA0012 Airfoil*: Evolution of C_D (objective) and C_L (constraint) during the optimization for the (a) coarse, (b) medium and (c) fine grid based on "standard" continuous and *TDDC* adjoint. Each optimization cycle includes the solution of one primal and three adjoint (for C_D, C_L and area) problem solutions.



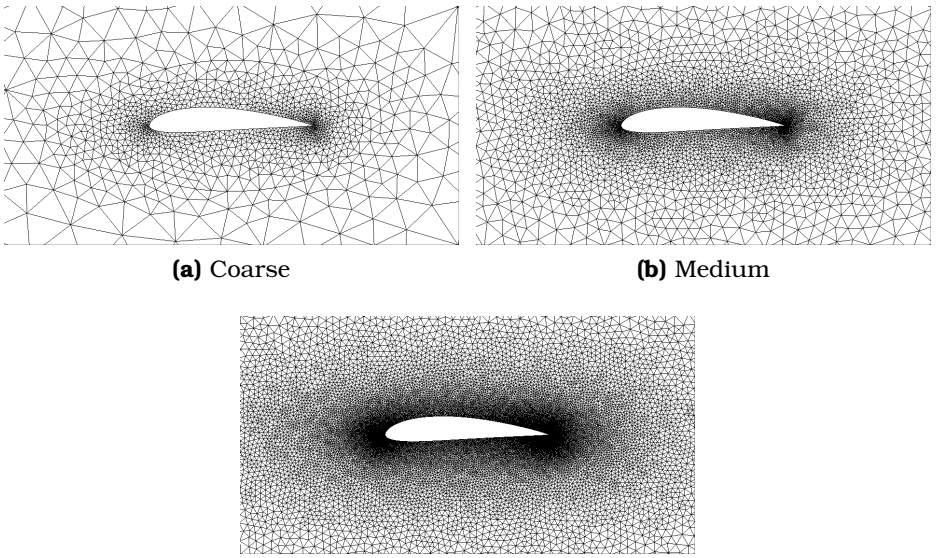
(a) Baseline **(b)** Stand Continuous **(c)** TDDC

Figure 4.5: *Optimization of the NACA0012 Airfoil*: Mach number iso-areas for the three grids of Fig. 4.2 for (a) the baseline and the optimized airfoils computed using (b) "standard" continuous and (c) *TDDC* adjoint. Computations on coarse to fine grids, from top to bottom.

4.5.1.2 Optimization of the NACA4415 Isolated Airfoil

The shape optimization of the NACA4415 airfoil is carried out for min. C_D with the constraint that C_L remains close to the baseline value (within $\pm 1‰$) and the airfoil area does not drop below $\sim 85\%$ of the initial one. Three grids, Fig. 4.6, are used to investigate the grid density effect on the computed SDs and a single flow analysis takes ~ 5, 15 and 24 s on a single NVIDIA V100 GPU, respectively.

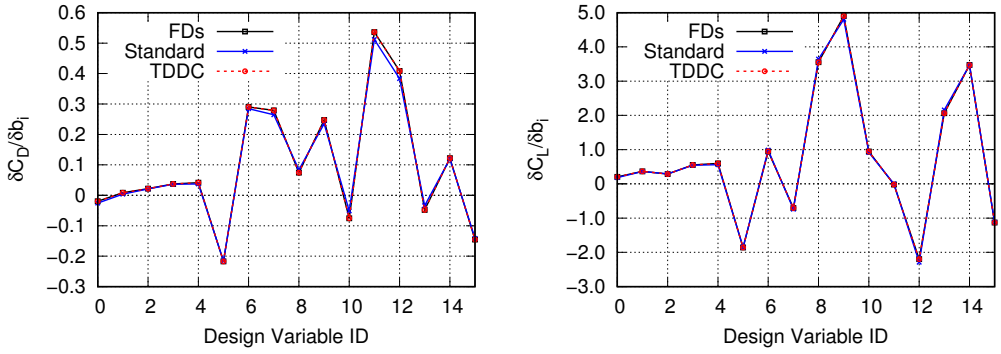
The SDs of C_D and C_L computed based on the "standard" and the *TDDC* adjoint are compared with FDs in Fig. 4.7, for the three grids. The SDs computed by the *TDDC* adjoint have, at least, a six significant digit accuracy, on all three grids; this verifies the accuracy of the proposed discretization schemes. On the other hand, small discrepancies (two significant digits) exist when the "standard" continuous adjoint is used; these are more intense for the C_D value and become less pronounced with the grid refinement. Fig. 4.8 presents the convergence history of the optimizations; a reduction between 50 and 80% is achieved while maintaining the C_L value close to the baseline one. The proposed discretization schemes are beneficial for the optimization: for the same number of optimization cycles, the *TDDC* adjoint results in a better quality solution than "standard" continuous adjoint (on the coarse grid); alternatively, it may give a solution of the same quality using less cycles (on the medium and fine grids). The Mach number fields around the baseline and optimized airfoils are presented in Fig. 4.9; the strong shock wave on the baseline airfoil weakens and thus the C_D is reduced.



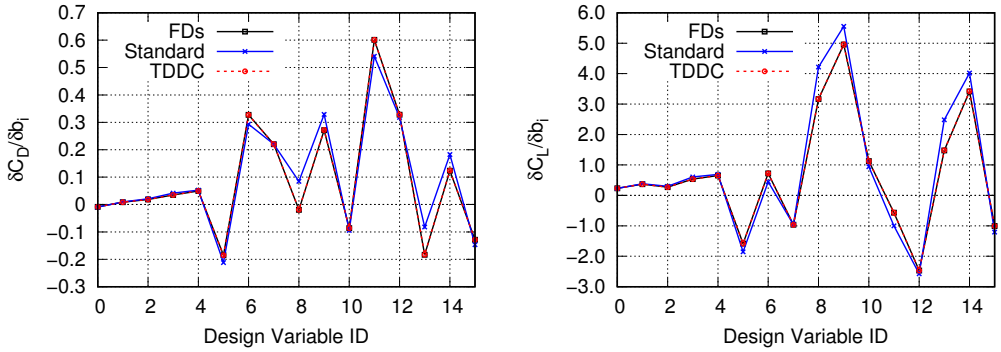
(a) Coarse **(b)** Medium

(c) Fine

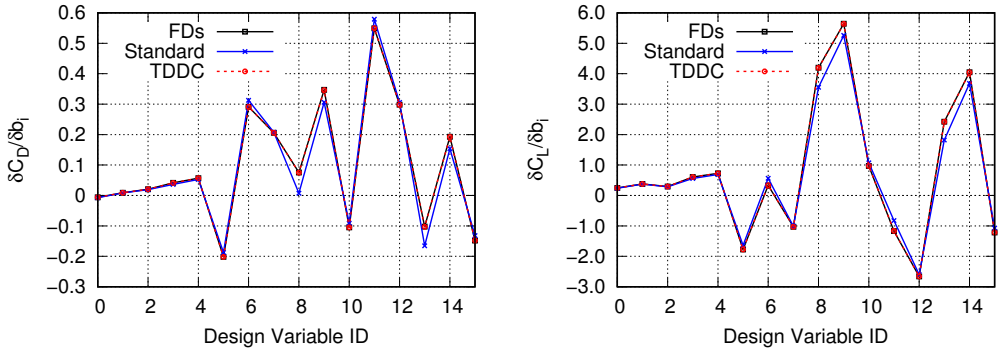
Figure 4.6: *Optimization of the NACA4415 Airfoil*: Unstructured grids of different fineness. (a) Coarse ~ 1500, (b) Medium ~ 6500, (c) Fine ~ 27000 nodes.



(a) Coarse

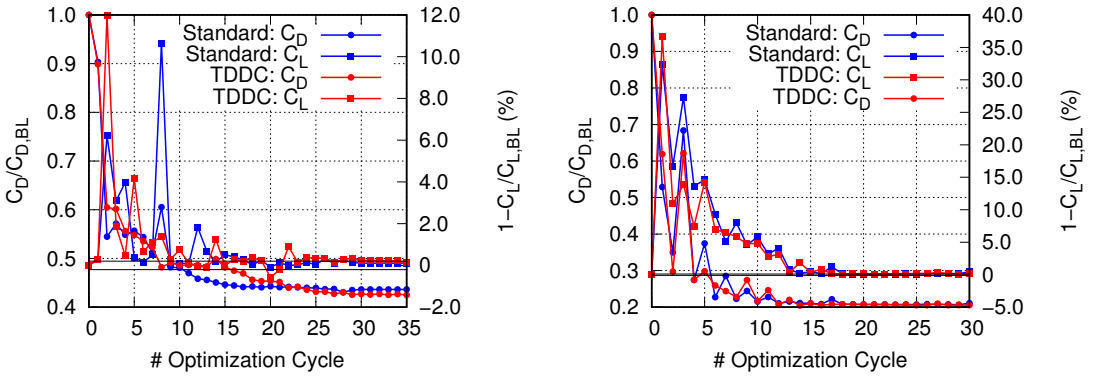


(b) Medium

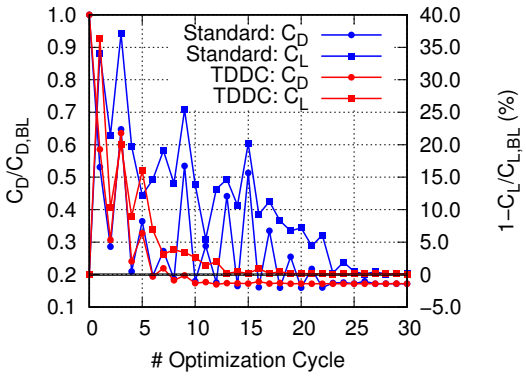


(c) Fine

Figure 4.7: *Optimization of the NACA4415 Airfoil*: SDs of C_D (left) and C_L (right) computed on the (a) coarse, (c) medium and (c) fine grid of Fig. 4.6 using FDs (black), "standard" continuous (blue) and *TDDC* adjoint (red).

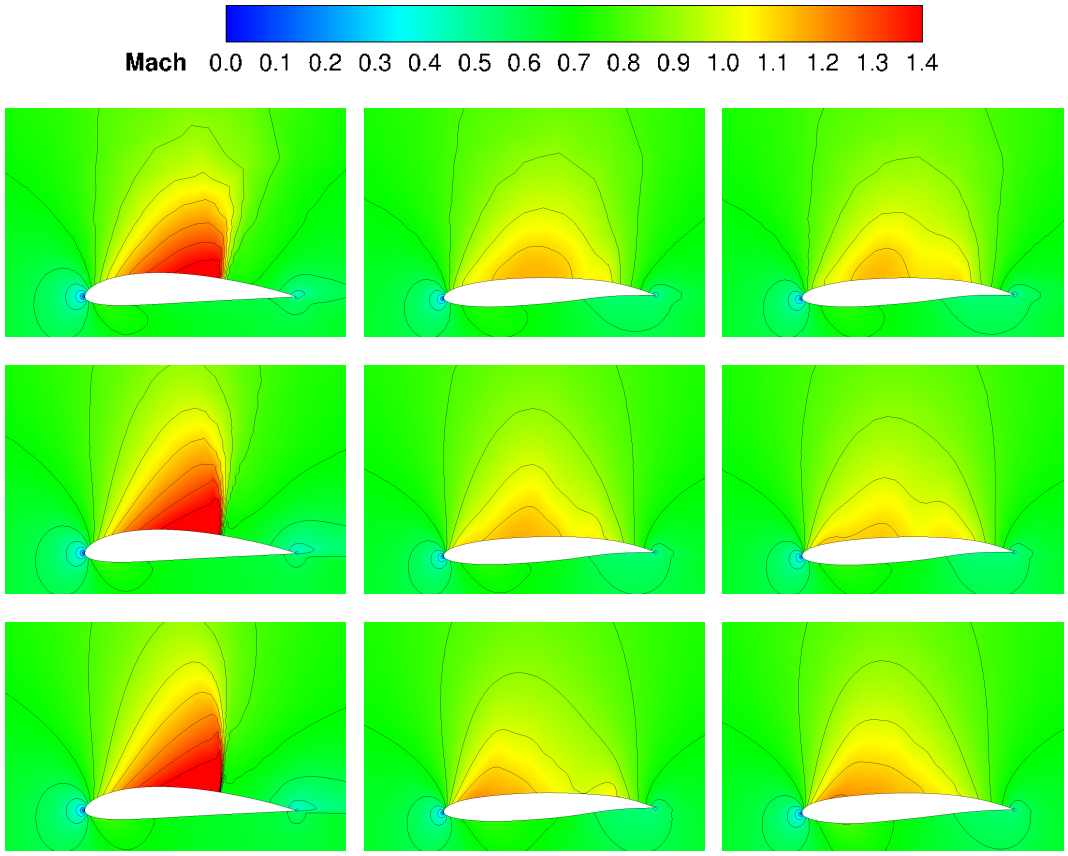


(a) Coarse **(b)** Medium



(c) Fine

Figure 4.8: *Optimization of the NACA4415 Airfoil*: Evolution of C_D (objective) and C_L (constraint) during the optimization for the (a) coarse, (b) medium and (c) fine grid based on the "standard" continuous and the *TDDC* adjoint. Each optimization cycle includes one primal and three adjoint (for C_D, C_L and area) problem solutions.



(a) Baseline **(b)** Stand Continuous **(c)** TDDC

Figure 4.9: *Optimization of the NACA4415 Airfoil*: Mach number iso-areas for the three grids of Fig. 4.6 for (a) the baseline and the optimized airfoils computed using (b) "standard" continuous and (c) *TDDC* adjoint. Results computed on the coarse to fine grid, from top to bottom.

4.5.2 Shape Optimization of a Transonic Wing

The shape optimization of a transonic wing is presented; the geometry of the ONERA M6 wing, [136], is used as the baseline one. The flow is inviscid with the freestream conditions of Table 4.2. A grid of ~ 70000 nodes is generated around the wing and a single flow simulation takes ~ 20 s on an NVIDIA V100 GPU.

Quantity	Symbol	Value
Freestream Mach Number	M_∞	0.8395
Angle of Attack (°)	AoA	3.06
Yaw Angle (°)	α_{yaw}	0.0

Table 4.2: *Optimization of a Transonic Wing*: Frestream flow conditions.

The wing is optimized for max. lift (L), with the constraint that drag (D) should not exceed that of the baseline wing. The wing shape is controlled by the $8\times7\times5$ volumetric NURBS control grid of Fig. 4.10; the 18 control points in red are allowed to move in the chordwise and the normal-to-the-planform directions resulting in 36 design variables in total. The accuracy of the SDs of the two quantities of interest computed by the "standard" and the *TDDC* adjoint is assessed against FDs, used as reference, in Fig. 4.11. The first 18 values correspond to the x and the second 18 to the y coordinates of the control points. The SDs of L are computed with high accuracy both with the "standard" and the *TDDC* adjoint. On the other hand, for D, the *TDDC* yields a precision of -at least- six significant digits accuracy in all SDs, while for the "standard" continuous adjoint the second significant digit may differ. In particular, for the design variables with ID 8 and 18, the SDs computed by the "standard" discretization scheme are even wrongly signed.

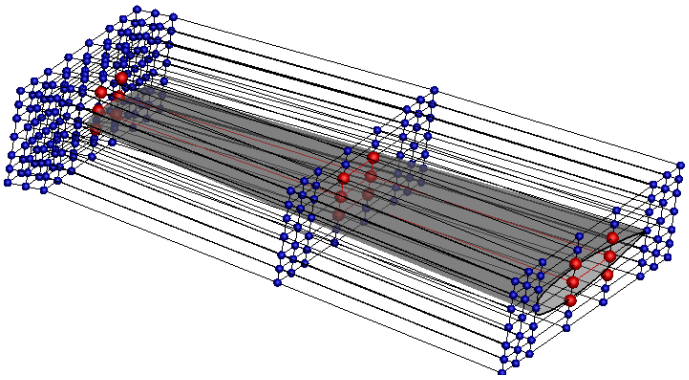


Figure 4.10: *Optimization of a Transonic Wing*: Parameterization. Control points in blue are fixed, red ones are allowed to move in the chordwise and the normal-to-the-planform directions resulting in 36 design variables, in total.

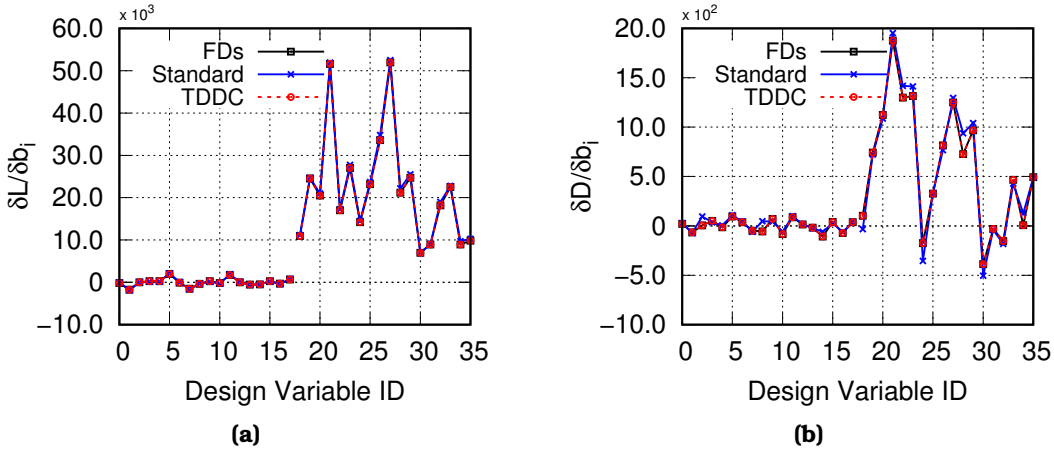


Figure 4.11: *Optimization of a Transonic Wing*: SDs of (a) lift (L) and (b) drag (D) computed by the "standard" continuous and *TDDC* adjoint as well as FDs.

After verifying the SDs accuracy, the shape optimization of the transonic wing follows; two optimization runs are performed based on the "standard" continuous and the *TDDC* adjoint. The optimization convergence histories are presented in Fig. 4.12, the two optimizations based on the "standard" continuous and the *TDDC* adjoint lead to $\sim 19.2\%$ and $\sim 21.5\%$ increase in the L value, respectively, at the same number of optimization cycles meeting the constraint on D. The Mach number fields for the baseline and optimized wings are presented in Fig. 4.13.

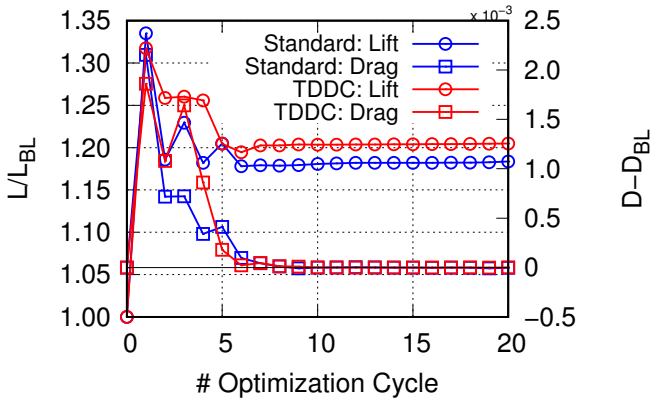
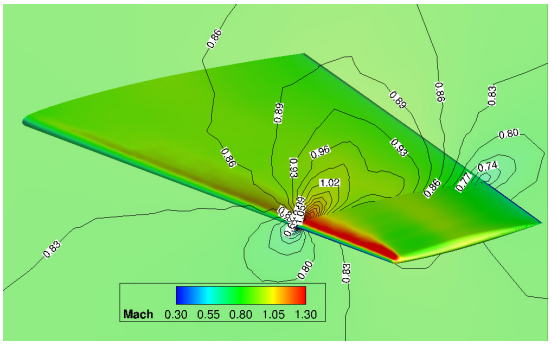
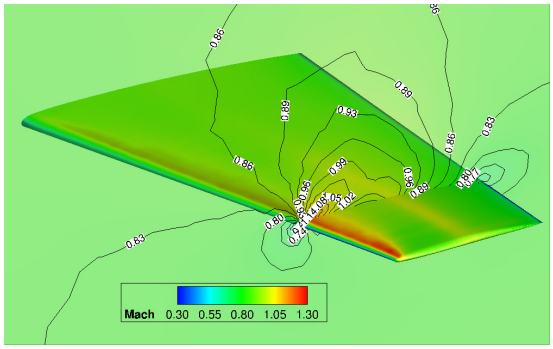


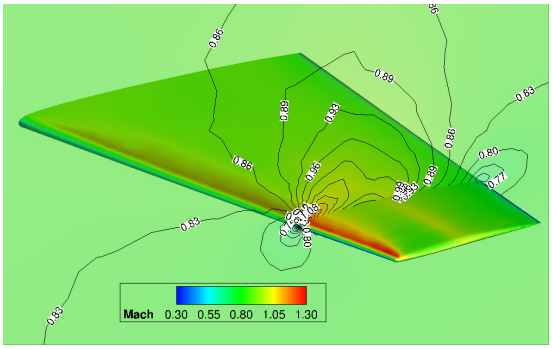
Figure 4.12: *Optimization of a Transonic Wing*: Convergence histories of the objective (L) and constraint (D) functions based on the "standard" continuous and *TDDC* adjoint. Each optimization cycle involves one primal and two adjoint (for L and D) problem solutions.



(a)



(b)



(c)

Figure 4.13: *Optimization of a Transonic Wing*: Mach number fields around the baseline (a) and the optimized wings computed using the "standard" continuous (b) and the *TDDC* adjoint (c). Spanwise cuts at 75% of the wing span.

4.5.3 Shape Optimization of a High-Pressure Turbine Cascade

This application is dealing with the shape optimization of a 2D high-pressure turbine cascade, for max. inlet capacity (F_1) constrained by the flow turning (F_2) which is allowed to change by no more than $\pm 1‰$ from the baseline value. The two quantities of interest are expressed as

$$F_1 = \dot{m}^{\text{in}} \frac{\sqrt{\tilde{T}_t^{\text{in}}}}{\tilde{p}_t^{\text{in}}}, \quad F_2 = \tan^{-1}\left(\frac{\tilde{v_2}^{\text{out}}}{\tilde{v_1}^{\text{out}}}\right) \tag{4.115}$$

where $\dot{m}^{\text{in}}$, $\tilde{T}_t^{\text{in}}$, $\tilde{p}_t^{\text{in}}$ are the massflow and the mass-averaged total temperature and total pressure at the inlet, respectively, and $\tilde{v_1}^{\text{out}}$, $\tilde{v_2}^{\text{out}}$ are the velocity components at the exit. The geometry of [52] is used as the baseline one. The flow conditions are summarized in Table 4.3; turbulence is modeled by means of the Spalart–Allmaras model without a transition model. An unstructured grid of ~ 95000 nodes is generated and used and a single flow simulation takes ~ 5 min on an NVIDIA V100 GPU. The case is parameterized using the 7×3 volumetric NURBS lattice of Fig. 4.14.

Quantity	Symbol	Value
Inlet Total Pressure (Pa)	p_t^{in}	243700.0
Inlet Total Temperature (K)	T_t^{in}	800.0
Outlet Static Pressure (Pa)	p^{out}	142530.0
Inlet Flow Angle (°)	α^{in}	0.0
Dynamic Viscosity $(10^{-5}\,\text{kg}\,\text{m}^{-1}\,\text{s}^{-1})$	μ	1.7831
Inlet Viscosity Ratio	$(\mu_t/\mu)^{\text{in}}$	30.0

Table 4.3: *Optimization of a Turbine Cascade*: Flow conditions.

The SDs of the objective (F_1) and constraint (F_2) functions are computed based on the "standard" continuous and the *TDDC* adjoint and are compared with FDs in Fig. 4.15. The first 19 values correspond to the x and the next 19 to the y coordinates of the control points. The SDs are computed with high accuracy with both adjoint variants; practically, no difference from FDs can be seen with the naked eye. Nevertheless, the *TDDC* adjoint yields an accuracy of at least six significant digits, which decreases to three with "standard" continuous adjoint. The shape optimization of the turbine cascade follows. The optimization convergence histories based on the "standard" continuous and the *TDDC* adjoint are presented in Fig. 4.16; the inlet capacity is increased by 0.979% and 1.197%, respectively, satisfying the flow turning constraint. It should be mentioned that, since the inlet total pressure and total temperature are fixed, the increase in the inlet capacity corresponds to an increase in the massflow. So, for the selected, with limited

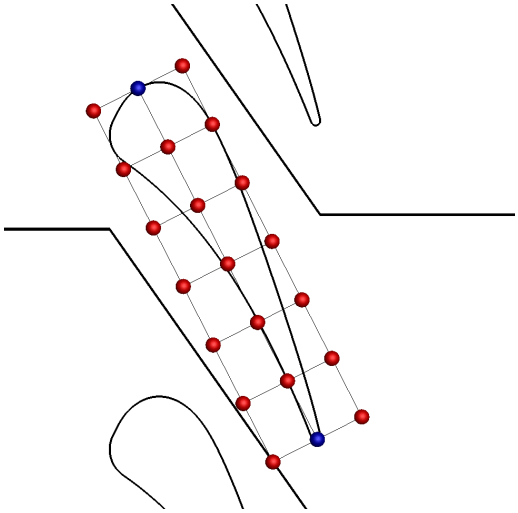


Figure 4.14: *Optimization of a Turbine Cascade*: Parameterization. Blue control points are kept fixed, whereas red ones are allowed to move in the axial and pitchwise directions.

room for improvement, objective function, this small gain obtained from the *TDDC* adjoint is appreciated.

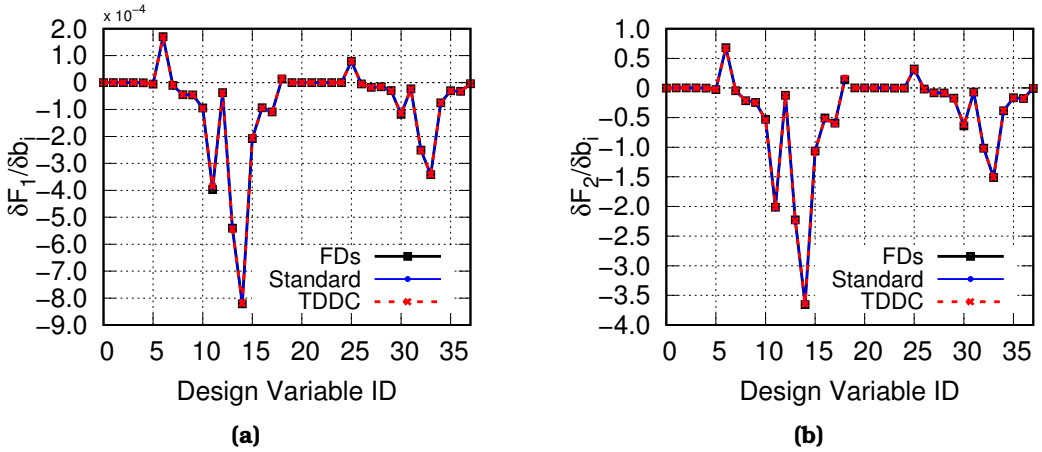


Figure 4.15: *Optimization of a Turbine Cascade*: SDs of (a) inlet capacity (F_1) and (b) flow turning (F_2) computed by the "standard" continuous and *TDDC* adjoint as well as FDs.

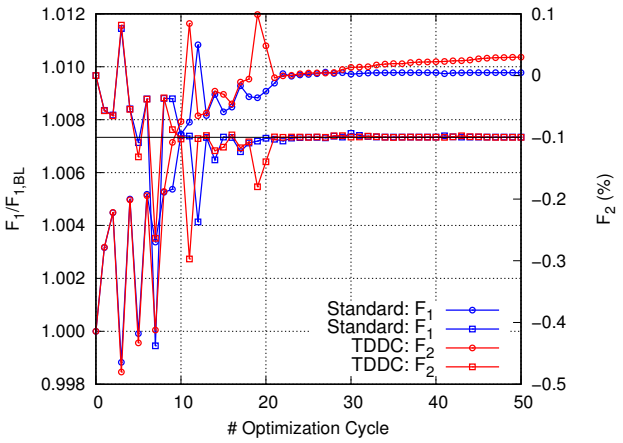


Figure 4.16: *Optimization of a Turbine Cascade*: Evolution of the objective (inlet capacity F_1) and constraint (flow turning F_2) functions based on the "standard" continuous and *TDDC* adjoint. In each optimization cycle, one primal and two adjoint problems (for F_1 and F_2) are numerically solved.

4.6 Conclusions

This chapter presented the development and assessment of new consistent discretization schemes for the continuous adjoint PDEs and their boundary conditions, with a clear physical insight and low memory requirements of the adjoint code. The discretization schemes were inspired by the hand–differentiated discrete adjoint equations which were firstly derived. The proposed *Think-Discrete Do-Continuous* or *TDDC* adjoint was developed for the time–marching hyperbolic–type solver for compressible fluid flows of PUMA. Based on the *TDDC* adjoint, consistent discretization schemes for the mean flow field adjoint equations, the adjoint Spalart–Allmaras, the adjoint γ–$\tilde{R}e_{\theta t}$ equations and their boundary conditions, as obtained by the continuous adjoint method in Chapter 3, were proposed.

The consistency and accuracy of the SDs with the *TDDC* adjoint, and its impact on the optimization convergence, were demonstrated in a series of verification cases; comparison with results from simulations with "standard" discretization schemes, widely used in the literature, were also made. The studies concerned the shape optimization of two isolated airfoils and a transonic wing (inviscid flow) and that of a high–pressure turbine cascade (turbulent flow). They revealed the high accuracy of the *TDDC* at the computation of the SDs and, consequently, the improved optimization behavior (in terms of convergence and quality of the optimized solution) compared to the "standard" continuous adjoint. Overall, the *TDDC* adjoint proved to be a nice way to bridge the gap between the two adjoint variants (discrete and continuous) as ensured high accuracy in the computed SDs (as in discrete adjoint) with clear physical insight into the adjoint problem and minimal memory footprint (as in continuous adjoint).

Chapter 5

Shape Optimization using Continuous Adjoint for Transitional Flows

In this Chapter, the continuous adjoint method for transitional flows (developed in Chapter 3) that applies the consistent *TDDC* discretization schemes proposed in Chapter 4, is used for the shape optimization of two real-world problems.

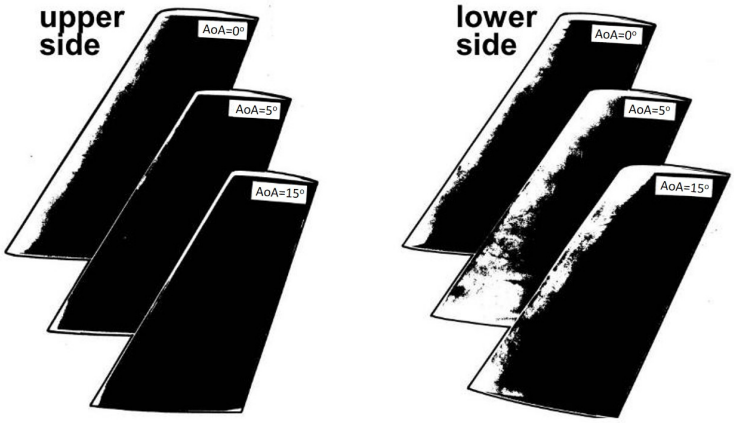
5.1 Shape Optimization of a Wing

This application is dealing with the shape optimization of the ONERA M6 wing in transitional flow; the geometry of [136] is used as the baseline one. The optimization aims at reducing the drag coefficient (C_D) while maintaining the lift coefficient (C_L) close to its baseline value; for the latter, a margin of $\pm 1‰$ is set. This is the same geometry used in Sec. 4.5.2 at different, though, flow conditions; there the flow was considered to be inviscid, while, herein, is transitional. Here, the simulations are performed based on experiments carried out in [121]; the freestream flow conditions for three Angles of Attack (AoA) are summarized in Table 5.1. A hybrid unstructured grid of $\sim 1.3M$ nodes is generated around the wing and the max. y^+ value of the first nodes off the wall is $y^+ \simeq 0.3$, in all performed simulations. This grid differs from the one used in Sec. 4.5.2 as, in order to capture the transitional effects, a finer grid is required. The SA-sLM2015 transition model is used and the flow analysis for each AoA takes ~ 30 min on an NVIDIA A100 GPU.

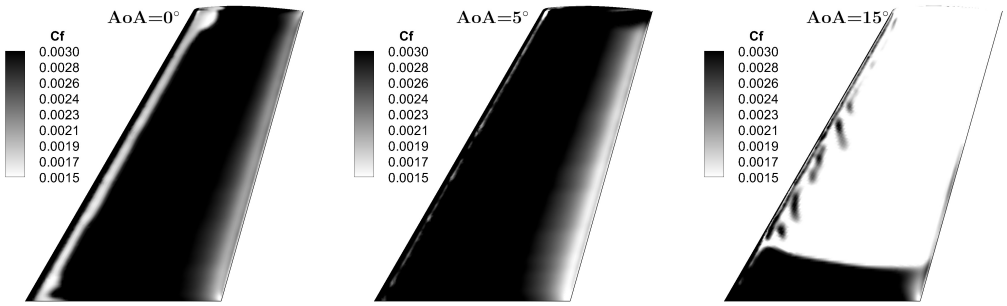
The transition location over the pressure and suction sides of the wing as visualized using sublimation of naphthalene in [121] and the skin friction coefficient (C_f) field resulted from the simulations based on the SA-sLM2015 transition model are presented in Fig. 5.1 for the three AoA. For AoA$=5^\circ$ and 15°, the laminar area is overestimated compared to the experiments, especially on the pressure side. Similar numerical results were also obtained for the same AoA in [29] with the SST-2003-LM2015 transition model. For AoA$=0^\circ$, the numerical results are in good agreement with the experimental data regarding the transition location.

Quantity	Symbol	Value
Freestream Mach Number	M_∞	0.262
Reynolds Number ($\cdot 10^6$)	Re	3.5
Angle of Attack (°)	AoA	$0.0, 5.0, 15.0$
Yaw Angle (°)	α_{yaw}	0.0
Turbulence Intensity (%)	Tu	0.2
Surface Roughness (μm)	h_{rms}	5.0

Table 5.1: *Optimization of a Wing*: Flow conditions.



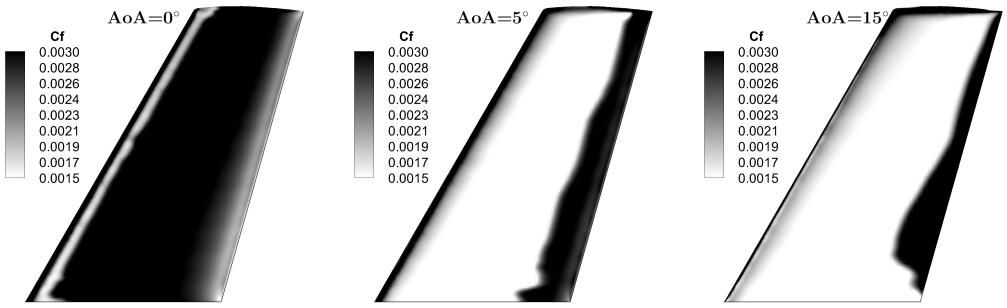
(a) Experimental Data



(b) Numerical Simulations - Suction (upper) Side

Figure 5.1: *Optimization of a Wing*: (a) Experimental transition locations using sublimation of naphthalene, [121], and skin friction coefficient (C_f) fields on the (b) suction and (c) pressure sides of the wing, for the three AoA.

The optimization is performed for AoA $= 5°$. The wing is parameterized using the $9\times7\times7$ NURBS lattice of Fig. 5.2; control points in blue are kept fixed whereas the red ones are allowed to move in the z direction, resulting in 45 design variables in total. The convergence histories of the objective and constraint functions are presented in Fig. 5.3; the optimization reduced the C_D by $\sim 3.3\%$, meeting



(c) Numerical Simulations - Pressure (lower) Side

Figure 5.1: Concluded

the imposed constraint. The C_f fields on the pressure and suction sides of the optimized wing and the C_f distributions at different spanwise cuts (0%, 25%, 50%, 75%, 87.5% and 100%) for the baseline and the optimized wings are presented in Figs. 5.4 and 5.5, respectively. It can be seen that the transition location line is moved upstream for the cuts that are close to the root (Figs. 5.5a and 5.5b) and downstream for those close to the tip (Figs. 5.5c, 5.5d, 5.5e and 5.5f); the displaced transition line, overall, leads to C_D reduction.

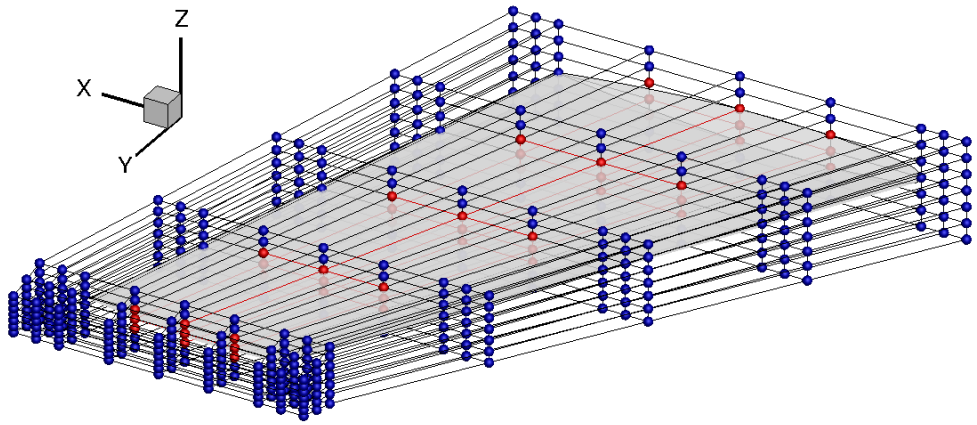


Figure 5.2: *Optimization of a Wing*: NURBS control lattice around the wing. The control points in blue remain constant while the red ones are allowed to be displaced in the z direction.

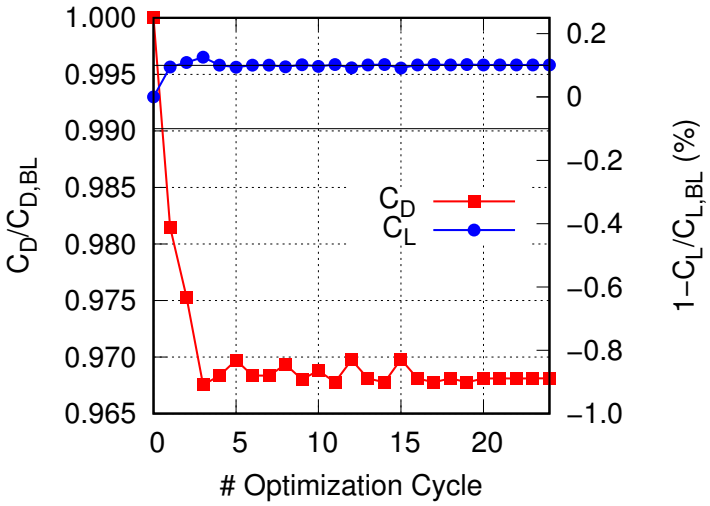
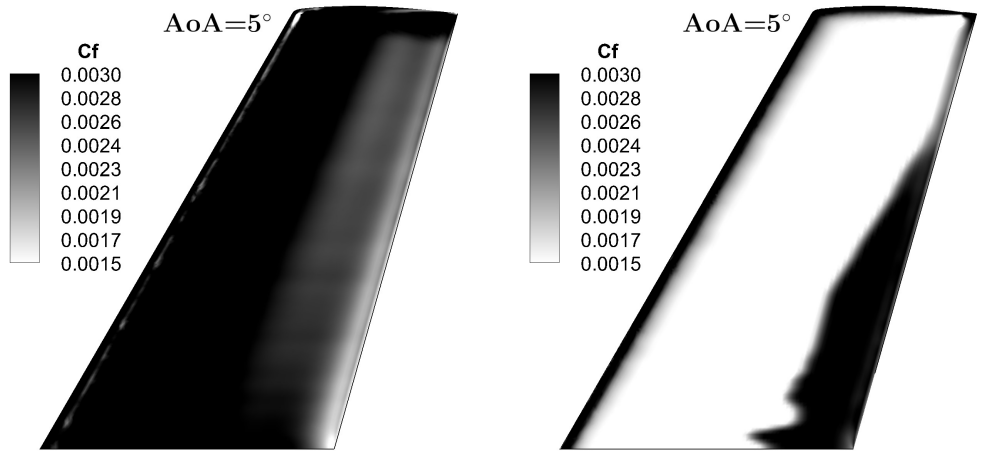
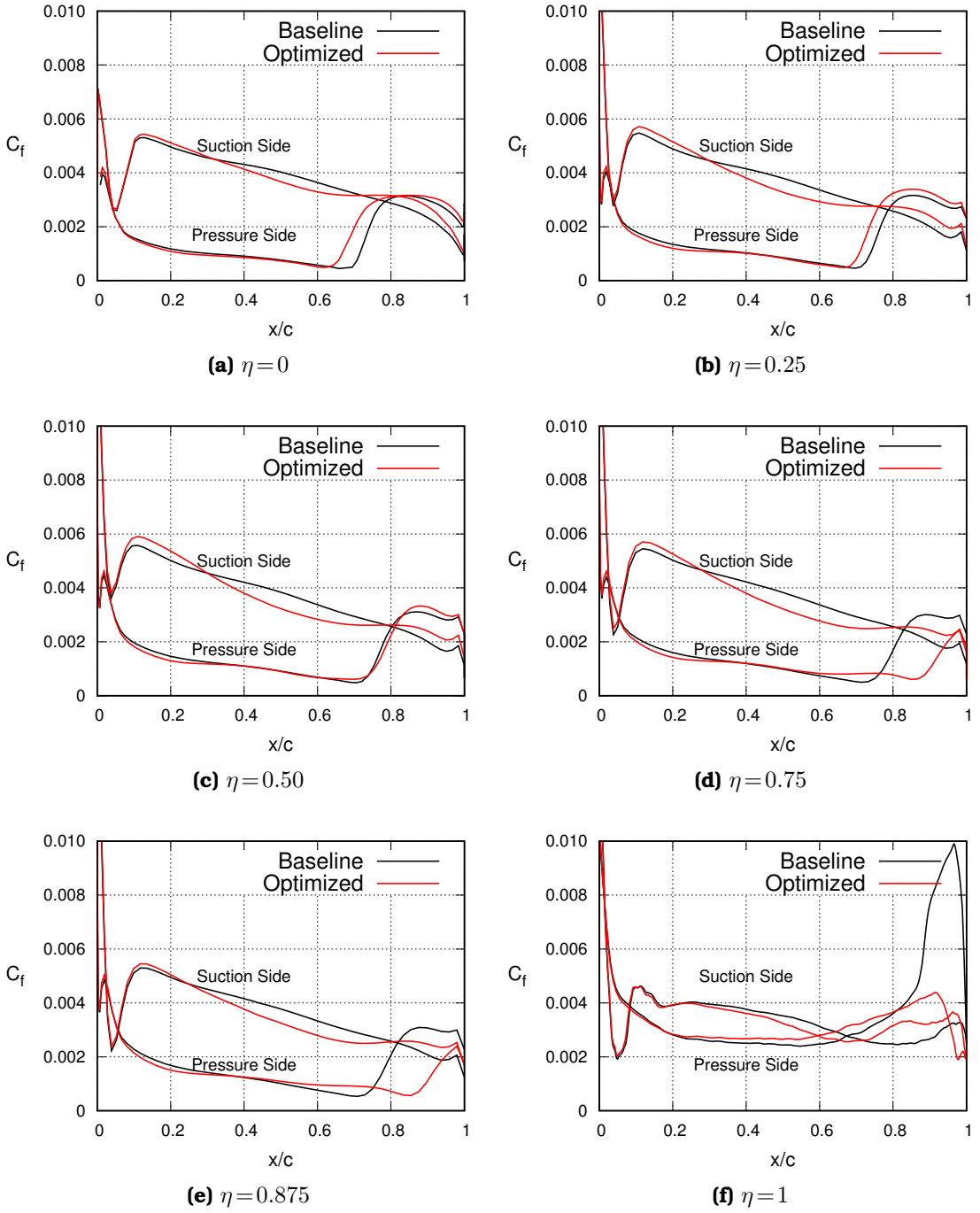


Figure 5.3: *Optimization of a Wing*: Convergence histories of the objective (C_D) and constraint (C_L) functions based on the SA-sLM2015 transition model and AoA=5°.



(a) Suction Side **(b)** Pressure Side

Figure 5.4: *Optimization of a Wing*: Computed skin friction coefficient (C_f) field on the (a) suction and (b) pressure sides of the optimized wing for AoA=5° (to be compared with Figs. 5.1b and 5.1c center).



(a) $\eta = 0$ **(b)** $\eta = 0.25$

(c) $\eta = 0.50$ **(d)** $\eta = 0.75$

(e) $\eta = 0.875$ **(f)** $\eta = 1$

Figure 5.5: *Optimization of a Wing*: Skin friction coefficient (C_f) distribution over the pressure and suction sides of the wing for the baseline and optimized geometries at spanwise cuts (a) $\eta = 0$, (b) $\eta = 0.25$, (c) $\eta = 0.50$, (d) $\eta = 0.75$, (e) $\eta = 0.875$ and (f) $\eta = 1$, for AoA$= 5^\circ$, with the SA-sLM2015 transition model.

5.2 Shape Optimization of a HARW Business Jet

This application is concerned with the shape optimization of a High Aspect-Ratio Wing (HARW) business jet model. The geometry of the aircraft (wing-body) and the boundary conditions were provided by Dassault Aviation in the framework of the NEXTAIR project (multi-disciplinary digital-enablers for NEXT-generation AIRcraft design and operations) funded by the European Union under Grant Agreement No. 101056732. The HARW model is expected to extend the natural laminar flow over the wing reducing fuel consumption and, thus, emissions. The optimization aims at minimizing the drag coefficient (C_D), constrained by the lift coefficient (C_L) that should be within the range 0.5 ± 10^{-4}. Herein, C_L refers to the full aircraft model and is computed as a linear combination of the lift ($C_{L,WB}$) and moment ($C_{m,WB}$) coefficients of the wing-body as follows

$$C_L = C_{L,WB} + \lambda C_{m,WB} \tag{5.1}$$

where λ is an empirical coefficient defined by the aforementioned industry.

5.2.1 Flow Analysis of the HARW Business Jet

Prior to the optimization, the flow analysis of the HARW business jet is performed. The aircraft is studied for three different Mach numbers, $M_\infty = 0.76$, 0.78 and 0.80 and four Angles of Attack AoA $\in [0°, 3°]$; the flow conditions (12 in total) are summarized in Table 5.2. A hybrid unstructured CFD grid of $\sim 12.4M$ nodes is generated around the aircraft with $y^+ < 0.7$ for the first nodes off the wall, Fig. 5.6. The flow is transitional and the SA-sLM2015 transition model is used. Computations are carried out for the half of the aircraft, using symmetry conditions. A single CFD run takes $\sim$2.5 h on an NVIDIA A100 GPU.

Quantity	Symbol	Value
Freestream Mach Number	M_∞	0.76, 0.78, 0.80
Reynolds Number ($\cdot 10^6$)	Re	4.24
Angle of Attack (°)	AoA	0.0, 1.0, 2.0, 3.0
Yaw Angle (°)	α_{yaw}	0.0
Turbulence Intensity (%)	Tu	0.1
Surface Roughness (µm)	h_{rms}	0.01

Table 5.2: *HARW Business Jet*: Flow Conditions.

The polar diagram of C_L vs. C_D for AoA$= [0°, 1°, 2°, 3°]$ is presented in Fig. 5.7, for the three Mach numbers and the SA-sLM2015 transition model; results from

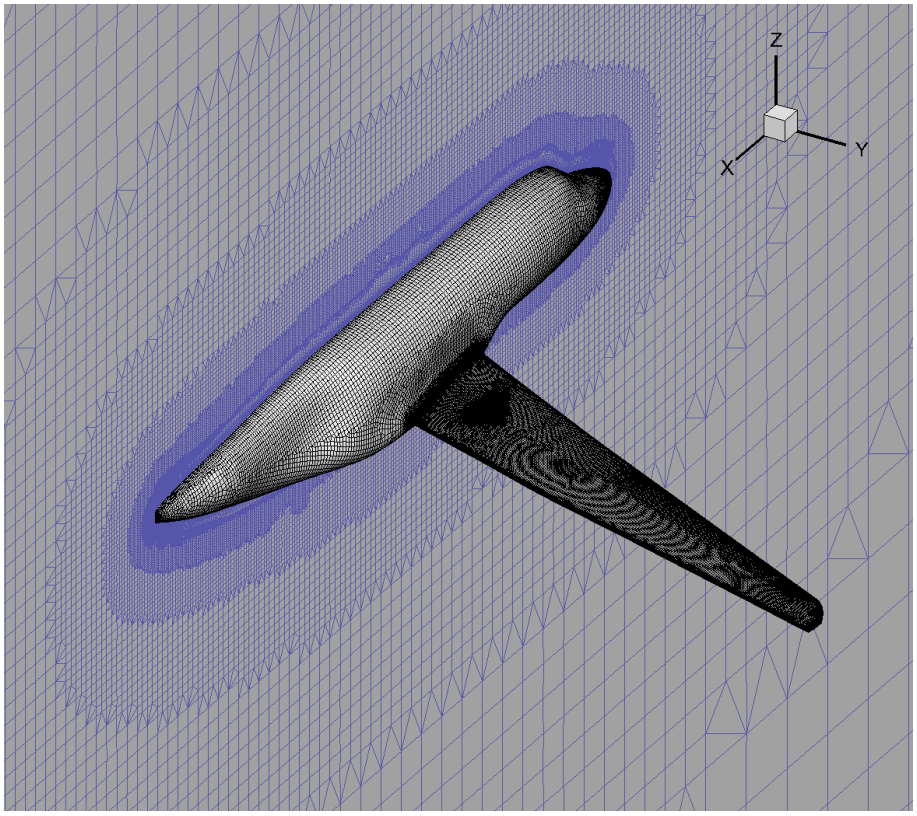
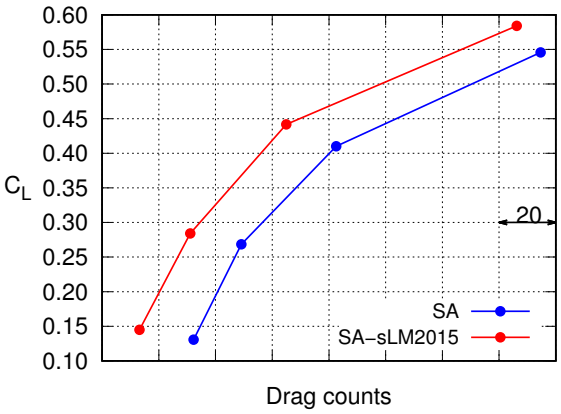
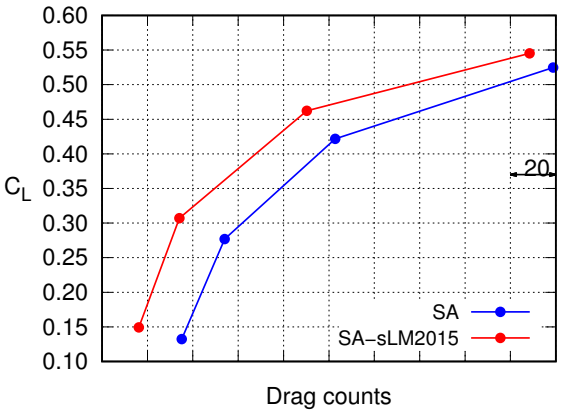


Figure 5.6: *HARW Business Jet*: Hybrid unstructured grid on the wing and fuselage surface.

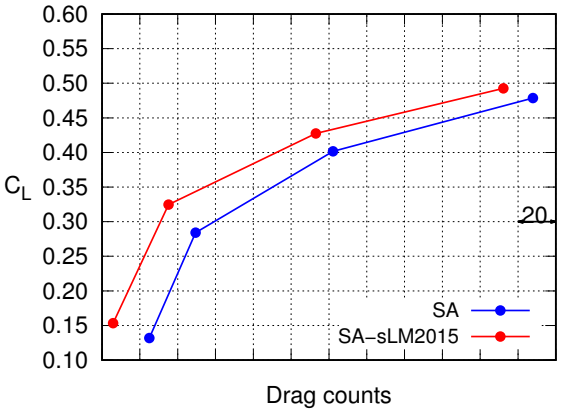
fully turbulent simulations with the Spalart–Allmaras model (without transition effect) are also included. For all Mach numbers, the use of the transition model shifts the polar diagram to smaller C_D values. The pressure (C_p) and skin friction (C_f) distributions on the wing-body surface are presented in Fig. 5.8 for the upper and lower sides of the aircraft, for $M_\infty = 0.78$ and AoA $= 2°$. The top-half aircraft corresponds to a simulation performed with the Spalart–Allmaras turbulence model without a transition model, while the bottom-half to a simulation performed with the SA-sLM2015 transition model. The C_p fields are similar for both simulations, Fig. 5.8a. The significance of using the transition model is verified by examining the C_f field, Fig. 5.8b, where only the use of transition modeling (SA-sLM2015) can capture the laminar area. In fact, this area is extended compared to that predicted with the Spalart–Allmaras model (without the transition model) and should be taken into account during the shape optimization of the wing. The wing laminar area can also be seen in the γ plot, Fig. 5.9.



(a) $M_\infty = 0.76$

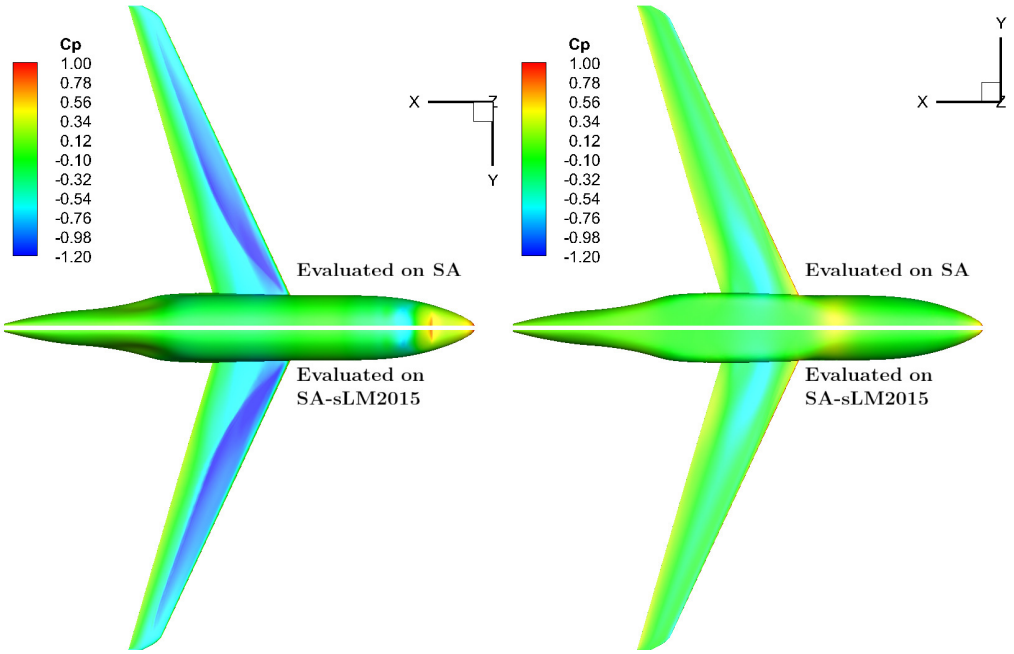


(b) $M_\infty = 0.78$

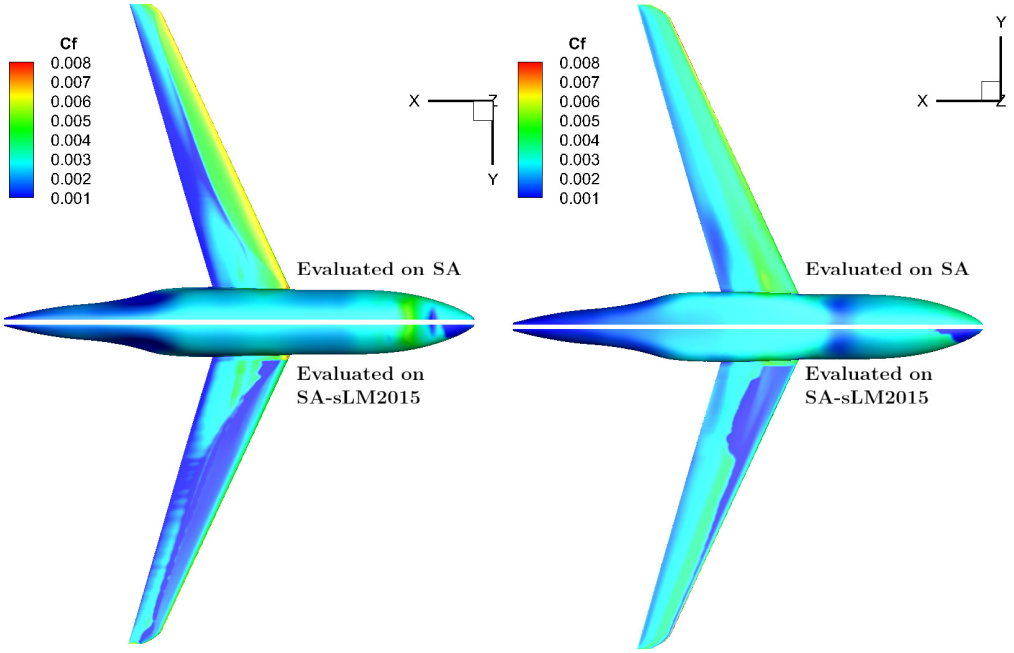


(c) $M_\infty = 0.80$

Figure 5.7: *HARW Business Jet*: Polar diagram C_L vs. C_D resulted from the Spalart–Allmaras (no transition model) and the SA-sLM2015 transition model for (a) $M_\infty = 0.76$, (b) $M_\infty = 0.78$ and (c) $M_\infty = 0.80$. Drag values on the horizontal axis are not shown on purpose. The spacing indicates a difference of 20 drag counts.

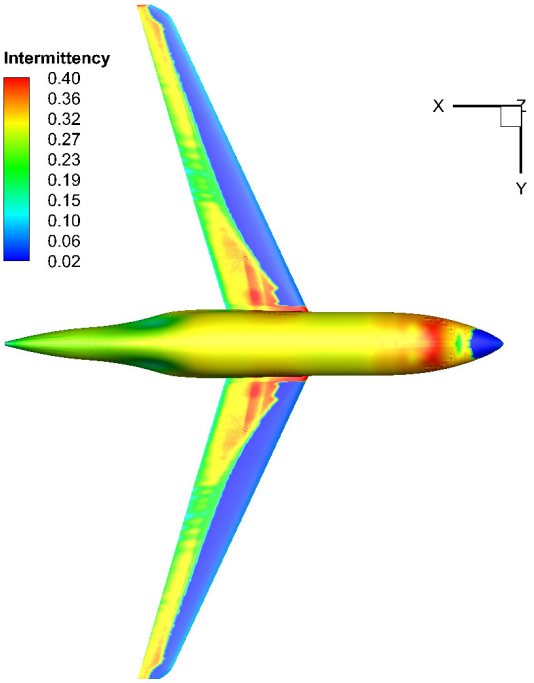


(a) C_p field

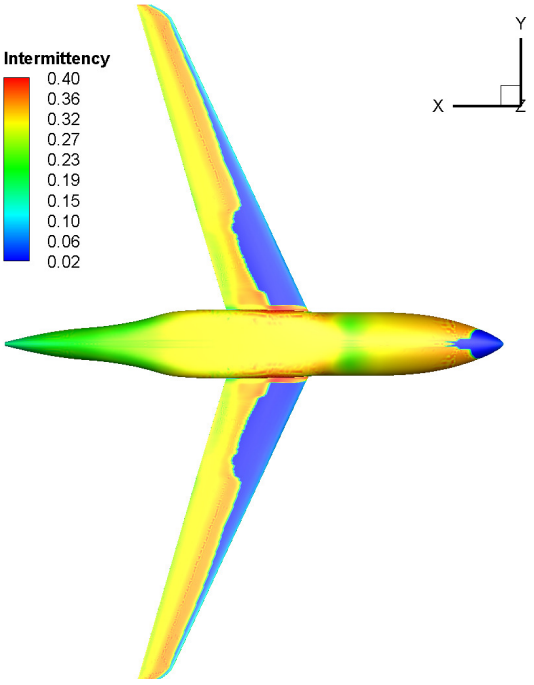


(b) C_f field

Figure 5.8: *HARW Business Jet*: (a) Pressure (C_p) and (b) skin friction coefficient (C_f) fields on the aircraft surface for $M_\infty = 0.78$ and AoA$=2°$. Computations based on the Spalart–Allmaras without and with the SA-sLM2015 transition model are simultaneously presented. Left: upper side of the aircraft. Right: lower side of the aircraft.



(a) Top View



(b) Bottom View

Figure 5.9: *HARW Business Jet*: Intermittency (γ) field on the aircraft surface for $M_\infty = 0.78$ and AoA$=2^\circ$. Low values of γ indicate the laminar zone.

5.2.2 Optimization of the HARW Business Jet

For the optimization, the wing is parameterized using the $4\times9\times2$ control box of Fig. 5.10. The control points in red are allowed to move in pairs in the z direction; in specific, every pair of control points with the same x and y coordinates performs the same displacement in the z direction, i.e. their distance is fixed. Such a way of displacement in the z direction ensures constant wing's chord distribution during the optimization. The control points in blue remain fixed. The 24 z coordinates of the control points constitute the design variables of the shape optimization.

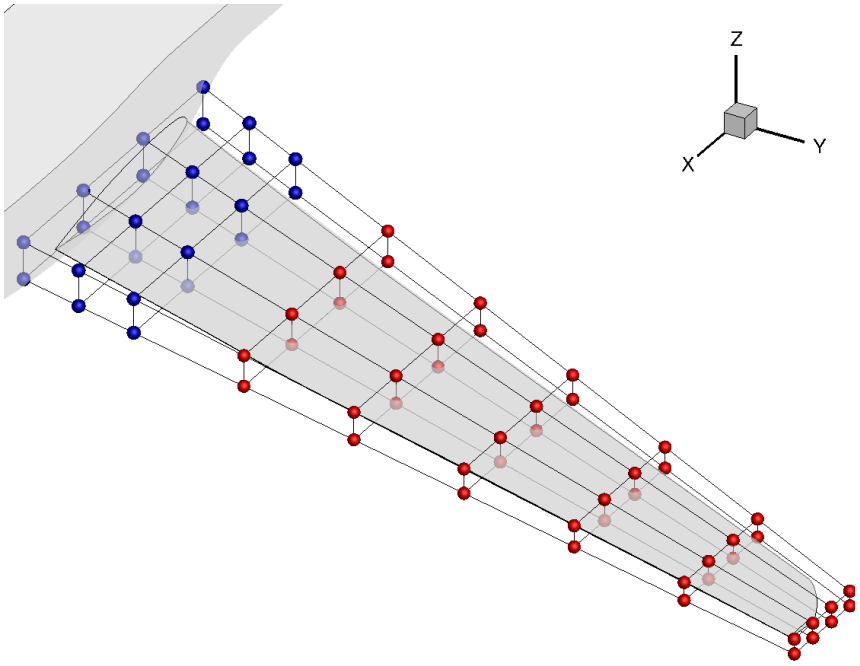


Figure 5.10: *Optimization of the HARW Business Jet*: NURBS control lattice formed around the wing. Control points in blue are fixed whereas the red are allowed to move in pairs (for the same x and y position) along the z-direction. The wing's chord remains constant during the optimization.

The optimization is performed for $M_\infty = 0.78$ and is carried out in steps, using different flow models, as an attempt to reduce its overall cost. Initially, the Spalart–Allmaras turbulence model (without transition) is used. Since a constraint on the C_L value is imposed (desired value equal to 0.5), the first step is to define the AoA that meets this constraint for the baseline geometry with the Spalart–Allmaras turbulence model. Then, the wing shape optimization follows for this AoA and with the same model. When this optimization converges, the optimized solution is evaluated on the SA-sLM2015 transition model, and the optimization continues with the transition model active in the evaluation tool. The optimization steps are described below:

1. Define the AoA value for which $C_L = 0.5$ by means of the Spalart–Allmaras model without implementing a transition model.
2. By fixing this AoA value, perform shape optimization using the adjoint code to the model mentioned before.
3. Evaluate the optimized aircraft shape on the Spalart–Allmaras model with the SA-sLM2015 transition model.
4. Continue the optimization with the SA-sLM2015 transition model and its adjoint.

Turbulent Optimization of the HARW Business Jet

The two first optimization steps are performed with the Spalart–Allmaras turbulence model (without implementing transition model). At the first optimization step, after 5 adjoint-based cycles (in which the objective is to reach the desired C_L value and the only free variable is the AoA, i.e. with a fixed wing shape) the AoA that gives $C_L = 0.5$ is found to be equal to $2.67°$. Next step is the shape optimization of the wing for this AoA. The optimization results in a reduction in C_D by $\sim 17\%$, meeting the constraint on C_L, Fig. 5.11. The C_p distribution on the aircraft surface for the baseline and the optimized wings is shown in Fig. 5.12. The main difference between the two geometries appears over the suction side of the wing where the area with low C_p values is increased leading to the C_D reduction. Small differences appear at the pressure side, mainly close to the wing root, which are though dominated from those of the suction side.

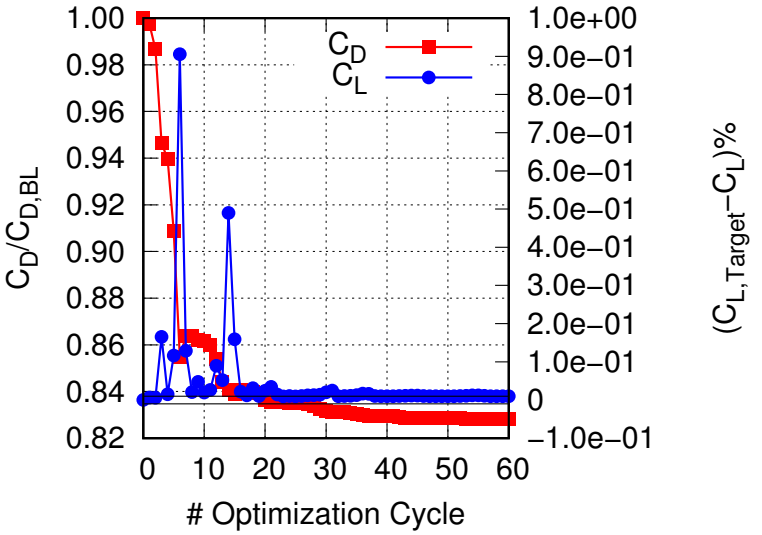
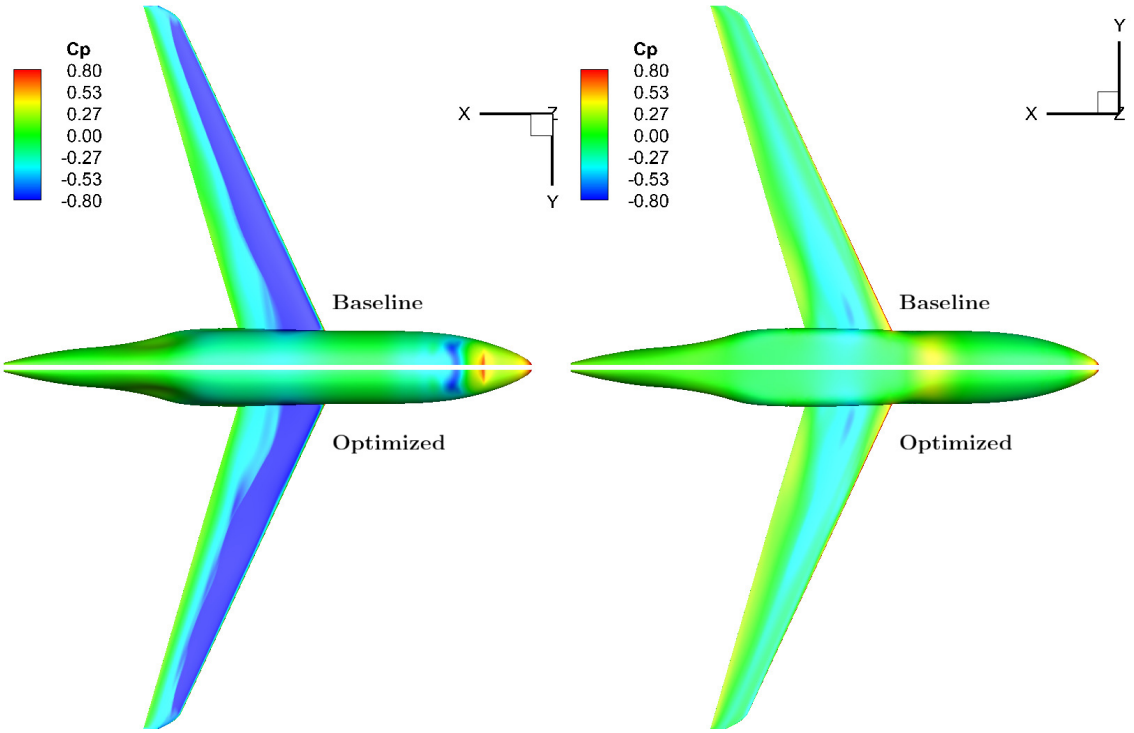


Figure 5.11: *Optimization of the HARW Business Jet*: Evolution of the objective (C_D) and constraint (C_L) functions during the optimization based on the Spalart–Allmaras turbulence model (without transition effects) for $M_\infty = 0.78$ and AoA$\simeq 2.67°$.



(a) Top View **(b)** Bottom View

Figure 5.12: *Optimization of the HARW Business Jet*: C_p fields on the aircraft surface for the baseline and the optimized with the Spalart–Allmaras model (without transition) geometries for $M = 0.78$ and $C_L = 0.50$. Left and right sides of the aircraft correspond to the baseline and optimized geometries, respectively.

Optimization of the HARW Business Jet using a Transition Model

In order to take into account transitional effects, the optimized geometry of Sec. 5.2.2 is then evaluated on the SA-sLM2015 transition model. This is the third step of the optimization. Though the resulting C_D value remains less than the initial one, the $C_L = 0.5$ constraint is no more satisfied. For this reason, an additional optimization is performed (step four) so as to meet the desired C_L value by altering the AoA; the resulting value in this case is AoA $\simeq 2.52°$. The overall procedure is schematically presented in the C_L vs. C_D polar diagram, Fig. 5.13; the C_D reduction is $\sim 17\%$ compared to the baseline geometry on the same model with $C_L = 0.5$. The skin friction coefficient (C_f) field and the contribution on a per face basis of the C_D normalized with the corresponding area are presented for the baseline and the optimized by the SA-sLM2015 transition model aircraft in Figs. 5.14a and 5.14b, respectively. The transition location line at the suction side of the wing is moved downstream for the optimized geometry; as a result the laminar area on the wing is increased leading to the C_D reduction, Fig. 5.14b. This was

in fact the idea behind the HARW configuration, to design a wing where the C_D reduction is achieved by increasing the laminar zone (instead of just modifying the shape) making the use of transition modeling necessary. This is confirmed by the per face C_D values normalized by the area, Fig. 5.14b; the main regions where the C_D reduction comes from are those where the flow has become laminar.

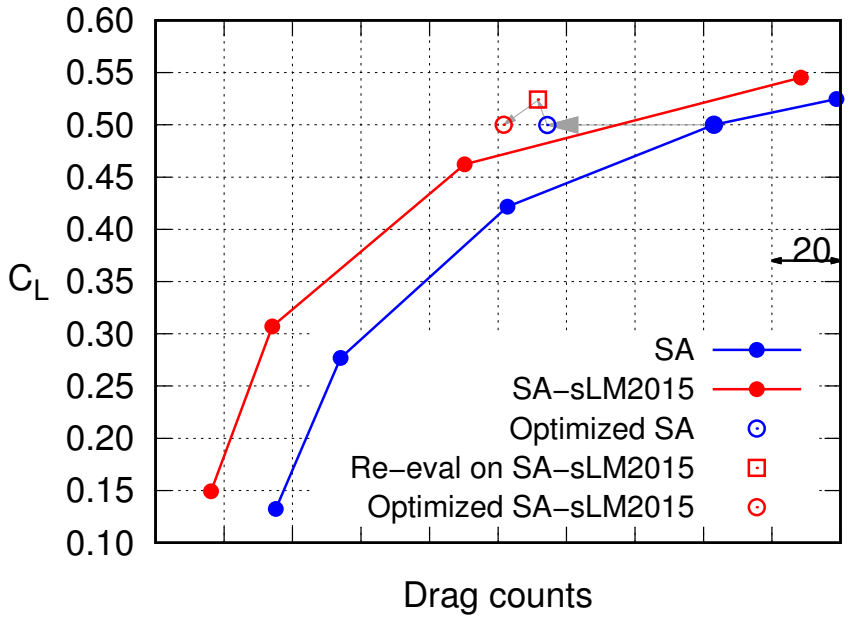
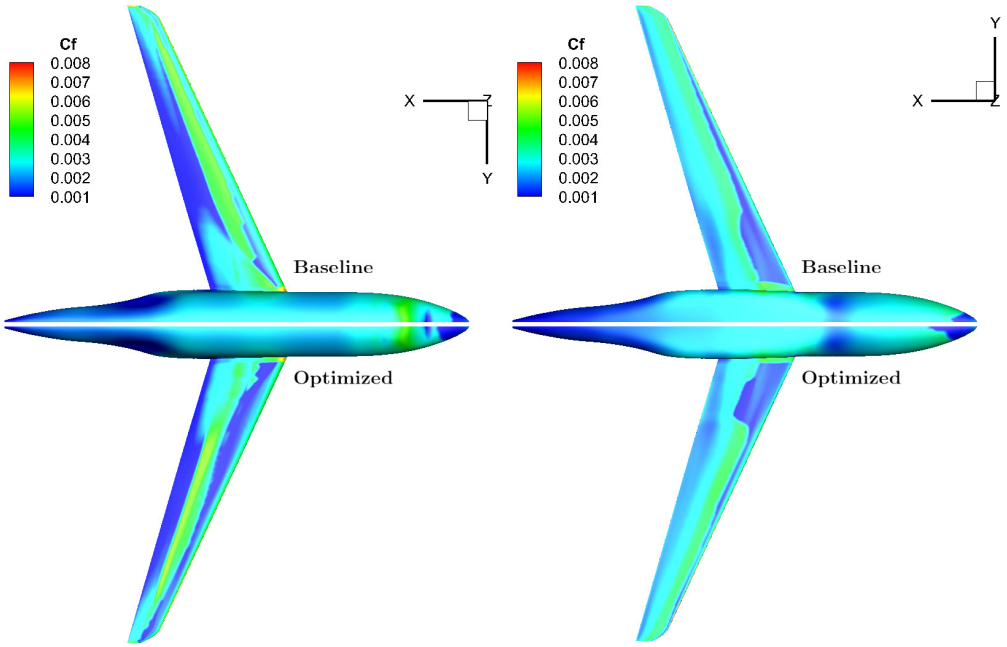
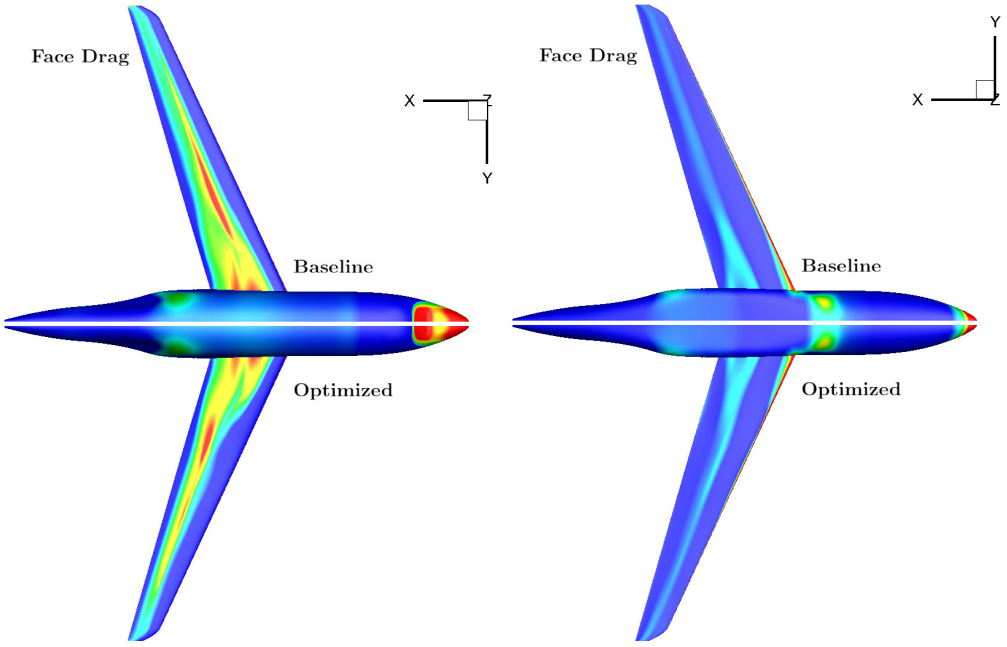


Figure 5.13: *Transitional Optimization of the HARW Business Jet*: Optimization on the SA-sLM2015 transition model. The optimized geometry based on the Spalart–Allmaras turbulence model (blue empty circle) is re-evaluated on the SA-sLM2015 transition model (red empty square). An additional optimization is performed in order to meet the C_L constraint (red empty circle). The polar curve for the baseline geometry for $M_\infty = 0.78$ is also presented for the Spalart–Allmaras and the SA-sLM2015 models.



(a) C_f field



(b) Per face C_D

Figure 5.14: *Transitional Optimization of the HARW Business Jet*: (a) Skin friction coefficient C_f field and (b) per face C_D normalized with the corresponding area on the aircraft surface for the baseline and the optimized with the SA-sLM2015 model geometries for $M=0.78$ and $C_L=0.50$. Left and right sides of the aircraft correspond to the baseline and optimized geometries, respectively. C_D values are not shown on purpose.

Chapter 6

DNN Surrogates for Turbulence & Transition Closure in Shape Optimization using EAs

The solution of the RANS equations coupled with turbulence and transition models is being used in CFD analysis and optimization tasks. As it has already showcased in Sec. 2.6, CFD methods can generate highly accurate results but can be computationally expensive. Despite the fact that the PUMA code exploits the GPU processing power and speed, which dramatically decreases the cost of the simulation, this can still be high. Given that, nowadays, optimization methods to solve industrial engineering problems are routinely used, possessing and using a dependable flow prediction tool at low computational cost is essential. On the other hand, Deep Neural Networks (DNNs) penetrate into various complex problems due to their ability to detect input data properties and patterns and make low cost predictions; this characteristic makes them suitable for assisting mono/multi-disciplinary analysis and optimization problems. Additionally, it is worth noting that for consecutive years industries have been trying to design a better version of the same product; thereby creating databases with the corresponding CFD simulations which can be used to train DNNs to replicate (part of) the CFD solver with negligible cost.

Herein, a DNN-based surrogate for turbulence and transition closure to be used during the numerical solution of the RANS equations is proposed; the role of the DNN is to provide the turbulent viscosity field in each pseudo-time iteration of the solver of the mean flow equations by replacing the turbulence and transition models (Spalart–Allmaras, $k-\omega\ SST$ and $\gamma-\tilde{Re}_{\theta t}$). The target is to reduce the computational cost of the CFD analysis by avoiding the solution of the turbulence-related PDEs (one, two or four, if the transition model is also active) with a DNN, with satisfactory prediction accuracy. The DNN configuration, i.e. the so–called hyperparameters (such as the number of layers, number of neurons, type of activation functions) and the DNN input data are optimized

for each case. The resulting code (combination of CFD with DNN) is, then, used as the evaluation tool for shape optimizations; these are carried out by means of Metamodel-Assisted Evolutionary Algorithm (MAEA) enhanced by the Principal Components Analysis (the EASY platform, [4], developed by PCOpt/NTUA). A brief description of the EASY platform can be found in Appendix B. This MAEA has a dual role as it is used both for the definition of the optimal DNN configurations and the shape optimizations.

6.1 Basics of DNNs

As already discussed in Chapter 1, the Artificial Neurons (Fig. 1.1b), are inspired by the way the biological neurons operate. The biological neuron's synapses correspond to weight signals received from other neurons which are delivered to the cell body by dendrites. The cell body's major operations are the summations of the weighted signals; if this signal exceeds a threshold value, it is delivered to the output down the axon. In Artificial Neurons, the threshold is represented by activation functions, which add non-linearities assisting the model to deal with complicated problems. There are various types of activation functions, in this PhD Thesis, the Rectified Linear Unit (ReLU), the Gaussian Error Linear Unit (GELU), the tangent hyperbolic (*tanh*) and the sigmoid functions are used:

- ReLU $g(z) = \max(0, z)$
- GELU $g(z) = \frac{z}{2}\left[1 + erf\left(\frac{z}{\sqrt{2}}\right)\right]$
- *tanh* $g(z) = \tanh(z)$
- sigmoid $g(z) = \frac{1}{1+e^{-z}}$

An ANN consists of multiple neurons and layers. An artificial neuron performs a transformation of its input to an output, Eq. 6.1

$$o_j = g\left(\sum_i^r w_{ij}x_i + b_j\right) \tag{6.1}$$

where o_j is the output of the j^{th} neuron, w_{ij} are the weights connecting the i^{th} with the j^{th} neuron, x_i are the signals of the i^{th} neuron, b_j is the bias of the j^{th} neuron, g is the activation function and r are the number of neurons at the current layer. The weights and biases of all neurons are computed during the training phase and affect the prediction capability of the ANN. To train an ANN, a database consisting of pairs of inputs and corresponding outputs should be available. During the training phase, some training patterns of the database are fed to the network which learns to predict the output regarding the input by appropriately adjusting the weights and biases. For DNNs, the training phase

corresponds to an optimization (for the weight and biases) which minimizes a cost function C. This is a metric of the DNN's prediction capability, and is expressed as the difference between the DNN prediction and the exact output, Eq. 6.2

$$C = \frac{1}{N} \sum_{i=1}^{N} C_i \left(w, b, \vec{x}^{(i)}, \vec{y}^{(i)} \right) \tag{6.2}$$

where C_i is the cost function for a single, of the total N, training patterns, $\vec{x}, \vec{y}$ are the input and output units. C_i can be, among others, the absolute/relative difference or the mean square value of the exact output of the i^{th} pattern w.r.t. the DNN's prediction.

Since DNNs can be expressed as analytical functions, the derivative of the cost function (C) w.r.t. the design parameters (v, e.g. the weights and the biases) ($\frac{\partial C}{\partial v}$) can be analytically computed. Gradient-based algorithms can utilize the computed gradient to carry out the optimization. The main problem is, though, that this analytical method proves to be extremely slow for the DNNs. For this reason, the back-propagation algorithm is used. Back-propagation, [118], is a key functionality in ML as it allows the information from the cost function to propagate backwards through the DNN architecture, in order to compute the gradients of each neuron through each layer. This procedure is applied repetitively from the output layer to the input layer, so as to compute all the necessary gradients.

For the optimization, the Adaptive Momentum estimation (Adam), [66], algorithm is used in this PhD Thesis. Adam is a stochastic optimization algorithm that combines the advantages of the Root Mean Square Propagation (RMSProp), [132], and Adaptive Gradient Algorithm (AdaGrad), [33], methods. It uses estimations of the first and second moments of the gradient to adapt the learning rate for each weight of the neural network; it is an efficient algorithm, well suited for large problems in terms of data and/or parameters.

6.1.1 Fully Connected Neural Network Architecture

A DNN consists of multiple neurons arranged into layers and connected with each other. A DNN architecture consists of an input layer, a number of hidden layers and an output layer, Fig. 6.1. Each hidden layer contains different numbers of neurons and different activation functions. The input layer is the first layer of the DNN and contains all of the input neurons (there is where the information is provided). No computations are performed at the input layer; the hidden layers are where all computations are performed. They are intermediate layers between the input and output transferring the information between them. Finally, the output layer contains the outcome of the DNNs processes w.r.t. the provided input and transfers the information from the DNN to the outside environment.

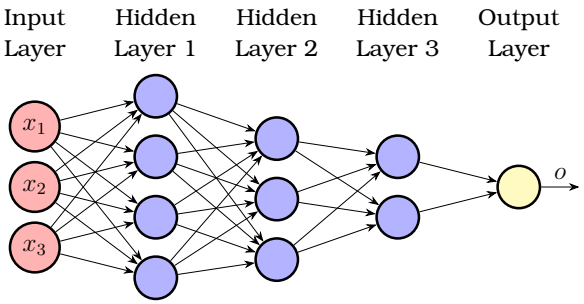


Figure 6.1: DNN architecture with 3 input units (x_1, x_2, x_3), 3 hidden layers and a single output unit (o).

The most widely used DNN types are the Fully Connected, the Convolutional, the Long Short-Term Memory, the Recurrent and the Generative Adversarial Networks. Herein, the fully connected neural networks are used. Fully Connected Neural Networks (FCNNs) consist of a series of fully connected layers; every neuron of a layer is connected to every neuron of the previous layer. Every input of the input layer influences every output of the output layer, but not all weights affect all outputs. The main advantage of FCNNs is that they are "structure agnostic" meaning that no special assumption needs to be made regarding the type of input. This "structure agnostic" feature makes them capable of being used in many different problems.

6.2 The proposed DNN-based Surrogate for Turbulence Closure

In this chapter, a DNN-based surrogate for turbulence and transition closure of the RANS equations is proposed. The key idea is to use the DNNs for replicating the numerical solution of the turbulence and/or transition model PDEs (described in Secs. 2.1.2, 2.1.3 and 2.1.4) in CFD problems, in order to reduce the computational cost. To do so, the turbulence and/or transition models are replaced with Fully Connected Neural Networks, trained to provide the turbulent viscosity fields with affordable accuracy. The model is supported by an automated procedure based on Metamodel-Assisted Evolutionary Algorithms (MAEAs) for defining the optimal DNN configuration. The ultimate goal is to introduce this into a stochastic (herein, MAEA-based) optimization which requires a great number of calls to the CFD code.

The first step is to sample the design space to generate a set of different geometries (samples) for which the RANS equations coupled with the turbulence and, if necessary, the transition model (to be referred as the RANS-TM model) are solved. The design space is defined by the ranges of the coordinates of the control points of the NURBS boxes which are used to parameterize the shape of each case. In all cases, a Latin hypercube sampling is used and the computed

flow fields constitute the database used for training the DNN (DB_{DNN}).

For the selected set of samples, the optimal DNN configuration must be found (see Sec. 6.2.1). After training, the DNN is used to predict the μ_t field in each pseudo–time iteration during the solution of the mean flow equations (by computing stresses in Eq. 2.3). The proposed algorithm, for each pseudo–time iteration of the solver, is shown in Fig. 6.2.

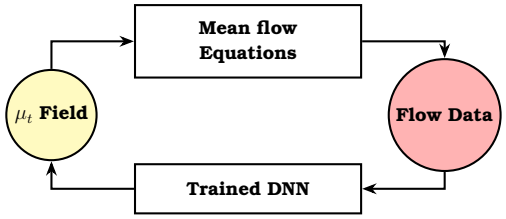


Figure 6.2: An iterative solver of the RANS equations using the DNN replicate of turbulence (and transition) model. The loop corresponds to a single pseudo-time iteration of the RANS-DNN model.

6.2.1 Defining the Optimal DNN Configuration

To determine the optimal DNN configuration, its hyperparameters (number of layers, number of neurons per layer, type of activation functions, etc.) as well as the inputs to the network should be carefully selected. While, in the literature, the selection of both the hyperparameters and the input data is often based on experience and/or a trial-and-error procedure, the nature and mechanisms of turbulence to be replaced by the DNN asks for a targeted approach.

Herein, the optimal DNN configuration results from a MAEA-based optimization (i.e. the same tool used to perform the shape optimizations). The target is to minimize the DNNs' prediction error. For each layer (hidden or output), the number of neurons is defined in powers of 2 and the ReLU, GELU, *tanh* and sigmoid activation functions can be selected. Local flow and geometrical quantities are used in the DNN input layer, when working with data-driven turbulence models. The pool of possible quantities includes geometrical information, such as nodal coordinates (x_k) and distances (d) from the nearest wall, and flow data, such as the primitive flow variables, i.e. density (ρ), velocity components (v_k), pressure (p) and quantities related to the gradient of the flow data, such as vorticity (ζ) and strain rate magnitude (S). In total, 11 quantities in 3D (9 in 2D) cases can be selected for the input layer.

In order to further reduce the cost for finding the optimal network configuration (hyperparameters and input layer data), each candidate network is trained for a few epochs, enough though to obtain a good indication of its prediction accuracy. Only upon completion of the corresponding MAEA run, the DNN with the optimal

configuration is trained to convergence. Regarding the training and validation datasets, a rule of $80-20\%$ is applied.

The development and training of DNNs is carried out in the TensorFlow framework (v2.6.0) using Python. The trained DNNs are then directly implemented into the PUMA software in order to suppress the data transfer cost between the CFD solution and the DNN prediction and exploit the GPU capabilities. This is achieved using the TensorFlow C++ API. The previously trained network (architecture/weights) is loaded in the starting phase of PUMA and uninterruptedly used during the solution of the RANS equations.

6.2.2 Assessment of the DNN in the MAEA framework

The trained DNN becomes the surrogate for turbulence closure within the numerical solution of the RANS equations and this is implemented as evaluation software in a MAEA-based shape optimization. The new evaluation model is faster compared to the standard solution of the RANS equations with the turbulence (and transition) models. One time unit corresponds to the computational cost of the RANS-TM, which includes the solution of the turbulence and, if necessary, the transition model PDEs; of course, time unit is different from case to case.

During the evolution, when outliers with respect to the DB_{DNN} appear or the MAEA stagnates, the DB_{DNN} is automatically enriched and the DNN is updated/re-trained. To do so, individuals from the current elite population are re-evaluated on the RANS-TM model and appended to the DB_{DNN}. Individuals close (in the design space) to an existing DB_{DNN} entry are excluded since DNN predictions in their niche are good enough. The re-training cost is negligible compared to the initial training cost. The overall procedure, from the initial DNN setup to the optimized geometry is presented in Fig. 6.3.

6.3 Shape Optimization of the NACA4318 Isolated Airfoil

The first optimization case is dealing with the re-design of the NACA4318 isolated airfoil, by changing its shape, for max. lift (C_L) and min. drag coefficient (C_D). The flow conditions are summarized in Table 6.1; turbulence is modeled by means of the Spalart-Allmaras model. A structured grid of $\sim 30K$ is generated around the airfoil. The airfoil is controlled by the 5×3 NURBS box of Fig. 6.4; 13 (in red) out of the 15 control points are allowed to move by $\pm 10\%$ of their reference position in the chordwise and normal-to-the chord directions resulting in 26 design variables in total. A single CFD run (RANS-TM) takes ~ 1 min on an NVIDIA A100 GPU.

Based on the Latin hypercube sampling technique, 75 new airfoil shapes are created by altering the coordinates of the control points of the NURBS box. Each new airfoil is then evaluated on the RANS-TM building the DB_{DNN}. In this study,

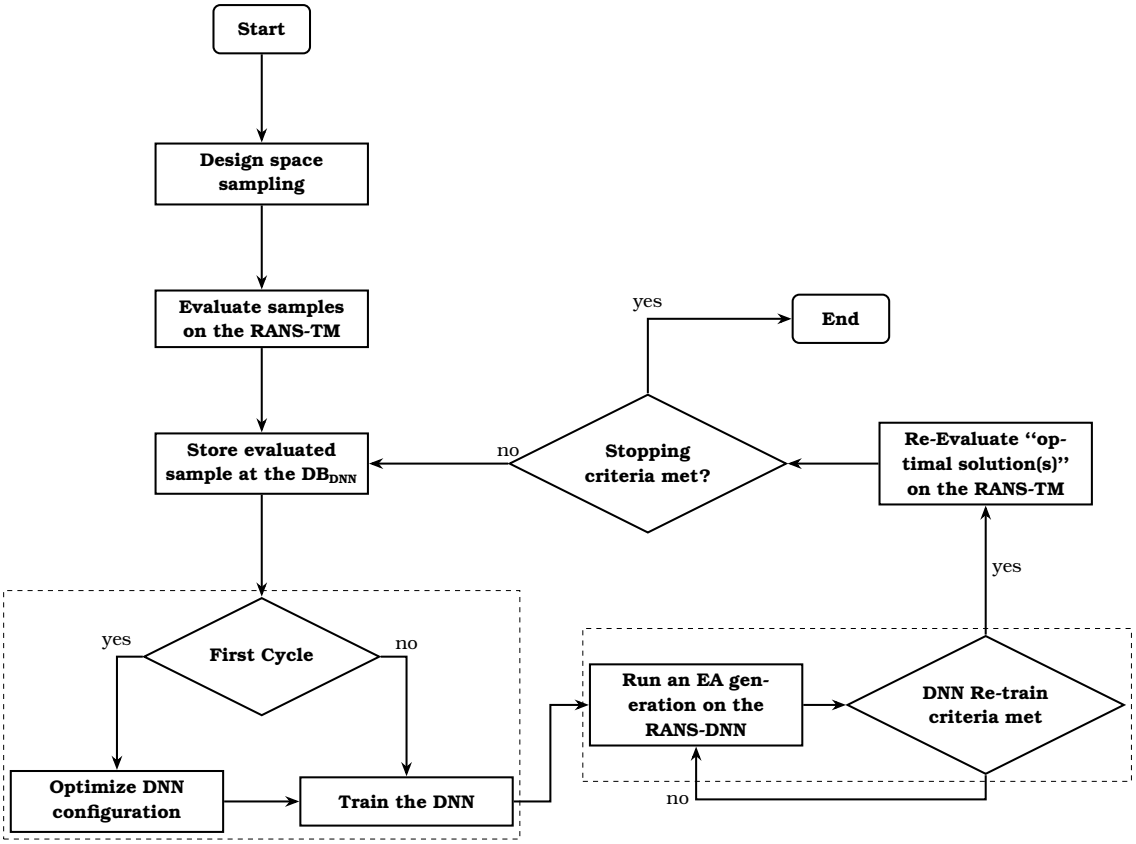


Figure 6.3: Flowchart of an evolutionary algorithm optimization based on the RANS-DNN model.

Quantity	Symbol	Value
Freestream Mach Number	M_∞	0.13
Reynolds Number $(\cdot 10^6)$	Re	3.8
Angle of Attack (°)	AoA	2.2

Table 6.1: *Optimization of the NACA4318 Airfoil*: Flow conditions.

the Spalart–Allmaras turbulence model is to be replaced by a properly trained DNN by predicting the turbulent viscosity μ_t field in each pseudo-time iteration of the RANS-DNN model. The optimal DNN configuration, resulted from the MAEA optimization, consists of 6 fully connected layers with $64, 128, 256, 1024, 4096, 512$ neurons each. Regarding the geometrical input data, the x coordinate and the distance from the nearest wall node (d) are selected. The absence of the y coordinate from the input data is probably due to the fact that is does not offer additional information compared to the wall distance. On the other hand, all available quantities related to the velocity vector and its gradient are selected (v_k, ζ, S), Finally, the ReLU and the $tanh$ activation functions are selected for all the hidden and

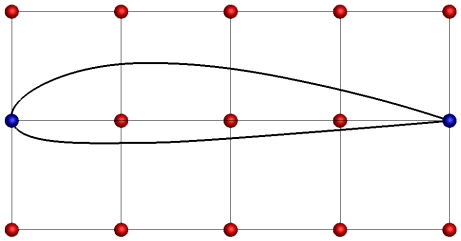


Figure 6.4: *Optimization of the NACA4318 Airfoil*: Parameterization of the NACA4318 airfoil. Control points in blue remain fixed, while the red ones are allowed to move in the chordwise and normal-to-the chord directions.

the output layer, respectively. The cost of defining and training the optimal DNN configuration is $\sim$2 h on an NVIDIA A100 GPU. The DNN replaces the numerical solution of the Spalart–Allmaras model, giving rise to the RANS-DNN solver. A CFD run based on the RANS-DNN solver costs $\sim$0.48 s on an NVIDIA A100 GPU. The Mach number and the turbulent viscosity fields based on the solution of the RANS-TM and the RANS-DNN solver, as well as their relative error, are presented in Fig. 6.5. The maximum prediction error appears in the airfoil boundary layer and wake areas.

Layers	**Neurons/Layer**	**Input Data**	**Act. Functions**
6	$64, 128, 256, 1024, 4096, 512$	x, d, v_k, ζ, S	ReLU/*tanh*

Table 6.2: *Optimization of the NACA4318 Airfoil*: MAEA-based optimal DNN configuration.

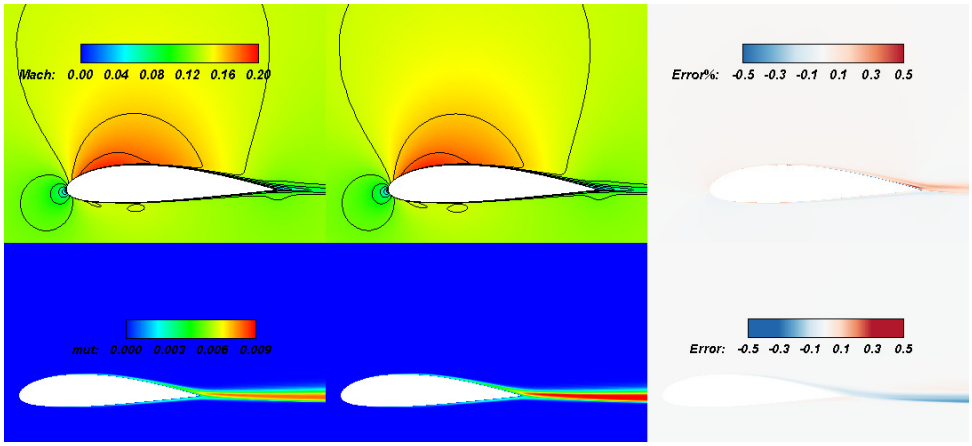


Figure 6.5: *Optimization of the NACA4318 Airfoil*: Mach number (top) and turbulent viscosity field (bottom) iso-areas computed by the RANS-TM (left) and RANS-DNN (center) models; relative errors between the two fields (right). This is a new airfoil geometry, not seen by the DNN before.

The shape optimization of the NACA4318 isolated airfoil for max. C_L and min. C_D follows, seeking for a Pareto front. Optimization studies using both the RANS-TM and RANS-DNN models as evaluation software are performed in order to qualify the RANS-DNN capabilities. For a fair comparison, both runs are identically configured: a $(10, 30)$ MAEA is used; the RBF type metamodels in the MAEA start being used after having evaluated $T^{MM} = 90$ airfoils and, during the LCPE phase, only $\lambda_e = 2$ members in each generation are re-evaluated on the software.

The termination criterion for the whole optimization is set to 500 time units, being the cost to run 500 evaluations on the RANS-TM solver. A single RANS-DNN run costs 0.8 and the DNN training 120 time units. Thus, the above termination criterion corresponds to 450 evaluations on the RANS-DNN (360 time units), since 20 more time units are spent to re-evaluate the best solution(s) on the RANS-TM before re-training the DNN and, also, after completion of the optimization. During the evolution, at the 20^{th} generation of the MAEA (after ~ 200 evaluations) an outlier w.r.t. the DB_{DNN} appears, a snapshot of the MAEA convergence at that point is presented in Fig. 6.6. The current front of non-dominated solutions from the RANS-DNN model is shown in filled red squares along with the members of the DB_{DNN} in black empty circles. The member identified as outlier is the one with the highest C_L value. The existence of the outlier is an indication to assess the DNN's accuracy by re-evaluating the individuals of the current front on the RANS-TM resulting in the blue filled circles. Indeed, the prediction accuracy of the DNN in the region of high C_L values (almost 20% higher than members included in the DB_{DNN}) is questionable. For that reason, the DB_{DNN} is enriched with the re-evaluated members and the network is updated/re-trained. The re-training cost is negligible compared to the initial training, while the re-evaluation cost is included in the overall training cost stated before.

The optimization continues using the updated network and results in the front of non-dominated solutions of Fig. 6.7 (filled red squares). After their re-evaluation on the RANS-TM, the front with filled blue circles is obtained. In the area of high lift (and high drag) values, the error in the C_L, C_D values is between 2% and 4% while, in all other areas, the error is below 1%. However, one should keep in mind that the DNN predicts the μ_t field rather than the objective function(s). In any case, the error values observed are affordable and the re-evaluated airfoils remain non-dominated; this demonstrates the capability of the DNN to replace the Spalart–Allmaras model during the EA-based shape optimization and this reduces the computational cost.

Fig. 6.7 also presents the front resulted from an optimization run exclusively based on the RANS-TM (green triangles). This is close to the one resulted from the RANS-DNN (after re-evaluation on the RANS-TM model; blue circles) at the same computational cost; both fronts yield similar hypervolume indicator[1] values.

[1]The hypervolume indicator quantifies the percentage of the area (or volume) dominated by the front members within a "hyperbox" defined by pre-selected nadir and zenith points.

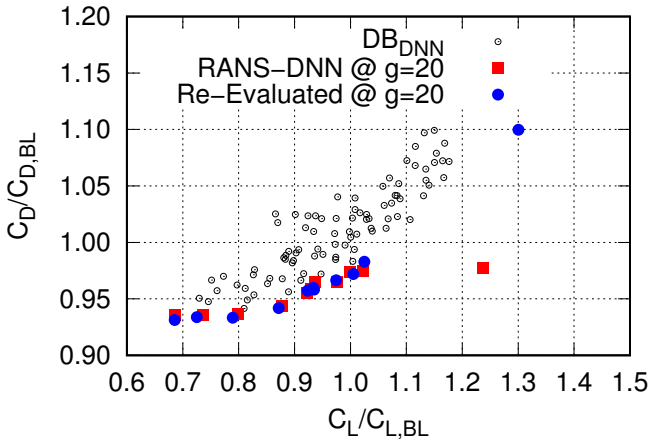


Figure 6.6: *Optimization of the NACA4318 Airfoil*: Front of non-dominated solutions at the 20^{th} generation (~ 200 evaluations) of the MAEA; these are presented twice, with performances computed on the RANS-DNN (filled red squares) and the RANS-TM model (filled blue circles). The DB_{DNN} (initial 75 members) is also shown with empty black circles.

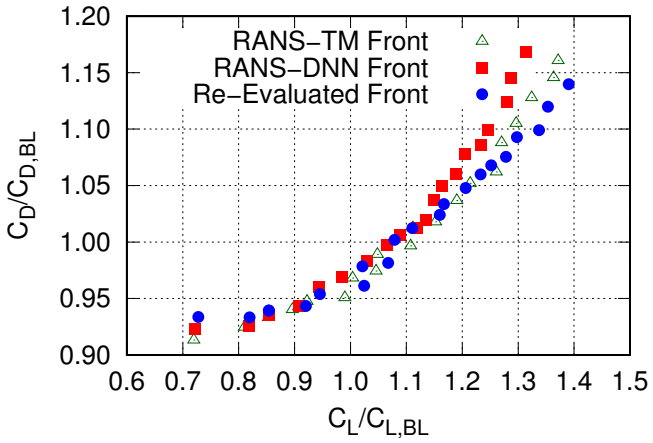
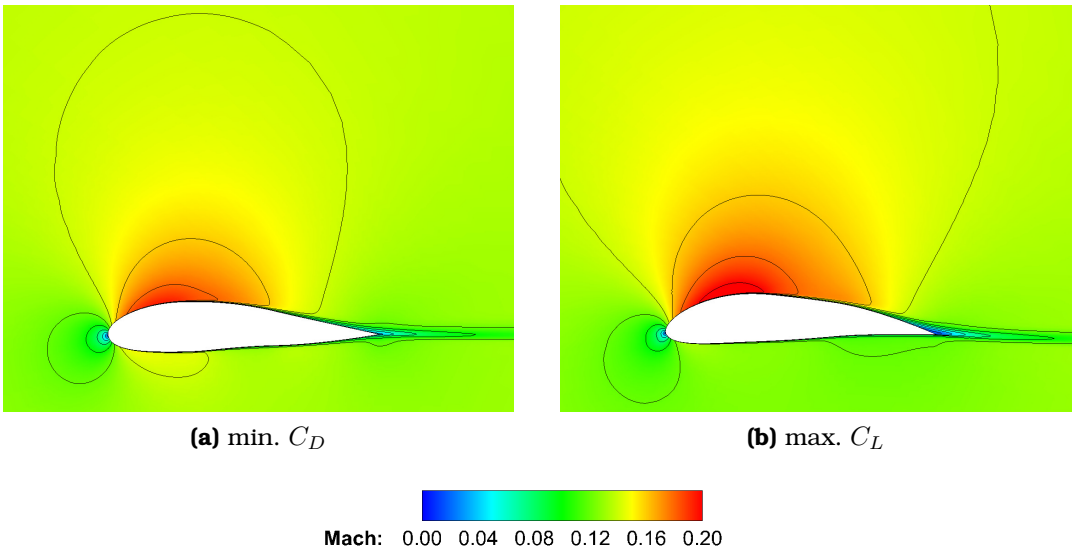


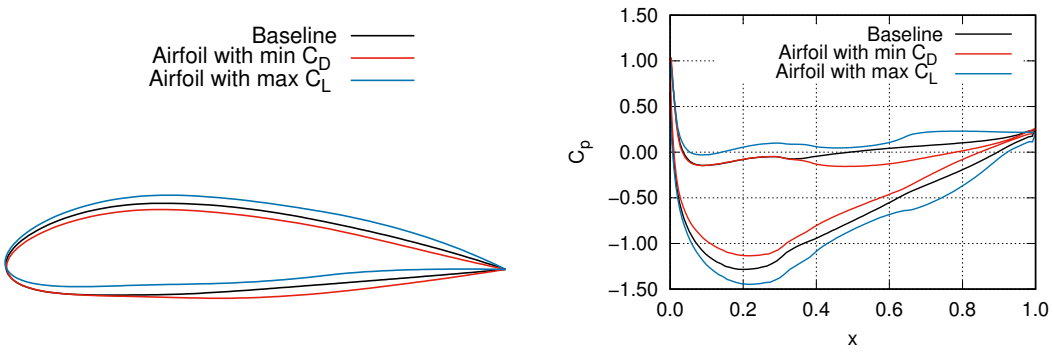
Figure 6.7: *Optimization of the NACA4318 Airfoil*: Front of non-dominated solutions from the MAEA on the RANS-DNN (filled red squares) and after its re-evaluation on the RANS-TM model (filled blue circles). The front resulted from another optimization run on the RANS-TM, at the same cost (500 time units) is shown with green empty triangles.

The Mach number fields around the airfoils that correspond to the two edges of the front (max. C_L and min. C_D) are shown in Fig. 6.8. The corresponding airfoil shapes and pressure coefficients are compared with the baseline geometry in Fig. 6.9. For the airfoil with max. C_L, the increased pressure difference between the pressure and suction side can easily be seen.



(a) min. C_D **(b)** max. C_L

Figure 6.8: *Optimization of the NACA4318 Airfoil*: Mach number fields for the airfoils at the two edges of the front of non-dominated solutions of Fig. 6.7; (a): airfoil with min. C_D, (b): airfoil with max. C_L.



(a) **(b)**

Figure 6.9: *Optimization of the NACA4318 Airfoil*: (a) Optimized geometries airfoils and (b) pressure coefficient distributions at the two edges of the front of Fig. 6.7 along with that of the baseline airfoil.

Since a stochastic optimization method is used, the same optimization procedure is repeated with two additional random number generator (RNG) seeds; the resulted fronts of non-dominated solutions are presented in Fig. 6.10. As in Fig. 6.7, in these cases too, the fronts from the optimization runs on RANS-DNN, after re-evaluations on the RANS-TM model, have comparable quality with those of the optimization on RANS-TM. Thus, it is reconfirmed that the DNN can successfully support an EA-based shape optimization process. Results from the use of the

RANS-DNN model in shape optimization studies are very promising, even if the cost of the RANS-TM run is very low (~ 1 min) in this case. Despite the small gain in the wall-clock time of the optimization, this is a first clear indication that, in cases with more expensive evaluations, the gain might be higher.

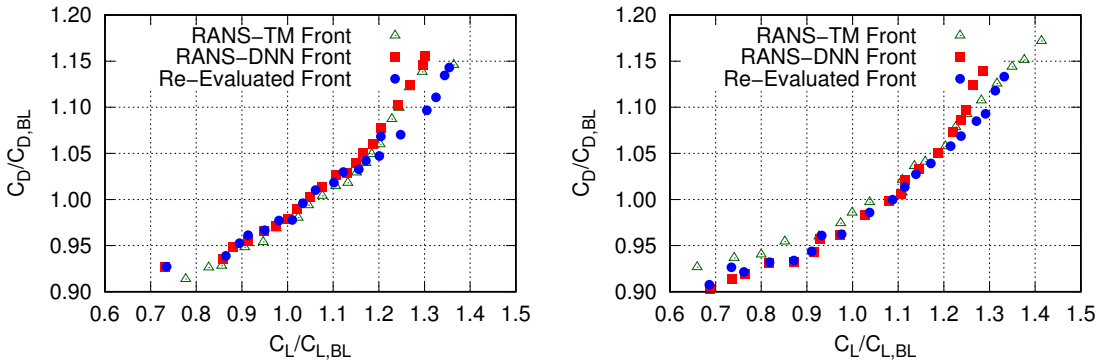


Figure 6.10: *Optimization of the NACA4318 Airfoil*: Two fronts of non-dominated solutions computed at the same computational cost and for different RNG seeds, from the MAEA with evaluations on the RANS-DNN (filled red squares) and after being re-evaluated on the RANS-TM model (filled blue circles). The front which resulted from another optimization run on the RANS-TM is shown in green empty triangles.

6.4 Shape Optimization of the LS89 Transonic Turbine Blade

This application concerns the optimization of a turbine blade in transonic flow conditions for mass-averaged total pressure losses minimization under the constraint that the exit angle (α^{out}) remains close to its reference (baseline) value; a threshold of $\pm 0.5^{\circ}$ is set. The highly loaded transonic linear turbine LS89, [7] (Case 116), validated and verified against experimental and numerical data in Sec. 2.6.2, is used as the baseline geometry. A single CFD run takes ~ 15 min on an NVIDIA A100 GPU.

The turbine blade airfoil shape is encapsulated within the 7×4 NURBS box of Fig. 6.11. The control points in red are allowed to move in the chordwise and the normal-to-the-chord directions within $\pm 10\%$ of their reference values. The 8 control points in blue along the left- and right-most sides of the control box are allowed to move only along the normal-to-the-chord direction, resulting in 48 design variables in total. The target of the study is to use a DNN as surrogate for both the $k-\omega$ SST and the $\gamma - \tilde{Re}_{\theta t}$ equations (4 equations in total) by predicting the turbulent viscosity field.

A DB_{DNN} consisting of 100 new blade shapes evaluated on the RANS-TM is formed. The optimal DNN configuration (Table 6.3) consists of 5 layers with

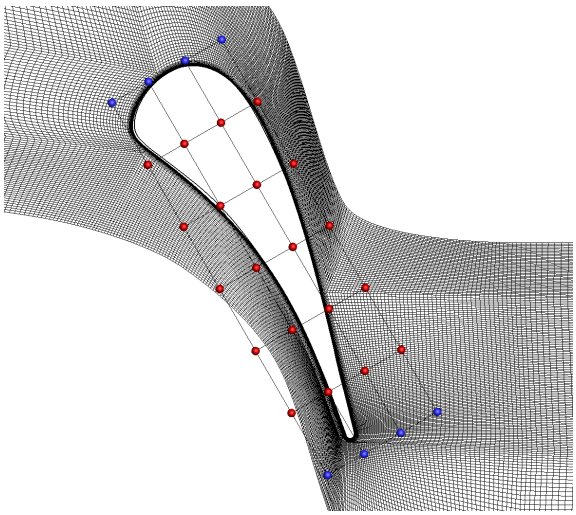


Figure 6.11: *Optimization of the LS89 Turbine Blade*: Computational grid around the transonic turbine blade. The turbine blade and the surrounded grid are parameterized by a 7×4 NURBS box. Red points are allowed to move in either direction, while blue points can only be displaced in the normal-to-the-chord direction.

$512, 512, 4096, 4096, 256$ neurons each. For all the hidden and the output layers the ReLU and the $tanh$ activation functions are used, respectively. Regarding the input data, all the available geometrical (x_k, d) and flow data (ρ, v_k, p, ζ, S) are selected by the MAEA. Despite the fact that the pressure (p) itself is not directly present at the expressions of the turbulence and transition models, it is present as an input to the DNN. This can be explained since the pressure can be used as an additional identifier of the position of the shock wave, in this transonic case. The cost for defining and training the optimal DNN configuration is $\sim$2.3 h on an NVIDIA A100 GPU. The Mach number fields for a new turbine blade (not included in the DB_{DNN}) computed by the solution of the RANS-TM and the RANS-DNN along with their relative errors are presented in Fig. 6.12. Apart from the wake region, an error of $\sim1\%$ can also be seen near the shock wave flow discontinuity.

Layers	**Neurons/Layer**	**Input Data**	**Act. Functions**
5	$512, 512, 4096, 4096, 256$	$x_k, d, \rho, v_k, p, \zeta, S$	ReLU/*tanh*

Table 6.3: *Optimization of the LS89 Turbine Blade*: MAEA-based optimal DNN configuration.

The LS89 turbine blade airfoil cascade is optimized for min. mass-averaged total pressure losses between the inlet and the outlet, subjected to constraints on the exit angle (α^{out}) which should remain close to its reference value. A (10,30) MAEA with the LCPE phase starting after $T^{MM} = 50$ individuals (blade airfoils), with $\lambda_\epsilon = 2$ members re-evaluated on each generation, is used. The PCA of the

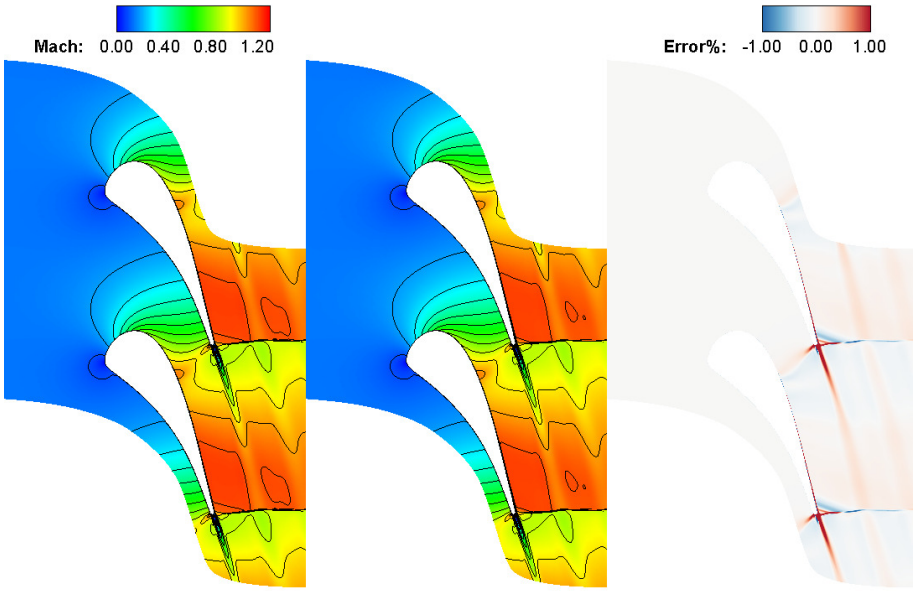


Figure 6.12: *Optimization of the LS89 Turbine Blade*: Mach number iso-areas computed by the RANS-TM (left) and RANS-DNN (center) models; relative error between the two fields (right). This is a new geometry, not seen by the DNN before.

offspring population before the application of the evolution operators in the feature space is also activated after the second generation.

The termination criterion is set to 250 time units; 100 and 9.2 are spent to form the $\mathrm{DB}_{\mathrm{DNN}}$ and train the network. A single run on the RANS-DNN costs ~ 9 min on an NVIDIA A100 GPU, i.e. 0.62 time units. The RANS-DNN is significantly faster compared to the RANS-TM since the DNN replaces the numerical solution of 4 equations in total. The convergence history of the optimization relying on the RANS-DNN model is shown in Fig. 6.13a.

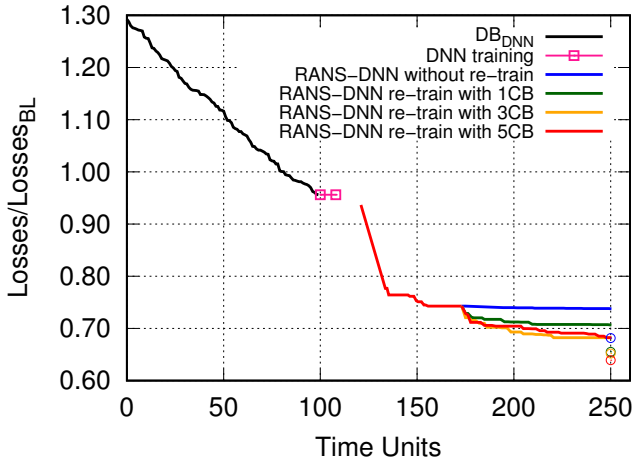
After ~ 100 evaluations on the RANS-DNN, the optimization stagnates after 6 successive generations during which the MAEA can not improve the best-so-far solution. Thus, the best-so-far solution(s) are re-evaluated on the RANS-TM, the $\mathrm{DB}_{\mathrm{DNN}}$ is enriched and the DNN is re-trained. A parametric study regarding the number of solutions that must be re-evaluated on RANS-TM is performed; one, three and five top individuals (blade airfoils) are used. An option without re-training the DNN is also used. Fig. 6.13a compares the convergence histories of the optimization runs using all the aforementioned parameters. Without re-training/updating the DNN, the optimization converges very slowly, while re-training the DNN directly boosts the optimization algorithm. It is interesting to compare the solutions resulted from the RANS-DNN by adding one (green curve), three (orange curve) and five (red curve) new individuals in the $\mathrm{DB}_{\mathrm{DNN}}$ after their re-evaluation on the RANS-TM. These results, compared with the best solution of the optimization run exclusively using the RANS-TM model, are summarized in

Table 6.4. Without updating the DNN, the solution are worse compared to any other result, even the one from the run on the RANS-TM model; this indicates the importance of re-training the DNN during the evolution when stagnation occurs. The re-evaluated solutions from the green and orange curves lead to solutions of similar quality; practically, the addition of only one new member in the DB_{DNN} leads to increased prediction error compared to the case with three new members. Adding five new members in the DB_{DNN} leads to the better solution for the same computational cost. It is important that, with the addition of a few new members (no more than 5% of the initial samples), one may significantly speed-up the optimization turnaround time. As shown in Fig. 6.13b, to obtain a solution of the same quality as that of the RANS-DNN run, with one or three (green and orange circles) members considered at the re-training phase, the RANS-TM model (black line) needs 100 more evaluations, i.e. cost increases by 40%. Moreover, an optimization on the RANS-TM model computes almost the solution obtained from the RANS-DNN model after adding five members during the re-training phase (red circle), at twice the computational cost.

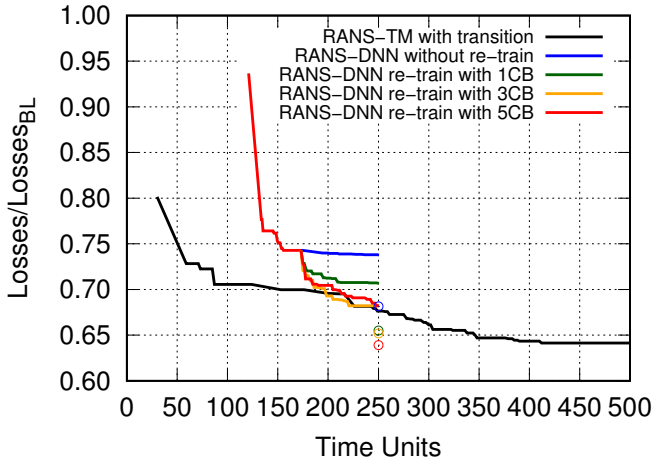
	Optimization on RANS-TM	Optimization on RANS-DNN			
		No re-training	1 Top	3 Top	5 Top
$Losses/Losses_{BL}$	0.6762	0.6812	0.6548	0.6517	**0.6389**

Table 6.4: *Optimization of the LS89 Turbine Blade*: Comparison of the optimal solutions resulted from the optimizations on the RANS-TM and RANS-DNN (the latter after being re-evaluated on the RANS-TM model).

The optimized blade geometries, for each parametric study, are compared with the baseline LS89 turbine in Fig. 6.14. The Mach number and the total pressure losses fields, i.e. the difference in total pressure from that of the inlet boundary, are plotted in Figs. 6.15a and 6.15b for the baseline and the optimized with the RANS-DNN solver (re-training with five current best solutions). The strength of the shock wave is decreased and its discontinuity (in the field of total pressure losses) is eliminated.

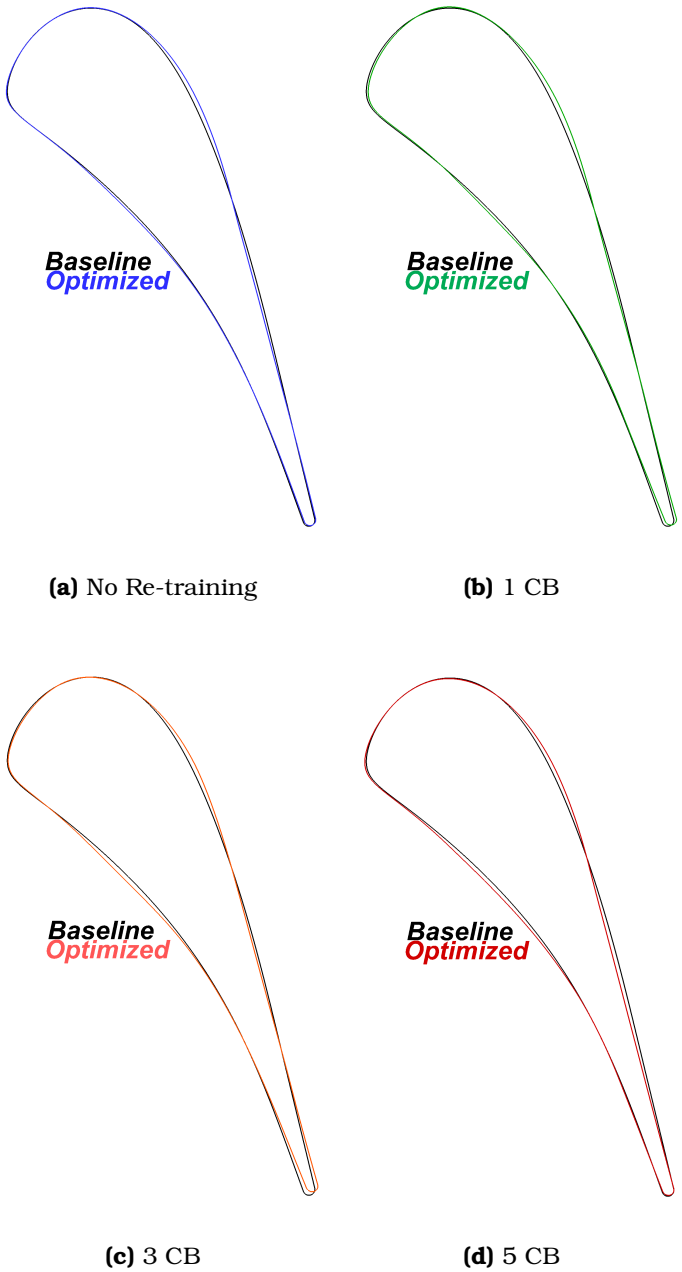


(a)



(b)

Figure 6.13: *Optimization of the LS89 Turbine Blade*: Convergence histories of the MAEA-based optimizations relying on the RANS-DNN model. Parametric study on the number of re-evaluated individuals: no re-training (blue curve), one (green curve), three (orange curve) and five (red curve) current-best blade/airfoil(s) considered, along with the re-evaluation of the optimal solution (empty circles) on the RANS-TM. (a) The DB_{DNN} plotted in descending order (black curve) and the DNN training cost (pink horizontal line segment). (b) The convergence history of the MAEA-based optimization relying on RANS-TM (black curve).



(a) No Re-training **(b)** 1 CB

(c) 3 CB **(d)** 5 CB

Figure 6.14: *Optimization of the LS89 Turbine Blade*: Optimized blade geometry compared with the baseline for the optimizations with (a) no re-training and re-training with (b) one, (c) three and (d) five current-best blade/airfoil(s).

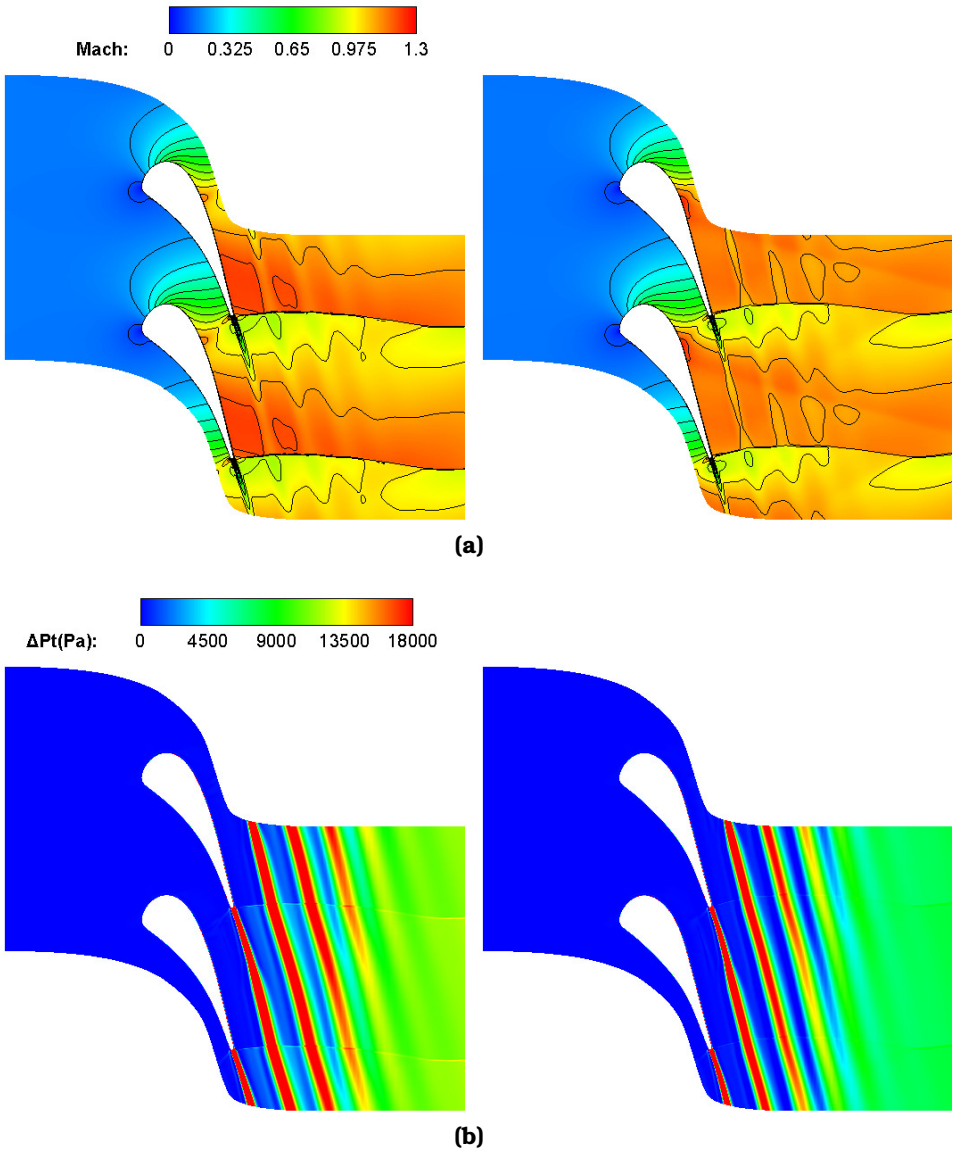


Figure 6.15: *Optimization of the LS89 Turbine Blade*: (a) Mach number and (b) total pressure losses fields for the baseline (left) and the optimized blade/airfoil re-evaluated with five current best solutions (right). The strength of the shock wave is reduced contributing to the decrease of the total pressure losses.

6.5 Shape Optimization of the DrivAer Car Model

In this application, the shape optimization of the DriVaer car model (fastback model with smooth underbody, mirrors and wheels, [51]) is carried out for min. drag coefficient (C_D). Flow conditions are summarized in Table 6.5. The Spalart–Allmaras turbulence model with wall functions is used. This case is studied with the incompressible fluid variant of the PUMA code that uses the pseudo–compressibility method. Computations are carried out for the half car, using symmetry conditions. A computational grid of $\sim 1.4M$ nodes is generated. The top of the rear part of the car is parameterized using the $7{\times}5{\times}6$ NURBS control box shown in Fig. 6.16. The control points in red are allowed to move by changing their reference coordinates by $\pm 25\%$ in the longitudinal and by $\pm 60\%$ in the normal-to-the ground directions, resulting in 96 design variables in total. A single CFD run takes ~ 1 h on an NVIDIA A100 GPU.

Quantity	Symbol	Value
Freestream Flow Velocity ($\mathrm{m\,s^{-1}}$)	U_∞	38.85
Air Kinematic Viscosity ($\cdot 10^{-5}\mathrm{m^2/s}$)	ν	1.5
Angle of Attack (°)	AoA	0.0

Table 6.5: *Oprtimization of a DrivAer Car Model*: Flow Conditions.

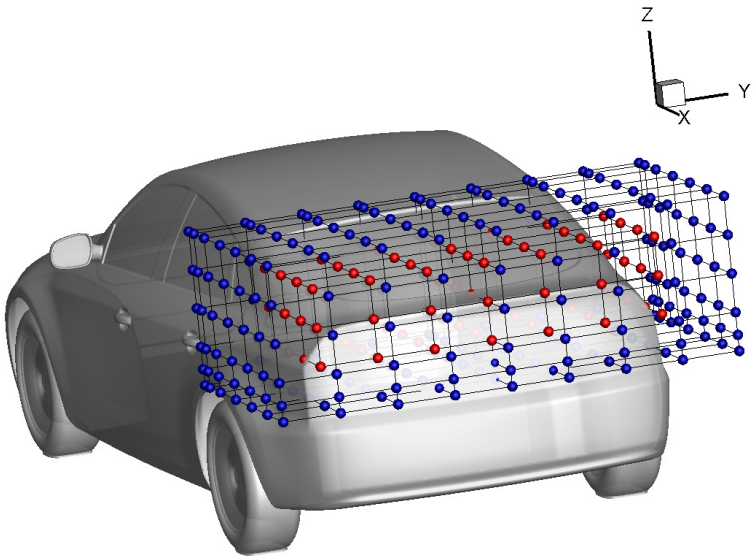


Figure 6.16: *Oprtimization of a DrivAer Car Model*: Parameterization of the DrivAer spoiler. Control points in blue remain fixed. The red ones are allowed to move in the longitudinal direction and normal-to-the ground.

In this computationally expensive application, the DNN is used to replace the numerical solution of the Spalart–Allmaras turbulence model by predicting the ν_t

field during the shape optimization; a total cost of 200 units is set. The DB_{DNN} consists of 100 different car geometries for which the RANS equations coupled with the Spalart–Allmaras turbulence model are solved. The design space is determined by the coordinate ranges of the NURBS box control points and the Latin hypercube sampling is selected as the sampling technique. The optimal DNN configuration, as resulted from the MAEA, is summarized in Table 6.6 and graphically presented in Fig. 6.17. The DNN consists of 4 layers with $64, 1024, 256, 1024$ neurons, respectively. The ReLU activation function is selected for all the hidden layers and the $tanh$ for the output one. The inputs to the DNN include all the available geometrical data (nodal coordinates and distances from the nearest wall nodes) and flow data related to the velocity and velocity gradients (velocity components, strain rate and vorticity magnitude). This is a relatively compact DNN architecture considering the application. The wall-clock time for finding and training the optimal DNN configuration is 6.5 h on an NVIDIA A100 GPU.

Layers	**Neurons/Layer**	**Input Data**	**Act. Functions**
4	$64, 1024, 256, 1024$	x_k, d, v_k, ζ, S	ReLU/*tanh*

Table 6.6: *Oprtimization of a DrivAer Car Model*: MAEA-based optimal DNN configuration.

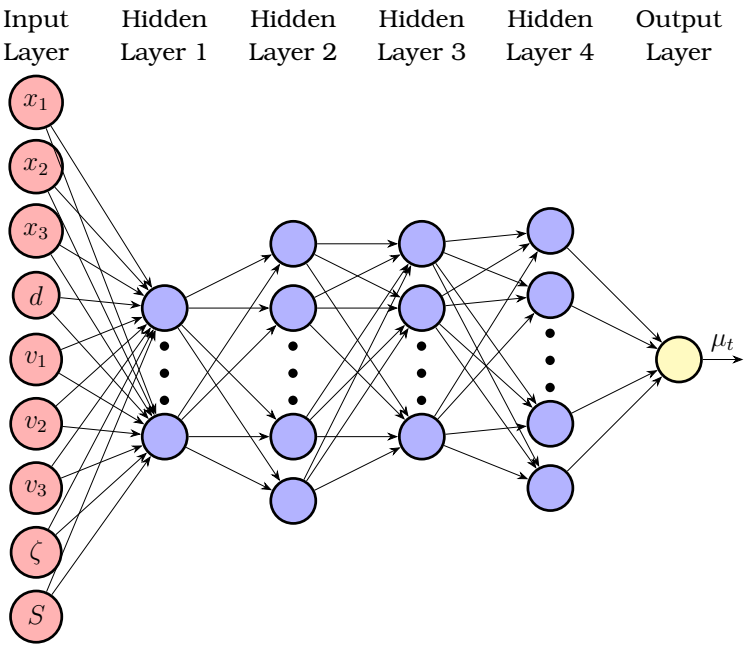


Figure 6.17: *Oprtimization of a DrivAer Car Model*: MAEA-based optimal DNN configuration. The 9 input units, properly marked with the corresponding physical quantity, and the 4 hidden layers are shown.

The prediction accuracy of the optimized network is assessed in Fig. 6.18. The pressure fields from the solution of the RANS-TM and the RANS solver with the DNN-based surrogate for turbulence closure (RANS-DNN), as well as the relative error between the two solutions are plotted, on a new car geometry not included in the DB_{DNN}. The maximum error appears close to the spoiler; while discrepancies are also present close to the hand of the mirrors.

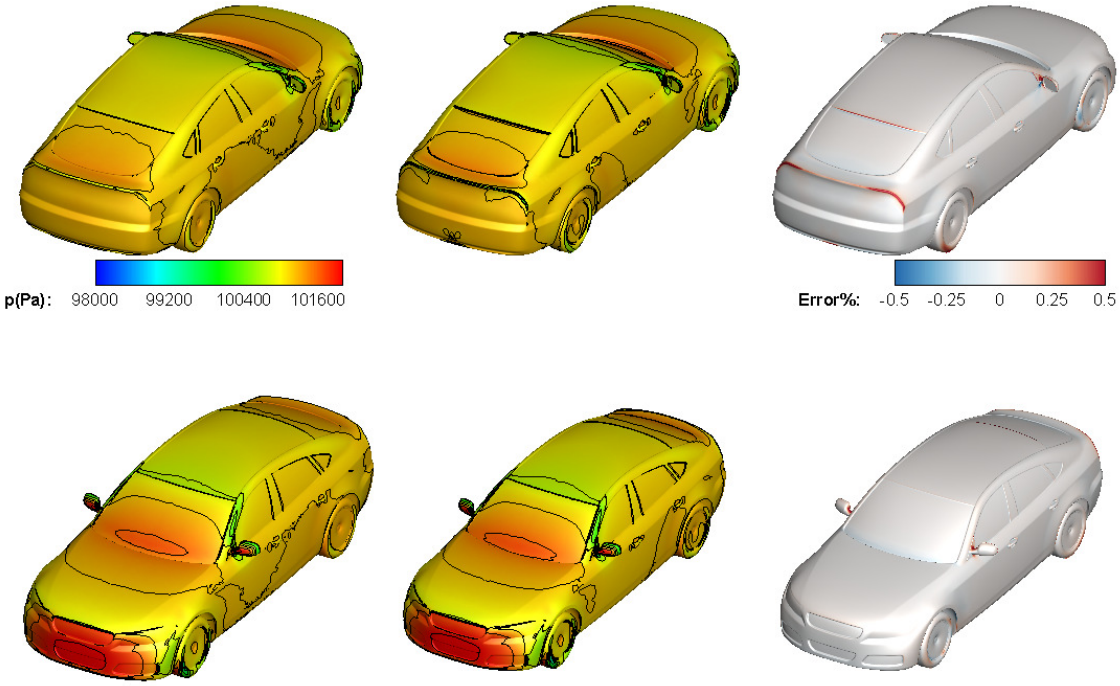


Figure 6.18: *Oprtimization of a DrivAer Car Model*: Pressure field on the car surface computed by the RANS-TM (left) and RANS-DNN (center) models; relative error between the two fields (right). This is a new car geometry, not seen by the DNN before.

The shape optimization for min. C_D follows. Two optimization runs are carried out at the total cost of 200 units; the first relies exclusively on the RANS-TM solver, while the second one at the RANS-DNN. For the latter, 106.5 time units have already been spent to form the DB_{DNN} (100) and train the DNN (6.5). A single run based on the RANS-DNN takes 48 min on an NVIDIA A100 GPU, i.e. the two tools have cost ratio of 0.8. The "best" individual from the DB_{DNN} is included in the initial population of the MAEA in both optimization runs.

The same configuration is used for both optimization runs. A (10,30) MAEA is used, the RBF metamodels are activated after evaluating $T^{MM} = 50$ candidate cars and $\lambda_\epsilon = 2$ members are re-evaluated in each generation on the selected model. In either run, with 96 design variable, the PCA is used both for applying the evolution operators in the feature space and for truncating the metamodel's input units during the LCPE phase.

The convergence histories of the optimizations with evaluations on the RANS-TM and RANS-DNN models are presented in Fig. 6.19. In this application, no stagnation is experienced in the convergence and, thus, there is no need to update the DNN; DB_{DNN} appears to be suitable for the optimization procedure. The optimization with the RANS-TM model results in a car geometry with 0.39% decrease in C_D, while with the RANS-DNN with 0.91% decrease, both at the same computational cost. The normal displacement of the car surface and the pressure distribution over the shapes resulted from both optimizations are presented in Fig. 6.20. Compared to the baseline; the pressure on the rear windshield surface is increased for both optimized geometries, contributing to lower drag force.

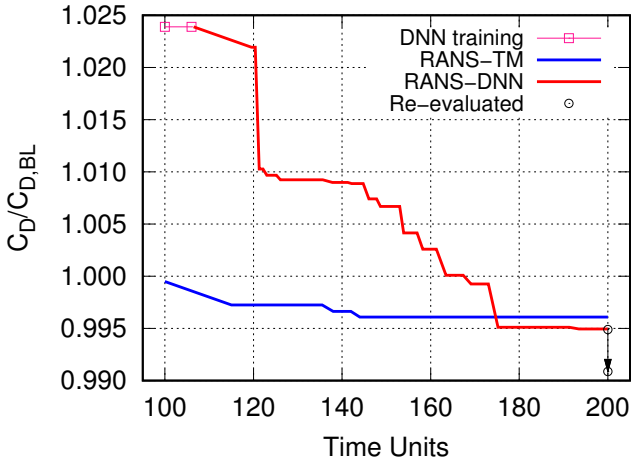


Figure 6.19: *Oprtimization of a DrivAer Car Model*: Convergence histories of the MAEA-based optimizations relying on the RANS-DNN (red) and RANS-TM (blue) models. DNN training of a total cost of 6.5 units is also presented in pink. The optimal solution of the RANS-DNN optimization is re-evaluated on the RANS-TM (empty black circle).

6.6 Conclusions

In this chapter, a DNN-based surrogate for turbulence and transition closure to be used during the numerical solution of the RANS equations was proposed; the role of the DNN was to provide the turbulent viscosity field in each pseudo-time iteration of the solver of the mean flow equations by replacing the turbulence and transition models (Spalart–Allmaras, $k-\omega$ SST and $\gamma-\tilde{Re}_{\theta t}$). The DNN surrogate was supported by an automated procedure based on MAEAs for finding the optimal configuration, i.e. the hyperparameters and the input data, in each case. The resulting RANS-DNN code was, then, used as the evaluation tool, in the framework of a MAEA-based search algorithm, in three aerodynamic shape optimization

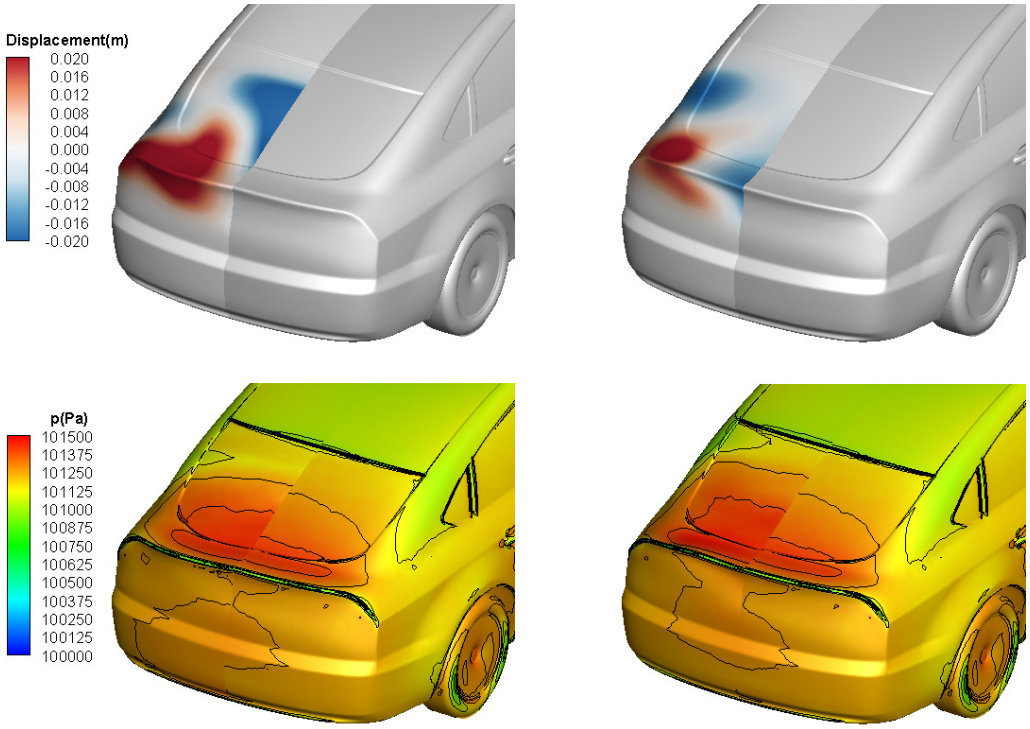


Figure 6.20: *Oprtimization of a DrivAer Car Model*: Normal displacement of the surface and pressure distribution for the optimized geometries (port side) of the RANS-TM (left) and RANS-DNN (right) models (the latter are re-evaluated on the former model) compared with the baseline geometry (starboard side of the car).

problems; an airfoil, a turbomachinery cascade and a car case. Occasionally, retraining the DNN with the best-so-far solution(s), when outliers with respect to the DB_{DNN} were observed or stagnation was experienced in the convergence of the MAEA, was mandatory and boosted the optimization. Indicatively, the use of RANS-DNN reduced the optimization cost, in comparison to that when the RANS-TM model was used, by up to 50%, for reaching a solution of same quality (e.g. the turbine cascade case) and resulted in a better performing solution (e.g. $\times 2$ for the car case) for a given computational cost.

It is clear that the more computationally demanding the application and the turbulent/transition model(s) to be replaced, the bigger the gain from the use of the RANS-DNN model during the optimization. This is due to the fact that the cost of forming the DB_{DNN} and finding/training the optimal DNN configuration cannot practically be depreciated in cases with low cost. Even if the gain in computational cost depends on the application itself, a conclusion can safely be extracted: the RANS-DNN model can replicate the turbulence and/or transition model(s) in CFD runs without jeopardizing the quality of the simulations.

Chapter 7

DNNs in Multi-Disciplinary and Multi-Row Shape Optimization using EAs

In this chapter, the use of DNNs in single-disciplinary optimization problems presented in Chapter 6 is extended to multi-disciplinary and multi-row problems. For the multi-disciplinary problems, the use of the two-branch $\lambda-$DNN architecture is proposed. This consists of two separate input branches connected to the main one towards its output. In multi-disciplinary problems, the $\lambda-$DNN can replicate the numerical solution of one or more disciplines. Herein, it is used in conjugate heat transfer problems to optimize an S-shaped cooling channel and an internally cooled turbine blade surrounded by hot gas. Then, the DNNs are used as a means to reduce the number of domains that need to be simulated in multi-row CFD runs in context of a shape optimization based on stochastic, population-based algorithms. This is demonstrated in the shape optimization of a hydraulic turbine consisting of stay vanes, guide vanes, runner blades and the draft tube, where a DNN-based surrogate to the mixing plane technique is used to avoid the CFD solution into all but the runner blade domains. The EASY, [4], platform developed by PCOpt/NTUA is used for the shape optimizations, as in Chapter 6.

7.1 The $\lambda-$DNN in Multi-Disciplinary Shape Optimization

7.1.1 The Conjugate Heat Transfer (CHT) Problem

In conjugate heat transfer (CHT) analysis problems, the flow equations in the fluid domain Ω^F, extensively described in Sec. 2.1, are coupled with the heat conduction one in the solid domain Ω^S. The heat conduction equation over the

solid domain is expressed as

$$R^S = -\frac{\partial}{\partial x_k}\left(\kappa^S \frac{\partial T^S}{\partial x_k}\right) = 0 \tag{7.1}$$

where κ^S is the thermal conductivity of the solid. The flow and heat conduction solvers communicate across the fluid-solid interface (FSI), along which heat flux conservation and temperature equality must be satisfied

$$\begin{aligned} T^F\big|_{S_{FSI^F}} &= T^S\big|_{S_{FSI^S}} \\ q_k^F \mathsf{n}_k^F\Big|_{S_{FSI^F}} &= q_k^S \mathsf{n}_k^S\Big|_{S_{FSI^S}} \end{aligned} \tag{7.2}$$

where $q_k^S = \kappa^S \frac{\partial T^S}{\partial x_k^S}$ is the heat flux at the solid domain, S_{FSI^F} and S_{FSI^S} corresponds to the fluid and solid side of the FSI boundary and n_k^F, n_k^S are the normal vectors to the FSI pointing toward Ω^F and Ω^S, respectively ($\mathsf{n}_k^F = -\mathsf{n}_k^S$).

The flow and heat conduction equation solvers are loosely coupled by means of Aitken's dynamic relaxation formula. A schematic representation of the coupling between the flow and heat conduction equation solver is illustrated in Fig. 7.1. Temperature fields computed by the heat conduction equation solver in Ω^S after performing a number of internal iterations are scaled according to the Aitken's formula and imposed as boundary conditions to the flow solver. The flow solver performs a number of pseudo-time steps, and the computed heat fluxes on the FSI are communicated back to the heat conduction solver. The stability of the coupling scheme is enhanced by computing and communicating the Jacobian of the computed heat fluxes ($q_k^F \mathsf{n}_k^F$) w.r.t. T^F along the FSI to the heat conduction solver. Heat fluxes along with their Jacobians are computed and communicated at each pseudo-time iteration of the heat conduction solver. The convergence criterion is based on the communicated, along the FSI boundary, data, Eq. 7.2.

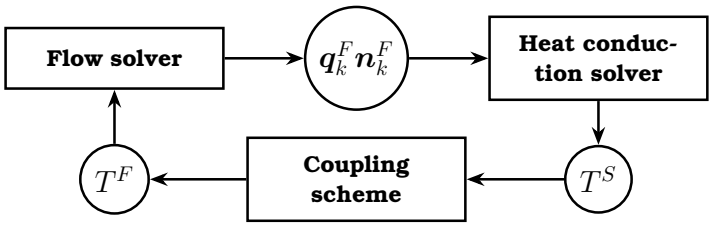


Figure 7.1: Schematic representation of the coupling of the flow and heat conduction solvers.

7.1.2 The $\lambda-$DNN Architecture

An evaluation using a CHT solver can be computationally expensive since it requires many iterative cycles with calls to the flow (for the fluid domain) and the heat conduction solver (for the solid domain). As described in Sec. 7.1.1, each domain is solved separately and the interaction of the solvers takes place at their interface, so a surrogate model trained on the contribution of the one solver to the interface can be very helpful for cost reduction. By doing so, the cost of one of the solvers is significantly reduced (practically negligible), together with the overall cost of the CHT evaluation.

In order for a DNN to act as a surrogate for the solver(s), it should be trained to correlate the body geometry, the heat flux and the temperature fields/distributions, as the (exact) PDEs do. For this purpose, the use of the multi-branch $\lambda-$DNN architecture is proposed. The network architecture named $\lambda-$DNN after its two-branch shape which looks like the Greek letter λ, Fig. 7.2. It is based on fully connected neural networks and consists of two separate branches merged into the main one. Each branch first passes through fully connected layers and then connects to the main branch which further processes signals through its own fully connected layers and concludes to the network output(s). The two branches allow the separate processing of input data with different physical meaning. This leads to extraction of different features for each branch, before merging them to produce the final output. The $\lambda-$DNN architecture is expected to be compact with relatively low numbers of trainable parameters compared to FCNN.

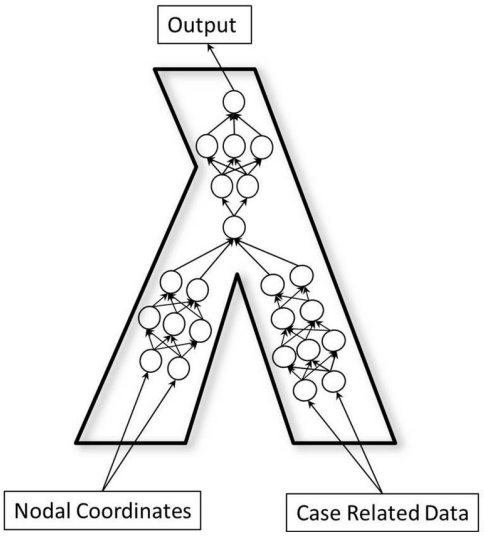


Figure 7.2: The $\lambda-$DNN architecture, named after its shape that looks like the Greek letter λ. For visualization purposes, each circle/layer comprises a number of neurons/layers. The number of inputs/outputs and that of fully connected layers may vary among the cases.

By construction, the $\lambda-$DNN architecture is suitable for CHT (or any other multi-disciplinary) problems as the one branch can handle data related to the geometry (nodal coordinates) and the other one can handle data related to physical quantities computed by the other discipline (case related data). The $\lambda-$DNN can reconstruct either fields or predict distributions of flow quantities over the contour of a body. In the cases studied in this PhD Thesis, the $\lambda-$DNN is used to replace the numerical solution of the heat conduction solver. As a result, the CHT cycle which was initially based on the coupling of the flow and heat conduction solvers, Fig. 7.1, is now based on the coupling of the flow solver and the $\lambda-$DNN, Fig. 7.3. Since the cost of the $\lambda-$DNN prediction is practically negligible compared to that of the heat conduction equation, the cost of a RANS/$\lambda-$DNN simulation is no more than the cost of the solving the fluid domain. The new RANS/$\lambda-$DNN solver is going to be used into an EA-based optimization framework, aiming at reducing the optimization turnaround time. For sake of fairness, the cost of forming the DB_{DNN} on which the $\lambda-$DNN is trained on and the training itself are also included. It is important, though, to keep in mind that, in the examined cases, this gain is sought to be achieved even by replicating the less computationally expensive discipline (i.e. the heat conduction equation solver).

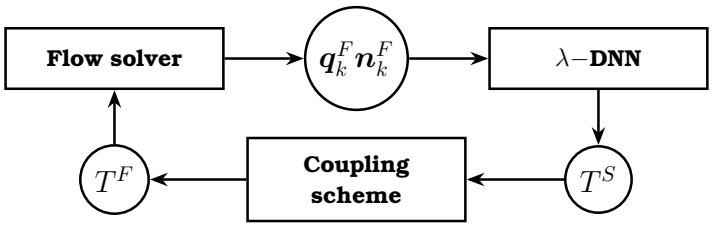


Figure 7.3: Schematic representation of a CHT cycle where the heat conduction equation solver is replaced by the $\lambda-$DNN.

7.1.3 CHT Shape Optimization of an S-Shaped Cooling Channel

The first CHT application is dealing with a high-temperature solid domain cooled by a low-temperature fluid flowing in the adjacent S-shaped cooling channel, Fig. 7.4. The flow conditions for the fluid domain, Ω^F, and the thermal characteristics of the solid domain, Ω^S, are summarized in Table 7.1. Turbulence is modeled by means of the Spalart–Allmaras model. Along the FSI boundary, Ω^F and Ω^S interchange heat flux (from Ω^F to Ω^S) and temperature (in the opposite direction) distributions. A mono-block structured grid of $330K$ nodes is generated; $90K$ of them in the fluid and $240K$ in the solid domain. A single CHT run takes $\sim$20 min on an NVIDIA K40 GPU.

In this CHT problem, the role of the $\lambda-$DNN is to replace the numerical solution of the heat conduction equation in Ω^S. The geometry is encapsulated within

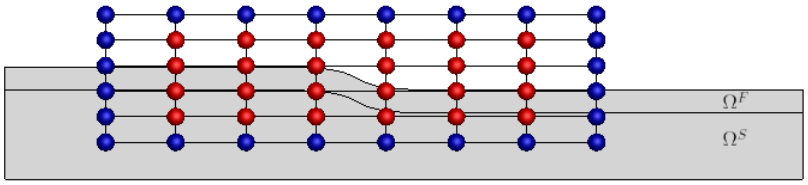


Figure 7.4: *Optimization of an S-Shaped Cooling Channel*: Control points of the volumetric NURBS morphing box, parameterizing the S-bend duct and a small part of the solid domain. Blue points are kept fixed, whereas red ones can be displaced in either direction.

Quantity	**Domain**	**Symbol**	**Value**
Inlet Total Pressure (Pa)	Fluid	p_t^{in}	105000.0
Inlet Total Temperature (K)	Fluid	T_t^{in}	300.0
Outlet Static Pressure (Pa)	Fluid	p^{out}	100000.0
Inlet Flow Angle (°)	Fluid	α^{in}	0.0
Dynamic Viscosity $(10^{-5}\,\text{kg}\,\text{m}^{-1}\,\text{s}^{-1})$	Fluid	μ	1.5
Inlet Viscosity Ratio	Fluid	$(\mu_t/\mu)^{\text{in}}$	20.0
Density (kg/m^3)	Solid	ρ^S	7854.0
Heat Capacity (J kg^{-1} K^{-1})	Solid	C^S	434.0
Thermal Conductivity (W m^{-1} K^{-1})	Solid	κ^S	60.0
Temperature at the Bottom Wall Boundary (K)	Solid	T_{BW}^S	500.0

Table 7.1: *Optimization of an S-Shaped Cooling Channel*: Flow conditions for the fluid Ω^F and the solid Ω^S domain. The rest non-FSI boundaries are considered adiabatic.

the 8×6 volumetric NURBS control lattice of Fig. 7.4; 24 control points are allowed to move within $\pm10\%$ of their reference coordinate values in either direction, resulting in 48 design variables in total and affecting both domains. Then, a DB_{DNN} consisting of 30 geometries evaluated on the exact CHT tool is generated by changing the position of the pre-defined control points of the morphing box through a Latin hypercube sampling; each geometry is associated with a pair of heat flux and temperature fields. Two neural networks with similar numbers of neurons and layers (and, also, trainable parameters) are used; the first is a FCNN architecture and the second one is a $\lambda-$DNN, Table 7.2. The FCNN consists of two layers with 1024 and 512 neurons each. Regarding the $\lambda-$DNN, each of the two branches consists of a single layer with 512 neurons, the two branches merge into a single layer of 512 neurons that leads to the final output. The input data to both networks are the (x, y) coordinates and the heat fluxes at the 600 FSI nodes as computed by the RANS solver running within the iterative decoupled CHT solver, yielding 1800 inputs in total. Regarding the $\lambda-$DNN, the first branch processes the (x, y) coordinates, while the second the heat fluxes. The output is the temper-

ature field (T^S) on the entire solid domain (240000 outputs in total), replacing the outcome of the heat conduction solver. For both networks, the ReLU and the $tanh$ activation functions are used for the hidden and the output layers, respectively. The Adam optimizer, [66], is used for minimizing the networks prediction error.

	Layers	**Neurons/Layer**	**Input Data**	**Act. Functions**
FCNN	2	$1024, 512$	$x_k, q_k \mathsf{n}_k$	ReLU/*tanh*
$\lambda-$DNN				
Left Branch	1	512	x_k	ReLU
Right Branch	1	512	$q_k \mathsf{n}_k$	ReLU
Main Branch	1	512	Left & Right Branch	*tanh*

Table 7.2: *Optimization of an S-Shaped Cooling Channel*: Configuration of the FCNN and the $\lambda-$DNN architectures.

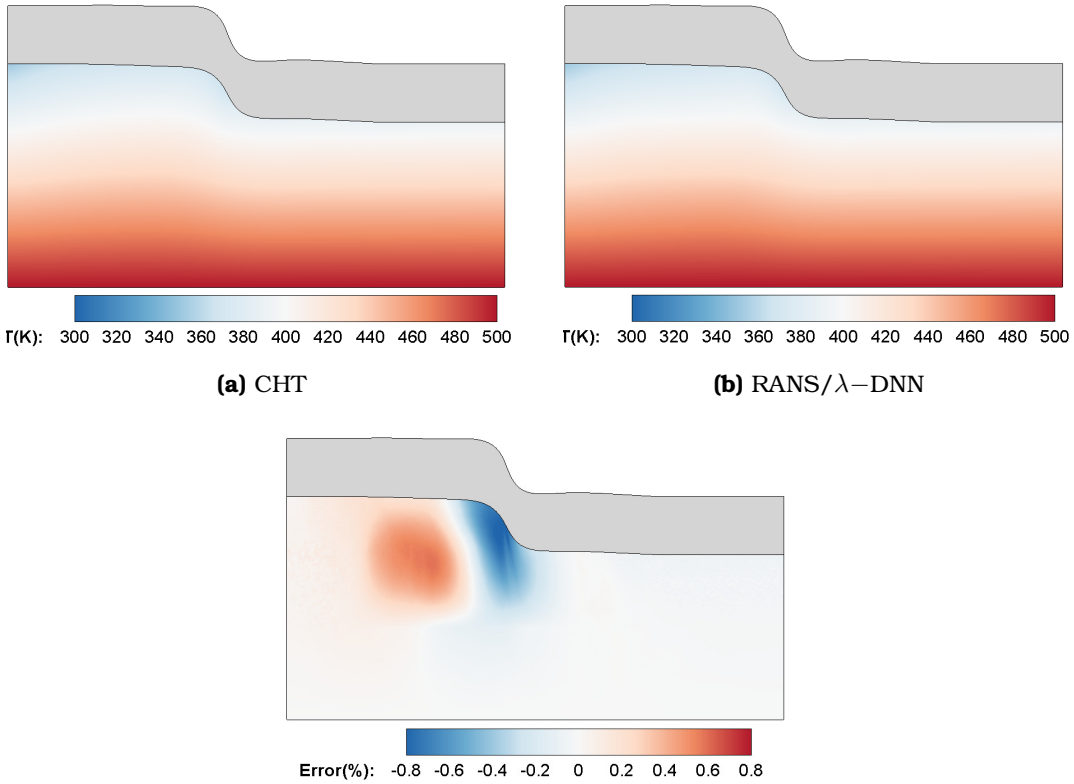
For both networks, the training cost is limited to ~ 6.5 h on an NVIDIA K40 GPU. The prediction quality of the FCNN and the $\lambda-$DNN on two new geometries generated by displacing the control points within the aforementioned bounds, not seen by the DNNs before is investigated. Temperature predictions over Ω^S using the two trained networks are compared with the CHT evaluations; the average error is $\sim 0.8\%$ for the FCNN and $\sim 0.3\%$ for the $\lambda-$DNN, demonstrating the superiority of the latter in multi-disciplinary problems. The temperature field over Ω^S as resulted from an evaluation based on the CHT and the RANS/$\lambda-$DNN solver, as well as their relative error, are presented in Fig. 7.5.

After confirming the suitability of the $\lambda-$DNN for replacing the heat conduction equation solver in CHT analysis, it is then used in optimization loops. Three MAEA-based CHT shape optimization runs are carried out

- Optimization Run 1: Single-Objective Optimization. Target : min. mass-averaged total pressure losses between the inlet and outlet of the S-shaped duct (F_1), Eq. 7.3. Evaluation tool: CHT solver.

- Optimization Run 2: Same target with Optimization Run 1. Evaluation tool: RANS/$\lambda-$DNN solver.

- Optimization Run 3: Two-Objective Optimization. Target : min. mass-averaged total pressure losses between the inlet and outlet of the S-shaped duct (F_1), Eq. 7.3, and min. overheated area within Ω^S (i.e. the percentage of area of Ω^S over which T^S exceed a threshold value (F_2), Eq. 7.4. Evaluation tool: RANS/$\lambda-$DNN solver.



(a) CHT **(b)** RANS/$\lambda-$DNN

(c)

Figure 7.5: *Optimization of an S-Shaped Cooling Channel*: Temperature field at the solid domain Ω^S as computed by (a) the CHT and (b) the RANS/$\lambda-$DNN solver. (c) Relative error between the two fields. This is a new geometry not included in the DB_{DNN}.

$$F_1 = \frac{\int\limits_{S_I} p_t \rho v_k^A \mathsf{n}_k \mathrm{d}S + \int\limits_{S_O} p_t \rho v_k^A \mathsf{n}_k \mathrm{d}S}{\int\limits_{S_I} \rho v_k^A \mathsf{n}_k \mathrm{d}S} \tag{7.3}$$

$$F_2 = \frac{1}{\Omega^S} \int\limits_{\Omega^S} \max\left(T^S - T^S_{\text{thres}}, 0\right) \mathrm{d}\Omega, \quad T^S_{\text{thres}} = 450\,\text{K} \tag{7.4}$$

For the sake of comparison, the same (10,20) MAEA configuration is used in all optimization runs. The MAEA is supported by on-line trained RBF networks with $T^{MM} = 40$ and $\lambda_\epsilon = 3$, irrespective of whether the $\lambda-$DNN is used or not. Run_1 and Run_2 perform the same CHT shape optimization, but with a different evaluation tool, the former is based on the solution of the governing PDEs in both

domains, while the latter uses the $\lambda-$DNN as surrogate for Ω^S. A total budget of 250 CHT evaluations (time units) is decided for both of them. For Run_2, this budgets splits into 30 time units to evaluate the training patterns which form the DB_{DNN}, 24 for training the $\lambda-$DNN and 196 for the optimization; an evaluation on the RANS/$\lambda-$DNN tool costs about 0.76 time units, i.e. the RANS/$\lambda-$DNN is 24% faster compared to the CHT solver. The optimization convergence histories of the MAEAs are presented in Fig. 7.6. The optimal solution of the optimization based on the RANS/$\lambda-$DNN is re-evaluated on the exact CHT solver and the error at F_1 is less than 0.008% verifying the $\lambda-$DNN's accuracy. The optimized, based on the RANS/$\lambda-$DNN solver, geometry and the heat flux distribution along the FSI boundary are compared with the baseline ones in Fig. 7.7. Overall, for the same computational budget of 250 time units, both runs lead to a reduction in F_1 by $\sim 11\%$. It should be mentioned though that, for the run based on the RANS/$\lambda-$DNN, this reduction is achieved earlier by ~ 50 time units.

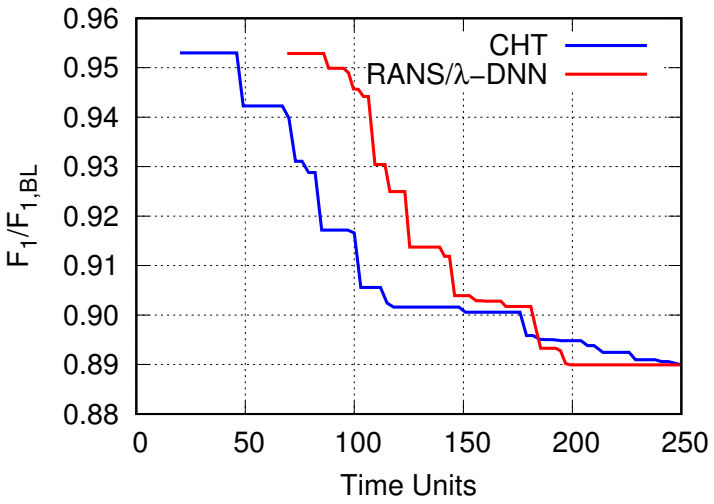


Figure 7.6: *Optimization of an S-Shaped Cooling Channel*: Convergence histories of the MAEA-based optimization relying on the CHT solver of PUMA (blue curve) and the RANS/$\lambda-$DNN solver (red curve). For the latter, the cost of forming the DB_{DNN} and training the $\lambda-$DNN is included.

Then, the two-objective optimization (Run_3) is performed by adding a second target defined over Ω^S. This optimization is most challenging for the $\lambda-$DNN, since F_2 solely relies on the prediction capability of the $\lambda-$DNN upgrading its role. The computational budget is still 250 CHT evaluations distributed as for the single-objective optimization Run_2. The front of non-dominated solutions is presented in Fig. 7.8; each front member is re-evaluated on the exact CHT solver to verify the network accuracy. All re-evaluated members remain non-dominated and the average errors in F_1 and F_2 are $\sim 0.008\%$ and $\sim 0.09\%$, respectively. The pressure and temperature fields in Ω^F and Ω^S for the baseline and the geometries

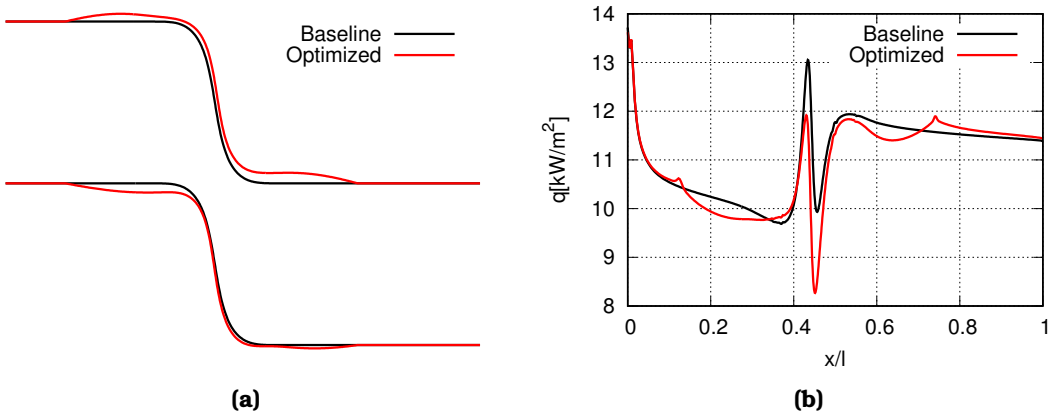


Figure 7.7: *Optimization of an S-Shaped Cooling Channel*: Baseline and optimized solutions using the RANS/$\lambda-$DNN. (a) Side-walls of the fluid domain Ω^F; the lower wall corresponds to the FSI boundary (x and y axes are not in scale). (b) Heat flux distribution along the FSI; abscissa is normalized by the duct's length.

corresponding to the two edges of the front are presented in Fig. 7.9. According to Figs. 7.6 and 7.8, the prediction accuracy of the $\lambda-$DNN is absolutely satisfactory. Therefore, this may readily replicate the heat conduction equation solver in Ω^S and reduce the optimization turnaround time.

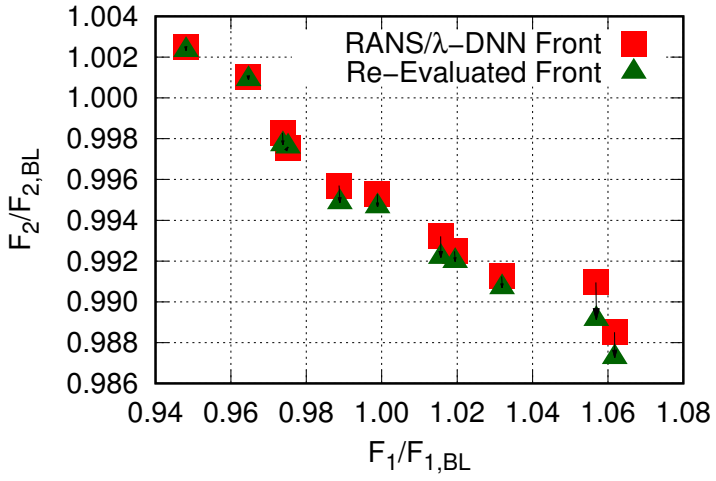
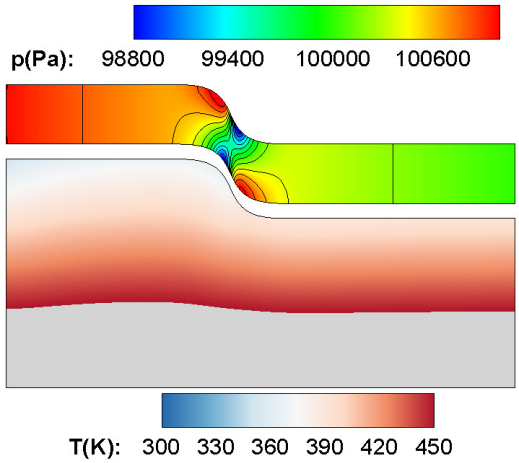
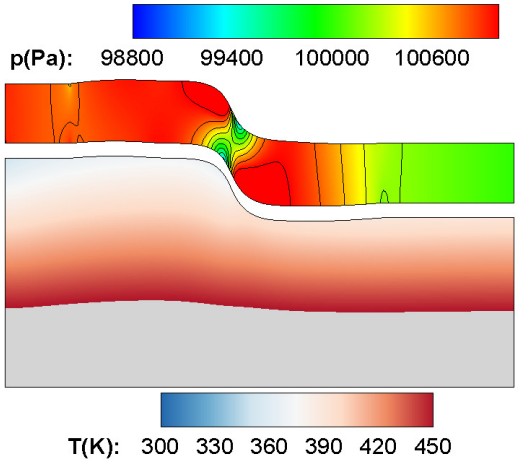


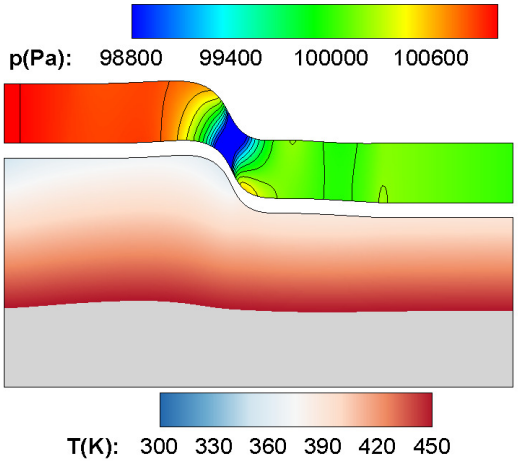
Figure 7.8: *Optimization of an S-Shaped Cooling Channel*: Front of non-dominated solutions from the MAEA computed by RANS/$\lambda-$DNN (filled red squares). The front members are re-evaluated on the CHT solver resulting in a new non-dominated front (green triangles). The computed "optimal" front members remain non-dominated.



(a) Baseline



(b) Optimized geometry with min. F_1



(c) Optimized geometry with min. F_2

Figure 7.9: *Optimization of an S-Shaped Cooling Channel*: Pressure and temperature fields for Ω^F and Ω^S for (a) the baseline and the optimized with (b) min. mass-averaged total pressure losses at Ω^F (F_1) and (c) min. area of Ω^S over which $T^S > 450K$ (F_2). The area with $T^S > 450K$ is presented in gray (axes not in scale).

7.1.4 CHT Shape Optimization of the C3X Internally Cooled Turbine Blade

The second CHT application is dealing with the shape optimization of the C3X internally cooled turbine blade, [52], aiming at minimizing both the mass-averaged total pressure losses in the hot gas and the highest temperature over the solid

blade. The geometry was also optimized in Sec. 4.5.3 for aerodynamic criteria, though. The linear cascade is cooled by 10 radial channels with circular cross sections at fixed locations. Case 158 is studied; the flow conditions of the hot gas, Ω^F, and the thermal characteristics of the solid blade (made of stainless steel), Ω^S, are summarized in Table 7.3. For each cooling channel, experimental data for the average temperature and the coolant flow rate are available; these are also shown in Table 7.3.

Quantity	**Domain**	**Symbol**	**Value**
Inlet Total Pressure (Pa)	Hot Gas	p_t^{in}	243700.0
Inlet Total Temperature (K)	Hot Gas	T_t^{in}	808.0
Outlet Static Pressure (Pa)	Hot Gas	p^{out}	142530.0
Inlet Flow Angle (°)	Hot Gas	α^{in}	0.0
Inlet Viscosity Ratio	Hot Gas	$(\mu_t/\mu)^{\text{in}}$	30.0
Inlet Turbulence Intensity	Hot Gas	$Tu^{\text{in}}(\%)$	8.3
Density (kg/m^3)	Solid Blade	ρ^S	7900.0
Heat Capacity (J kg^{-1} K^{-1})	Solid Blade	C^S	586.15
Thermal Conductivity (W m^{-1} K^{-1})	Solid Blade	κ^S	6.811+0.020716T^S

Hole ID	**Bulk Temperature** (K)	**Flow Rate** (kg s^{-1})	**Diameter** (m)	C_r
1	358.14	0.0167	0.0063	1.118
2	359.37	0.0174	0.0063	1.118
3	349.97	0.0148	0.0063	1.118
4	351.51	0.0165	0.0063	1.118
5	342.56	0.0175	0.0063	1.118
6	371.85	0.0165	0.0063	1.118
7	351.85	0.0161	0.0063	1.118
8	385.96	0.0055	0.0031	1.056
9	413.22	0.00349	0.0031	1.056
10	454.87	0.00171	0.00198	1.025

Table 7.3: *Optimization of the internally cooled C3X turbine blade*: Flow conditions for the hot gas, the solid blade and the 10 cooling channels. Dynamic viscosity is computed based on the Sutherland's law.

When a 2D analysis is performed, the average temperature and coolant flow rate per cooling channel are replaced by heat transfer coefficient values (as the 2D analysis refrains from computing the coolant flow). The heat transfer coefficient (HTC) is a function of the coolant thermal conductivity (κ), the channel diameter (D), and the Nusselt number (Nu_D) (for turbulent flows in smooth pipes) as follows

$$HTC = \frac{\kappa Nu_D}{D} \tag{7.5}$$

$$Nu_D = 0.022 Cr Pr^{0.5} Re_D^{0.8} \tag{7.6}$$

C_r is a correction for a fully developed thermal boundary layer to account for thermal entrance region effects.

A computational grid of $\sim 312K$ nodes is generated; $\sim 183K$ nodes for the hot gas and $\sim 129K$ nodes for the solid blade-airfoil. A single CHT run takes ~ 15 min on an NVIDIA V100 GPU. Prior to the shape optimization, the validation of the CHT solver against experimental data is assessed. Two CHT runs are performed, the first one is based on the $k-\omega\ SST$ turbulence and the second one on the SST-2003-LM2015 transition model. The pressure, temperature and heat transfer coefficient distributions along the blade contour are compared with experimental data in Fig. 7.10. A good agreement is obtained for the pressure distribution. On the other hand, the temperature along the suction side is overestimated when the $k-\omega\ SST$ model is used; the use of a transition model improves the behavior of the CHT solver compared to experimental data.

The CHT optimization of the C3X turbine blade follows for min. mass-averaged total pressure losses between the inlet and outlet of the hot gas, Eq. 7.7 and min. highest temperature over the solid blade, Eq. 7.8. The blade-airfoil is parameterized using the 7×3 volumetric NURBS control lattice of Fig. 7.11. The closest to the leading and trailing edge control points remain fixed, while the rest are free to move in the axial and pitchwise directions, resulting in $19\times 2=38$ design variables in total. The parameterization affects only the airfoil contour, while the positions and diameters of the cooling holes are kept fixed.

$$F_1 = \frac{\int\limits_{S_I} p_t \rho v_k^A \mathsf{n}_k \mathrm{d}S + \int\limits_{S_O} p_t \rho v_k^A \mathsf{n}_k \mathrm{d}S}{\int\limits_{S_I} \rho v_k^A \mathsf{n}_k \mathrm{d}S} \tag{7.7}$$

$$F_2 = \max\left(T_i^S\right), \quad i = 1, \ldots, N^S \tag{7.8}$$

In this application, two different ways of implementing the $\lambda-$DNN are presented. In the first, the $\lambda-$DNN is used as a surrogate for the heat conduction equation solver in Ω^S and communicates with the RANS code providing the temperature distribution along the blade contour during all the CHT cycles. This usage mode is the same as described in the shape optimization of the cooling channel of Sec. 7.1.3. In the second, a similar $\lambda-$DNN is trained to provide the T^S distribution along the interface which though corresponds to converged solutions of the FSI loop. Once this distribution is made available, a single call to the RANS equations solver computes the total pressure losses. Practically, in the second usage mode, the $\lambda-$DNN is trained to replace all the calls to Ω^S solver and all but the last call to the Ω^F solver.

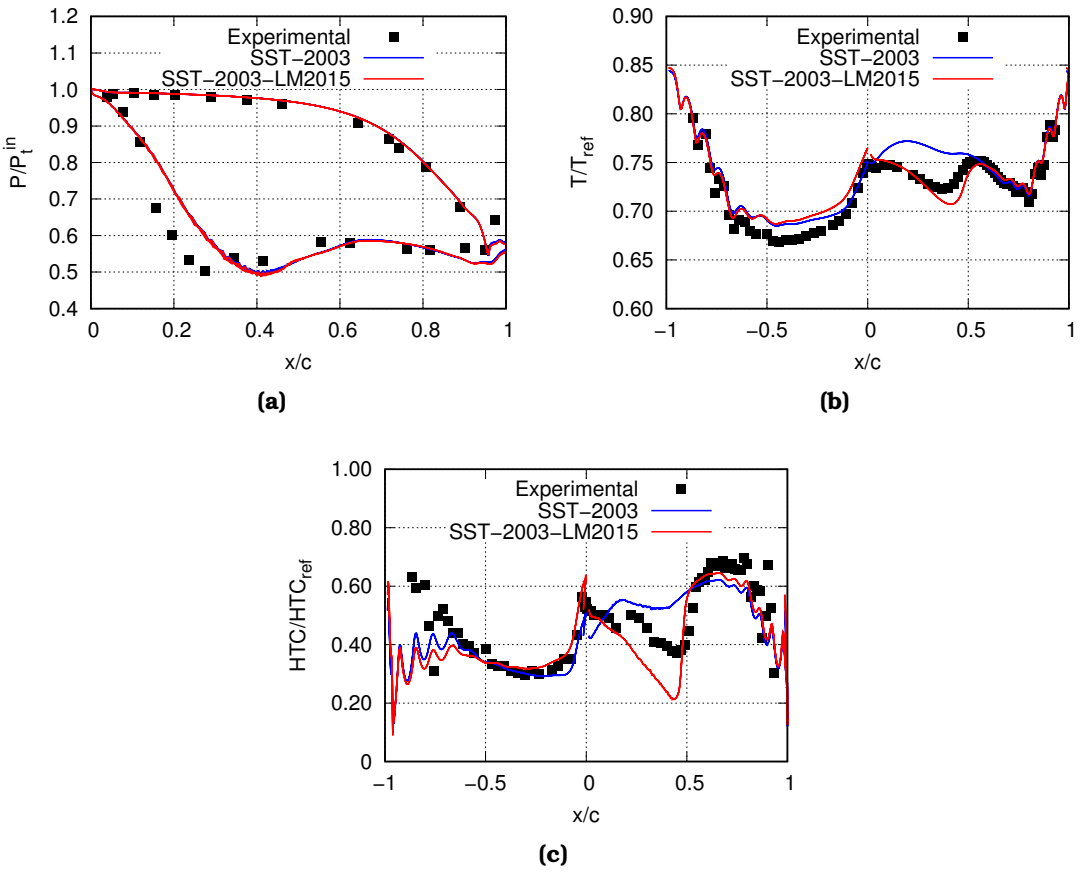


Figure 7.10: *Optimization of the internally cooled C3X turbine blade*: Comparison of the (a) pressure, (b) temperature and (c) heat transfer coefficient distributions across the blade compared with experimental data, [52]. Pressure is non-dimensionalized by the inlet total pressure of $p_t^{in} = 243\,700\,\text{Pa}$, temperature by $T_{ref} = 811\,\text{K}$ and the heat transfer coefficient by $HTC_{ref} = 1135.6\,\text{W/m}^2\text{/K}$.

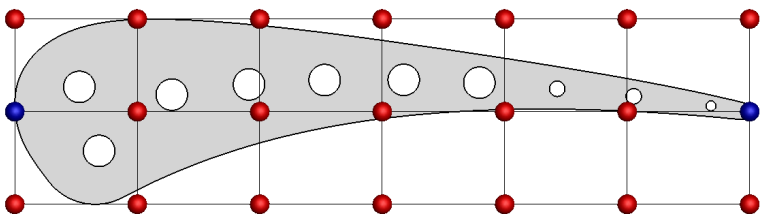


Figure 7.11: *Optimization of the internally cooled C3X turbine blade*: Control points of the volumetric NURBS morphing box, parameterizing the blade-airfoil. Blue control points are kept fixed, whereas red ones may move in the axial and pitchwise directions.

The DB_{DNN} for training the $\lambda-$DNN is built by evaluating 160 blade geometries on the CHT solver generated by moving the control points within $\pm 5\%$ of their reference coordinate values through a Latin hypercube sampling technique. The two input branches of the $\lambda-$DNN consist of a single layer with 64 neurons each; after their merging, a single layer of 128 neurons leads to the final output. The few parameters of the $\lambda-$DNN (number of layers and number of neurons) demonstrate the simplicity of the architecture. For comparison, an FCNN formed by two layers with 128 neurons each is used, too. The blade geometries included in the DB_{DNN} and the network architectures are common in both usage modes, what alters is the network inputs. These are summarized in Table 7.4 and will be further discussed in the description of the optimizations.

	Layers	**Neurons/Layer**	**Input Data**	**Act. Functions**
	First Usage Mode			
FCNN_1	2	$128, 128$	$x_k, q_k \mathsf{n}_k$	ReLU/*tanh*
$\lambda-\text{DNN}_1$				
Left Branch	1	64	x_k	ReLU
Right Branch	1	64	$q_k \mathsf{n}_k$	ReLU
Main Branch	1	128	Left & Right Branch	*tanh*
	Second Usage Mode			
FCNN_2	2	$128, 128$	x_k, Coordinates of CPs	ReLU/*tanh*
$\lambda-\text{DNN}_2$				
Left Branch	1	64	x_k	ReLU
Right Branch	1	64	Coordinates of CPs	ReLU
Main Branch	1	128	Left & Right Branch	*tanh*

Table 7.4: *Optimization of the internally cooled C3X turbine blade*: Configuration of the FCNN and the $\lambda-$DNN architectures for the two usage modes.

First Usage Mode

In the first mode, the $\lambda-$DNN is used as a surrogate for the solver of the blade (heat conduction) in all CHT cycle. The $\lambda-\text{DNN}_1$ and FCNN_1 are trained on the 446 FSI nodes of the pressure and suction side of the airfoil. The input data to the networks are the nodal coordinates and the heat fluxes at the FSI nodes (1338 inputs in total) and the output is the T^S distribution at the same nodes (446 outputs in total). The first branch of the $\lambda-\text{DNN}_1$ processes the (x, y) nodal coordinates, while the second one the heat fluxes. For a fair comparison, the training cost for both networks is limited to ~ 30 min on an NVIDIA V100 GPU. Two new blade geometries, not seen by the networks before, are used to assess the DNNs. The average error values are $\sim 0.18\%$ and $\sim 1.52\%$ for the $\lambda-\text{DNN}_1$ and the FCNN_1, showing the better prediction accuracy of the $\lambda-$DNN architecture.

In addition, in order to evaluate the dependency of the network architecture on the selection of training and validation datasets, a 16-fold cross-validation, [50],

is performed, with 10 pattern in each fold. The mean value of the computed errors is $\sim 0.19\%$ with standard deviation of $\sim 0.024\%$; the mean value is close to the one previously computed with a small range, due to the low standard deviation. Since the DB_{DNN} is based on a Latin hypercube sampling technique, the 16-fold cross-validation results confirm the randomness of the data and the independence of the architecture from the initial selection.

Two (10,20) MAEA optimizations assisted by on-line trained RBF networks ($T^{MM}=40$ and $\lambda_\epsilon=3$) are performed. The first run relies exclusively on the exact CHT solver, while the second one on the RANS/$\lambda-\text{DNN}_1$. During the second optimization run, the $\lambda-\text{DNN}_1$ acts as a surrogate to the heat conduction equation solver of Ω^S, in all CHT cycles, by predicting the T^S distribution over the blade airfoil. Since the highest temperature over the blade area lies along its contour, in contact with the hot gas, the maximum value of the predicted simulations is practically F_2.

A total budget of 250 CHT evaluations is decided for both runs. For the optimization based on the RANS/$\lambda-\text{DNN}_1$, these correspond to 160 time units to form the DB_{DNN}, 2 to train the network and 88 for the optimization itself; an evaluation on the RANS/$\lambda-\text{DNN}_1$ is about 0.90 time units. During the optimization, some candidate solutions from the fronts of non-dominated solutions are re-evaluated with the exact CHT solver, the DB_{DNN} is enriched and the $\lambda-\text{DNN}_1$ updated. Candidates too close to an already evaluated individual are not re-evaluated, since their available objective function values are considered to be dependable. The re-training cost is significantly less than that of the initial training (~ 0.1 time units). The cost for the re-evaluations is included in the 88 time units.

The fronts of non-dominated solutions resulted from the two optimization runs are presented in Fig. 7.12; the members of the run based on the RANS/$\lambda-\text{DNN}_1$ are re-evaluated on the exact CHT tool, and the errors in F_1 and F_2 are $\sim 0.0017\%$ and $\sim 0.10\%$, respectively. At the same computational cost, the front resulted from the optimization based on RANS/$\lambda-\text{DNN}_1$ dominates the one based on the CHT solver, highlighting that this usage of the $\lambda-$DNN improves the efficiency of the (already very fast) MAEA. In Figs. 7.13, 7.14 and 7.15, the optimized geometries compared with the baseline, the Mach number field and the error in the total pressure field of the three optimized solutions of the RANS/$\lambda-\text{DNN}_1$ run are presented. It is clear that the reduction in total pressure drop is mostly achieved by reducing the flow velocity in the passage throat region (**A** geometry), while for the **C** geometry, the reduction at the temperature is achieved by moving the pressure side closer to the cooling holes.

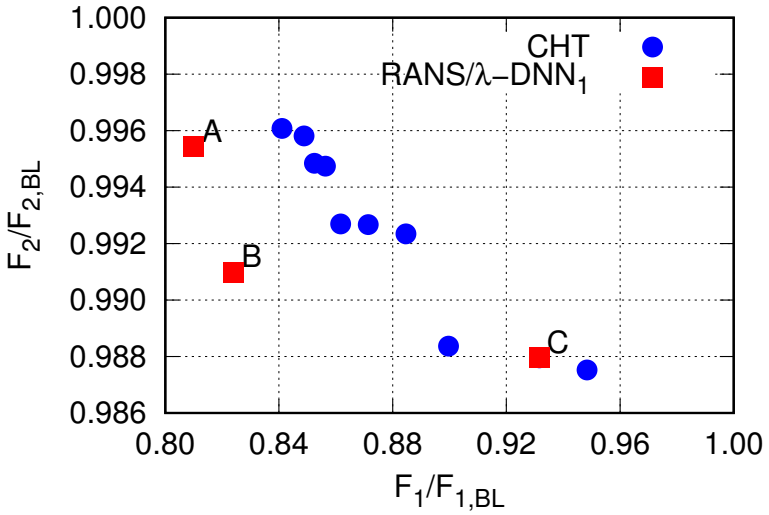


Figure 7.12: *Optimization of the internally cooled C3X turbine blade*: Front of non-dominated solutions as resulted from the optimization based on the CHT and the RANS/λ−DNN$_1$ solver.

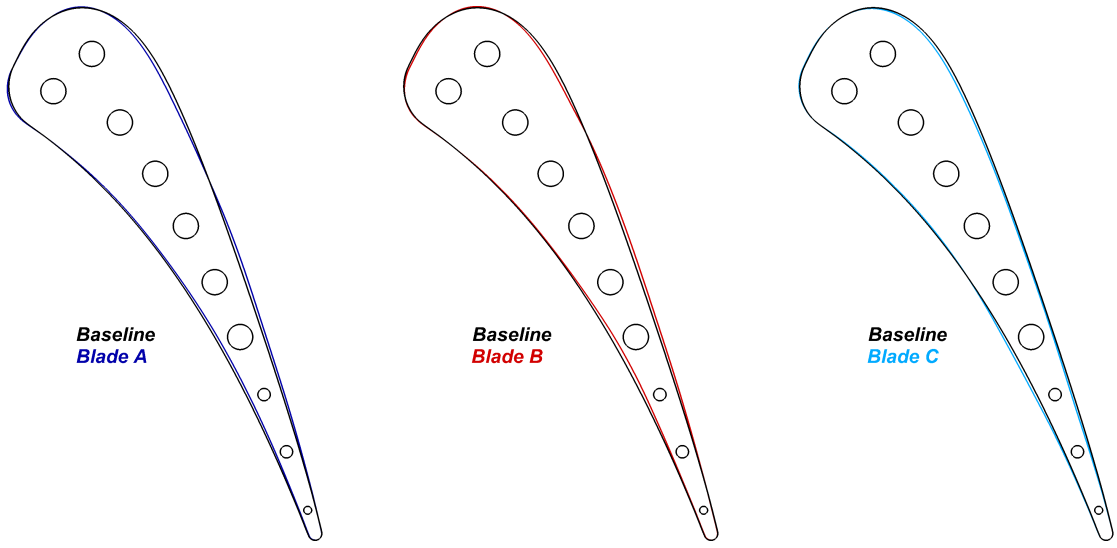
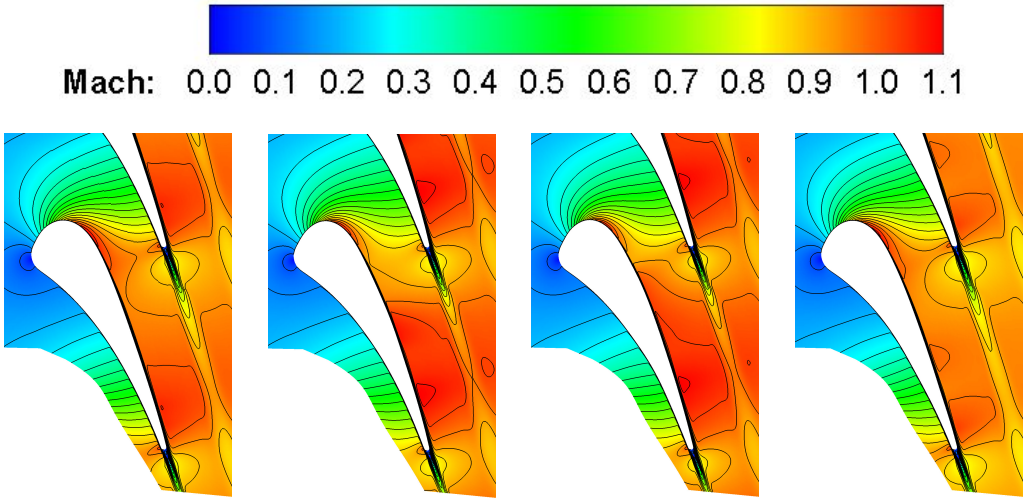
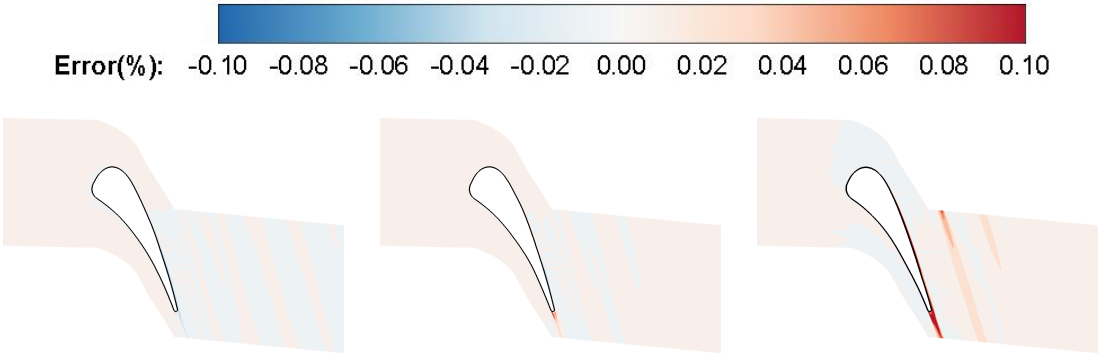


Figure 7.13: *Optimization of the internally cooled C3X turbine blade*: Baseline and optimized blade-airfoil geometries corresponding to members of the front of Fig. 7.12, resulted from the optimization based on the λ−DNN$_1$ solver.



(a) Baseline **(b)** Blade A **(c)** Blade B **(d)** Blade C

Figure 7.14: *Optimization of the internally cooled C3X turbine blade*: Mach number fields for the baseline and three members of the front of Fig. 7.12, resulted from the optimization based on the $\lambda-$DNN$_1$ solver.



(a) Blade A **(b)** Blade B **(c)** Blade C

Figure 7.15: *Optimization of the internally cooled C3X turbine blade*: Error in the total pressure fields for three members of the front of Fig. 7.12, resulted from the optimization based on the $\lambda-$DNN$_1$ solver.

Second Usage Mode

In this usage mode, the $\lambda-$DNN provides the final/converged temperature distribution at the blade-airfoil contour without performing CHT cycles, i.e. without iterating with the RANS solver for Ω^F. The RANS solver, having the final temperature distribution, is called once to compute the total pressure losses of the hot gas. The architecture used is the same as in the previous approach. The first branch processes the 446 (x, y) FSI nodal coordinates, while the second the 38 coordinates of the free-to-move control points. The T^S distribution along the blade-airfoil contour is the network's output. The $\lambda-$DNN$_2$ and the FCNN$_2$ are trained on the same blade airfoils (same DB$_{\text{DNN}}$), though by using different data per blade airfoil and with the same cost as in the previous approach. The average error values on two new geometries are $\sim 0.12\%$ and $\sim 1.24\%$ for $\lambda-$DNN$_2$ and the FCNN$_2$.

A third MAEA-based optimization run is carried out using as evaluation tool the RANS/$\lambda-$DNN$_2$ solver; the MAEA configuration is kept the same. The front of non-dominated solutions is compared with that resulted from the optimization based on the exact CHT solver in Fig. 7.16; all plotted members are re-evaluated on CHT. Once again, the front of the run assisted by the $\lambda-$DNN$_2$ prevails on that resulted from the CHT solver. Moreover, it is important to notice that the optimization based on the $\lambda-$DNN$_2$ converged to this front at $\sim 73\%$ of the total budget and the errors at the re-evaluated members are $\sim 0.05\%$ for F_1 and $\sim 0.12\%$ for F_2. The fact that the optimization run based on $\lambda-$DNN$_2$ converges to a front dominating that of the run based on CHT and this is achieved with less computational cost, demonstrates the effectiveness and efficiency of the $\lambda-$DNN architecture within a multi-disciplinary optimization. The optimized geometries, the Mach number field and the error in the total pressure field are compared with the baseline one for the two edges of the front in Figs. 7.17, 7.18 and 7.19.

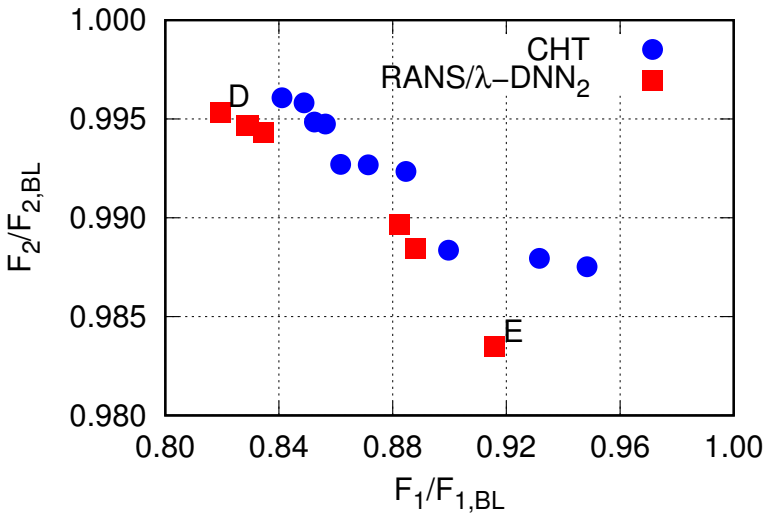


Figure 7.16: *Optimization of the internally cooled C3X turbine blade*: Front of non-dominated solutions as resulted from the optimization based on the CHT and the RANS/$\lambda-$DNN$_2$ solver.

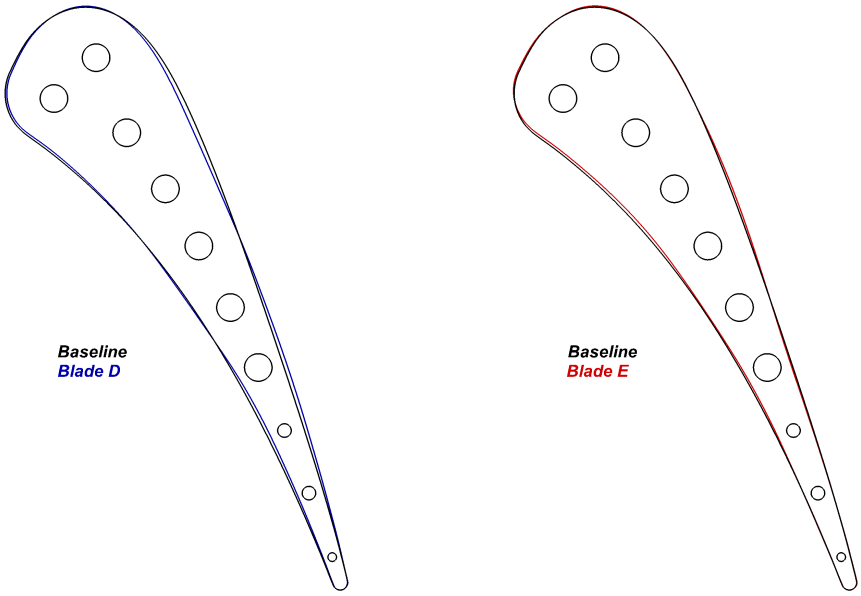
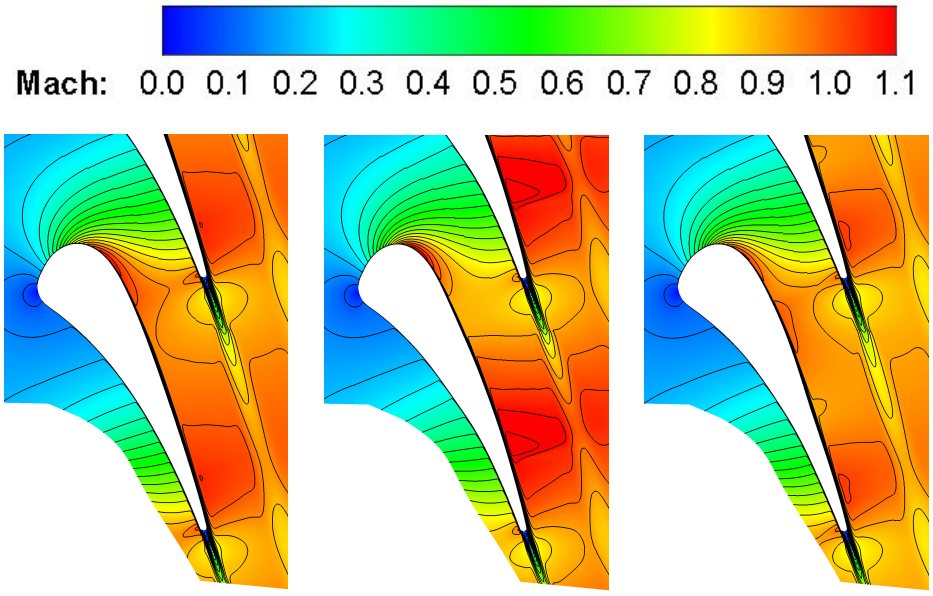
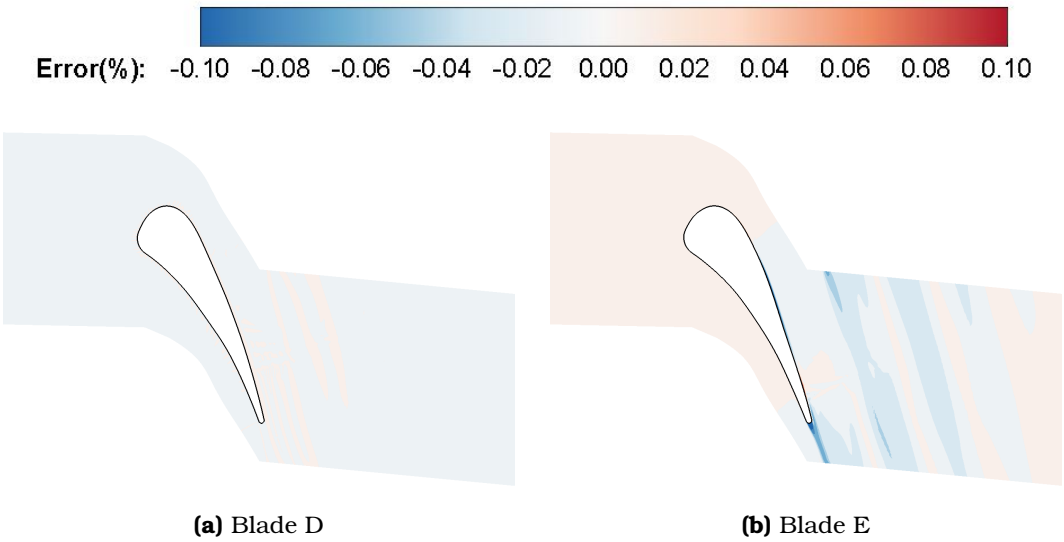


Figure 7.17: *Optimization of the internally cooled C3X turbine blade*: Baseline and the two extreme solutions selected from the front of non-dominated solutions of Fig. 7.16, resulted from the optimization based on the $\lambda-$DNN$_2$ solver.



(a) Baseline **(b)** Blade D **(c)** Blade E

Figure 7.18: *Optimization of the internally cooled C3X turbine blade*: Mach number fields for the baseline and the two extreme solutions selected from the front of non-dominated solutions of Fig. 7.16, resulted from the optimization based on the $\lambda-\text{DNN}_2$ solver.



(a) Blade D **(b)** Blade E

Figure 7.19: *Optimization of the internally cooled C3X turbine blade*: Error in the total pressure field, for the two extreme solutions from the front of non-dominated solutions of Fig. 7.16, resulted from the optimization based on the $\lambda-\text{DNN}_2$ solver.

7.2 DNNs in Multi-row Shape Optimization Problems

In this section, the shape optimization of a hydraulic turbine consisting of 21 stay vanes, 21 guide vanes, 9 runner blades and the draft tube, Fig. 7.20, is carried out. This case was studied in the REGALE project (An Open Architecture to Equip Next Generation HPC Applications with Exascale Capabilities) co-funded by the General Secreteriat for Research and Innovation of the Ministry of Development and Investments under the EuroHPC JU, EU Grant Agreement No. 956560. Geometry and flow conditions were provided by Andritz Hydro. The optimization aims at reducing the amplitude of the pressure pulsations between the runner and the guide vanes while also ensuring that no cavitation occurs in the runner. DNNs are used as a mean to reduce the number of domains that need to be simulated in this multi-row run in a MAEA-based shape optimization.

The mixing plane technique is used to model guide vanes-runner and runner-draft tube interactions when running the CFD code through the entire turbine (this code will be referred to as the CFD^{F}). The idea is to train and use DNN-based surrogates to the mixing plane technique, before and after the only domain to be solved by the CFD tool (that of the runner), to overcome the simultaneous CFD solution into the domains of stay and guide vanes as well as the draft tube. The new model will be abbreviated to CFD^{R} as it simulates the flow by the CFD code only in the runner (=R) domain; the two DNNs are indispensable parts of CFD^{R}.

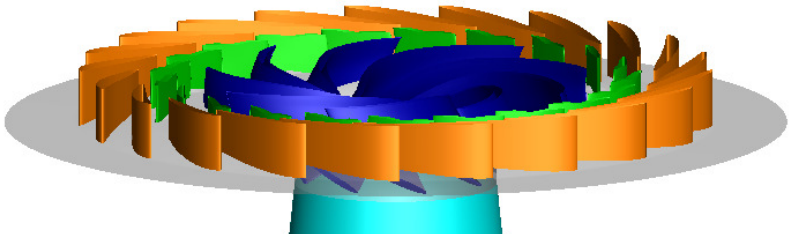


Figure 7.20: *Optimization of a Hydraulic Turbine*: Perspective view of the baseline turbine geometry with the stay vanes (orange), guide vanes (green), runner blades (blue) and (just the very first part of) the draft tube (ice blue).

7.2.1 The Mixing Plane Technique for Incompressible Flows

For the flow analysis, the incompressible variant of the flow solver described in Sec. 2.1.1 is used. For completeness sake, the Navier–Stokes equations are presented, herein, for incompressible fluid flows. The steady residuals of the flow

equations read

$$R_n^{\text{MF}} = \underbrace{\frac{\partial f_{nk}^{\text{inv}}}{\partial x_k}}_{\text{MF}^{inv}} - \underbrace{\frac{\partial f_{nk}^{\text{vis}}}{\partial x_k}}_{\text{MF}^{vis}} + \underbrace{S_n}_{\text{MF}^{s}} = 0 \tag{7.9}$$

where $U_n = \begin{bmatrix} p & v_1^A & v_2^A & v_3^A \end{bmatrix}^T$ stands for the flow variables with p being the kinematic pressure and v_k^A the absolute Cartesian velocity components. The inviscid fluxes f_{nk}^{inv}, viscous fluxes f_{nk}^{vis} and source terms S_n are written as

$$f_{nk}^{\text{inv}} = \begin{bmatrix} \beta^2 v_k^R \\ v_k^R v_1^A + p\delta_{1k} \\ v_k^R v_2^A + p\delta_{2k} \\ v_k^R v_3^A + p\delta_{3k} \end{bmatrix} \quad f_{nk}^{\text{vis}} = \begin{bmatrix} 0 \\ \tau_{1k} \\ \tau_{2k} \\ \tau_{3k} \end{bmatrix} \quad S_n = \begin{bmatrix} 0 \\ \rho\varepsilon_{1\ell k}\omega_\ell v_k^A \\ \rho\varepsilon_{2\ell k}\omega_\ell v_k^A \\ \rho\varepsilon_{3\ell k}\omega_\ell v_k^A \end{bmatrix} \tag{7.10}$$

where β is a parameter corresponding to a (constant) artificial speed of sound. The relative velocity components v_m^R are linked to the absolute ones v_m^A through the equation $v_m^A = v_m^R + v_m^F$, with $v_m^F = \varepsilon_{m\ell k}\omega_k \left(x_k - x_k^C\right)$ being the rotating/non-inertial frame velocity and x_k^C the position vector of the center of rotation. In Eqs. 7.10 τ_{km} is the viscous stress tensor for a Newtonian fluid defined by:

$$\tau_{km} = \frac{\nu + \nu_t}{\text{Re}_0}\left(\frac{\partial v_k^A}{\partial x_m} + \frac{\partial v_m^A}{\partial x_k}\right) \tag{7.11}$$

where ν and ν_t stand for the kinematic and the turbulent viscosities, respectively. Eqs. 7.9 are coupled with the Spalart–Allmaras turbulence model.

The rotor-stator interaction (RSI) is modeled using the mixing plane technique according to which the spanwise distribution of circumferentially averaged flow quantities is iteratively exchanged between the adjacent domains. Exchanged quantities are the mixed-out flow variables $\hat{V} = \begin{bmatrix} \hat{p} & \hat{v}_r & \hat{v}_\theta & \hat{v}_a & \hat{\tilde{\nu}} \end{bmatrix}^T$ computed based on the area-averaged fluxes crossing the iso-span RSI zones, [143]. v_r, v_θ, v_a stand for the radial, peripheral and axial components of the absolute velocity vector and the hat (ˆ) symbol indicates circumferentially averaged quantities. $\tilde{\nu}$ is the turbulence model variable. The area-averaged fluxes crossing each RSI zone are computed leading to the corresponding mixed-out flow variables, $\hat{\mathbf{V}}$, to be communicated to the other side at the same span. Upon convergence of the flow equations, the numerical fluxes crossing the interface are conserved.

7.2.2 DNN-assisted Shape Optimization of a Hydraulic Turbine

The optimization aims at minimizing the difference of the max. and min. pressure, Eq. 7.12, at a point (probe) located between the guide vanes and the runner (at a specified radial and axial position). The temporal variation of this point can

easily be seen as a spatial distribution along the corresponding circumference. Constraints that (a) ensure that no cavitation occurs on the runner blade, and (b) retain the torque (T) produced by the turbine are imposed, Eqs. 7.13 and 7.14. The turbine is considered cavitation free when the min. pressure computed over the runner blade is higher than that on the runner of the baseline geometry.

$$F_1 = [P_{\text{max}} - P_{\text{min}}]_{Probe} \tag{7.12}$$

$$F_2 = P^R_{\text{min,BL}} - P^R_{\text{min}} \leq 0 \tag{7.13}$$

$$F_3 = T^{BL} - T \leq 0 \tag{7.14}$$

Given that all the aforementioned quantities of interest (objective and constraints) result from flow data computed in the runner domain, the idea is to simulate the flow only in the runner than in the entire turbine. As a result, the cost of a flow simulation is reduced, reducing at the same time the overall optimization turnaround time. To do so, boundary conditions that replicate the presence of the stators (stay and guide vanes) and the draft tube must be imposed at the inlet and outlet of the runner domain. Since the RSI is modeled via the mixing plane technique (schematically illustrated in Fig. 7.21), two DNN-based surrogates of the RSI are used to predict the distributions of the flow quantities communicated between the adjacent domains at these interfaces. In specific, the two DNNs undertake the prediction of the axial distribution of the runner inlet flow variables (DNN_1) and the radial one of the runner exit flow variables (DNN_2), Fig. 7.22.

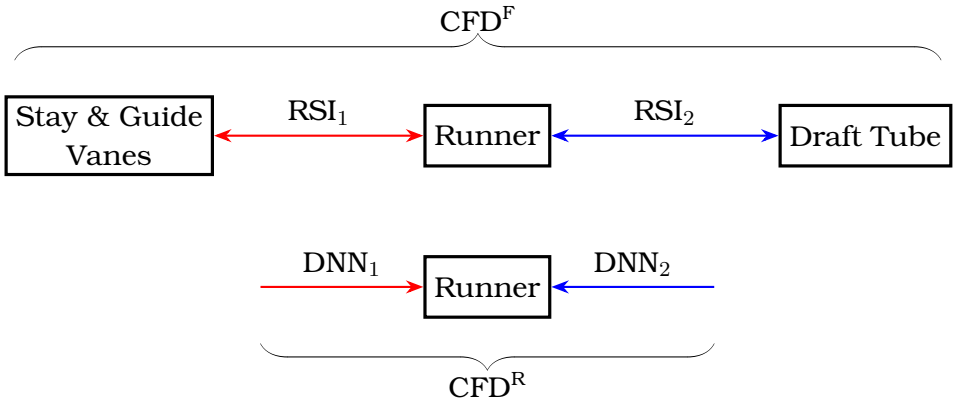


Figure 7.21: *Optimization of a Hydraulic Turbine*: Schematic representation of the exchange of information (numerical fluxes) between adjacent rotating and stationary domains at the two interfaces (RSI) in the case of CFD^{F} (top), and CFD^{R} supported by the DNNs (bottom).

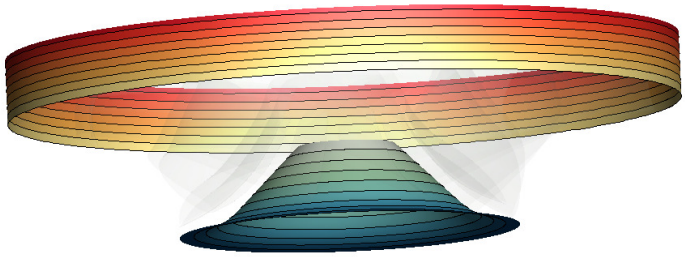


Figure 7.22: *Optimization of a Hydraulic Turbine*: The radial inflow and axial outflow boundaries of the runner domain. The two interfaces of the runner with the guide vanes (top) and the draft tube (bottom) along with the zones used for averaging the flow quantities.

Prior to the creation of the two DNNs that compromise the CFD^R tool to be used during the optimization, the parameterization of the turbine is required. The $11\times3\times5$ NURBS control lattice of Fig. 7.23 is used to parameterize the runner blade. The coordinates of some of these control points are selected as the design variables. In specific, 9 out of the 11 series of control points in the streamwise direction are allowed to vary in both the streamwise and pitchwise directions giving rise to 180 design variables in total.

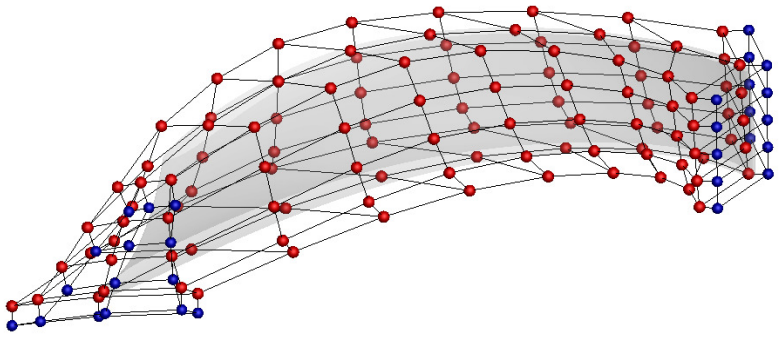


Figure 7.23: *Optimization of a Hydraulic Turbine*: The runner blade enclosed into the NURBS control lattice. Red points are allowed to vary during the optimization while blue ones remain constant.

The first step to be taken is to sample the runner's blade design space and generate the DB_{DNN} to be used for the DNNs training. For each value set of design variables, corresponding to a different runner geometry, the full turbine is simulated using the CFD^F software. The large number of design variables (180) makes the sampling procedure challenging. A small number of samples which though ensures that the so-generated DB_{DNN} is suitable for the DNN training would be ideal. The Latin hypercube sampling technique is used and three DB_{DNN}

sizes consisting of 50, 75 and 100 samples are generated. These will be denoted as $\text{DB}^{50}_{\text{DNN}}$, $\text{DB}^{75}_{\text{DNN}}$ and $\text{DB}^{100}_{\text{DNN}}$, respectively.

The DNN training uses the coordinates of the NURBS' lattice control points that are allowed to vary as inputs and predicts the distributions of the flow quantities (velocity vector and pressure) at the RSI, to be used as the inlet and outlet conditions for the runner domain. FCNN are considered; the optimal configuration of each network results from a MAEA-based optimization procedure, as described in Sec. 6.2.1.

The optimal DNNs configurations are summarized in Table 7.5. In all cases, the optimization selected the GELU activation function for all the hidden layers and the sigmoid one for the output layer. Though the number of layers is not affected by the number of training patterns, this is not the case for the number of neurons per layer. Nevertheless, the total number of the trainable parameters of each DNN are comparable. For the DNN_1 these are about $20M$, while for the DNN_2 about $5M$. This means that the DNN hyperparameters could have been optimized only once (for any DB_{DNN} size), followed by an independent training for each DB_{DNN} size, since the number of trainable parameters is, more or less, the same for all optimal configurations. The use of three different DNN architectures is decided to ensure a fair comparison between the performances of the DNNs during the shape optimization of the hydraulic turbine.

Position	DB_{DNN}	Layers	Neurons/Layer	Parameters
Inlet (DNN_1)	50	7	$4096, 4096, 1024, 64, 128, 256, 1024$	22M
	75	6	$2048, 4096, 1024, 4096, 256, 512$	19M
	100	7	$128, 64, 4096, 4096, 256, 1024, 64$	22M
Outlet (DNN_2)	50	4	$2048, 1024, 256, 1024$	3M
	75	3	$1024, 1024, 1024$	5M
	100	4	$2048, 2048, 32, 256$	4.5M

Table 7.5: *Optimization of a Hydraulic Turbine*: Optimal configurations for DNN_1 (inlet) and DNN_2 (outlet).

The velocities and pressure distributions at the RSI zones of the turbine as predicted by the DNNs trained on the $\text{DB}^{50}_{\text{DNN}}$ are compared with those resulted from a CFD^{F} simulation, in Fig. 7.24. The comparison refers to the baseline geometry which is not included in the $\text{DB}^{50}_{\text{DNN}}$, The agreement is absolutely satisfactory; small discrepancies exist in some of the zones of the axial velocity profile at which values are overestimated by the DNN. The flow simulation at the runner domain using either CFD^{F} or the DNN (predicted) distributions yield similar objective function values; a percentage error of about 1.1%, with the DNN overestimating the objective function value is observed. Overall, the DNNs are considered reliable for use together with the CFD^{R}, in the shape optimization.

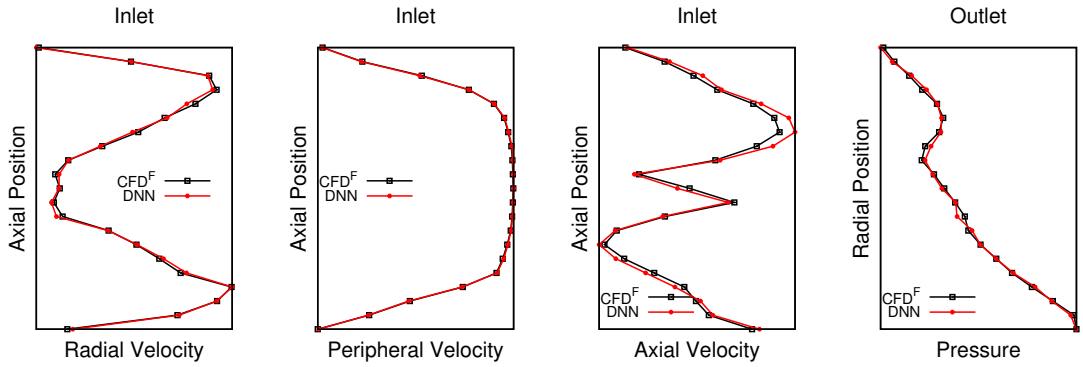


Figure 7.24: *Optimization of a Hydraulic Turbine*: Comparison of the radial, peripheral, axial velocity and pressure (from left to right) between the CFD^{F} (black) and the DNN prediction (red).

The shape optimization of the hydraulic turbine using the proposed DNN-based surrogates for the mixing plane follows. For the sake of comparison, an additional run simulating the flow in the entire turbine domain (using the CFD^{F}) is also presented. The target is to suppress pressure pulsations occurring between the guide vanes and the runner by additionally retaining the torque produced by the turbine and ensuring that no cavitation occurs on the runner blade. The objective is quantified by the amplitude of the pressure field defined at a constant radius between the guide vane-runner interface and the runner leading edge. To avoid cavitation, the optimization should ensure that the min. pressure over the optimal runner blade exceeds that of the non-cavitated runner of the baseline geometry; this is a relative, rather than an absolute cavitation criterion.

A $(\mu, \lambda) = (10, 20)$ MAEA is used, the metamodels are activated after the evaluations of at least $T^{MM} = 50$ geometries and $\lambda_\epsilon \in [2, 4]$, provided that at least 30 of them meet the constraints. The PCA is activated after the 3^{rd} generation and the metamodels are trained using the 45 first principal components, pinpointed by the PCA of the current offspring population.

The computational cost of an evaluation based on CFD^{F} (the entire turbine) is ~ 15 min on an NVIDIA A100 GPU (one time unit). The CFD^{R} simulation that solves only the runner domain using the DNN-based boundary conditions takes ~ 9 min on the same GPU, i.e. the two tools have a cost ratio of 0.6. The computational budget for any optimization run is set to 150 time units; this includes the cost for creating the DB_{DNN}, configuring and training the DNNs. The overall cost of the optimization of the hyperparameters of both DNNs, including the training itself, sums up to no more than 2 time units. The convergence histories of all optimization runs as well as a close up view to the optimal (re-evaluated on CFD^{F}) solutions are presented in Fig. 7.25. The optimization on the CFD^{F} decreased the objective function by $\sim 61\%$. Regarding the runs based on CFD^{R} which makes use of $\text{DB}^{50}_{\text{DNN}}$, $\text{DB}^{75}_{\text{DNN}}$ and $\text{DB}^{100}_{\text{DNN}}$ a reduction of $\sim 68\%$, $\sim 65\%$ and $\sim 61\%$ is obtained

(after re-evaluations on the CFD^F model), respectively.

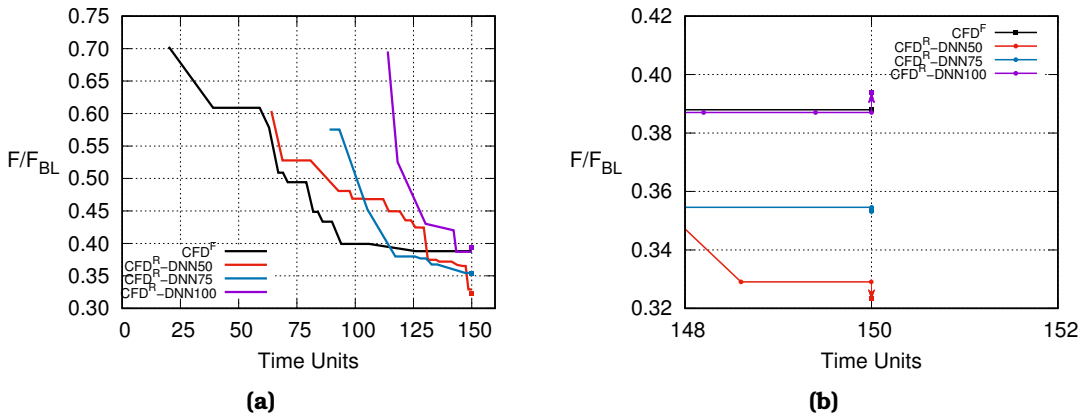


Figure 7.25: *Optimization of a Hydraulic Turbine*: (a) Convergence histories of the MAEA-based optimizations; CFD^F (black), CFD^R with DB^{50}_{DNN} (red), DB^{75}_{DNN} (blue) and DB^{100}_{DNN} (purple). Time units from 1 to the beginning of the colored lines correspond to the evaluation of training patterns, the training of DNNs as well as the cost of the very first MAEA generation. (b) Close-up view of the end of the optimization in which the optimized solutions are re-evaluated on the CFD^F (filled squares).

The (non-dimensionalized) objective and constraint values of the optimized solutions are summarized in Table 7.6. It can be seen that the optimizations relying on CFD^R with DB^{50}_{DNN} and DB^{75}_{DNN} outperform the CFD^F-based one. With the decided computational budget, the run with DB^{100}_{DNN} does not have enough time units left for the optimization, given that 102 time units have already been spent for populating the DB_{DNN} and training the DNNs. Regarding the cavitation constraint (second row of Table 7.6), the minimum pressure on the runner blades resulted from all optimization runs is higher compared to that of the baseline geometry, i.e. the cavitation risk is less. As far as the torque constraint is concerned, CFD^R with DB^{75}_{DNN} and DB^{100}_{DNN} yield more or less the same torque with the baseline geometry, while CFD^R with DB^{50}_{DNN} yields more torque. In all cases studied with the CFD^R the objective function value is computed with adequate accuracy; the percentage errors can be found after re-evaluations on the CFD^R are 1.7%, 0.3% and 1.1% for the 50, 75 and 100 training patterns, respectively. The very small error values confirm that the use of an optimization algorithm to select the DNNs' hyperparameters is absolutely useful.

A comparison of the circumferential pressure distribution, used for computing the objective function, between the baseline and the optimized from the CFD^R model with DB^{50}_{DNN}, is presented in Fig. 7.26. A comparison of the pressure on a constant radius surface used for the definition of the objective function between the baseline and optimized geometries is shown in Fig. 7.27. Apart from the amplitude itself, the pressure values are also decreased in the optimized geometry.

	Optimization on CFD^F	Optimizations on CFD^R		
		DB^{50}_{DNN}	DB^{75}_{DNN}	DB^{100}_{DNN}
Objective	0.388	**0.323**	0.353	0.394
Min. pressure	1.01	**1.24**	1.14	1.14
Torque	1.039	**1.012**	0.994	1.007

Table 7.6: *Optimization of a Hydraulic Turbine*: Comparison of the optimized (within the pre-decided computational budget) solutions resulted from the optimizations on CFD^F and CFD^R (the tabulated values of the latter have been computed by means of CFD^F re-evaluations). All quantities are non-dimensionalized by the corresponding values of the baseline geometry. The "min. pressure" is a measure for possible cavitation; this should exceed 1. Figures in bold highlight the best values obtained from the CFD^R-based runs.

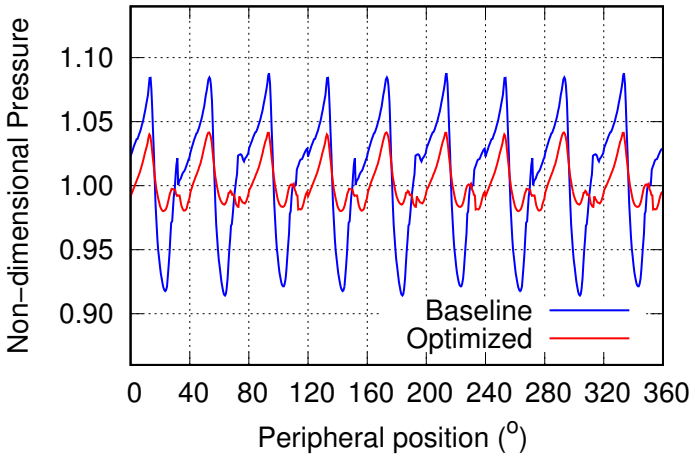


Figure 7.26: *Optimization of a Hydraulic Turbine*: Comparison of the circumferential pressure distribution on the baseline (blue) and the optimized by CFD^R with DB^{50}_{DNN} (red) geometries. Pressure values are non-dimensionalized with the mean pressure value of the baseline geometry.

7.3 Conclusions

In this chapter, DNNs were used to reduce the cost of multi-disciplinary and multi-row optimization problems by means of evolutionary algorithms. Regarding the CHT problem, the use of the $\lambda-$DNN was proposed for the multi-objective CHT shape optimization of an S-shaped cooling channel and an internally cooled turbine blade-airfoil. For the former, the optimization aimed at min. mass-averaged total pressure losses between the inlet and the outlet and min. overheated area within the solid domain, while for the latter, the turbine was optimized for min. mass-averaged total pressure losses and min. highest temperature over the solid

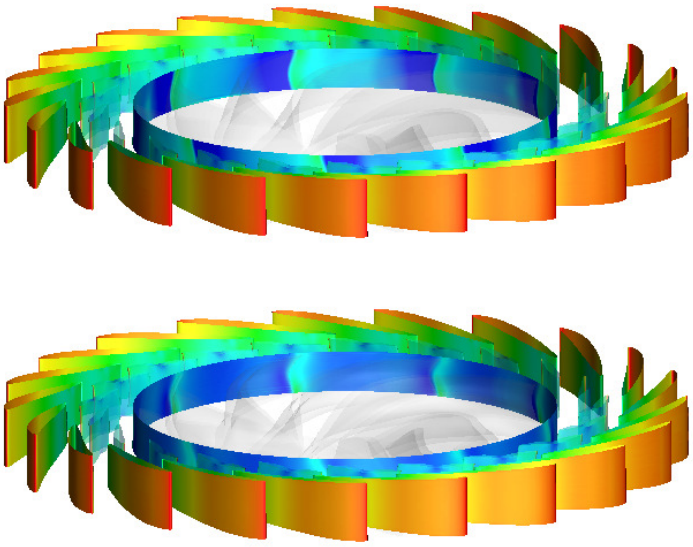


Figure 7.27: *Optimization of a Hydraulic Turbine*: Pressure fields on the surface at a constant radius used for the definition of the objective function for the baseline (top) and the optimized (bottom) geometries.

blade. It was demonstrated that the $\lambda-$DNN is capable of replicating the heat conduction equation solver during a MAEA optimization as the resulted fronts of non-dominated solutions managed to prevail over those of an optimization based on the exact CHT solver in terms of computational cost.

Regarding the multi-row problem, a DNN-based surrogate to the mixing plane technique was used and the gain in computational cost was showcased in the shape optimization of a hydraulic turbine. The DNNs were used to "simulate" the presence of the stationary domains namely the stay vanes, the guide vanes and the draft tube. The optimization studies revealed that the DNN-based surrogate is a reliable tool and when used in the context of an optimization algorithm may lead to an optimized solution of better quality, for a given computational cost.

Chapter 8

Closure

8.1 Summary-Conclusions

The overall target of this PhD Thesis was to further extend and assess computational tools for aero/hydrodynamic shape optimization problems, based on CFD methods occasionally combined with Machine Learning techniques. These include (a) the development of the continuous adjoint for transitional flows of compressible fluids, (b) the development of consistent discretization schemes for the adjoint equations, as these appear in continuous adjoint methods and (c) the development and use of DNNs in flow predictions. Depending on their extra capabilities, the above were used in gradient-based and evolutionary algorithm (EA)-based shape optimization problems; in all examined cases, target was to perform a cost-efficient shape optimization.

This PhD Thesis relies on the in–house GPU-enabled flow and adjoint solver PUMA of PCOpt/NTUA. All tools developed were incorporated into the PUMA code; software programming was performed in CUDA C/C++ and, for the DNNs, the TensorFlow C++ API was utilized. The DNN training was performed in Python.

The $\gamma - \tilde{R}e_{\theta t}$ transition model was implemented in the PUMA solver so as to extend its capabilities for capturing and simulating transitional flows. The original model was coupled with the $k-\omega$ SST turbulence model and is denoted as SST-2003-LM2015. Then, various modifications were proposed to its source terms so as to interact with the Spalart–Allmaras turbulence model resulting in the SA-noft2-Gamma-Retheta, SA-LM2015 and SA-sLM2015 models. In total, the aforementioned four model variants were programmed. Modifications to the Spalart–Allmaras and $k-\omega$ SST turbulence models were also necessary so as to interact with the transition models. The so-implemented transition models were successfully verified and/or validated against experimental data and numerical results in a wide range of applications in internal and external aerodynamics, subsonic and transonic flows, both 2D and 3D. The results from all transition model

variants were in good agreement with the experimental data. An interesting finding related to the numerical robustness of the transition model variants is the improved convergence of the SA-sLM2015 model due to the smooth approximations that replaced the min./max. operators and the conditional statements that are present at the rest models. This is an important finding and the development of the continuous adjoint proved to also benefit of it.

Then, the continuous adjoint method for transitional flows was mathematically developed, programmed and verified for the three SA-based transition model variants, the SA-noft2-Gamma-Retheta, the SA-LM2015 and the SA-sLM2015, for the first time in the corresponding literature. The FI adjoint approach, in the terminology introduced in [62] (in incompressible flows though), in which the gradients of the objective function are expressed, among other, in terms of field integrals of grid sensitivities, was followed. In order to accurately derive the Field Adjoint Equations and the Adjoint Boundary Conditions, variations from the turbulence and transition models as well as from the distance from the wall were taken into account. As demonstrated in isolated airfoil cases, the developed method and the programmed software was capable of reproducing the reference Sensitivity Derivatives (i.e. those computed by Finite Differences). The most important message, though, was that, in this kind of problems, making the "frozen transition" assumption should be avoided. Recall that "frozen transition" refers to the omission of formulating and solving the adjoint to the transition model adjoint equations. In the examined cases, it was shown that this omission leads to erroneous gradients, despite the small reduction in computational cost and, of course, the great reduction in investment cost for developing, programming and debugging the corresponding adjoint method. The continuous adjoint method that takes into account transitional effects, was incorporated into an optimization loop for the constrained optimization of an airfoil with the three transition model variants; all runs improved the aerodynamic behavior of the airfoil, meeting also the imposed constraints. The optimization convergence of the SA-sLM2015 transition model is noteworthy, as the outcome of this run outperformed those of the other models. This is due to the numerical robustness of the SA-sLM2015 model that was reflected on the optimization procedure.

The continuous adjoint method results in the Field Adjoint Equations and the Adjoint Boundary Conditions; these should be discretized and numerically solved. When working with continuous adjoint, the discretization schemes to be selected (occasionally decided intuitively) can significantly affect the accuracy of the computed Sensitivity Derivatives as they are not necessarily consistent with the discretized primal problem. Thus, different discretization schemes would result in different Sensitivity Derivatives. On the other hand, when working with discrete adjoint, consistency is not an issue as the formulation begins with the discretized flow problem. Its main flaw, though, is the possibly excessive memory footprint of the adjoint code. For the first time in the corresponding literature,

this PhD Thesis bridged the gap between the discrete and continuous adjoint variants under the so–called *Think-Discrete Do-Continuous* (*TDDC*) adjoint. Based on the *TDDC* adjoint, consistent discretization schemes for the continuous adjoint equations, with a clear physical meaning, were proposed; these were inspired by the discrete adjoint equations which were necessarily derived first. The accuracy of the *TDDC* on the computation of the Sensitivity Derivatives was compared with Finite Differences and with "standard" discretization schemes widely used in the literature, in a wide number of applications. It was demonstrated that the *TDDC* adjoint is capable of computing Sensitivity Derivatives with at least 6 significant digit accuracy regardless of the fineness of the computational grid (as the discrete adjoint) with a clear physical insight and low memory footprint (as the continuous adjoint).

Overall, the PUMA code, extended with the continuous adjoint method for transitional flows and consistent discretization schemes was applied in a number of industrial applications. These included the shape optimization of an isolated wing and that of a high aspect-ratio wing business jet configuration. It was concluded that the PUMA code with the upgraded adjoint capabilities is a reliable and computationally affordable tool to be used in real-world optimization problems.

DNN-based techniques were also developed to reduce the cost in computationally expensive analysis or optimization tasks assisted by EAs. Initially, DNNs were used for turbulence and transition closure; their role was to replace the numerical solution of the corresponding PDEs and predict the turbulent viscosity field in each pseudo-time iteration of the RANS solver. In order to increase the reliability of the DNN, its configuration, i.e. architecture and input (geometrical and flow) data, resulted from a Metamodel-Assisted EA (MAEA) optimization. Three representative aerodynamic cases from 2D to 3D, internal to external and compressible to incompressible flows were selected to demonstrate the capabilities of the new, less computationally expensive solver; an isolated airfoil, a turbomachinery cascade and a car model were included. Re-training the DNN when outliers or stagnation was observed proved to be of paramount importance for the optimization convergence. Overall, a gain of up to 50% was obtained in terms of computational cost. The dual role of the MAEA should be noticed, as it was used not only to carry out the optimization runs, but also to define the optimal DNN configuration. For the sake of fairness, in all cases, the cost for selecting and evaluating the training patterns on which the DNN was trained and define/train the configuration was included in the overall cost.

Another use of DNNs was in multi-disciplinary shape optimization problems and, in specific, in Conjugate Heat Transfer (CHT) ones. This action involved a new DNN architecture, called $\lambda-$DNN due to its shape that looks like the Greek letter λ. Its distinguishing feature is the separate processing of input data of different nature, i.e. geometrical and case data, ensuring high accuracy and low computational cost. The $\lambda-$DNN was used in two CHT shape optimization problems,

that of an S-shaped cooling duct and that of an internally cooled turbine-blade airfoil. The role of the $\lambda-$DNN was to replace the heat conduction equation solver of the solid domain during the CHT cycles. At the same computational cost, the MAEAs that relied on the new, cost-effective evaluation tool resulted in fronts of non-dominated solutions that prevailed those of the exact CHT solver.

A last application of DNNs was related to the shape optimization of a hydraulic turbine. In this multi-row problem, the DNN acted as a surrogate to the mixing plane technique, applied at the interface of rotating and stationary flow domains, decreasing the number of domains to be simulated. During a MAEA-based shape optimization, it was shown that for the same computational cost, the use of the DNNs results in an optimal solution of better quality (compared to the exact solver) maintaining high accuracy. Overall, it was concluded that DNNs can be used to support stochastic, population-based algorithms, such as EAs, for the design of optimal shape in computational mechanics.

8.2 Novel Contributions

The novel contributions of this PhD Thesis are summarized below:

- Mathematical formulation, programming and assessment of the continuous adjoint method for transitional flows of compressible fluids. To the author knowledge, it is the first time in the corresponding literature that the continuous adjoint method is developed by taking transition from laminar-to-turbulent flow into account.

- Development of consistent discretization schemes for the continuous adjoint PDEs and their boundary conditions, under the so–called *Think-Discrete Do Continuous* (*TDDC*) adjoint. Discretization schemes for the adjoint inviscid and viscous fluxes, as well as for the adjoint turbulent and transition models were proposed. This is actually the first attempt in the corresponding literature that bridges the gap between the continuous and discrete adjoint variants.

- The capabilities of the PUMA code regarding turbulence and transition modeling were upgraded as DNNs can alternatively be used.

- The development of a fully automated procedure based on MAEAs in order to define the optimal DNN configuration. This stands not only for the DNN architecture but also for the input to the DNN data.

- The use of the $\lambda-$DNN architecture was extended to multi-disciplinary shape optimization problems.

- The use of DNNs as surrogates to the mixing plane technique in turbomachinery shape optimization problems.

8.3 Publications & Conference Presentations

The work carried out during this PhD Thesis resulted in publications in scientific journals and presentations in international conferences. In specific,

Journal Publications:

- V. Asouti, M. Kontou, K. Giannakoglou, "Radial Basis Function surrogates for uncertainty quantification and aerodynamic shape optimization under uncertainties". Fluids 2023; 8(11).
- M.G. Kontou, V.G. Asouti, K.C. Giannakoglou, "DNN surrogates for turbulence closure in CFD-based shape optimization". Applied Soft Computing 2023; 143:110013.
- X. Trompoukis, K. Tsiakas, V. Asouti, M. Kontou, K. Giannakoglou, "Optimization of an internally cooled turbine blade - Mathematical development and application". International Journal of Turbomachinery, Propulsion and Power 2021; 6(2):20.
- M. Kontou, D. Kapsoulis, I. Baklagis, X. Trompoukis, K.C. Giannakoglou, "$\lambda-$DNNs and their implementation in conjugate heat transfer shape optimization". Neural Computing and Applications 2022; 34:843-854.

Conference Papers:

- M.G. Kontou, X.S. Trompoukis, V.G. Asouti, K.C. Giannakoglou. "On the discretization of the continuous adjoint to the Euler equations in shape optimization". ADMOS 2023, International Conference on Adaptive Modeling and Simulation, Gothenburg, Sweden, June 19-21, 2023.
- K.G. Kovani, M.G. Kontou, V.G. Asouti, K.C. Giannakoglou. "DNN-driven gradient-based shape optimization in fluid mechanics". EANN 2023, International Conference on Engineering Applications of Neural Networks, Leon, Spain, June 14-17, 2023.
- K.C. Giannakoglou, V.G. Asouti, E.M. Papoutsis-Kiachagias, N. Galanos, M.G. Kontou, X.S. Trompoukis. "The Think-Discrete Do-Continuous adjoint in aerodynamic shape optimization". EUROGEN 2023, 15th International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control, Chania, Greece, June 1-3, 2023.
- M.G. Kontou, V.G. Asouti, X.S. Trompoukis, K.C. Giannakoglou. "DNN-assisted EA-based shape optimization of a hydraulic turbine". EUROGEN 2023, 15th International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control, Chania, Greece, June 1-3, 2023.

- X.S. Trompoukis, K.T. Tsiakas, V.G. Asouti, M.G. Kontou and K.C. Giannakoglou. "Continuous adjoint shape optimization of internally cooled turbine blade". 14th European Conference on Turbomachinery Fluid Dynamics & Thermodynamics (ETC14), Gdansk, Poland, April 12-16, 2021.

- M. Kontou, D. Kapsoulis, I. Baklagis and K. Giannakoglou, "$\lambda-$DNNs and their implementation in aerodynamic and conjugate heat transfer optimization". 21st Engineering Applications of Neural Networks (EANN) Conference, Halkidiki, Greece, June 5-7, 2020.

Invited Lectures:

- K.C. Giannakoglou, V.G. Asouti, M.G. Kontou and X.S. Trompoukis. "Low-cost evolutionary optimization, with engineering applications". von Karman Institute Lectures Series on Optimization Methods for Computational Fluid Dynamics, May 16-20, 2022.

Awards:

- **Best Student Paper Award:** M. Kontou, D. Kapsoulis, I. Baklagis and K. Giannakoglou, "$\lambda-$DNNs and their implementation in aerodynamic and conjugate heat transfer optimization". 21st Engineering Applications of Neural Networks (EANN) Conference, Halkidiki, Greece, June 5-7, 2020.

8.4 Future Work

Some ideas and suggestions for future work are given below:

- Extension of the continuous adjoint method for transitional flows to the SST-2003-LM2015 transition model. Compressibility corrections [112] (both to the primal and the adjoint solver) could also be implemented into the PUMA code for transition modeling.

- Extension of the *TDDC* adjoint to all the discretization schemes and features of the PUMA code which were not used in the framework of this PhD Thesis. Regarding the discretization of the inviscid fluxes, the *TDDC* could also be developed and programmed for the Barth and Jespersen [11] and the Venkatakrishnan [140] limiters. For turbulence modeling, the extension of the *TDDC* adjoint to the $k-\omega$ SST turbulence model is suggested; in addition, consistent discretization schemes for the wall functions that can be used both with the Spalart–Allmaras and the $k-\omega$ SST models can be proposed. As far as the additional features of the PUMA code are concerned, the *TDDC* adjoint can also be formulated for the rotor-stator-interaction through the mixing or sliding plane technique.

- The experience gained for compressible fluid flows can be extended to incompressible ones. This can be done either with the low-Mach number preconditioning technique on the equations of a compressible fluid solver or with the incompressible variant of the PUMA code (based on pseudo-compressibility technique). The ultimate goal is to make the *TDDC* adjoint globally used in the PUMA code.

- Regarding turbulence closure, DNNs could be trained on high fidelity simulation data (LES or DNS) and then incorporate this information to the RANS equations so as to reproduce the outcome of the former. By doing so, the accuracy of the LES simulations could be achieved by solving the less computationally demanding RANS equations.

- A different approach regarding the DNNs is their use in gradient–based shape optimization problems. DNNs, trained to predict the objective/constraint function(s) of the optimization, could be differentiated to deliver the required derivatives w.r.t. the design variables. Consequently, DNNs could be used in gradient–based optimization problems by replacing, for instance, the adjoint code. The use of higher order derivatives should also be considered. Ongoing research on this topic has shown encouraging results, see [67].

- The $\lambda-$DNN architecture could be used for other real–world multi-disciplinary optimization problems, such as aerostructural or aeroelastic ones. Results from its use in other disciplines could generalize those extracted in this PhD Thesis regarding aerothermal optimization.

Appendix A

Differentiation of the Absolute Jacobian Matrix

The formulation of the *TDDC* adjoint for the inviscid terms, Sec. 4.1.1, requires the computation of the derivative of the absolute Jacobian w.r.t. the primitive flow variables, $V=[\rho\ v_1\ v_2\ v_3\ p]^T$. For the Roe scheme, since all involved quantities are Roe-averaged, the chain rule is used. In specific, these derivatives are computed as the derivatives of the absolute Jacobian w.r.t. the Roe-averaged quantities, $\tilde{V}=[\tilde{\rho}\ \tilde{v}_1\ \tilde{v}_2\ \tilde{v}_3\ \tilde{h}_t]^T$ times the derivatives of the Roe-averaged quantities w.r.t. the primitive variables.

The absolute Jacobian matrix $|\tilde{A}^{LR}_{nmk}\mathsf{n}_k|$ is diagonalized as $|\tilde{A}^{LR}_{nmk}\mathsf{n}_k|=\tilde{P}_{n\ell}|\tilde{\Lambda}_{\ell r}|\tilde{P}^{-1}_{rm}$ where $\tilde{\Lambda}$ is a diagonal matrix containing the absolute eigenvalues of the Jacobian matrix computed using the Roe-averaged quantities between the L and R states. Matrices P, P^{-1} include the right (column-wise) and left (row-wise) eigenvectors of the Jacobian matrix. Thus, $|\tilde{\Lambda}|$ stands for $diag\left(|\tilde{\lambda}_1|,|\tilde{\lambda}_2|,|\tilde{\lambda}_3|,|\tilde{\lambda}_4|,|\tilde{\lambda}_5|\right)$ with

$$\begin{aligned}\tilde{\lambda}_{1,2,3}&=\vec{\tilde{v}}\cdot\vec{\mathsf{n}}\\ \tilde{\lambda}_{4,5}&=\vec{\tilde{v}}\cdot\vec{\mathsf{n}}\pm+\tilde{c}|\vec{\mathsf{n}}|\end{aligned}$$

and

$$\tilde{P}=\begin{bmatrix}\hat{\mathsf{n}}_1 & \hat{\mathsf{n}}_2 & \hat{\mathsf{n}}_3 & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}} & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\\ \tilde{v}_1\hat{\mathsf{n}}_1 & \tilde{v}_1\hat{\mathsf{n}}_2-\tilde{\rho}\mathsf{n}_3 & \tilde{v}_1\hat{\mathsf{n}}_3+\tilde{\rho}\mathsf{n}_2 & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{v}_1+\tilde{\lambda}_{15}\hat{\mathsf{n}}_1\right) & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{v}_1+\tilde{\lambda}_{14}\hat{\mathsf{n}}_1\right)\\ \tilde{v}_2\hat{\mathsf{n}}_1+\tilde{\rho}\hat{\mathsf{n}}_3 & \tilde{v}_2\hat{\mathsf{n}}_2 & \tilde{v}_2\hat{\mathsf{n}}_3-\tilde{\rho}\hat{\mathsf{n}}_1 & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{v}_2+\tilde{\lambda}_{15}\hat{\mathsf{n}}_2\right) & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{v}_2+\tilde{\lambda}_{14}\hat{\mathsf{n}}_2\right)\\ \tilde{v}_3\hat{\mathsf{n}}_1-\tilde{\rho}\hat{\mathsf{n}}_2 & \tilde{v}_3\hat{\mathsf{n}}_2+\tilde{\rho}\hat{\mathsf{n}}_1 & \tilde{v}_3\hat{\mathsf{n}}_3 & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{v}_3+\tilde{\lambda}_{15}\hat{\mathsf{n}}_3\right) & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{v}_3+\tilde{\lambda}_{14}\hat{\mathsf{n}}_3\right)\\ \frac{1}{2}\tilde{v}_k^2\hat{\mathsf{n}}_1+\tilde{\rho}\varepsilon_{iq1}\tilde{v}_i\hat{\mathsf{n}}_q & \frac{1}{2}\tilde{v}_k^2\hat{\mathsf{n}}_2+\tilde{\rho}\varepsilon_{iq2}\tilde{v}_i\hat{\mathsf{n}}_q & \frac{1}{2}\tilde{v}_k^2\hat{\mathsf{n}}_3+\tilde{\rho}\varepsilon_{iq3}\tilde{v}_i\hat{\mathsf{n}}_q & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{\alpha}\tilde{h}_t+\tilde{\lambda}_{15}\vec{\tilde{v}}\vec{\hat{\mathsf{n}}}\right) & \frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\left(\tilde{\alpha}\tilde{h}_t+\tilde{\lambda}_{14}\vec{\tilde{v}}\vec{\hat{\mathsf{n}}}\right)\end{bmatrix}$$

where $\tilde{\lambda}_{14} = \frac{\tilde{\lambda}_1 - \tilde{\lambda}_4}{|\vec{\mathsf{n}}|}$, $\tilde{\lambda}_{15} = \frac{\tilde{\lambda}_1 - \tilde{\lambda}_5}{|\vec{\mathsf{n}}|}$, $\tilde{\lambda}_{45} = \frac{\tilde{\lambda}_4 - \tilde{\lambda}_5}{|\vec{\mathsf{n}}|}$, $\vec{\mathsf{n}}$ the dimensional normal vector with magnitude equal to the area of the interface between two adjacent finite volumes and $\hat{\mathsf{n}}_q$ the unit normal vector.

The differentiation of the absolute Jacobian matrix w.r.t. the Roe-averaged quantities reads

$$\frac{\partial |\tilde{A}^{LR}_{nmk} \mathsf{n}_k|}{\partial \tilde{V}_j} = \frac{\partial \tilde{P}_{nk}}{\partial \tilde{V}_j} |\tilde{\lambda}_k| \tilde{P}^{-1}_{km} + \tilde{P}_{nk} \frac{\partial |\tilde{\lambda}_k|}{\partial \tilde{V}_j} \tilde{P}^{-1}_{km} + \tilde{P}_{nk} |\tilde{\lambda}_k| \frac{\partial \tilde{P}^{-1}_{km}}{\partial \tilde{V}_j} \tag{A.1}$$

The computation of the derivatives of the eigenvalue matrix in Eq. A.1 follows. It is

$$\frac{\partial |\tilde{\lambda}_{1,2,3}|}{\partial \tilde{V}_j} = sign\left(\tilde{\lambda}_1\right) \begin{bmatrix} 0 \\ \mathsf{n}_1 \\ \mathsf{n}_2 \\ \mathsf{n}_3 \\ 0 \end{bmatrix}, \quad \frac{\partial |\tilde{\lambda}_{4,5}|}{\partial \tilde{V}_j} = sign\left(\tilde{\lambda}_{4,5}\right) \begin{bmatrix} 0 \\ \mathsf{n}_1 \pm |\vec{\mathsf{n}}| \frac{\partial \tilde{c}}{\partial \tilde{V}_j} \\ \mathsf{n}_2 \pm |\vec{\mathsf{n}}| \frac{\partial \tilde{c}}{\partial \tilde{V}_j} \\ \mathsf{n}_3 \pm |\vec{\mathsf{n}}| \frac{\partial \tilde{c}}{\partial \tilde{V}_j} \\ \pm |\vec{\mathsf{n}}| \frac{\partial \tilde{c}}{\partial \tilde{V}_j} \end{bmatrix} \tag{A.2}$$

where

$$sign\,(x) = \begin{cases} +1 & , \text{ if } x \geq 0 \\ -1 & , \text{ if } x < 0 \end{cases}$$

The speed of sound that corresponds to the Roe-averaged quantities reads $\tilde{c} = \sqrt{(\gamma - 1)\left(\tilde{h}_t - \frac{1}{2}\tilde{v}_k^2\right)}$ and

$$\frac{\partial \tilde{c}}{\partial \tilde{V}_j} = \frac{\gamma - 1}{2\tilde{c}} \begin{bmatrix} 0 \\ -\tilde{v}_1 \\ -\tilde{v}_2 \\ -\tilde{v}_3 \\ 1 \end{bmatrix} \tag{A.3}$$

The derivative of the matrix with the right eigenvectors of the Jacobian is com-

puted as

$$\frac{\partial \tilde{P}}{\partial \tilde{V}_j} = \begin{bmatrix} 0 & 0 & 0 & b_j & b_j \\ \hat{\mathsf{n}}_1\delta_{2j} & -\hat{\mathsf{n}}_3\delta_{1j}+\hat{\mathsf{n}}_2\delta_{2j} & \hat{\mathsf{n}}_2\delta_{1j}+\hat{\mathsf{n}}_3\delta_{2j} & d^5_{2j} & d^4_{2j} \\ \hat{\mathsf{n}}_3\delta_{1j}+\hat{\mathsf{n}}_1\delta_{3j} & \hat{\mathsf{n}}_2\delta_{3j} & -\hat{\mathsf{n}}_1\delta_{1j}+\hat{\mathsf{n}}_3\delta_{3j} & d^5_{3j} & d^4_{3j} \\ -\hat{\mathsf{n}}_2\delta_{1j}+\hat{\mathsf{n}}_1\delta_{4j} & \hat{\mathsf{n}}_1\delta_{1j}+\hat{\mathsf{n}}_2\delta_{4j} & \hat{\mathsf{n}}_3\delta_{4j} & d^5_{3j} & d^4_{4j} \\ f_{1j} & f_{2j} & f_{3j} & e^5_j & e^4_j \end{bmatrix}$$

with

$$b_j = \begin{cases} \frac{1}{\tilde{\lambda}_{45}} & , \ j=1 \\ -\frac{1\tilde{\rho}}{\tilde{\lambda}^2_{45}}\frac{\partial \tilde{\lambda}_{45}}{\partial \tilde{V}_j} & , \ j=2,3,4,5 \end{cases}$$

$$d^m_{ij} = \begin{cases} b_j\tilde{v}_{i-1}+\frac{\tilde{\lambda}_{1m}}{\tilde{\lambda}_{45}}\hat{\mathsf{n}}_{i-1} & , \ j=1 \\ b_j\tilde{v}_{i-1}+\frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\delta_{ij}+\tilde{\rho}\hat{\mathsf{n}}_{i-1}\frac{\partial}{\partial \tilde{V}_j}\left(\frac{\tilde{\lambda}_{1m}}{\tilde{\lambda}_{45}}\right) & , \ j=2,3,4 \\ b_j\tilde{v}_{i-1}+\tilde{\rho}\hat{\mathsf{n}}_{i-1}\frac{\partial}{\partial \tilde{V}_j}\left(\frac{\tilde{\lambda}_{1m}}{\tilde{\lambda}_{45}}\right) & , \ j=5 \end{cases}$$

$$e^m_j = \begin{cases} b_j\tilde{h}_t+\frac{\tilde{\lambda}_{1m}}{\tilde{\lambda}_{45}}\vec{\tilde{v}}\vec{\hat{\mathsf{n}}} & , \ j=1 \\ b_j\tilde{h}_t+\frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\tilde{\lambda}_{1m}\hat{\mathsf{n}}_{j-1}+\frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\vec{\tilde{v}}\vec{\hat{\mathsf{n}}}\frac{\partial}{\partial \tilde{V}_j}\left(\frac{\tilde{\lambda}_{1m}}{\tilde{\lambda}_{45}}\right) & , \ j=2,3,4 \\ b_j\tilde{h}_t+\frac{\tilde{\rho}}{\tilde{\lambda}_{45}}+\frac{\tilde{\rho}}{\tilde{\lambda}_{45}}\vec{\tilde{v}}\vec{\hat{\mathsf{n}}}\frac{\partial}{\partial \tilde{V}_j}\left(\frac{\tilde{\lambda}_{1m}}{\tilde{\lambda}_{45}}\right) & , \ j=5 \end{cases}$$

$$f_{rj} = \begin{cases} \varepsilon_{iqr}\tilde{v}_i\hat{\mathsf{n}}_q & , \ j=1 \\ \tilde{v}_{j-1}\hat{\mathsf{n}}_r-\tilde{\rho}\varepsilon_{k(j-1)r}\hat{\mathsf{n}}_k & , \ j=2,3,4 \\ 0 & , \ j=5 \end{cases}$$

Then, the derivative of the matrix with the left eigenvectors can be computed by differentiating $\tilde{P}^{-1}\tilde{P}=I$, as follows

$$\frac{\partial \tilde{P}^{-1}_{km}}{\partial \tilde{V}_j} = -\tilde{P}^{-1}_{k\ell}\frac{\partial \tilde{P}_{\ell r}}{\partial \tilde{V}_j}\tilde{P}^{-1}_{rm}$$

The derivatives of the Roe-averaged quantities w.r.t. the primitive variables of the L statement are

$$\frac{\partial \tilde{V}_j}{\partial V_\ell^L} = \begin{bmatrix} \frac{1}{2}\frac{\tilde{\rho}}{\rho^L} & 0 & 0 & 0 & 0 \\ r_1\left(v_1^L - \tilde{v}_1\right) & r_2 & 0 & 0 & 0 \\ r_1\left(v_2^L - \tilde{v}_2\right) & 0 & r_2 & 0 & 0 \\ r_1\left(v_3^L - \tilde{v}_3\right) & 0 & 0 & r_2 & 0 \\ r_1\left(h_t^L - \tilde{h}_t\right) - r_2\frac{\gamma p^L}{{\rho^L}^2(\gamma-1)} & r_2 v_1^L & r_2 v_2^L & r_2 v_3^L & r_2\frac{\gamma}{\rho^L(\gamma-1)} \end{bmatrix}$$

where

$$r_1 = \frac{1}{2\sqrt{\rho^L}}\frac{1}{\sqrt{\rho^L}+\sqrt{\rho^R}}, \quad r_2 = \frac{\sqrt{\rho^L}}{\sqrt{\rho^L}+\sqrt{\rho^R}}$$

Appendix B

Evolutionary Algorithms / The EASY Platform

A brief description of the EASY platform, [4], developed by the PCOpt/NTUA that was used to carry out the optimization studies of Chapters 6 and 7 follows. In the EASY platform, search is based on a (μ, λ) EA, with μ parents and λ offspring. Its distinguishing feature is the unique way of implementing and using on-line trained personalized metamodels to replace a great number of calls to the (usually) computationally expensive evaluation tool, giving rise to the so-called Metamodel–Assisted EA (MAEA), [60]. Without loss in generality, the built-in Radial Basis Function (RBF) networks are used as surrogate models.

In each generation (g) of the MAEA, starting from the offspring population evaluated in the previous generation (S_{λ}^{g-1}), the parent population (S_{μ}^{g}) results from the parent selection and elitism operators. From S_{μ}^{g}, through crossover and mutation, the new offspring population (S_{λ}^{g}) results. In each generation, the elite population (S_{ε}^{g}, with ε members at most) includes the best-so-far individual(s) or the current front of non-dominated solutions. The so–called low–cost pre–evaluation (LCPE) phase of the MAEA starts after having evaluated and stored in the EA database (DB_{EA}) a user-defined minimum number (T^{MM}) of individuals evaluated on the problem specific evaluation model, during the first generations. During the LCPE phase, for each new offspring, a "personalized" metamodel is trained on the "closest" already evaluated individuals, [60] and only the few most promising ($\lambda_e \ll \lambda$; λ_e is also user-defined) of them are re–evaluated on the problem-specific evaluation model. It is obvious that population members evaluated only on the metamodel do not enter DB_{EA}. The small size of the neighboring training patterns used per personalized metamodel makes the overall training cost practically negligible (compared, for instance, to the cost of a single CFD evaluation). Using a MAEA, cost reduction by even one order of magnitude (compared to a plain EA) is achieved.

Even by using MAEAs, performance degradation, in real-world problems with

many design variables is expected and this is avoided through the additional use of the kernel Principal Component Analysis (kPCA) of the offspring population in each generation. The PCA is used to draw information that improves the evolution operators and/or reduces the number of input units of metamodels, [59]. Briefly, in each generation of the MAEA, the parent population is mapped onto a new feature space with ordered variances, the evolution operators generate offspring in this space and these are, then, transformed back into the design space. During the LCPE, the PCA is additionally used to selectively prune metamodels' inputs. For each population member, once the training patterns for its "personalized" metamodel have been selected, these are mapped onto the feature space and the metamodel inputs along the direction with the smaller variances are pruned. By doing so, the metamodels are trained using less inputs and yields better predictions.

For a detailed presentation of all EASY features, the interested reader should refer to [4], [60] and [59].

Bibliography

[1] AIAA Transition Modeling Workshop-I. https://transitionmodeling.larc.nasa.gov/workshop_i/?doing_wp_cron=1689003551.6611080169677734375000. Accessed: July 2021.

[2] INRIA Sophia-Antipolis, TAPENADE. https://www-sop.inria.fr/tropics/tapenade.html.

[3] Mathematics, Argonne National Laboratory Computer Science Division, and Rice University Center for Research on Parallel Computation. https://www.mcs.anl.gov/research/projects/adifor/.

[4] The EASY (Evolutionary Algorithms SYstem) software. http://velos0.ltt.mech.ntua.gr/EASY, 2008.

[5] W. Anderson, J. Newman, D. Whitfield, and E. Nielsen. Sensitivity analysis for Navier-Stokes equations on unstructured meshes using complex variables. *AIAA Journal*, 39(1):56–63, 2001.

[6] W. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. *Computers & Fluids*, 28(4):443–480, 1999.

[7] T. Arts, M. Lambert de Rouvroit, and A. Rutherford. Aero-thermal investigation of a highly loaded transonic linear turbine guide vane cascade. a test case for inviscid and viscous flow computations. Technical Note 174, von Karman Institute for Fluid Dynamics, Belgium, 1990.

[8] V. Asouti. *Aerodynamic analysis and design methods at high and low speed flows, on multiprocessor platforms (in Greek)*. PhD thesis, National Technical University of Athens, 2009.

[9] V. Asouti, X. Trompoukis, I. Kampolis, and K. Giannakoglou. Unsteady CFD computations using vertex-centered finite volumes for unstructured grids on Graphics Processing Units. *International Journal for Numerical Methods in Fluids*, 67(2):232–246, 2011.

[10] B. Baldwin and H. Lomax. Thin-layer approximation and algebraic model for separated turbulent flows. In *16th Aerospace Sciences Meeting*, 1978.

[11] T. Barth and D. Jespersen. The design and application of upwind schemes on unstructured meshes. In *27th Aerospace Sciences Meeting*, 1989.

[12] A. Bhalla, M. Nikhila, and P. Singh. Simulation of self-driving car using deep learning. In *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, pages 519–525, 2020.

[13] C. Bishop and N. Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[14] K. Boopathy and M. Rumpfkeil. A multivariate interpolation and regression enhanced Kriging surrogate model. *21st AIAA Computational Fluid Dynamics Conference*, 2013.

[15] S. Cakmakcioglu, O. Bas, and U. Kaynak. A correlation-based algebraic transition model. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 232(21):3915–3929, 2018.

[16] C. Castro, C. Lozano, F. Palacios, and E. Zuazua. Systematic continuous adjoint approach to viscous aerodynamic design on unstructured grids. *AIAA Journal*, 45(9):2125–2139, 2007.

[17] J. Coder. Standard test cases for CFD-based laminar-transition model verification and validation. In *2018 AIAA Aerospace Sciences Meeting*, 2018.

[18] J. Coder. Further development of the amplification factor transport transition model for aerodynamic flows. In *AIAA Scitech 2019 Forum*, 2019.

[19] J. Coder and M. Maughmer. Computational fluid dynamics compatible transition modeling using an amplification factor transport equation. *AIAA Journal*, 52:2506–2512, 2014.

[20] J. Coder, T. Pulliam, and J. Jensen. Contributions to HiLiftPW-3 using structured, overset grid methods. In *2018 AIAA Aerospace Sciences Meeting*, 2018.

[21] J. Dagenhart and W. Saric. Crossflow stability and transition experiments in swept-wing flow. Technical Report TP-1999-209344, NASA, 1999.

[22] V. D'Alessandro, S. Montelpare, and R. Ricci. Assessment of a Spalart-Allmaras model coupled with local correlation based transition approaches for wind turbine airfoils. *Applied Sciences*, 11(4):1872, 2021.

[23] V. D'Alessandro, S. Montelpare, R. Ricci, and A. Zoppi. Numerical modeling of the flow over wind turbine airfoils by means of Spalart-Allmaras local correlation based transition model. *Energy*, 130:402–419, 2017.

[24] C. Darwin. *On the origin of species by means of natural selection*. Murray, London, 1859. or the preservation of favored races in the struggle for life.

[25] R. Dechter. Learning while searching in constraint-satisfaction-problems. In *Proceedings of the 5th National Conference on Artificial Intelligence. Philadelphia*, pages 178–185, 1986.

[26] Z. Deng, C. He, and Y. Liu. Deep neural network-based strategy for optimal sensor placement in data assimilation of turbulent flow. *Physics of Fluids*, 33(2):025119, 2021.

[27] S. Dhawan and R. Narasimha. Some properties of boundary layer flow during the transition from laminar to turbulent motion. *Journal of Fluid Mechanics*, 3:418–436, 1958.

[28] D. Di Pasquale and A. Rona. A selective review of transition modelling for CFD. In *39th AIAA Fluid Dynamics Conference*, 2009.

[29] K. Diakakis. *Computational analysis of transitional and massively separated flows with application to wind turbines*. PhD thesis, National Technical University of Athens, 2020.

[30] K. Diakakis, G. Papadakis, and S. Voutsinas. Assessment of transition modeling for high Reynolds flows. *Aerospace Science and Technology*, 85, 2019.

[31] E. Dick and S. Kubacki. Transition models for turbomachinery boundary layer flows: A review. *International Journal of Turbomachinery, Propulsion and Power*, 2(2), 2017.

[32] M. Drela and M. Giles. Viscous-inviscid analysis of transonic and low Reynolds number airfoils. *AIAA Journal*, 25(10):1347–1355, 1987.

[33] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[34] K. Duraisamy, Z. Zhang, and A. Singh. New approaches in turbulence and transition modeling using data-driven techniques. In *53rd AIAA Aerospace Sciences Meeting*, 2015.

[35] P. Durbin, R. Jacobs, and X. Wu. *DNS of Bypass Transition*, pages 449–463. Cambridge University Press, Cambridge, England, U.K., 2001.

[36] M. Duta, S. Shahpar, and M. Giles. Turbomachinery design optimization using automatic differentiated adjoint code. In *Turbo Expo: Power for Land, Sea, and Air*, volume 6, 2007.

[37] H. Elbanna and L. Carlson. Determination of aerodynamic sensitivity coefficients in the transonic and supersonic regimes. In *27th Aerospace Sciences Meeting*, page 532, 1989.

[38] J. Elliott and J. Peraire. Practical three-dimensional aerodynamic design and optimization using unstructured meshes. *AIAA Journal*, 35(9):1479–1485, 1997.

[39] H. Emmons. The laminar-turbulent transition in a boundary layer-Part I. *Journal of the Aeronautical Sciences*, 18(7):490–498, 1951.

[40] R. Eppler. A computer program for the design and analysis of low-speed airfoils, supplement. Technical Memorandum NASA-TM-81862, 1980.

[41] N. Foster and G. Dulikravich. Three-dimensional aerodynamic shape optimization using genetic and gradient search algorithms. *Journal of Spacecraft and Rockets*, 34(1):36–42, 1997.

[42] K. Giannakoglou. Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. *Progress in Aerospace Sciences*, 38(1):43–76, 2002.

[43] M. Giles, M. Duta, and J. Muller. Adjoint code developments using the exact discrete approach. *15th AIAA Computational Fluid Dynamics Conference*, 2001.

[44] M. Giles and N. Pierce. Adjoint equations in CFD - duality, boundary conditions and solution behaviour. In *13th Computational Fluid Dynamics Conference*, 1997.

[45] M. Giles and N. Pierce. An introduction to the adjoint approach to design. *Flow, Turbulence and Combustion*, 65, 2000.

[46] A. Griewank and A. Walther. *Evaluating derivatives: Principles and techniques of algorithmic differentiation*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2008.

[47] G. Halila, J. Martins, and K. Fidkowski. Adjoint-based aerodynamic shape optimization including transition to turbulence effects. *Aerospace Science and Technology*, 107:106243, 2020.

[48] J. Hammond, N. Pepper, F. Montomoli, and V. Michelassi. Machine learning methods in CFD for turbomachinery: A review. *International Journal of Turbomachinery, Propulsion and Power*, 7:16, 2022.

[49] Z. Han. Improving adjoint-based aerodynamic optimization via gradient-enhanced Kriging. *50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2012.

[50] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning, second edition: data mining, inference, and prediction*. Springer, 2016.

[51] A. Heft, Th. Indinger, and N. Adams. Experimental and numerical investigation of the DrivAer model. In *ASME 2012 Fluids Engineering Summer Meeting*, Rio Grande, Puerto Rico, 2012.

[52] L. Hylton, M. Mihelc, E. Turner, D. Nealy, and R. York. Analytical and experimental evaluation of the heat transfer distribution over the surfaces of turbine vanes. *NASA-CR-168015*, 1983.

[53] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3, 1988.

[54] A. Jameson and J. Reuther. Control theory based airfoil design using the euler equations. *5th Symposium on Multidisciplinary Analysis and Optimization*, 1994.

[55] I. Kampolis and K. Giannakoglou. Distributed evolutionary algorithms with hierarchical evaluation. *Engineering Optimization*, 41(11):1037–1049, 2009.

[56] I. Kampolis and K. Giannakoglou. Synergetic use of different evaluation, parameterization and search tools within a multilevel optimization platform. *Applied Soft Computing*, 11(1):645–651, 2011.

[57] I. Kampolis, X. Trompoukis, V. Asouti, and K. Giannakoglou. CFD-based analysis and two-level aerodynamic optimization on graphics processing units. *Computer Methods in Applied Mechanics and Engineering*, 199(9-12):712–722, 2010.

[58] D. Kapsoulis. *Low-Cost Metamodel-Assisted Evolutionary Algorithms with applications in shape optimization in fluid dynamics*. PhD thesis, National Technical University of Athens, 2019.

[59] D. Kapsoulis, K. Tsiakas, X. Trompoukis, V. Asouti, and K. Giannakoglou. Evolutionary multi-objective optimization assisted by metamodels, kernel PCA and multi-criteria decision making techniques with applications in aerodynamics. *Applied Soft Computing*, 64:1–13, 2018.

[60] M. Karakasis and K. Giannakoglou. On the use of metamodel-assisted, multi-objective evolutionary algorithms. *Engineering Optimization*, 38(8):941–957, 2006.

[61] M. Karakasis, A. Giotis, and K. Giannakoglou. Inexact information aided, low-cost, distributed genetic algorithms for aerodynamic shape optimization. *International Journal for Numerical Methods in Fluids*, 43(10-11):1149–1166, 2003.

[62] I. Kavvadias. *Continuous adjoint methods for steady and unsteady turbulent flows with emphasis on the accuracy of sensitivity derivatives*. PhD thesis, National Technical University of Athens, 2016.

[63] I. Kavvadias, E. Papoutsis-Kiachagias, G. Dimitrakopoulos, and K. Giannakoglou. The continuous adjoint approach to the k–ω SST turbulence model with applications in shape optimization. *Engineering Optimization*, 47(11):1523–1542, 2015.

[64] I. Kavvadias, E. Papoutsis-Kiachagias, and K. Giannakoglou. On the proper treatment of grid sensitivities in continuous adjoint methods for shape optimization. *Journal of Computational Physics*, 301:1–18, 2015.

[65] P. Khayatzadeh and S. Nadarajah. Aerodynamic shape optimization via discrete viscous adjoint equations for the k-wSST turbulence and y-Re0 transition models. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2011.

[66] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, 2015.

[67] K. Kovani, M. Kontou, V. Asouti, and K. Giannakoglou. DNN-driven gradient-based shape optimization in fluid mechanics. In L. Iliadis, I. Maglogiannis, S. Alonso, C. Jayne, and E. Pimenidis, editors, *Engineering Applications of Neural Networks*, pages 379–390, Cham, 2023. Springer Nature Switzerland.

[68] S. Kubacki and E. Dick. An algebraic model for bypass transition in turbomachinery boundary layer flows. *International Journal of Heat and Fluid Flow*, 58:68–83, 2016.

[69] S. Kubacki, P. Jonak, and E. Dick. Evaluation of an algebraic model for laminar-to-turbulent transition on secondary flow loss in a low-pressure turbine cascade with an endwall. *International Journal of Heat and Fluid Flow*, 77:98–112, 2019.

[70] S. Kyriacou, V. Asouti, and K. Giannakoglou. Efficient PCA-driven EAs and metamodel-assisted EAs, with applications in turbomachinery. *Engineering Optimization*, 46(7):895–911, 2014.

[71] R. Langtry. Extending the Gamma-Rethetat correlation based transition model for crossflow effects (invited). In *45th AIAA Fluid Dynamics Conference*, 2015.

[72] R. Langtry and F. Menter. Transition modeling for general CFD applications in aeronautics. In *43rd AIAA Aerospace Sciences Meeting and Exhibit - Meeting Papers*, 2005.

[73] R. Langtry and F. Menter. Correlation-based transition modeling for unstructured parallelized computational fluid dynamics codes. *AIAA Journal*, 47:2894–2906, 2009.

[74] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin. Exploring strategies for training deep neural networks. *Journal of Machine Learning Research*, 10(1), 2009.

[75] B. Lee and C. Kim. Automated design methodology of turbulent internal flow using discrete adjoint formulation. *Aerospace Science and Technology*, 11(2):163–173, 2007.

[76] M. Lesoinne and C. Farhat. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Computer Methods in Applied Mechanics and Engineering*, 134:71–90, 1996.

[77] J. Ling, A. Kurzawski, and J. Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, 807:155{166, 2016.

[78] J. Lions. *Optimal Control of Systems Governed by Partial Differentiable Equations*. Springer-Verlang, New York, NY, USA, 1971.

[79] Q. Liu, N. Zhang, W. Yang, W. Sile, Z. Cui, X. Chen, and L. Chen. A review of image recognition with deep convolutional neural network. In De-S. Huang, V. Bevilacqua, P. Premaratne, and P. Gupta, editors, *Intelligent Computing Theories and Application*, pages 69–80, Cham, 2017. Springer International Publishing.

[80] C. Lozano. On the properties of solutions of the 2D adjoint Euler equations. *Computational Methods in Applied Sciences*, pages 77–91, 2019.

[81] C. Lozano, E. Andrés, M. Martín, and P. Bitrián. Domain versus boundary computation of flow sensitivities with the continuous adjoint method for aerodynamic shape optimization problems. *International Journal for Numerical Methods in Fluids*, 70(10):1305–1323, 2012.

[82] A. Lozano-Durán, M. J. P. Hack, and P. Moin. Modeling boundary-layer transition in direct and large-eddy simulations using parabolized stability equations. *Phys. Rev. Fluids*, 3:023901, 2018.

[83] M. Lynde, R. Campbell, M. Rivers, S. Viken, D. Chan, A. Watkins, and S. Goodliff. Preliminary results from an experimental assessment of a natural laminar flow design method. *AIAA Scitech 2019 Forum*, 2019.

[84] M. Lynde, R. Campbell, and S. Viken. Additional findings from the common research model natural laminar flow wind tunnel test. *AIAA Scitech 2019 Forum*, 2019.

[85] Y. Mahmoud, Y. Okuyama, T. Fukuchi, T. Kosuke, and I. Ando. Optimizing deep-neural-network-driven autonomous race car using image scaling. *SHS Web of Conferences*, 77:04002, 2020.

[86] P. Malan, K. Suluksna, and E. Juntasaro. Calibrating the $\gamma - Re_\theta$ transition model for commercial CFD. *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2009.

[87] M. Malik, J. Crouch, W. Saric, J. Lin, and E. Whalen. *Application of drag reduction techniques to transport aircraft*, pages 1–10. John Wiley & Sons, Ltd, 2015.

[88] E. Malkiel and R. Mayle. Transition in a separation bubble. *Journal of Turbomachinery*, 118(4):752–759, 1996.

[89] P. Mall, P. Singh, S. Srivastav, V. Narayan, M. Paprzycki, T. Jaworska, and M. Ganzha. A comprehensive review of deep neural networks for medical image processing: Recent developments and future opportunities. *Healthcare Analytics*, 4:100216, 2023.

[90] Y. Marioni, E. Ortiz, A. Cassinelli, F. Montomoli, P. Adami, and R. Vazquez-Diaz. A machine learning approach to improve turbulence modelling from dns data using neural networks. *International Journal of Turbomachinery, Propulsion and Power*, 6:17, 2021.

[91] O. Marxen and U. Rist. DNS and LES of the transition process in a laminar separation bubble. In R. Friedrich, B. Geurts, and O. Métais, editors, *Direct and Large-Eddy Simulation V*, pages 231–240, Dordrecht, 2004. Springer Netherlands.

[92] R. Maulik, H. Sharma, S. Patel, B. Lusch, and E. Jennings. A turbulent eddy-viscosity surrogate modeling framework for Reynolds-averaged Navier-Stokes simulations. *Computers & Fluids*, 227:104777, 2021.

[93] R. Mayle. The 1991 IGTI Scholar Lecture: The role of laminar-turbulent transition in gas turbine engines. *Journal of Turbomachinery*, 113(4):509–536, 1991.

[94] S. Medida and J. Baeder. Application of the correlation-based Gamma-Re Thetat transition model to the Spalart-Allmaras turbulence model. In *20th AIAA Computational Fluid Dynamics Conference*, 2011.

[95] F. Menter, M. Kuntz, and R. Langtry. Ten years of industrial experience with the SST turbulence model. *Turbulence, Heat and Mass Transfer*, 4, 2003.

[96] F. Menter, P. Smirnov, T. Liu, and R. Avancha. A one-equation local correlation-based transition model. *Flow, Turbulence and Combustion*, 95:1–37, 2015.

[97] A. Mohamed, A. Hemeida, and M. Hassan. Image classification based deep learning: A review. *Aswan University Journal of Sciences and Technology*, 2, 2022.

[98] M. Monfaredi. *Continuous Adjoint-based aeroacoustic shape optimization by coupling the Ffowcs-Williams Hawkings analogy with URANS equations*. PhD thesis, National Technical University of Athens, 2021.

[99] D. Montgomery. *Design and analysis of experiments*. John Wiley & Sons, 12th edition, 2017.

[100] M. Morkovin. On the many faces of transition. In S. Wells, editor, *Viscous Drag Reduction*, pages 1–31, Boston, MA, 1969. Springer US.

[101] S. Nadarajah and A. Jameson. Studies of the continuous and discrete adjoint approaches to viscous automatic aerodynamic shape optimization. *15th AIAA Computational Fluid Dynamics Conference*, 25(30), 2001.

[102] K. Nakagawa, T. Tsukahara, and T. Ishida. DNS study on turbulent transition induced by an interaction between freestream turbulence and cylindrical roughness in swept flat-plate boundary layer. *Aerospace*, 10(2), 2023.

[103] E. Nielsen and W. Anderson. Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. *AIAA Journal*, 37(11):1411–1419, 1999.

[104] E. Nielsen, J. Lu, M. Park, and D. Darmofal. An implicit, exact dual adjoint solution method for turbulent flows on unstructured grids. *Computers & Fluids*, 33(9):1131–1155, 2004.

[105] J. Nocedal and S. Wright. *Numerical optimization*. Springer, 1999.

[106] D. Papadimitriou and K. Giannakoglou. A continuous adjoint method with objective function derivatives based on boundary integrals for inviscid and viscous flows. *Computers & Fluids*, 36(2):325–341, 2007.

[107] E. Papoutsis-Kiachagias. *Adjoint methods for turbulent flows, applied to shape or topology optimization and robust design*. PhD thesis, National Technical University of Athens, 2013.

[108] E. Papoutsis-Kiachagias, V. Asouti, K. Giannakoglou, K. Gkagkas, S. Shimokawa, and E. Itakura. Multi-point aerodynamic shape optimization of cars based on continuous adjoint. *Structural and Multidisciplinary Optimization*, 59(2):675–694, 2019.

[109] E. Papoutsis-Kiachagias, A. Zymaris, I. Kavvadias, D. Papadimitriou, and K. Giannakoglou. The continuous adjoint approach to the k–ε turbulence model for shape optimization and optimal active control of turbulent flows. *Engineering Optimization*, 47(3):370–389, 2015.

[110] R. Perrin and E. Lamballais. Assessment of implicit LES modelling for bypass transition of a boundary layer. *Computers & Fluids*, 251:105728, 2023.

[111] N. Pierce and M. Giles. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review*, 42, 2000.

[112] M. Piotrowski and D. Zingg. Investigation of a local correlation-based transition model in a Newton-Krylov algorithm. In *AIAA SCITECH 2022 Forum*, 2019.

[113] M. Piotrowski and D. Zingg. Smooth local correlation-based transition model for the Spalart-Allmaras turbulence model. *AIAA Journal*, 2020.

[114] O. Pironneau. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64(1):97–110, 1974.

[115] Q. Rao and J. Frtunikj. Deep learning for self-driving cars: Chances and challenges. In *2018 IEEE/ACM 1st International Workshop on Software Engineering for AI in Autonomous Systems (SEFAIAS)*, pages 35–38, 2018.

[116] J. Reuther, A. Jameson, American Institute of Aeronautics, and Astronautics. *Control theory based airfoil design for potential flow and a finite volume discretization*. AIAA-94/0499. AIAA, 1994.

[117] P. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2):357–372, 1981.

[118] D. Rumelhart, G. Hinton, and R. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[119] A. Savill. Further progress in the turbulence modelling of by-pass transition. In *Engineering Turbulence Modelling and Experiments*, pages 583–592. Elsevier, Oxford, 1993.

[120] H. Schlichting. *Boundary-Layer Theory*. Springer Berlin, Heidelberg, 1979.

[121] V. Schmitt and J. Cousteix. Etude de la couche limite tridimensionnelle sur une aile en fleche. Technical Report No 14/1713 AN, ONERA, 1975.

[122] G. Schubauer and P. Klebanoff. Contributions on the mechanics of boundary-layer transition. Technical Report NACA-TR-1289, NASA, 1955.

[123] S. Shamshirband, M. Fathi, A. Dehzangi, A. Chronopoulos, and H. Alinejad-Rokny. A review on deep learning approaches in healthcare systems: Taxonomies, challenges, and open issues. *Journal of Biomedical Informatics*, 113:103627, 2021.

[124] A. Shrestha and A. Mahmood. Optimizing deep neural network architecture with enhanced genetic algorithm. In *18th IEEE International Conference On Machine Learning and Applications (ICMLA)*, 2019.

[125] S. Singh, A. Kumar, H. Darbari, L. Singh, A. Rastogi, and S. Jain. Machine translation using deep learning: An overview. In *2017 International Conference on Computer, Communications and Electronics (Comptelix)*, pages 162–167, 2017.

[126] A.M.O. Smith, Douglas Aircraft Company, N. Gamberoni, Aeronautical Research Council. Fluid Motion Sub-Committee, and Douglas Aircraft Company. El Segundo Division. Engineering Department. *Transition, Pressure Gradient and Stability Theory*. ARC-19322. Douglas Aircraft Company, El Segundo Division, 1956.

[127] D. Somers. Design and Experimental Results for a Natural-Laminar-Flow Airfoil for General Aviation Applications. *NASA Technical Paper 1861*, 1981.

[128] P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. *La Recherche Aérospatiale*, 1:5–21, 1994.

[129] J. Steger and R. Warming. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. *Journal of Computational Physics*, 40(2):263–293, 1981.

[130] W. Sutherland. The Viscosity of Gases and Molecular Force. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 36(223):507–531, 1893.

[131] A. Taebi. Deep learning for computational hemodynamics: A brief review of recent advances. *Fluids*, 7(6):197, 2022.

[132] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[133] X. Trompoukis. *Numerical solution of aerodynamic-aeroelastic problems on Graphics Processing Units (in Greek)*. PhD thesis, National Technical University of Athens, 2012.

[134] X. Trompoukis, K. Tsiakas, V. Asouti, and K. Giannakoglou. Continuous adjoint-based optimization of a turbomachinery stage using a 3D volumetric parameterization. *International Journal for Numerical Methods in Fluids*, 2023.

[135] K. Tsiakas. *Development of shape parameterization techniques, a flow solver and its adjoint, for optimization on GPUs. Turbomachinery and external aerodynamics applications*. PhD thesis, National Technical University of Athens, 2019.

[136] F. Charpin V. Schmitt. Pressure distributions on the ONERA M6 wing at transonic Mach numbers, experimental data base for computer program assessment. *Technical report*, AGARD 138, 1979.

[137] J. van Ingen. *A Suggested Semi-empirical Method for the Calculation of the Boundary Layer Transition Region*. ARC-19337. TH Delft, Delft, 1956.

[138] J. van Ingen. The eN method for transition prediction. Historical review of work at TU Delft. In *38th Fluid Dynamics Conference and Exhibit*, 2008.

[139] B. van Leer. Flux-vector splitting for the Euler equations. In *Eighth International Conference on Numerical Methods in Fluid Dynamics*, pages 507–512, 1982.

[140] V. Venkatakrishnan. On the accuracy of limiters and convergence to steady state solutions. *AIAA Paper 1993-880*, 1993.

[141] T. Verstraete, L. Müller, and J.-D. Müller. Adjoint-based design optimisation of an internal cooling channel U-bend for minimised pressure losses. *International Journal of Turbomachinery, Propulsion and Power*, 2(2), 2017.

[142] A. Walther. Getting started with ADOL-C. *Combinatorial Scientific Computing*, 2009.

[143] D. Wang. An improved mixing-plane method for analyzing steady flow through multiple-blade-row turbomachines. *Journal of Turbomachinery*, 136(8):081003, 2014.

[144] J. Weiss, J. Maruszewski, and W. Smith. Implicit solution of preconditioned Navier-Stokes equations using algebraic multigrid. *AIAA Journal*, 37(1):29–36, 1999.

[145] Y. Wu and et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *ArXiv*, abs/1609.08144, 2016.

[146] W. Yamazaki, K. Matsushima, and K. Nakahashi. Aerodynamic design optimization using the drag-decomposition method. *AIAA Journal*, 46:1096–1106, 2008.

[147] T. Zervogiannis. *Optimization methods in aerodynamics and turbomachinery based on the adjoint technique, hybrid grids and the exact Hessian matrix (in Greek)*. PhD thesis, National Technical University of Athens, 2011.

[148] J. Zhang and C. Zong. Deep neural networks in machine translation: An overview. *IEEE Intelligent Systems*, 30:16–25, 2015.

[149] X. Zheng, C. Liu, F. Liu, and C. Yang. Turbulent transition simulation using the k–ω model. *International Journal for Numerical Methods in Engineering*, 42(5):907–926, 1998.

[150] H. Zhu, M. Akrout, B. Zheng, A. Pelegris, A. Jayarajan, A. Phanishayee, B. Schroeder, and G. Pekhimenko. Benchmarking and analyzing deep neural network training. In *2018 IEEE International Symposium on Workload Characterization (IISWC)*, pages 88–100, 2018.

[151] A. Zymaris. *Adjoint methods for the design of shapes with optimal aerodynamic performance in laminar and turbulent flows*. PhD thesis, National Technical University of Athens, 2010.

[152] A. Zymaris, D. Papadimitriou, K. Giannakoglou, and C. Othmer. Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. *Computers & Fluids*, 38(8):1528–1538, 2009.

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Τομέας Ρευστών
Εργαστήριο Θερμικών Στροβιλομηχανών
Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης

Η Συνεχής Συζυγής Μέθοδος με Συμβατά Σχήματα Διακριτοποίησης για Ροές με Μετάβαση και η Χρήση Βαθέων Νευρωνικών Δικτύων στη Βελτιστοποίηση Μορφής στη Ρευστομηχανική

Διδακτορική Διατριβή
(Εκτεταμένη Περίληψη)

Μαρίνα Γ. Κοντού

Επιβλέπων: Κυριάκος Χ. Γιαννάκογλου,
Καθηγητής ΕΜΠ

Αθήνα, 2023

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Τομέας Ρευστών
Εργαστήριο Θερμικών Στροβιλομηχανών
Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης

Η Συνεχής Συζυγής Μέθοδος με Συμβατά Σχήματα Διακριτοποίησης για Ροές με Μετάβαση και η Χρήση Βαθέων Νευρωνικών Δικτύων στη Βελτιστοποίηση Μορφής στη Ρευστομηχανική

Διδακτορική Διατριβή

Μαρίνα Γ. Κοντού

Εξεταστική Επιτροπή:

1. Κυριάκος Γιαννάκογλου*(Επιβλέπων), Καθηγητής, ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
2. Νικόλαος Αρετάκης*, Αναπληρωτής Καθηγητής, ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
3. Κωνσταντίνος Μαθιουδάκης*, Καθηγητής, ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
4. Σπυρίδων Βουτσινάς, Καθηγητής, ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
5. Francesco Montomoli, Professor, Imperial College London, Department of Aeronautics
6. Βασίλειος Ριζιώτης, Αναπληρωτής Καθηγητής, ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
7. Gilbert Rogé, HDR, Dassault Aviation, Département de Mathématiques Appliquées et Informatique Numérique, Sorbonne Université

* *Μέλος της Συμβουλευτικής Επιτροπής*

Αθήνα, 2023

Περίληψη

Η Διδακτορική αυτή Διατριβή πραγματεύεται την ανάπτυξη, πιστοποίηση και εφαρμογή μεθόδων και λογισμικού για προβλήματα βελτιστοποίησης μορφής στη ρευστοδυναμική. Έμφαση δίνεται (α) στη μαθηματική διατύπωση, προγραμματισμό και πιστοποίηση της συνεχούς συζυγούς μεθόδου για βελτιστοποίηση μορφής σε ροές συμπιεστού ρευστού με μετάβαση από στρωτή σε τυρβώδη ροή, (β) την ανάπτυξη συμβατών σχημάτων διακριτοποίησης για τις εξισώσεις και οριακές συνθήκες της συνεχούς συζυγούς μεθόδου και (γ) τη χρήση Βαθέων Νευρωνικών Δικτύων (ΒΝΔ) για προβλέψεις ροής και την αποτίμηση του κέρδους στη βελτιστοποίηση μορφής. Τα δύο πρώτα θέματα σχετίζονται με μεθόδους αιτιοκρατικής βελτιστοποίησης, ενώ το τελευταίο με τη μείωση κόστους στους Εξελικτικούς Αλγορίθμους.

Στη Διατριβή, χρησιμοποιείται ο οικείος επιλύτης PUMA των εξισώσεων (U)RANS που εκτελείται σε επεξεργαστές καρτών γραφικών και έχει αναπτυχθεί από τη Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης (ΜΠΥΡ&Β) του ΕΜΠ. Ο επιλύτης συμπιεστών τυρβωδών ροών του PUMA επεκτείνεται για να προσομοιώσει και τη μετάβαση από στρωτή σε τυρβώδη ροή. Εφαρμόζεται το μοντέλο μετάβασης $\gamma{-}\tilde{R}e_{\theta t}$, συζευγμένο με τα μοντέλα τύρβης $k{-}\omega$ SST και Spalart–Allmaras. Η πιστοποίηση του μοντέλου μετάβασης πραγματοποιείται σε 2Δ και 3Δ εφαρμογές με διαθέσιμα πειραματικά και υπολογιστικά αποτελέσματα. Αυτές περιλαμβάνουν ροές με μετάβαση σε επίπεδες πλάκες αλλά και σε πτερύγωση διηχητικού στροβίλου. Επιπλέον, στην εξωτερική αεροδυναμική, μελετώνται μια μεμονωμένη αεροτομή και δύο πτέρυγες σε συνθήκες φυσικά στρωτής ροής.

Στην αιτιοκρατική βελτιστοποίηση μορφής, η συνεχής συζυγής μέθοδος αποτελεί τον πιο αποδοτικό τρόπο υπολογισμού των παραγώγων ευαισθησίας μιας συνάρτησης-στόχου ως προς τις μεταβλητές σχεδιασμού. Εδώ, αναπτύσσεται, για πρώτη φορά στη σχετική βιβλιογραφία, η συνεχής συζυγής μέθοδος για το μοντέλο μετάβασης $\gamma{-}\tilde{R}e_{\theta t}$ συζευγμένου με το μοντέλο τύρβης Spalart–Allmaras. Δείχνεται ότι υπερτερεί της συζυγούς μεθόδου με 'παγωμένη μετάβαση' (χωρίς, δηλαδή, την επίλυση του συζυγούς προβλήματος του μοντέλου μετάβασης) ως προς την ακρίβεια υπολογισμού των παραγώγων ευαισθησίας.

Το κύριο πλεονέκτημα της συνεχούς συζυγούς μεθόδου είναι η φυσική εποπτεία των συζυγών ΜΔΕ και των οριακών συνθηκών τους, και, ειδικότερα, οι χαμηλές απαιτήσεις σε μνήμη του σχετικού λογισμικού. Τα σχήματα διακριτοποίησης που χρησιμοποιούνται για τις συζυγείς εξισώσεις, μπορούν να επηρεάσουν σημαντικά την ακρίβεια υπολογισμού των παραγώγων ευαισθησίας, όσο αυτά δεν είναι συμβατά με το διακριτοποιημένο ροϊκό (πρωτεύον) πρόβλημα. Η διακριτή συζυγής μέθοδος εξασφαλίζει αυτή τη συμβατότητα, αλλά με ενδεχομένως υπέρογκες απαιτήσεις μνήμης, ενώ υστερεί στην ερμηνεία των συζυγών εξισώσεων. Η Διατριβή γεφυρώνει το χάσμα ανάμεσα στις δύο παραλλαγές αναπτύσσοντας συμβατά σχήματα διακριτοποίησης (τα οποία εμπνέονται από τη διακριτή μέθοδο) για τις συνεχείς συζυγείς εξισώσεις και τις οριακές τους συνθήκες, εστιάζοντας ταυτόχρονα στη φυσική τους

σημασία. Η νέα συζυγής μέθοδος, επονομαζόμενη *Think-Discrete Do-Continuous* (*TDDC*), αναπτύσσεται για τον επιλύτη εξισώσεων υπερβολικού τύπου συμπιεστών ρευστών του PUMA. Σύμφωνα με αυτή, οι παράγωγοι ευαισθησίας υπολογίζονται με τη λογική της συνεχούς αλλά την ακρίβεια της διακριτής μεθόδου με χαμηλές απαιτήσεις μνήμης και με φυσική εποπτεία του συζυγούς προβλήματος. Η *TDDC* συζυγής μέθοδος υπερτερεί της συζυγούς μεθόδου με τα 'κλασικά' σχήματα διακριτοποίησης και οι ακριβέστερες παράγωγοι οδηγούν σε καλύτερη σύγκλιση της βελτιστοποίησης.

Η συνεχής συζυγής μέθοδος για ροές με φαινόμενα μετάβασης από στρωτή σε τυρβώδη, με τα προτεινόμενα *TDDC* σχήματα διακριτοποίησης, χρησιμοποιείται σε δύο βελτιστοποιήσεις μορφής βιομηχανικού ενδιαφέροντος, μιας μεμονωμένης πτέρυγας και μιας πτέρυγας αεροσκάφους.

Στην κατεύθυνση της μείωσης του υπολογιστικού κόστους σε προβλήματα βελτιστοποίησης, στο τελευταίο τμήμα της, η Διατριβή εστιάζει στη χρήση ΒΝΔ για προβλέψεις ροής και χρήση σε βελτιστοποίηση μορφής στην αεροδυναμική. Μια πρώτη χρήση των ΒΝΔ είναι ως υποκατάστατα της επίλυσης των εξισώσεων μοντέλων τύρβης και μετάβασης παρέχοντας το πεδίο της τυρβώδους συνεκτικότητας κατά την επαναληπτική επίλυση των εξισώσεων RANS. Το βέλτιστο ΒΝΔ (αρχιτεκτονική και δεδομένα εισόδου) απορρέει από ένα Εξελικτικό Αλγόριθμο Υποβοηθούμενο από Μεταπρότυπα (Metamodel–Assisted Evolutionary Algorithm, MAEA). Ο νέος υπολογιστικά αποδοτικός επιλύτης χρησιμοποιείται για τη βελτιστοποίηση μορφής μεμονωμένης αεροτομής, πτερυγίου στροβιλομηχανής και μοντέλου αυτοκινήτου με τον MAEA του λογισμικού βελτιστοποίησης EASY.

Μια δεύτερη προσέγγιση είναι η χρήση των ΒΝΔ σε πολυπεδιακά προβλήματα και προβλήματα βελτιστοποίησης στροβιλομηχανών που εμπλέκουν αλληλεπίδραση περιστρεφόμενων και σταθερών πτερυγώσεων. Προτείνεται η χρήση της δίκλαδης αρχιτεκτονικής $\lambda-$DNN, με χαρακτηριστικό γνώρισμά της τη χωριστή επεξεργασία δεδομένων εισόδου διαφορετικής φύσης. Εδώ, χρησιμοποιείται σε προβλήματα βελτιστοποίησης συζευγμένης μεταφοράς θερμότητας αντικαθιστώντας τον επιλύτη της εξίσωσης αγωγής της θερμότητας για το στερεό υλικό. Ο νέος οικονομικότερος επιλύτης χρησιμοποιείται ως λογισμικό αξιολόγησης για τη βελτιστοποίηση μορφής ενός αγωγού ψύξης και ενός εσωτερικά ψυχόμενου πτερυγίου στροβίλου. Τέλος, προτείνεται ένα υποκατάστατο μοντέλο βασιζόμενο σε ΒΝΔ στο πλαίσιο της τεχνικής επιφάνειας ανάμειξης μεταξύ περιστρεφόμενων και σταθερών πτερυγώσεων για εφαρμογή σε προβλήματα βελτιστοποίησης πτερυγώσεων στροβιλομηχανών. Συνολικά, η χρήση των ΒΝΔ ως υποκατάστατα λογισμικά μειώνει το υπολογιστικό κόστος βελτιστοποιήσεων μορφής με MAEA και της πλατφόρμας EASY, ειδικότερα.

Λέξεις Κλειδιά: Αεροδυναμική Βελτιστοποίηση Μορφής, Συνεχής Συζυγής Μέθοδος, Σχήματα Διακριτοποίησης των Συζυγών Εξισώσεων, Μοντέλα Μετάβασης, Βαθέα Νευρωνικά Δίκτυα, Μεταπρότυπα Μοντέλων Τύρβης, Εξελικτικοί Αλγόριθμοι

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

Η ανάπτυξη των αερομεταφορών οδηγεί σε αυξημένη κατανάλωση καυσίμου άρα και σε αυξημένες εκπομπές αερίων του θερμοκηπίου. Σήμερα γίνονται σημαντικές προσπάθειες για την επίτευξη των στόχων περί αειφόρου ανάπτυξης το 2030[1] και κλιματικής ουδετερότητας το 2050[2]. Στην κατεύθυνση αυτή, γίνεται προσπάθεια τα τελευταία χρόνια για το σχεδιασμό πτερύγων αεροσκαφών με φυσικά στρωτή ροή. Η βασικά ιδέα είναι η αύξηση της επιφάνειας της πτέρυγας όπου η ροή είναι στρωτή κάτι το οποίο οδηγεί σε μειωμένη οπισθέλκουσα, μειωμένη κατανάλωση καυσίμου και, άρα, περιβαλλοντική βιωσιμότητα. Επομένως, κρίνεται απαραίτητη η ανάπτυξη υπολογιστικών εργαλείων που μπορούν να προβλέψουν με ακρίβεια τα φαινόμενα μετάβασης με χαμηλές υπολογιστικές απαιτήσεις. Τα φαινόμενα αυτά θα πρέπει φυσικά να ληφθούν υπόψη και να ενσωματωθούν και στη βελτιστοποίηση. Ταυτόχρονα, η Τεχνητή Νοημοσύνη και τα Βαθέα Νευρωνικά Δίκτυα (ΒΝΔ) μπορούν να εντοπίζουν ιδιότητες και μοτίβα και να πραγματοποιούν ακριβείς προβλέψεις με χαμηλό υπολογιστικό κόστος. Πέρα από τις συνήθεις εφαρμογές, [51], η χρήση τους σε προβλήματα ρευστοδυναμικής πρέπει να θεωρείται ως φυσικό επακόλουθο, [45].

Η Διατριβή αυτή εστιάζει στα προαναφερθέντα στοιχεία και στοχεύει στην ανάπτυξη αξιόπιστων και οικονομικά αποτελεσματικών αριθμητικών μεθόδων και υπολογιστικών εργαλείων για βιομηχανικά προβλήματα ανάλυσης και βελτιστοποίησης.

1.1 Μοντελοποίηση Μετάβασης από Στρωτή σε Τυρβώδη Ροή

Τα μοντέλα τύρβης της βιβλιογραφίας μπορούν να εκτιμήσουν με ακρίβεια την επίδραση της τύρβης σε ευρύ φάσμα εφαρμογών στην πλειονότητα των σύγχρονων λογισμικών Υπολογιστικής Ρευστοδυναμικής (ΥΡΔ). Ωστόσο, κάτι τέτοιο δεν ισχύει για τη μοντελοποίηση της μετάβασης της ροής από στρωτή σε τυρβώδη. Πέραν του επιπλέον κόστους, η κύρια αιτία της απουσίας της μοντελοποίησης της μετάβασης στις προσομοιώσεις με λογισμικό ΥΡΔ είναι πώς αυτή συμβαίνει μέσω διαφορετικών μηχανισμών ανάλογα με την εφαρμογή, καθιστώντας δύσκολη τη συμπερίληψή τους σε ένα ενιαίο μοντέλο.

Οι τρεις κύριοι μηχανισμοί της μετάβασης είναι η φυσική [42], η υποβοηθούμενη [34] και η μετάβαση λόγω (στρωτής) αποκόλλησης [31]. Η φυσική μετάβαση εμφα-

[1] https://undocs.org/A/RES/70/1

[2] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52019DC0640

νίζεται κυρίως σε καθαρά αεροδυναμικές ροές (πτέρυγες) λόγω κυμάτων η ενίσχυση των οποίων οδηγεί σε πλήρως αναπτυγμένη τυρβώδη ροή. Η υποβοηθούμενη μετάβαση εμφανίζεται στις στροβιλομηχανές λόγω της υψηλής έντασης της τύρβης στην είσοδο, ενώ στη μετάβαση λόγω αποκόλλησης, η ροή αποκολλάται ως στρωτή και, ενδεχομένως, προσκολλάται ως τυρβώδης και εμφανίζεται κυρίως σε αεριοστροβίλους, συμπιεστές και στροβίλους χαμηλής πίεσης. Στη βιβλιογραφία, έχουν γίνει ποικίλες προσπάθειες για τη μοντελοποίηση της μετάβασης της ροής. Αυτές περιλαμβάνουν αλγεβρικές εκφράσεις, [9], μοντέλα που βασίζονται σε πληροφορία του οριακού στρώματος, [49] ή που επιλύουν εξισώσεις μεταφοράς, [23, 6] αλλά και μοντέλα υψηλής πιστότητας, LES και DNS, [29].

Στο πλήρες κείμενο της Διατριβής παρουσιάζονται λεπτομερώς οι ανωτέρω τρόποι για την εκτίμηση της μετάβασης της ροής. Σε κάθε περίπτωση, είναι αναγκαία η ύπαρξη ενός μοντέλου συμβατού με ένα σύγχρονο/παραλληλοποιημένο λογισμικό ΥΡΔ, που θα εμπεριέχει όλους τους μηχανισμούς μετάβασης και θα είναι λογικού κόστους για εφαρμογές μεγάλης κλίμακας. Ένα τέτοιο μοντέλο (που χρησιμοποιείται σε αυτή τη Διατριβή) είναι το μοντέλο δύο εξισώσεων $\gamma - \tilde{R}e_{\theta t}$ [24, 22, 7, 39].

1.2 Μέθοδοι Βελτιστοποίησης Ρευστοδυναμικών Μορφών

Το προβλήματα βελτιστοποίησης στη ρευστοδυναμική επιλύονται είτε με στοχαστικές, πληθυσμιακές μεθόδους είτε με αιτιοκρατικές μεθόδους που βασίζονται σε παραγώγους. Οι πρώτες αναζητούν τη βέλτιστη λύση με συστηματικά τυχαίο τρόπο, ενώ οι δεύτερες αξιοποιούν την έννοια της παραγώγου της συνάρτησης-στόχου.

Η πιο αντιπροσωπευτική μέθοδος στοχαστικής βελτιστοποίησης είναι οι Εξελικτικοί Αλγόριθμοι (Evolutionary Algorithms, EAs). Οι EAs χειρίζονται πληθυσμούς υποψηφίων λύσεων που εξελίσσονται από γενιά σε γενιά μέσω τελεστών διασταύρωσης, εξέλιξης και ελιτισμού. Η βασική ιδέα είναι πως τα άτομα που προσαρμόζονται καλύτερα στο περιβάλλον τους είναι πιο πιθανό να αναπαραχθούν και οι απόγονοι της νέας γενιάς να είναι ακόμα καλύτεροι. Το κύριο πλεονέκτημά τους είναι πως μπορούν να διαχειριστούν οποιοδήποτε πρόβλημα βελτιστοποίησης χωρίς να απαιτείται πρόσβαση στον πηγαίο κώδικα. Αρκεί ένα λογισμικό στη λογική του μαύρου κουτιού για την αξιολόγηση των υποψήφιων λύσεων. Οι EAs μπορούν εύκολα να χειριστούν προβλήματα πολλών στόχων και να εντοπίσουν το ολικό ακρότατο εφόσον έχει αξιολογηθεί επαρκής αριθμός υποψηφίων λύσεων. Συνδέονται, όμως, με υψηλό υπολογιστικό κόστος βελτιστοποίησης λόγω του μεγάλου αριθμού κλήσεων στο λογισμικού αξιολόγησης που απαιτείται για τον εντοπισμού του ολικού ακρότατου. Σε προβλήματα μεγάλης κλίμακας, λ.χ. ρευστοδυναμικής, η χρήση των EAs είναι συχνά απαγορευτική. Έτσι, αναπτύχθηκαν τεχνικές βασισμένες σε τεχνητά νευρωνικά δίκτυα ή άλλα μοντέλα που λειτουργούν ως μεταπρότυπα, δημιουργώντας τους Εξελικτικούς Αλγορίθμους Υποβοηθούμενους από Μεταπρότυπα (Metamodel-Assisted EAs, MAEAs), [19, 20]. Οι τεχνικές αυτές υπάρχουν στο λογισμικό βελτιστοποίησης EASY (Evolutionary Algorithm SYstem) που έχει αναπτυχθεί στη ΜΠΥΡ&Β, [2], και

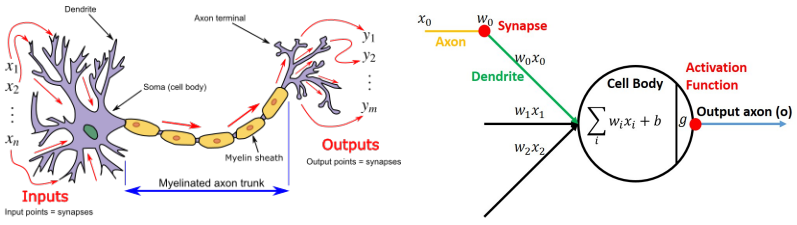
χρησιμοποιείται στη Διατριβή.

Αντίθετα, οι μέθοδοι βελτιστοποίησης που βασίζονται στην κλίση της συνάρτησης-στόχου είναι περισσότερο αποδοτικές υπολογιστικά, ωστόσο, μπορούν να εγκλωβιστούν σε τοπικό ακρότατο (ανάλογα με το σημείο εκκίνησης). Βασικό τους χαρακτηριστικό είναι το κόστος υπολογισμού των παραγώγων ευαισθησίας. Σε μεθόδους όπως οι πεπερασμένες διαφορές, η μέθοδος των μιγαδικών μεταβλητών [3] και η αυτόματη διαφόριση [12], το κόστος είναι ανάλογο του πλήθους των μεταβλητών σχεδιασμού (N). Σε βιομηχανικές εφαρμογές, το πλήθος N είναι αυξημένο οπότε αναζητείται μια μέθοδος όπου το κόστος υπολογισμού των παραγώγων ευαισθησίας είναι ανεξάρτητο του N. Μια τέτοια μέθοδος είναι η συζυγής (adjoint), [27, 40, 18].

Η συζυγής μέθοδος χωρίζεται σε δύο εκδοχές: τη διακριτή [14, 11] όπου η διακριτοποίηση των εξισώσεων ροής προηγείται της διαφόρισης και στη συνεχή [36, 38] όπου συμβαίνει το αντίστροφο. Το κύριο πλεονέκτημα της συνεχούς συζυγούς μεθόδου είναι η φυσική εποπτεία του συζυγούς προβλήματος και οι χαμηλές απαιτήσεις σε μνήμη. Όμως, η επιλογή των σχημάτων διακριτοποίησης επηρεάζει σημαντικά την ακρίβεια υπολογισμού των παραγώγων ευαισθησίας αφού αυτά δεν είναι κατ' ανάγκη συμβατά με το διακριτοποιημένο πρωτεύον πρόβλημα. Αντίθετα, η διακριτή συζυγής μέθοδος χαρακτηρίζεται από συνέπεια στον υπολογισμό των παραγώγων με ενδεχόμενες αυξημένες απαιτήσεις σε μνήμη. Στη Διατριβή αναπτύσσεται η συνεχής συζυγής μέθοδος. Σε τυρβώδεις ροές, η συζυγής μέθοδος για το μοντέλο τύρβης είναι απαραίτητη για τον ακριβή υπολογισμό των παραγώγων ευαισθησίας τόσο στη διακριτή [26] όσο και στη συνεχή, [53] εκδοχή. Για τις ροές με μετάβαση, η σχετική βιβλιογραφία είναι περιορισμένη και βασίζεται αποκλειστικά στη διακριτή συζυγή μέθοδο, [15]. Η ανάπτυξη της συνεχούς συζυγούς μεθόδου για ροές με μετάβαση πραγματοποιείται για πρώτη φορά στη σχετικά βιβλιογραφία.

1.3 Τα Βαθέα Νευρωνικά Δίκτυα στη Ρευστοδυναμική

Τα Βαθέα Νευρωνικά Δίκτυα (ΒΝΔ), [8], αποτελούν ένα σύστημα αλγορίθμων εμπνευσμένων από τον ανθρώπινο εγκέφαλο και το νευρικό σύστημα των θηλαστικών και μιμούνται τον τρόπο επεξεργασίας και μετάδοσης της πληροφορίας σε αυτά.



(α΄) Βιολογικός Νευρώνας **(β΄)** Τεχνητός Νευρώνας

Σχήμα 1.1: (α΄) Τυπική δομή βιολογικού νευρώνα (από https://commons.wikimedia.org/w/index.php?curid=72816083). (β΄) Δομή τεχνητού νευρώνα.

Ένας βιολογικός νευρώνας αποτελείται από συνάψεις, δενδρίτες, το κυτταρικό σώμα και τις νευροαξονικές απολήξεις, Σχ. 1.1α΄. Οι νευρώνες είναι υπεύθυνοι για την επεξεργασία και τη μετάδοση της πληροφορίας. Κάθε νευρώνας λαμβάνει σήματα από άλλους νευρώνες μέσω των δενδριτών, το κυτταρικό σώμα αθροίζει τα εισερχόμενα σήματα και, αν η άθροιση αυτή υπερβαίνει ένα όριο, μεταφέρεται ως έξοδος στις νευροαξονικές απολήξεις που με τη σειρά τους το μεταδίδουν σε άλλα κύτταρα. Ανάλογη είναι η δομή ενός τεχνητού νευρώνα, Σχ. 1.1β΄. Οι συνάψεις του βιολογικού νευρώνα αντιστοιχούν σε σταθμισμένα σήματα που λαμβάνονται από άλλους νευρώνες. Τα σήματα μεταφέρονται μέσω των δενδριτών στο κυρίως σώμα όπου αθροίζονται, και αν υπερβαίνουν ένα κατώφλι τότε αποστέλλονται ως έξοδος. Το κατώφλι μοντελοποιείται με τη χρήση συναρτήσεων ενεργοποίησης οι οποίες εισάγουν στοιχεία μη-γραμμικότητας, απαραίτητα για το χειρισμό πολύπλοκων προβλημάτων.

Τα τελευταία χρόνια, η χρήση των ΒΝΔ έχει επεκταθεί και στον τομέα της αεροδυναμικής βελτιστοποίησης ως εναλλακτικός τρόπος μείωσης του υπολογιστικού κόστους. Ενδεικτικά, τα ΒΝΔ έχουν χρησιμοποιηθεί για την πρόβλεψη πεδίων ροής, [20] καθώς και ως μέσο εξαγωγής δεδομένων από προσομοιώσεις υψηλής πιστότητας και χρήσης αυτών για μοντελοποίηση της τύρβης, [10]. Ακόμα, έχουν αναπτυχθεί μέθοδοι για την εύρεση και την αποτελεσματική εκπαίδευση των ΒΝΔ, [25].

1.4 Δομή της Διατριβής

Στο Κεφάλαιο 2 παρουσιάζεται η επέκταση του οικείου λογισμικού PUMA (Parallel Unstructured Adjoint Multi-row) της ΜΠΥΡ&Β για ροές με μετάβαση από στρωτή σε τυρβώδη με το μοντέλο $\gamma-\tilde{Re}_{\theta t}$, ενώ στο Κεφάλαιο 3 πραγματοποιείται η ανάπτυξη και πιστοποίηση της συνεχούς συζυγούς μεθόδου διαφορίζοντας το ίδιο μοντέλο. Έπειτα, στο Κεφάλαιο 4, αναπτύσσονται και προτείνονται συμβατά σχήματα διακριτοποίησης για τις συνεχείς συζυγείς εξισώσεις σύμφωνα με μια νέα συζυγή μέθοδο, την *Think-Discrete Do-Continuous* (*TDDC*). Η συνεχής συζυγής μέθοδος για ροές με μετάβαση με τα *TDDC* σχήματα διακριτοποίησης χρησιμοποιείται στη βελτιστοποίηση μορφής δύο βιομηχανικών εφαρμογών στο Κεφάλαιο 5. Τα δύο επόμενα κεφάλαια σχετίζονται με τη χρήση ΒΝΔ για πρόβλεψη ροής στη βελτιστοποίηση μορφής με ΕΑ. Στο Κεφάλαιο 6, τα ΒΝΔ υποκαθιστούν τα μοντέλα τύρβης και μετάβασης παρέχοντας το πεδίο της τυρβώδους συνεκτικότητας κατά την επίλυση των εξισώσεων ροής. Στο Κεφάλαιο 7, προτείνεται η χρήση της δίκλαδης αρχιτεκτονικής $\lambda-$DNN για πολυπεδιακά προβλήματα βελτιστοποίησης, καθώς και ένα υποκατάστατο για την τεχνική επιφάνειας ανάμειξης μεταξύ περιστρεφόμενων και σταθερών πτερυγώσεων σε προβλήματα βελτιστοποίησης πτερυγώσεων στροβιλομηχανών. Τέλος, στο Κεφάλαιο 8 εξάγονται συμπεράσματα και προτείνονται ιδέες για μελλοντική έρευνα.

Κεφάλαιο 2

Οι Εξισώσεις Navier–Stokes και η Αριθμητική τους Επίλυση

Το κυρίως θέμα της Διατριβής είναι η ανάπτυξη και πιστοποίηση υπολογιστικών μεθόδων και αριθμητικών εργαλείων για προβλήματα βελτιστοποίησης μορφής στη ρευστοδυναμική. Οι βελτιστοποιήσεις βασίζονται τόσο στην κλίση της συνάρτησης-στόχου όσο και σε εξελικτικούς αλγορίθμους. Στην πρώτη προσέγγιση, αναπτύσσεται η συνεχής συζυγής μέθοδος με συμβατά σχήματα διακριτοποίησης, ενώ στη δεύτερη χρησιμοποιούνται ΒΝΔ για τη μείωση του υπολογιστικού κόστους του λογισμικού αξιολόγησης. Πριν την ανάπτυξη της συνεχούς συζυγούς μεθόδου και των υπόλοιπων εργαλείων, παρουσιάζεται το μοντέλο ροής (εξισώσεις, σχήματα διακριτοποίησης και αριθμητικής επίλυσης) που χρησιμοποιείται για την αξιολόγηση κάθε υποψήφιας λύσης. Το μοντέλο ροής αποτελείται από τις εξισώσεις Navier–Stokes για συμπιεστές ροές, τα μοντέλα τύρβης Spalart–Allmaras και $k-\omega$ SST καθώς και μοντέλο μετάβασης $\gamma - \tilde{Re}_{\theta t}$. Η αριθμητική επίλυση της ροής γίνεται με το οικείο λογισμικό PUMA της ΜΠΥΡ&Β που έχει αναπτυχθεί σε προηγούμενες Διατριβές ([52],[5],[46],[47],[33]), εκτελείται σε επεξεργαστές καρτών γραφικών και το οποίο, εδώ, αναβαθμίζεται περαιτέρω.

2.1 Οι Εξισώσεις URANS για Συμπιεστές Ροές

Σε ένα σύστημα συντεταγμένων $O(x_1\ x_2\ x_3)$ που περιστρέφεται με σταθερή γωνιακή ταχύτητα $\omega_m\ (m=1,2,3)$, οι εξισώσεις Navier–Stokes για συμπιεστές ροές είναι

$$R_n^{\text{MF}} = \frac{\partial U_n}{\partial t} + \frac{\partial f_{nk}^{\text{inv}}}{\partial x_k} - \frac{\partial f_{nk}^{\text{vis}}}{\partial x_k} + S_n = 0 \tag{2.1}$$

Η Εξ. 2.1 επιλύεται για τις συντηρητικές μεταβλητές $U_n = \left[\rho\ \ \rho v_1^A\ \ \rho v_2^A\ \ \rho v_3^A\ \ \rho E\right]^T$, όπου ρ η πυκνότητα του ρευστού, $v_m^A\ (m=1,2,3)$ οι συνιστώσες της ταχύτητας στο απόλυτο/αδρανειακό σύστημα αναφοράς και E η ενέργεια του ρευστού ανά μονάδα μάζας. Στην Εξ. 2.1 $f_{nk}^{\text{inv}} = \left[\rho v_k^R\ \ \rho v_k^R v_1^A + p\delta_{1k}\ \ \rho v_k^R v_2^A + p\delta_{2k}\ \ \rho v_k^R v_3^A + p\delta_{3k}\ \ \rho v_k^R h_t\right]$ είναι τα διανύσματα ατριβούς και $f_{nk}^{\text{vis}} = \left[0\ \ \tau_{1k}\ \ \tau_{2k}\ \ \tau_{3k}\ \ v_\ell^A \tau_{\ell k} + q_k\right]$ τα διανύσματα συνεκτικής ροής. Οι όροι πηγής εκφράζονται ως $S_n = \left[0\ \ \rho\varepsilon_{1\ell k}\omega_\ell v_k^A\ \ \rho\varepsilon_{2\ell k}\omega_\ell v_k^A\ \ \rho\varepsilon_{3\ell k}\omega_\ell v_k^A\ \ 0\right]$ όπου p η στατική πίεση. Οι συνιστώσες της σχετικής ταχύτητας v_m^R συνδέονται με

εκείνες της απόλυτης v_m^A μέσω της έκφρασης $v_m^A = v_m^R + v_m^F$, όπου v_m^F οι συνιστώσες της ταχύτητας του συστήματος αναφοράς. Στο πλήρες κείμενο της Διατριβής παρουσιάζονται επιπλέον σχέσεις που συμπληρώνουν τις ανωτέρω εξισώσεις.

Στη Διατριβή, η τυρβώδης συνεκτικότητα μ_t υπολογίζεται είτε μέσω του μοντέλου τύρβης μίας εξίσωσης των Spalart–Allmaras [44] είτε μέσω του μοντέλου δύο εξισώσεων του Menter $k-\omega$ SST [32]. Η διαφορική εξίσωση των Spalart–Allmaras επιλύεται για το τυρβώδες πεδίο $\tilde{\nu}$ και εκφράζεται ως

$$\begin{aligned} R^{\tilde{\nu}} = & \frac{\partial\left(\rho\tilde{\nu}\right)}{\partial t} + \frac{\partial\left(\rho\tilde{\nu}v_k^R\right)}{\partial x_k} - \frac{\rho}{\mathrm{Re}_0\,\sigma}\left\{\frac{\partial}{\partial x_k}\left[\left(\nu+\tilde{\nu}\right)\frac{\partial\tilde{\nu}}{\partial x_k}\right] + c_{b_2}\frac{\partial\tilde{\nu}}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\right\} \\ & - \rho c_{b_1}\left(1-f_{t_2}\right)\tilde{S}\tilde{\nu} + \frac{\rho}{\mathrm{Re}_0}\left(c_{w_1}f_w - \frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\left(\frac{\tilde{\nu}}{\Delta}\right)^2 = 0 \end{aligned} \tag{2.2}$$

όπου με Δ συμβολίζεται η απόσταση από το στερεό τοίχωμα. Μετά την επίλυση της Εξ. 2.2, $\mu_t = \rho\tilde{\nu}f_{v_1}$. Σύμφωνα με το μοντέλο τύρβης $k-\omega$ SST, [32], επιλύονται δύο διαφορικές εξισώσεις για την τυρβώδη κινητική ενέργεια k και το ρυθμό καταστροφής της τύρβης ω, οι οποίες εκφράζονται ως

$$\begin{aligned} R^k &= \frac{\partial\left(\rho k\right)}{\partial t} + \frac{\partial\left(\rho k v_k^R\right)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[\left(\mu+\sigma_k\mu_t\right)\frac{\partial k}{\partial x_k}\right] - \tilde{P}_\kappa + \beta^*\rho k\omega = 0 \\ R^\omega &= \frac{\partial\left(\rho\omega\right)}{\partial t} + \frac{\partial\left(\rho\omega v_k^R\right)}{\partial x_k}\frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[\left(\mu+\sigma_\omega\mu_t\right)\frac{\partial\omega}{\partial x_k}\right] \\ & - \gamma\frac{P_\kappa}{\nu_t} + \beta\rho\omega^2 - 2(1-F_1)\frac{\rho\sigma_{\omega_2}}{\omega}\frac{\partial k}{\partial x_k}\frac{\partial\omega}{\partial x_k} = 0 \end{aligned} \tag{2.3}$$

και $\mu_t = \mathrm{Re}_0\frac{\rho a_1 k}{\max\left(a_1\omega, SF_2\right)}$. Οι εκφράσεις και οι σταθερές που συμπληρώνουν τα μοντέλα Spalart–Allmaras και $k-\omega$ SST δίνονται στο πλήρες κείμενο της Διατριβής.

Προκειμένου να ληφθούν υπόψη φαινόμενα μετάβασης από στρωτή σε τυρβώδη ροή, τα δύο μοντέλα τύρβης συνδυάζονται με το μοντέλο μετάβασης $\gamma-\tilde{Re}_{\theta t}$, [24]. Σύμφωνα με αυτό επιλύονται δύο επιπλέον διαφορικές εξισώσεις για τη διάλειψη της μετάβασης (transition intermittency) γ και τον αριθμό Reynolds του πάχους ορμής (transition momentum-thickness Reynolds) $\tilde{Re}_{\theta t}$,

$$\begin{aligned} R^\gamma &= \frac{\partial\left(\rho\gamma\right)}{\partial t} + \frac{\partial\left(\rho v_k^A\gamma\right)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma}{\partial x_k}\right] - P_\gamma + E_\gamma = 0 \\ R^{\tilde{Re}_{\theta t}} &= \frac{\partial\left(\rho\tilde{Re}_{\theta t}\right)}{\partial t} + \frac{\partial\left(\rho v_k^A\tilde{Re}_{\theta t}\right)}{\partial x_k} - \frac{1}{\mathrm{Re}_0}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\right] - P_{\theta,t} - D_{SCF} = 0 \end{aligned} \tag{2.4}$$

Η αλληλεπίδραση του αρχικού μοντέλου $\gamma-\tilde{Re}_{\theta t}$, [24, 22], γίνεται με το μοντέλο τύρβης $k-\omega\ SST$. Κατάλληλες τροποποιήσεις στους όρους πηγής, [7, 39], οδήγησαν στη σύζευξή του με το μοντέλο Spalart–Allmaras. Συνολικά, στη Διατριβή, εφαρμόζονται τέσσερις εκδοχές του μοντέλου, η αρχική που συνεργάζεται με το μοντέλο τύρβης $k-\omega\ SST$ και τρεις που συνεργάζονται με το μοντέλο των Spalart–Allmaras, και αναφέρονται ως SST-2003-LM2015 [24, 22], SA-noft2-Gamma-Retheta[7], SA-LM2015 [39] και SA-sLM2015 [39]. Οι εκφράσεις των όρων πηγής για τα μοντέλα SST-2003-LM2015, SA-noft2-Gamma-Retheta και SA-LM2015 βρίθουν από τελεστές ελάχιστου/μέγιστου και από πολύκλαδες συναρτήσεις. Αντίθετα, στο μοντέλο μετάβασης SA-sLM2015 οι όροι αυτοί έχουν αντικατασταθεί από λείες, συνεχείς και πλήρως διαφορίσιμες εκφράσεις με στόχο τη βελτίωση της αριθμητικής ευστάθειας, [39]. Κάτι τέτοιο είναι φυσικά ευπρόσδεκτο λόγω της επικείμενης ανάπτυξης της συνεχούς συζυγούς μεθόδου. Εδώ, ενδεικτικά, παρουσιάζονται οι εκφράσεις των όρων πηγής (P_γ, E_γ, $P_{\theta,t}$ και D_{SCF}) για την εκδοχή SA-sLM2015, στο πλήρες κείμενο αναφέρονται αναλυτικά οι εκφράσεις και για τα υπόλοιπες εκδοχές του μοντέλου μετάβασης $\gamma-\tilde{Re}_{\theta t}$ καθώς και οι απαραίτητες τροποποιήσεις στα μοντέλα τύρβης.

Οι όροι πηγής P_γ και E_γ για την εξίσωση του γ εκφράζονται ως

$$\begin{aligned} P_\gamma &= \rho c_{\alpha_1} F_{length} F_{onset} \left[\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\right] \sqrt{\gamma}\,(1-c_{e_1}\gamma) \\ E_\gamma &= \rho c_{\alpha_2} F_{turb} \left[\phi_{-300}\left(\zeta, \frac{M\sqrt{MRe}}{20}\right)\right] \gamma\,(c_{e_2}\gamma - 1) \end{aligned} \tag{2.5}$$

όπου $\phi_p(\alpha,\beta)$ είναι μία διαφορίσιμη εκδοχή του ελάχιστου και του μέγιστου δύο μεταβλητών α,β (με $p>0$ για το μέγιστο και $p<0$ για το ελάχιστο), ζ ο στροβιλισμός, M ο αριθμός Mach υπολογισμένος με τις επ' άπειρον συνθήκες και Re ο αριθμός Reynolds. Οι όροι F_{onset}, F_{turb}, F_{length} και $Re_{\theta c}$ δίνονται από

$$\begin{aligned} &F_{onset} = \frac{\tanh\left[6\left(F_{onset1}-1.35\right)\right]+1}{2}, \quad F_{onset1} = \sqrt{\left(\frac{Re_S}{2.6Re_{\theta c}}\right)^2 + (R_T)^2}, \quad Re_S = \frac{\rho\Delta^2 S}{\mu}\,\mathrm{Re}_0 \\ &F_{length} = 44 - \frac{44-\left(0.50-3\cdot10^{-4}\left(\tilde{Re}_{\theta t}-596\right)\right)}{\left(1+F_{length1}\right)^{1/6}}, \quad F_{length1} = \exp\left(-3\cdot10^{-2}\left(\tilde{Re}_{\theta t}-460\right)\right) \\ &Re_{\theta c} = 0.67\tilde{Re}_{\theta t} + 24\sin\left(\frac{\tilde{Re}_{\theta t}}{240}+0.5\right)+14, \quad F_{turb} = (1-F_{onset})\exp(-R_T)\,, \; R_T = \frac{\mu_t}{\mu} \end{aligned} \tag{2.6}$$

Στην εξίσωση του $\tilde{R}e_{\theta t}$, οι όροι πηγής $P_{\theta,t}$ και D_{SCF} εκφράζονται ως

$$P_{\theta,t}=\rho\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{R}e_{\theta t}\right)(1-F_{\theta t})\,,\quad D_{SCF}=c_{\theta,t}\frac{\rho}{\mathcal{T}}c_{crossflow}\phi_{-300}\left(Re_{SCF}-\tilde{R}e_{\theta t},0\right)F_{\theta t} \tag{2.7}$$

Οι εκφράσεις για τους όρους $F_{\theta t}$ και $\mathcal{T}$ δίνονται ως

$$\begin{aligned} &F_{\theta t}=F_{wake}\exp\left[-\left(\frac{\Delta}{\delta}\right)^4\right],\quad F_{wake}=\exp\left[-{}^{Re_S}/_{1.e^6}\right]\\ &\mathcal{T}=\frac{500\mu}{\rho\left|\mathbf{U}\right|^2}\frac{1}{\mathrm{Re}_0},\quad \delta=\frac{375\zeta\mu\tilde{R}e_{\theta t}\Delta}{\rho\left|\mathbf{U}\right|^2}\frac{1}{\mathrm{Re}_0},\quad \left|\mathbf{U}\right|=\sqrt{v^A_\ell v^A_\ell},\ (\ell=1,\ldots,3) \end{aligned} \tag{2.8}$$

Ο όρος $Re^{eq}_{\theta,t}$ εκφράζεται συναρτήσει της ισχύος της τύρβης (Tu) και της παραμέτρου για την κλίση της πίεσης λ_θ ως

$$Tu=100\frac{\sqrt{2k/3}}{\left|\mathbf{U}\right|},\quad \lambda_\theta=\frac{\rho\theta^2}{\mu}\frac{d\left|\mathbf{U}\right|}{ds}\mathrm{Re}_0,\quad \frac{d\left|\mathbf{U}\right|}{ds}=\frac{v^A_m v^A_n}{\left|\mathbf{U}\right|^2}\frac{\partial v^A_m}{\partial x_n} \tag{2.9}$$

$$Re^{eq}_{\theta,t}=\begin{cases}\left(1173.51-589.428Tu+0.2196/Tu^2\right)F\left(\lambda_\theta\right) & ,\text{if } Tu\leq 1.3\\ 331.5\left(Tu-0.5668\right)^{-0.671}F\left(\lambda_\theta\right) & ,\text{if } Tu>1.3\end{cases} \tag{2.10}$$

$$\begin{aligned} &F(\lambda_\theta)_1=1+0.275\left[1-\exp^{[-35\lambda_\theta]}\right]\exp^{-\frac{Tu}{0.5}},\quad F(\lambda_\theta)_2=\phi_{300}\left(F\left(\lambda_\theta\right)_1,1\right)\\ &F(\lambda_\theta)_3=1+\left[12.986\lambda_\theta+123.66\lambda_\theta^2+405.689\lambda_\theta^3\right]\exp^{\left[-(Tu/1.5)^{1.5}\right]},\ F(\lambda_\theta)=\phi_{-300}(F(\lambda_\theta)_2,F(\lambda_\theta)_3) \end{aligned} \tag{2.11}$$

Η έκφραση του $Re^{eq}_{\theta,t}$ είναι μια κλειστή συνάρτηση του θ λόγω της ύπαρξης του λ_θ αφού $Re^{eq}_{\theta,t}=\frac{\rho|\mathbf{U}|\theta}{\mu}\mathrm{Re}_0$. Οι Εξ. 2.9-2.11 επιλύονται επαναληπτικά μέσω του θ. Για λόγους αριθμητικής ευστάθειας, ισχύουν οι παρακάτω περιορισμοί για τα λ_θ, Tu και $Re^{eq}_{\theta,t}$: $-0.1\leq\lambda_\theta\leq 0.1$, $Tu\geq 0.027$, $Re^{eq}_{\theta,t}\geq 20$.

Ο όρος D_{SCF} εμπλέκει την τραχύτητα της επιφάνειας (h_{rms}) και εκφράζεται ως

$$\begin{aligned} &Re_{SCF}=\frac{\rho\theta_t\left|\mathbf{U}\right|}{0.82\mu}\mathrm{Re}_0=-35.088\ln\left(\frac{h_{rms}}{\theta_t}\right)+319.51+f\left(\mathcal{D}H_{CF+}\right)-f\left(\mathcal{D}H_{CF-}\right)\\ &f\left(\mathcal{D}H_{CF+}\right)=6200\mathcal{D}H_{CF+}+5000\left(\mathcal{D}H_{CF+}\right)^2,\quad f\left(\mathcal{D}H_{CF-}\right)=75\tanh\left(\frac{\mathcal{D}H_{CF-}}{0.0125}\right)\\ &\mathcal{D}H_{CF+}=\phi_{300}[+\left(0.1066-\mathcal{D}H_{CF},0\right)],\quad \mathcal{D}H_{CF-}=\phi_{300}[-\left(0.1066-\mathcal{D}H_{CF},0\right)] \end{aligned}$$

$$\mathcal{D}H_{CF} = H_{CF}\left[1+\phi_{-300}\left(\frac{\mu_t}{\mu}, 0.4\right)\right],\ H_{CF} = \frac{\zeta_{streamwise}\Delta}{|\mathbf{U}|},\ \zeta_{streamwise} = \left|\frac{v_k^A}{|\mathbf{U}|}\frac{\partial v_m^A}{\partial x_\ell}\varepsilon_{k\ell m}\right| \tag{2.12}$$

Η έκφραση του Re_{SCF} είναι μια κλειστή συνάρτηση του θ_t και ακολουθείται η ίδια επαναληπτική διαδικασία όπως στον όρο $Re_{\theta,t}^{eq}$. Οι σταθερές είναι $c_{\alpha_1} = 2$, $c_{\alpha_2} = 0.06$, $c_{\epsilon_1} = 1$, $c_{\epsilon_2} = 50$, $c_{\theta,t} = 0.03$, $\sigma_f = 1$, $\sigma_{\theta,t} = 2$, $c_{crossflow} = 0.6$. Για τη σύζευξη με το μοντέλο τύρβης Spalart–Allmaras, οι όροι πηγής του τελευταίου τροποποιούνται ως

$$\tilde{P}_{\tilde{\nu}} = \gamma\rho c_{b_1}\tilde{S}\tilde{\nu},\quad D_{\tilde{\nu}} = \frac{\rho}{\mathrm{Re}_0}c_{w_1}f_w\left(\frac{\tilde{\nu}}{\Delta}\right)^2 \tag{2.13}$$

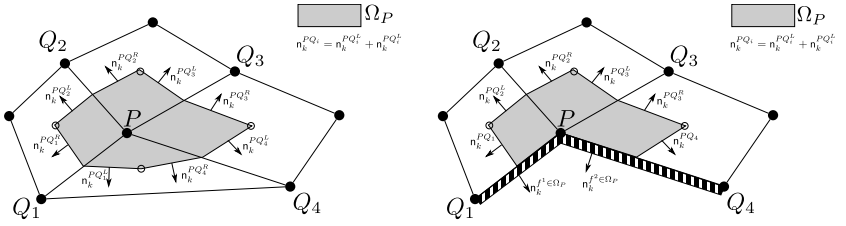
Τέλος, τις εξισώσεις ροής συμπληρώνει η εξίσωση Hamilton–Jacobi (Eikonal) η οποία υπολογίζει την απόσταση Δ κάθε εσωτερικού σημείου από το πλησιέστερο σημείο του στερεού τοιχώματος. Αυτή εκφράζεται ως

$$R^{\Delta} = \frac{\partial}{\partial x_k}\left(\Delta\frac{\partial\Delta}{\partial x_k}\right) - \Delta\frac{\partial}{\partial x_k}\left(\frac{\partial\Delta}{\partial x_k}\right) - 1 = 0 \tag{2.14}$$

Οι οριακές συνθήκες παρουσιάζονται στο πλήρες κείμενο της Διατριβής.

2.2 Διακριτοποίηση και Αριθμητική Επίλυση

Οι Εξ. 2.1-2.4 επιλύονται χρησιμοποιώντας την τεχνική της χρονο-προέλασης σε μη δομημένα πλέγματα με κεντροκομβική διατύπωση πεπερασμένων όγκων. Σε κάθε ψευδο–χρονικό βήμα, γύρω από κάθε κόμβο του πλέγματος, δημιουργείται ένας όγκος ελέγχου συνδέοντας τα μέσα των ακμών που πηγάζουν από αυτόν με τα βαρύκεντρα των πλευρών και τα βαρύκεντρα των γειτονικών πλεγματικών στοιχείων. Στο Σχ. 2.1 παρουσιάζεται ένας 2Δ όγκος ελέγχου γύρω από ένα κόμβο P.



Σχήμα 2.1: Πεπερασμένος όγκος ελέγχου γύρω από έναν εσωτερικό (αριστερά) και από έναν συνοριακό κόμβο P (δεξιά).

Εφαρμόζοντας το θεώρημα των Green-Gauss στους όρους ατριβούς ροής ενός πεπερασμένου όγκου P και διακριτοποιώντας το αποτέλεσμα προκύπτει

$$\int\limits_{\Omega_P^{t_{i+1}}} \frac{\partial f_{nk}^{\text{inv}}}{\partial x_k} \mathrm{d}\Omega \;=\; \int\limits_{\partial\Omega_P^{t_{i+1}}} f_{nk}^{\text{inv}} \hat{\mathsf{n}}_k \mathrm{d}\left(\partial\Omega\right) \simeq \sum_{Q\in\mathcal{N}(P)} \Phi_n^{\text{inv},PQ} + \sum_{f\in\mathcal{B}(P)} \Phi_n^{\text{inv},f\in\Omega_P} \tag{2.15}$$

όπου $\partial\Omega_P^{t_{i+1}}$ είναι το όριο του πεπερασμένου όγκου γύρω από τον κόμβο P στο χρονικό βήμα t_{i+1} και $\hat{\mathsf{n}}_k\left(t\right)$, $(k=1,2,3)$ το αντίστοιχο μοναδιαίο κάθετο διάνυσμα που πηγάζει από τον κόμβο P. Τα διανύσματα ατριβούς ροής που διέρχονται από τη διεπιφάνεια μεταξύ των πεπερασμένων όγκων των κόμβων P και Q διακριτοποιούνται σύμφωνα με το ανάντι σχήμα Roe [41], ως εξής

$$\Phi_n^{\text{inv},PQ} = \frac{1}{2}\left(A_{nmk}^P U_m^P + A_{nmk}^Q U_m^Q\right)\mathsf{n}_k^{PQ} - \frac{1}{2}\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|\left(U_m^R - U_m^L\right) \tag{2.16}$$

όπου $A_{nmk}{=}\frac{\partial f_{nk}^{\text{inv}}}{\partial U_m}$ τα Ιακωβιανά μητρώα , $\mathsf{n}_k^{PQ,t_{i+1}}$ το κάθετο διάνυσμα μεταξύ των πεπερασμένων όγκων των κόμβων P και Q στο χρονικό βήμα t_{i+1}. Για τον υπολογισμό της Ιακωβιανής ορίζουσας $\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k\right|$ απαιτείται ο υπολογισμός των ροϊκών μεγεθών στη λεγόμενη αριστερή (L) και δεξιά (R) κατάσταση με δεύτερης τάξης ακρίβεια χωρίς ή με περιοριστές [50]. Για δεύτερης τάξης ακρίβεια χωρίς περιοριστές ισχύει

$$V_m^L = V_m^P + \frac{1}{2}\mathsf{t}_\ell^{PQ}\frac{\partial V_m^P}{\partial x_\ell}, \qquad \frac{\partial V_m^P}{\partial x_\ell} = D_\ell^P V_m^P + \sum_{\Lambda\in\mathcal{N}(P)} Z_\ell^{P\Lambda} V_m^\Lambda \tag{2.17α'}$$

$$V_m^R = V_m^Q - \frac{1}{2}\mathsf{t}_\ell^{PQ}\frac{\partial V_m^Q}{\partial x_\ell}, \qquad \frac{\partial V_m^Q}{\partial x_\ell} = D_\ell^Q V_m^Q + \sum_{K\in\mathcal{N}(Q)} Z_\ell^{QK} V_m^K \tag{2.17β'}$$

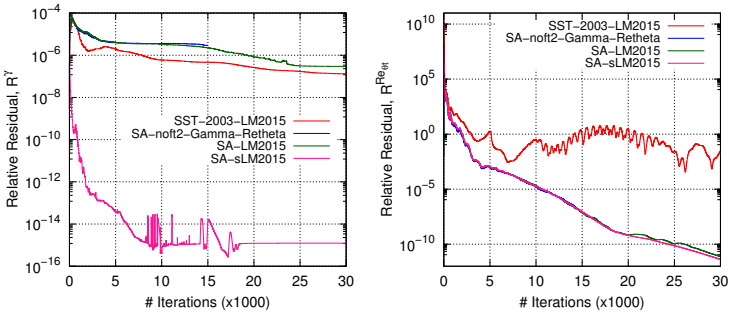
Η διακριτοποίηση των διανυσμάτων συνεκτικής ροής, των όρων πηγής, του χρονικού όρου και των όρων των μοντέλων τύρβης και μετάβασης καθώς και λεπτομέρειες σχετικά με την αριθμητική επίλυση αλλά και την υλοποίηση του κώδικα σε επεξεργαστές καρτών γραφικών αναφέρονται αναλυτικά στο πλήρες κείμενο της Διατριβής.

2.3 Πιστοποίηση και Επαλήθευση για Ροές με Μετάβαση

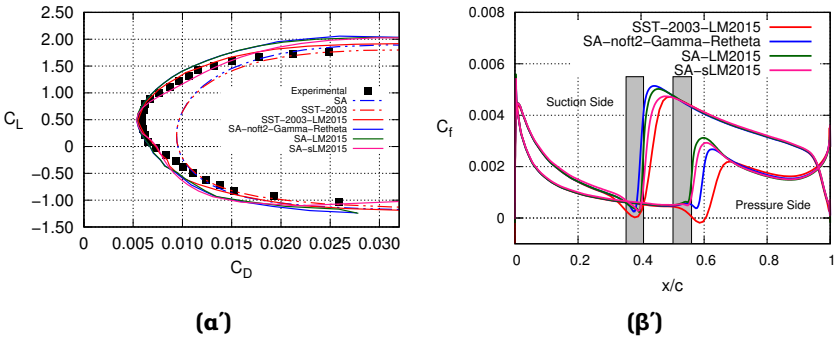
Η ενότητα αυτή, πιστοποιεί το νέο/εμπλουτισμένο λογισμικό PUMA ως προς την ικανότητα του να προσομοιώσει με ακρίβεια ροές με μετάβαση χρησιμοποιώντας τις τέσσερις εκδοχές του μοντέλου μετάβασης $\gamma - \tilde{Re}_{\theta t}$. Στο πλήρες κείμενο της Διατριβής περιλαμβάνονται πέντε εφαρμογές 2Δ και 3Δ ροών με πειραματικά ή/και υπολογιστικά δεδομένα. Εδώ, παρουσιάζονται ενδεικτικά οι δύο εξ αυτών.

Η πρώτη εφαρμογή αφορά τη φυσικά στρωτή ροή γύρω από την αεροτομή NLF(1)–0416. Ο αριθμός Mach της επ' άπειρο ροής, ο αριθμός Reynolds και η ισχύς της τύρβης είναι $M_\infty = 0.1$, $Re = 4\cdot 10^6$ και $Tu = 0.15\%$, αντίστοιχα. Μελέτη για

την ανεξαρτησία του πλέγματος και συγκρίσεις με τα πειραματικά δεδομένα, [43], για τις τέσσερις εκδοχές του μοντέλου $\gamma - \tilde{R}e_{\theta t}$ παρουσιάζονται στο πλήρες κείμενο της Διατριβής. Εδώ, στα Σχ. 2.2, 2.3α΄ και 2.3β΄ φαίνεται η πορεία σύγκλισης των εξισώσεων γ και $\tilde{R}e_{\theta t}$, η πολική των συντελεστών άνωσης (C_L) και οπισθέλκουσας (C_D) καθώς και η κατανομή του συντελεστή τριβής (C_f) στην πλευρά υπερπίεσης και υποπίεσης της αεροτομής. Παρατηρείται αυξημένη αριθμητική ευστάθεια του μοντέλου SA-sLM2015, καθώς και συμφωνία των αριθμητικών αποτελεσμάτων όλων των εκδοχών του $\gamma - \tilde{R}e_{\theta t}$ με τα πειραματικά δεδομένα.



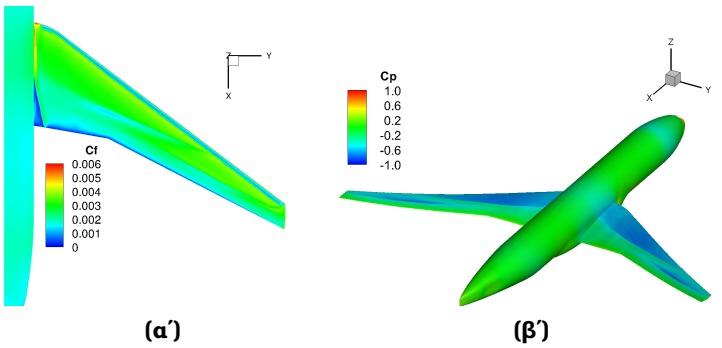
Σχήμα 2.2: *Αεροτομή NLF(1)-0416*: Πορεία σύγκλισης των εξισώσεων (α΄) γ και (β΄) $\tilde{R}e_{\theta t}$ των τεσσάρων εκδοχών του μοντέλου μετάβασης για AoA$=0^\circ$.



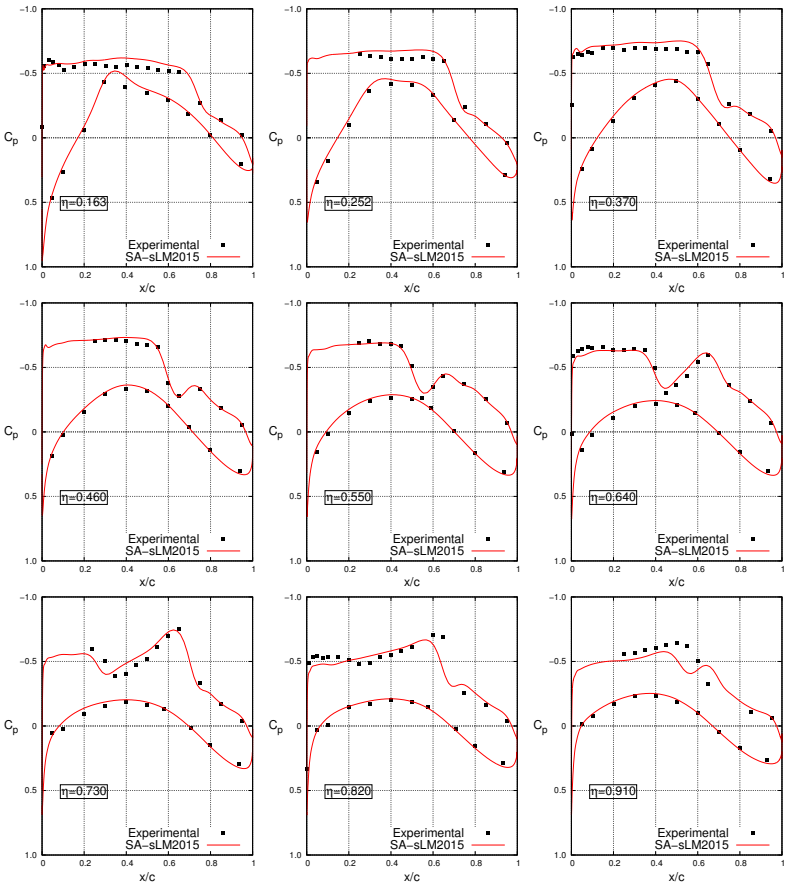
Σχήμα 2.3: *Αεροτομή NLF(1)-0416*: (α΄) Σύγκριση της πολικής C_L με C_D με πειραματικά δεδομένα, [43]. (β΄) Αντίστοιχη σύγκριση για την κατανομή του συντελεστή τριβής στην πλευρά υπερπίεσης και υποπίεσης της αεροτομής για AoA$=0^\circ$. Η γκρι περιοχή αντιπροσωπεύει πειραματικά δεδομένα, [43].

Η δεύτερη εφαρμογή μελετά τη ροή γύρω από το μοντέλο πτέρυγας αεροσκάφους CRM-NLF. Για την πτέρυγα υπάρχουν πειραματικά δεδομένα, [30], καθώς και (τέσσερις) συνθήκες για τους αριθμούς Mach και Reynolds και τη γωνία πρόσπτωσης, [1]. Η ένταση της τύρβης και η τραχύτητα της επιφάνειας είναι $Tu = 0.24\%$ και $h_{rms} = 0.0254\,\mu\text{m}$. Στο πλήρες κείμενο παρουσιάζονται συγκρίσεις για όλες τις

συνθήκες με το μοντέλο μετάβασης SA-sLM2015. Εδώ, ενδεικτικά οι συγκρίσεις γίνονται για $M_\infty = 0.856491$, $Re = 14.94591 \cdot 10^6$, AoA $= 1.980\,31°$, Σχ. 2.4, 2.5.



Σχήμα 2.4: *Πτέρυγα CRM-NLF*: Πεδίο του συντελεστή (α′) τριβής και (β′) πίεσης για AoA $\simeq 2.0°$.



Σχήμα 2.5: *Πτέρυγα CRM-NLF*: Κατανομή του συντελεστή πίεσης σε τομές στο εκπέτασμα της πτέρυγας (η) και σύγκριση με πειραματικά δεδομένα, [30], για AoA $\simeq 2.0°$.

Κεφάλαιο 3

Η Συνεχής Συζυγής Μέθοδος στη Βελτιστοποίηση Μορφής

Στο κεφάλαιο αυτό παρουσιάζεται η συνεχής συζυγής μέθοδος στη βελτιστοποίηση μορφής σε προβλήματα αεροδυναμικής. Η συνεχής συζυγής μέθοδος για τυρβώδεις ροές, στο λογισμικό PUMA, αναπτύχθηκε στη Διατριβή του Κ. Τσιάκα, [47], και εδώ επεκτείνεται σε ροές με μετάβαση διαφορίζοντας το μοντέλο $\gamma - \tilde{Re}_{\theta t}$.

Κάθε ροϊκό μέγεθος Φ μπορεί να εκφραστεί συναρτήσει των μεταβλητών ροής $\mathcal{U}_n, (n=1,\ldots,9$ με $\mathcal{U}_m := U_m$ $m=1,\ldots,5$, $\mathcal{U}_6 := \tilde{\nu}$, $\mathcal{U}_7 := \gamma$, $\mathcal{U}_8 := \tilde{Re}_{\theta t}$, $\mathcal{U}_9 := \Delta\Big)$ και των συντεταγμένων των κόμβων του πλέγματος x_k, $(k=1,\ldots,3)$. Η μεταβολή $\frac{\delta\Phi}{\delta b_i}$ εκφράζεται ως $\frac{\delta\Phi}{\delta b_i} = \frac{\partial\Phi}{\partial b_i} + \frac{\partial\Phi}{\partial x_k}\frac{\delta x_k}{\delta b_i}$, όπου $\frac{\partial\Phi}{\partial b_i}$ η μερική και $\frac{\delta\Phi}{\delta b_i}$ η ολική παράγωγος. Η συνεχής συζυγής μέθοδος εκκινεί με την επαύξηση της συνάρτησης-στόχου J με τα χωρικά ολοκληρώματα των υπολοίπων των εξισώσεων του πρωτεύοντος προβλήματος πολλαπλασιασμένα με τις συζυγείς μεταβλητές οδηγώντας στην επαυξημένη συνάρτηση-στόχο

$$J_{\text{aug}} = J + \int_\Omega \Psi_n R_n^{\text{MF}} \mathrm{d}\Omega + \int_\Omega \tilde{\nu_a} R^{\tilde{\nu}} \mathrm{d}\Omega + \int_\Omega \gamma_a R^{\gamma} \mathrm{d}\Omega + \int_\Omega \tilde{Re}_a R^{\tilde{Re}_{\theta t}} \mathrm{d}\Omega + \int_\Omega \Delta_a R^{\Delta} \mathrm{d}\Omega \quad (3.1)$$

Στην Εξ. 3.1 $\Psi_n (n=1, ..., 5)$ είναι τα συζυγή πεδία μέσης ροής και $\tilde{\nu_a}$, γ_a, $\tilde{Re}_a$ και Δ_a οι συζυγείς μεταβλητές Spalart–Allmaras, γ, $\tilde{Re}_{\theta t}$ και Δ, αντίστοιχα. Διαφορίζοντας την Εξ. 3.1 προκύπτουν χωρικά και επιφανειακά ολοκληρώματα που περιέχουν (υπολογιστικά ακριβούς) όρους $\frac{\delta\mathcal{U}_n}{\delta b_i}$. Για την αποφυγή υπολογισμού τους, οι όροι αυτοί απαλείφονται. Η απαλοιφή των χωρικών ολοκληρωμάτων οδηγεί στις συζυγείς πεδιακές εξισώσεις, ενώ εκείνη των επιφανειακών στις συζυγείς οριακές συνθήκες. Οι εναπομείναντες όροι συνεισφέρουν στην έκφραση των παραγώγων ευαισθησίας, το κόστος υπολογισμού των οποίων είναι ανεξάρτητο του πλήθους των μεταβλητών σχεδιασμού. Στη Διατριβή, η Field Integral (FI), [28, 37], διατύπωση της συνεχούς συζυγούς μεθόδου όπως αναπτύχθηκε για τυρβώδεις ροές [47], επεκτείνεται σε ροές με μετάβαση διαφορίζοντας τα μοντέλα SA-noft2-Gamma-Retheta, SA-LM2015 και SA-sLM2015. Η εκτενής ανάπτυξη της συζυγούς μεθόδου παρουσιάζεται στο πλήρες κείμενο της Διατριβής. Εδώ, παρουσιάζεται η τελική έκφραση των συζυγών πεδιακών εξισώσεων με το μοντέλο μετάβασης SA-sLM2015.

3.1 Συζυγείς Εξισώσεις

Οι συζυγείς εξισώσεις της μέσης ροής, των μοντέλων τύρβης Spalart–Allmaras και μετάβασης SA-sLM2015 και της εξίσωσης Hamilton–Jacobi εκφράζονται ως

$$R_m^{\Psi} = -A_{nmk}\frac{\partial \Psi_n}{\partial x_k} - \mathcal{K}_m + \mathcal{K}_m^{\text{SA}} + \mathcal{K}_m^{\gamma-\tilde{Re}_{\theta t}} + \mathcal{B}_m + \mathcal{B}_m^{\text{SA}} + \mathcal{B}_m^{\gamma-\tilde{Re}_{\theta t}} = 0 \tag{3.2α'}$$

$$R^{\tilde{\nu_a}} = -v_k^A\frac{\partial \tilde{\nu_a}}{\partial x_k} - \mathcal{G}^{\text{SA},diff} + \mathcal{G}^{\text{SA},src} + \left[\mathcal{G}^{\mu_t} + \mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}}\right]\frac{\partial \mu_t}{\partial \tilde{\mu}} = 0 \tag{3.2β'}$$

$$R^{\gamma_a} = -v_k^A\frac{\partial \gamma_a}{\partial x_k} - \mathcal{H}_{\gamma}^{\gamma-\tilde{Re}_{\theta t},diff} + \mathcal{H}_{\gamma}^{\gamma-\tilde{Re}_{\theta t},src} + \mathcal{H}_{\gamma}^{\text{SA},src} = 0 \tag{3.2γ'}$$

$$R^{\tilde{Re}_a} = -v_k^A\frac{\partial \tilde{Re}_a}{\partial x_k} - \mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},diff} + \mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},src} + \mathcal{H}_{\tilde{Re}_{\theta t}}^{\text{SA},src} = 0 \tag{3.2δ'}$$

$$R^{\Delta_a} = -2\frac{\partial}{\partial x_k}\left(\Delta_a\frac{\partial \Delta}{\partial x_k}\right) + \mathcal{M}^{\text{SA},src} + \mathcal{M}^{\gamma-\tilde{Re}_{\theta t},src} = 0 \tag{3.2ε'}$$

$n{=}1,\ldots,5,\ m{=}1,\ldots,5,\ k{=}1,\ldots,3$ όπου $\mathcal{K}_m,\ \mathcal{K}_m^{\text{SA}}$ και $\mathcal{K}_m^{\gamma-\tilde{Re}_{\theta t}}$ προέρχονται από τη διαφόριση των συνεκτικών όρων μέσης ροής, του μοντέλου τύρβης και μετάβασης,

$$\mathcal{K}_m = \left(\frac{\partial \tau_{kq}^{\text{adj}}}{\partial x_k} - \tau_{kq}\frac{\partial \Psi_5}{\partial x_k}\right)\frac{\partial v_q^A}{\partial U_m} + \frac{\partial q_k^{\text{adj}}}{\partial x_k}\frac{\partial T}{\partial U_m} \tag{3.3}$$

$$\mathcal{K}_m^{\text{SA}} = \mathcal{K}^{\text{SA},\rho}\frac{\partial \rho}{\partial U_m} + \mathcal{K}_k^{\text{SA},v_k^A}\frac{\partial v_k^A}{\partial U_m} \tag{3.4}$$

$$\mathcal{K}_m^{\gamma-\tilde{Re}_{\theta t}} = \mathcal{K}^{\gamma-\tilde{Re}_{\theta t},\rho}\frac{\partial \rho}{\partial U_m} + \mathcal{K}_k^{\gamma-\tilde{Re}_{\theta t},v_k^A}\frac{\partial v_k^A}{\partial U_m} \tag{3.5}$$

Για τις συζυγείς εξισώσεις μέσης ροής,

$$\begin{aligned}
\mathcal{K}^{\text{SA},\rho} = &-\frac{\tilde{\nu_a}}{\text{Re}_0\,\sigma}\left[\frac{\partial}{\partial x_k}\left[(\nu+\tilde{\nu})\frac{\partial \tilde{\nu}}{\partial x_k}\right] + c_{b_2}\frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\right] + \frac{1}{\text{Re}_0\,\sigma}\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\mathcal{P}\left(\nu,\rho\right)\\
&-\frac{1}{\text{Re}_0\,\sigma}\frac{\tilde{\nu}}{\rho}\left[\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k}\left(1+c_{b_2}\right) - \frac{\partial}{\partial x_k}\left[\left[\nu+\left(1+c_{b_2}\right)\tilde{\nu}\right]\frac{\partial\left(\rho\tilde{\nu_a}\right)}{\partial x_k}\right]\right]\\
&-\frac{c_{b_2}}{\text{Re}_0\,\sigma}\frac{\tilde{\nu}}{\rho}\left[\rho\tilde{\nu_a}\frac{\partial^2 \tilde{\nu}}{\partial x_k^2} + \frac{\partial}{\partial x_k}\left(\frac{\partial\left(\tilde{\nu_a}\tilde{\mu}\right)}{\partial x_k}\right)\right] + \frac{\tilde{\nu_a}}{\text{Re}_0}\left[-\left(c_{w_1}f_w - \frac{c_{b_1}}{\kappa^2}f_{t_2}\right)\left(\frac{\tilde{\nu}}{\Delta}\right)^2\right]\\
&+\tilde{\nu_a}\left[\mathcal{C}_r\mathcal{P}\left(r,\rho\right)+\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)\right] - \tilde{\nu_a}\frac{\gamma}{\rho}\mathcal{C}_{\gamma_{eff}}
\end{aligned} \tag{3.6}$$

$$\mathcal{K}_k^{\text{SA},v_k^A} = -\tilde{\mu}\frac{\partial \tilde{\nu_a}}{\partial x_k} - \frac{\partial}{\partial x_\ell}\left[\tilde{\nu_a}\frac{\mathcal{C}_\zeta^{\text{SA}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\right] - \frac{\partial}{\partial x_m}\left[\tilde{\nu_a}\frac{\mathcal{C}_S^{\text{SA}}}{S}2S_{km}\right]$$

$$
\begin{aligned}
\mathcal{K}^{\gamma-\tilde{Re}_{\theta t},\rho}= & \frac{1}{\mathrm{Re}_0}\frac{\gamma}{\rho}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\right]+\frac{1}{\mathrm{Re}_0}\frac{\tilde{Re}_{\theta t}}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_a}{\partial x_k}\right] \\
& -\gamma_a c_{\alpha_1}F_{length}F_{onset}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left(1-c_{\epsilon_1}\gamma\right) \\
& +\frac{1}{2}\gamma\gamma_a c_{\alpha_1}F_{length}F_{onset}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\left({}^1\!/\!_\gamma-3c_{\epsilon_1}\right) \\
& -\gamma_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{length}}\mathcal{P}\left(F_{length},F_{length1}\right)\mathcal{P}\left(F_{length1},\tilde{Re}_{\theta t}\right) \\
& +\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\gamma F_{turb}\left(c_{\epsilon_2}\gamma-1\right) \\
& -\gamma\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\left(2c_{\epsilon_2}\gamma-1\right) \\
& -\gamma_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{Re}_{\theta t}\right) \\
& -\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)+\rho\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\rho\right) \\
& -\tilde{Re}_a\tilde{Re}_{\theta t}\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)-\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t} \\
& +\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},\rho\right) \\
& +\tilde{Re}_{\theta t}\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big),\tilde{Re}_{\theta t}\right)F_{\theta t} \\
& +\tilde{Re}_a\mathcal{C}_{F_{\theta t}}\left[\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\mathcal{P}\left(Re_S,\rho\right)+\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\rho\right)\right] \\
& -\tilde{Re}_a\frac{\tilde{Re}_{\theta t}}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{Re}_{\theta t}\right) \\
& +\tilde{Re}_a\mathcal{C}_{Re^{eq}_{\theta,t}}\left[\mathcal{P}\left(\lambda_\theta,\rho\right)+\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\rho\right)\right]-\tilde{Re}_a\mathcal{C}_{Re_{SCF}}\frac{0.82\mu}{\rho^2\left|\mathbf{U}\right|} \qquad \textbf{(3.7)}
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{K}^{\gamma-\tilde{Re}_{\theta t},v^A_k}_k= & -\rho\left(\frac{\partial\gamma_a}{\partial x_k}\gamma+\frac{\partial\tilde{Re}_a}{\partial x_k}\tilde{Re}_{\theta t}\right)+\rho\tilde{Re}_a\frac{c_{\theta,t}}{\mathcal{T}^2}\left(Re^{eq}_{\theta,t}-\tilde{Re}_{\theta t}\right)\left(1-F_{\theta t}\right)\mathcal{P}\left(\mathcal{T},\left|\mathbf{U}\right|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k} \\
& +\tilde{Re}_a\rho\frac{c_{\theta,t}}{\mathcal{T}^2}c_{crossflow}\phi_{-300}\Big(Re_{SCF}-\tilde{Re}_{\theta t},0\Big)F_{\theta t}\mathcal{P}\left(\mathcal{T},\left|\mathbf{U}\right|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k} \\
& +\tilde{Re}_a\left[\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t,}\delta\right)\mathcal{P}\left(\delta,\left|\mathbf{U}\right|\right)+\mathcal{C}_{Re^{eq}_{\theta,t}}\mathcal{P}\left(\lambda_\theta,\theta\right)\mathcal{P}\left(\theta,\left|\mathbf{U}\right|\right)\right]\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k} \\
& +\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\left[-\frac{2}{\left|\mathbf{U}\right|}\frac{d\left|\mathbf{U}\right|}{ds}\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k}+\frac{v^A_m}{\left|\mathbf{U}\right|^2}\left(\frac{\partial v^A_k}{\partial x_m}+\frac{\partial v^A_m}{\partial x_k}\right)\right]-\frac{\partial}{\partial x_m}\left[\tilde{Re}_a\mathcal{C}_{d|\mathbf{U}|ds}\frac{v^A_k v^A_m}{\left|\mathbf{U}\right|^2}\right] \\
& -\tilde{Re}_a\mathcal{C}_{Re_{SCF}}\frac{0.82\mu}{\rho\left|\mathbf{U}\right|^2}\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k}+\tilde{Re}_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},H_{CF}\right)\mathcal{P}\left(H_{CF},\left|\mathbf{U}\right|\right)\frac{\partial\left|\mathbf{U}\right|}{\partial v^A_k}
\end{aligned}
$$

$$-\frac{\partial}{\partial x_q}\left[\tilde{Re}_a\mathcal{C}_{\zeta_{streamwise}}\frac{v_m^A}{|\mathbf{U}|}\varepsilon_{mqk}\right]+\tilde{Re}_a\frac{\mathcal{C}_{\zeta_{streamwise}}}{|\mathbf{U}|}\frac{\partial v_m^A}{\partial x_q}\left[\varepsilon_{kqm}+\frac{u_r}{|\mathbf{U}|}\varepsilon_{rqm}\frac{\partial|\mathbf{U}|}{\partial v_k^A}\right]$$
$$-\frac{\partial}{\partial x_\ell}\left[\frac{\mathcal{C}_\zeta^{\gamma-\tilde{Re}_{\theta t}}}{\zeta}\varepsilon_{m\ell k}\varepsilon_{mqr}\frac{\partial v_r^A}{\partial x_q}\right]-\frac{\partial}{\partial x_m}\left[\frac{\mathcal{C}_S^{\gamma-\tilde{Re}_{\theta t}}}{S}2S_{km}\right]$$

Οι όροι $\mathcal{B}_m, \mathcal{B}_m^{\text{SA}}$ και $\mathcal{B}_m^{\gamma-\tilde{Re}_{\theta t}}$ εμφανίζονται εφόσον διαφορίζεται ο νόμος που διέπει τις μεταβολές της δυναμικής συνεκτικότητας λόγω των μεταβολών της θερμοκρασίας. Σχετικά με τις συζυγείς εξισώσεις του μοντέλου Spalart–Allmaras, οι $\mathcal{G}^{\text{SA},diff}$ και $\mathcal{G}^{\text{SA},src}$ προκύπτουν από τη διαφόριση των όρων διάχυσης και πηγής του ίδιου μοντέλου. $\mathcal{G}^{\mu_t}$ και $\mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}}$ περιέχουν τη συνεισφορά των εξισώσεων μέσης ροής και του μοντέλου $\gamma-\tilde{Re}_{\theta t}$ στην τυρβώδη συνεκτικότητα μ_t. Οι όροι αυτοί εκφράζονται ως

$$\begin{aligned}
\mathcal{G}^{\text{SA},diff} &= -\frac{1}{\text{Re}_0}\frac{1}{\sigma}\frac{1}{\rho}\left[\frac{\partial(\rho\tilde{\nu_a})}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}(1+c_{b_2})-\frac{\partial}{\partial x_k}\left\{[\nu+(1+c_{b_2})\tilde{\nu}]\frac{\partial(\rho\tilde{\nu_a})}{\partial x_k}\right\}\right]\\
&\quad -\frac{c_{b_2}}{\text{Re}_0}\frac{1}{\sigma}\frac{1}{\rho}\left[\rho\tilde{\nu_a}\frac{\partial^2\tilde{\nu}}{\partial x_k^2}+\frac{\partial}{\partial x_k}\left(\frac{\partial(\tilde{\nu_a}\tilde{\mu})}{\partial x_k}\right)\right]\\
\mathcal{G}^{\text{SA},src} &= \tilde{\nu_a}\left[-\gamma c_{b_1}\tilde{S}+\frac{2}{\text{Re}_0}c_{w_1}f_w\frac{\tilde{\nu}}{\Delta^2}\right]+\tilde{\nu_a}\mathcal{C}_r\mathcal{P}(r,\tilde{\mu})+\frac{\tilde{\nu_a}}{\rho}\mathcal{C}_{\tilde{S}}\left[\mathcal{P}\left(\tilde{S},\tilde{\nu}\right)+\mathcal{C}_\chi\mathcal{P}(\chi,\tilde{\mu})\right]\\
\mathcal{G}^{\mu_t} &= \frac{\tau_{km}}{\mu+\mu_t}\left(\frac{\partial\Psi_{m+1}}{\partial x_k}+\frac{\partial\Psi_5}{\partial x_k}v_m^A\right)+\frac{\mathcal{C}_p}{\text{Re}_0}\frac{\partial\Psi_5}{\partial x_k}\frac{\partial T}{\partial x_k}\frac{1}{\text{Pr}_\text{t}}\\
\mathcal{G}^{\mu_t,\gamma-\tilde{Re}_{\theta t}} &= \frac{1}{\text{Re}_0}\left(\frac{\partial\gamma_a}{\partial x_k}\frac{\partial\gamma}{\partial x_k}\frac{1}{\sigma_f}+\frac{\partial\tilde{Re}_a}{\partial x_k}\frac{\partial\tilde{Re}_{\theta t}}{\partial x_k}\sigma_{\theta,t}\right)+\mathcal{C}_{\mu_t}^{\gamma-\tilde{Re}_{\theta t}}
\end{aligned}\tag{3.8}$$

Για τις συζυγείς εξισώσεις του μοντέλου μετάβασης γ–$\tilde{Re}_{\theta t}$, οι όροι $\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},diff}/\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},diff}$ $\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},src}/\mathcal{H}_{\tilde{Re}_{\theta t}}^{\gamma-\tilde{Re}_{\theta t},src}$ και $\mathcal{H}_\gamma^{\text{SA},src}/\mathcal{H}_{\tilde{Re}_{\theta t}}^{\text{SA},src}$ προέρχονται από η διαφόριση των όρων διάχυσης και πηγής του μοντέλου $\gamma-\tilde{Re}_{\theta t}$ και των όρων πηγής του μοντέλου Spalart–Allmaras και εκφράζονται ως

$$\begin{aligned}
\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},diff} &= \frac{1}{\text{Re}_0}\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\left(\mu+\frac{\mu_t}{\sigma_f}\right)\frac{\partial\gamma_a}{\partial x_k}\right]\\
\mathcal{H}_\gamma^{\gamma-\tilde{Re}_{\theta t},src} &= -\frac{1}{2}\gamma_a c_{\alpha_1}F_{length}F_{onset}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)\sqrt{\gamma}\,(1/\gamma-3c_{\epsilon_1})\\
&\quad +\gamma_a c_{\alpha_2}\phi_{-300}\left(\zeta,\frac{M\sqrt{MRe}}{20}\right)F_{turb}\,(2c_{\epsilon_2}\gamma-1)\\
\mathcal{H}_\gamma^{\text{SA},src} &= -\tilde{\nu_a}c_{b_1}\tilde{\nu}\tilde{S}
\end{aligned}$$

$$\mathcal{H}^{\gamma-\tilde{R}e_{\theta t},diff}_{\tilde{R}e_{\theta t}} = \frac{1}{\mathrm{Re}_0}\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\sigma_{\theta,t}\left(\mu+\mu_t\right)\frac{\partial \tilde{R}e_a}{\partial x_k}\right] \tag{3.9}$$

$$\begin{aligned}\mathcal{H}^{\gamma-\tilde{R}e_{\theta t},src}_{\tilde{R}e_{\theta t}} =& \gamma_a\frac{1}{\rho}\mathcal{C}_{F_{length}}\mathcal{P}\left(F_{length},F_{length1}\right)\mathcal{P}\left(F_{length1},\tilde{R}e_{\theta t}\right)\\ &+\gamma_a\frac{1}{\rho}\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_{\theta c}\right)\mathcal{P}\left(Re_{\theta c},\tilde{R}e_{\theta t}\right)+\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}\left(1-F_{\theta t}\right)\\ &-\tilde{R}e_a\frac{c_{\theta,t}}{\mathcal{T}}c_{crossflow}\mathcal{P}\left(\phi_{-300}\left(Re_{SCF}-\tilde{R}e_{\theta t},0\right),\tilde{R}e_{\theta t}\right)F_{\theta t}\\ &+\frac{\tilde{R}e_a}{\rho}\mathcal{C}_{F_{\theta t}}F_{wake}\mathcal{P}\left(F_{\theta t},\delta\right)\mathcal{P}\left(\delta,\tilde{R}e_{\theta t}\right)\end{aligned}$$

$$\mathcal{H}^{\mathrm{SA},src}_{\tilde{R}e_{\theta t}} = 0 \tag{3.10}$$

Αναφορικά με τη συζυγή εξίσωση Hamilton–Jacobi, οι όροι $\mathcal{M}^{\mathrm{SA},src}$ και $\mathcal{M}^{\gamma-\tilde{R}e_{\theta t},src}$ (οι οποίοι προέρχονται από τη διαφόριση του μοντέλου τύρβης Spalart–Allmaras και μετάβασης SA-sLM2015, αντίστοιχα) δίνονται ως

$$\begin{aligned}\mathcal{M}^{\mathrm{SA},src} =& \tilde{\nu_a}\left[\mathcal{C}_r\mathcal{P}\left(r,\Delta\right)-2\frac{\rho}{\mathrm{Re}_0}c_{w_1}f_w\frac{\tilde{\nu}^2}{\Delta^3}\right]+\tilde{\nu_a}\mathcal{C}_{\tilde{S}}\mathcal{P}\left(\tilde{S},\Delta\right)\\ \mathcal{M}^{\gamma-\tilde{R}e_{\theta t},src} =& \left[\gamma_a\mathcal{C}_{F_{onset1}}\mathcal{P}\left(F_{onset1},Re_S\right)+\tilde{R}e_a\mathcal{C}_{F_{\theta t}}\mathcal{P}\left(F_{\theta t},F_{wake}\right)\mathcal{P}\left(F_{wake},Re_S\right)\right]\mathcal{P}\left(Re_S,\Delta\right)\\ &+\tilde{R}e_a\mathcal{C}_{\mathcal{D}H_{CF}}\mathcal{P}\left(\mathcal{D}H_{CF},H_{CF}\right)\mathcal{P}\left(H_{CF},\Delta\right)\end{aligned} \tag{3.11}$$

Οι συζυγείς οριακές συνθήκες για τις Εξ. 3.2α′-3.2ε′ για κάθε τύπο ορίου, η έκφραση των παραγώγων ευαισθησίας καθώς και στοιχεία για την αριθμητική επίλυση των συζυγών πεδιακών εξισώσεων παρουσιάζονται αναλυτικά στο πλήρες κείμενο.

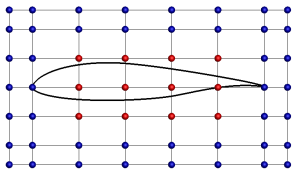
3.2 Πιστοποίηση της Συζυγούς Διατύπωσης

Στην ενότητα αυτή πιστοποιείται η συνεχής συζυγής μέθοδος για ροές με μετάβαση για τα μοντέλα SA-noft2-Gamma-Retheta, SA-LM2015 και SA-sLM2015 στην αεροτομή NLF(1)-0416 σε γωνία AoA $= 2.03°$, Σχ. 3.1. Υπολογίζονται οι παράγωγοι των συντελεστών άνωσης (C_L) και οπισθέλκουσας (C_D), Εξ. 3.12, με και χωρίς την παραδοχή της 'παγωμένης μετάβασης' και για τα τρία μοντέλα και συγκρίνονται με Πεπερασμένες Διαφορές (ΠΔ), Σχ. 3.2. Η συζυγής μέθοδος αναπαράγει το αποτέλεσμα των ΠΔ με μεγάλη ακρίβεια, αντίθετα η παραδοχή της 'παγωμένης μετάβασης' είναι επιζήμια για τον υπολογισμό των παραγώγων, ειδικά για τον C_D, καθώς ενίοτε

αυτές υπολογίζονται ακόμα και με αντίθετο πρόσημο (μεταβλητή σχεδιασμού με 21).

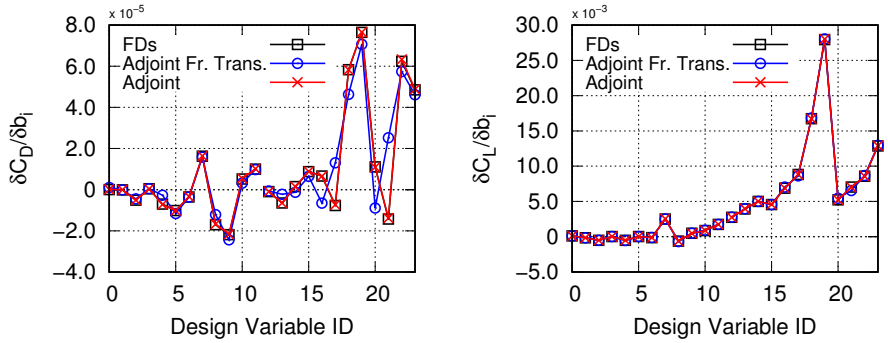
$$J = \frac{F}{\frac{1}{2}\rho_\infty U_\infty c}, \quad F = \int_{S_{\text{Obj}}} (p\mathsf{n}_k - \tau_{km}\mathsf{n}_m r_k)\, dS \tag{3.12}$$

όπου F η δύναμη (οπισθέλκουσα ή άνωση) και r_k η κατεύθυνση της δύναμης, για την οπισθέλκουσα $r_k^D = (-\sin\alpha_\infty, \cos\alpha_\infty)$ και για την άνωση $r_k^L = (\cos\alpha_\infty, \sin\alpha_\infty)$.

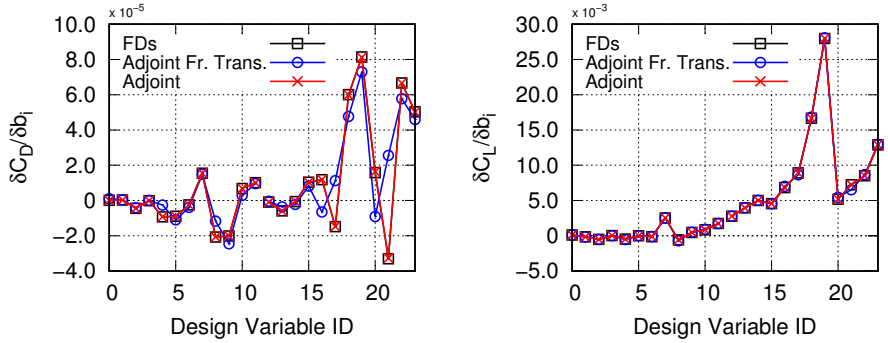


Σχήμα 3.1: *Βελτιστοποίηση της αεροτομής NLF(1)-0416*: Παραμετροποίηση.

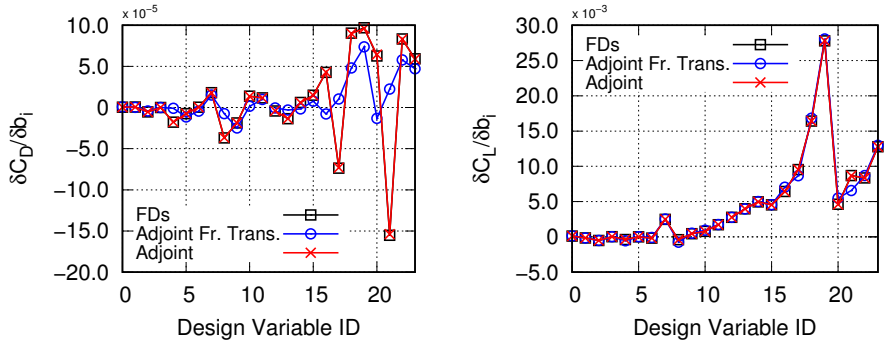
Η αεροτομή NLF(1)–0416 βελτιστοποιείται για ελάχιστο C_D υπό τον διπλό περιορισμό το C_L να είναι ίδιο με της αρχικής γεωμετρίας ($\pm 1\%$) καθώς και ο όγκος της αεροτομής να μην μεταβάλλεται πάνω από το 10% της αρχικής. Η σύγκλιση των βελτιστοποιήσεων για τα τρία μοντέλα μετάβασης παρουσιάζεται στο Σχ. 3.3, το C_D μειώθηκε κατά $\sim 7.5\%$, $\sim 17.4\%$ και $\sim 28.7\%$ για τα μοντέλα SA-noft2-Gamma-Retheta, SA-LM2015 και SA-sLM2015, αντίστοιχα, ικανοποιώντας τους περιορισμούς. Φαίνεται πως η βελτιστοποίηση με το μοντέλο μετάβασης SA-sLM2015 υπερτερεί των υπολοίπων, πιθανότατα λόγω των πλήρως διαφορίσιμων εκφράσεων του μοντέλου. Η γεωμετρία και η κατανομή του συντελεστή τριβής για την αρχική και τις τρεις βελτιστοποιημένες αεροτομές παρουσιάζονται στο Σχ. 3.4. Παρατηρείται η μετατόπιση του σημείου μετάβασης (τόσο στην πλευρά υπερπίεσης όσο και στην πλευρά υποπίεσης) κατάντι οδηγώντας σε μείωση του C_D. Εκτενής ερμηνεία των αποτελεσμάτων της βελτιστοποίησης παρουσιάζεται στο πλήρες κείμενο της Διατριβής.



(α΄) SA-noft2-Gamma-Retheta

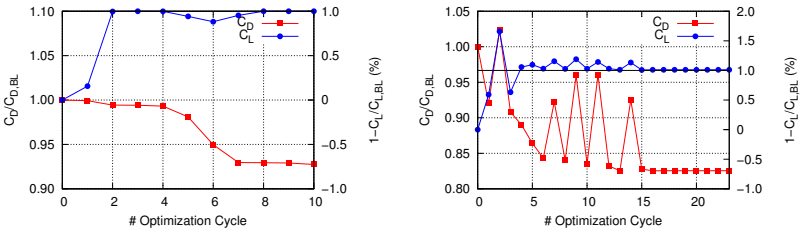


(β΄) SA-LM2015

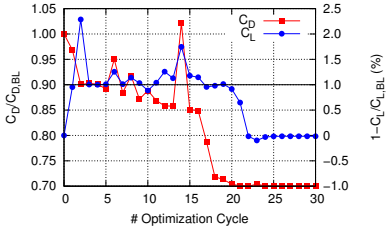


(γ΄) SA-sLM2015

Σχήμα 3.2: *Βελτιστοποίηση της αεροτομής NLF(1)-0416*: Σύγκριση των παραγώγων ευαισθησίας του C_D (αριστερά) και C_L (δεξιά) υπολογισμένες με τη συνεχή συζυγή μέθοδο, με και χωρίς την παραδοχή της 'παγωμένης μετάβασης', με ΠΔ. Παράγωγοι με τα μοντέλα μετάβασης (α΄) SA-noft2-Gamma-Retheta, (β΄) SA-LM2015 και (γ΄) SA-sLM2015.

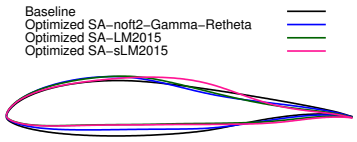


(α′) SA-noft2-Gamma-Retheta **(β′)** SA-LM2015

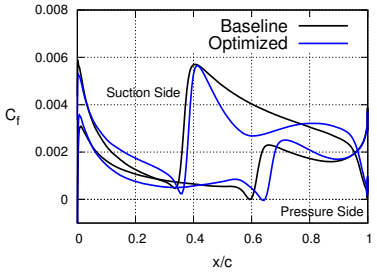


(γ′) SA-sLM2015

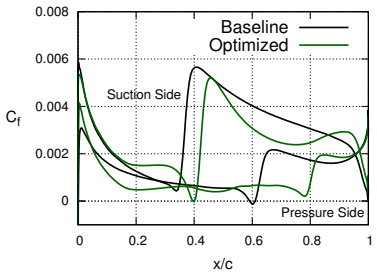
Σχήμα 3.3: *Βελτιστοποίηση της αεροτομής NLF(1)-0416*: Πορεία σύγκλισης της βελτιστοποίησης για τα μοντέλα μετάβασης (α′) SA-noft2-Gamma-Retheta, (β′) SA-LM2015 και (γ′) SA-sLM2015.



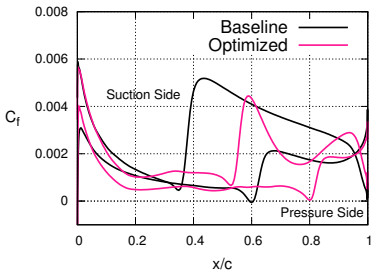
(α′)



(β′) SA-noft2-Gamma-Retheta



(γ′) SA-LM2015



(δ′) SA-sLM2015

Σχήμα 3.4: *Βελτιστοποίηση της αεροτομής NLF(1)-0416*: (α′) Αρχική και βελτιστοποιημένες γεωμετρίες. Κατανομή του συντελεστή τριβής στην επιφάνεια της αεροτομής για την αρχική και τη βελτιστοποιημένη γεωμετρία με το μοντέλο μετάβασης (β′) SA-noft2-Gamma-Retheta, (γ′) SA-LM2015 και (δ′) SA-sLM2015.

Κεφάλαιο 4

Συμβατά Σχήματα Διακριτοποίησης για τη Συνεχή Συζυγή Μέθοδο

Στο κεφάλαιο αυτό αναπτύσσονται και προτείνονται συμβατά σχήματα διακριτοποίησης για τις συνεχείς συζυγείς εξισώσεις και οριακές συνθήκες (Κεφάλαιο 3) με φυσική σημασία. Στη συνεχή συζυγή μέθοδο, η εύρεση κατάλληλων σχημάτων διακριτοποίησης αποτελεί πρόκληση, καθώς αυτά μπορούν να επηρεάσουν σημαντικά την ακρίβεια υπολογισμού των παραγώγων ευαισθησίας εφόσον δεν είναι απολύτως συμβατά με το διακριτοποιημένο πρωτεύον πρόβλημα. Εν αντιθέσει, στη διακριτή συζυγή, [13, 35], η συμβατότητα είναι εγγυημένη καθώς η μέθοδος βασίζεται στο διακριτοποιημένο πρωτεύον πρόβλημα, ενώ ο συζυγής επιλύτης υιοθετεί και τα χαρακτηριστικά του πρωτεύοντος αναφορικά με τη σύγκλιση. Το βασικό της μειονέκτημα όμως είναι οι ενδεχόμενες αυξημένες απαιτήσεις σε μνήμη του συζυγούς κώδικα και η πλήρης έλλειψη φυσικής εποπτείας του προβλήματος.

Εδώ, προτείνονται σχήματα διακριτοποίησης για τις συνεχείς συζυγείς εξισώσεις και οριακές συνθήκες τα οποία εμπνέονται από τη διακριτή μέθοδο και είναι συμβατά με το διακριτοποιημένο πρωτεύον πρόβλημα. Πρακτικά, τα πλεονεκτήματα της συνεχούς και της διακριτής μεθόδου ενώνονται σε μια νέα *Think-Discrete Do-Continuous* (*TDDC*) συζυγή μέθοδο. Με τον τρόπο αυτό, οι παράγωγοι ευαισθησίας υπολογίζονται με την ακρίβεια της διακριτής μεθόδου, χωρίς τις ενδεχόμενες αυξημένες απαιτήσεις σε μνήμη και με καθαρή εποπτεία του συζυγούς προβλήματος.

Στη διακριτή συζυγή μέθοδο, η συνάρτηση-στόχος J επαυξάνεται με το άθροισμα των γινομένων των διακριτοποιημένων υπολοίπων των εξισώσεων του πρωτεύοντος προβλήματος με τις συζυγείς μεταβλητές, οδηγώντας στην επαυξημένη συνάρτηση-στόχο J_{aug}.

$$J_{\text{aug}} = J + \sum_P \Psi_n^P R_n^{\text{MF},P} + \sum_P \tilde{\nu_a}^P R^{\tilde{\nu},P} + \sum_P \gamma_a^P R^{\gamma,P} + \sum_P \tilde{Re}_a^P R^{\tilde{Re}_{\theta t},P} + \sum_P \Delta_a^P R^{\Delta,P} \quad (4.1)$$

όπου $\Psi_n^P, (n = 1, ..., 5)$, $\tilde{\nu_a}^P$, γ_a^P, $\tilde{Re}_a^P$ και Δ_a^P είναι τα συζυγή πεδία μέσης ροής, η συζυγής μεταβλητή Spalart–Allmaras και οι συζυγείς μεταβλητές των γ, $\tilde{Re}_{\theta t}$ και Δ υπολογιζόμενες στον (εσωτερικό ή οριακό) κόμβο P. $R_n^{\text{MF},P}, R^{\tilde{\nu},P}, R^{\gamma,P}, R^{\tilde{Re}_{\theta t},P}, R^{\Delta,P}$ είναι τα διακριτοποιημένα υπόλοιπα υπολογισμένα στον κόμβο P. Διαφορίζοντας την Εξ. 4.1 ως προς τις μεταβλητές σχεδιασμού b_i προκύπτουν αθροίσματα που περιέχουν όρους $\frac{\delta \mathcal{U}_n}{\delta b_i}$. Όμοια με τη συνεχή συζυγή μέθοδο, οι πολλαπλασιαστές των

όρων αυτών τίθενται ίσοι με μηδέν.

Στο πλήρες κείμενο της Διατριβής παρουσιάζεται η με-το-χέρι-διαφόριση της Εξ. 4.1 και προτείνονται συμβατά σχήματα διακριτοποίησης για τις συζυγείς εξισώσεις (μέσης ροής, μοντέλων τύρβης και μετάβασης). Εδώ, παραλείπεται η μαθηματική ανάπτυξη και παρουσιάζονται ενδεικτικά τα προτεινόμενα *TDDC* σχήματα διακριτοποίησης για τους συζυγούς όρους ατριβούς ροής.

4.1 Διακριτοποίηση των Συζυγών Όρων Ατριβούς Ροής

Ολοκληρώνοντας τον όρο ατριβούς ροής της Εξ. 3.2α′ σε έναν πεπερασμένο όγκο Ω_P όπως στο Σχ. 2.1 και πραγματοποιώντας τις απαραίτητες προσεγγίσεις δημιουργείται ένα ισοζύγιο των συζυγών διανυσμάτων ατριβούς ροής γραμμένο ως

$$-\int_{\Omega_P} A_{nmk}\frac{\partial \Psi_n}{\partial x_k}d\Omega \simeq \sum_{Q\in\mathcal{N}(P)}\Phi_m^{\text{inv,adj},PQ}+\sum_{f\in\mathcal{B}(P)}\Phi_m^{\text{inv,adj},f} \tag{4.2}$$

Για τη διακριτοποίηση των συζυγών διανυσμάτων ατριβούς ροής στο εσωτερικό του χωρίου $\Phi_m^{\text{inv,adj},PQ}$, μπορεί να χρησιμοποιηθεί ένα σχήμα που απαντάται και εφαρμόζεται ευρέως στη σχετική βιβλιογραφία [18, 36], και, πρακτικά, εμπνέεται από εκείνο του πρωτεύοντος προβλήματος, Εξ. 2.16,

$$\Phi_{m,\text{std}}^{\text{inv,adj},PQ}=-\frac{1}{2}\left(\Psi_n^P+\Psi_n^Q\right)A_{nmk}^P\mathsf{n}_k^{PQ}-\frac{1}{2}\left|\tilde{A}_{nmk}^{LR}\mathsf{n}_k^{PQ}\right|\left(\Psi_n^R-\Psi_n^L\right) \tag{4.3}$$

Ο συζυγής επιλύτης που χρησιμοποιεί το ανωτέρω σχήμα διακριτοποίησης θα αναφέρεται ως ‘κλασικός’ συνεχής (δείκτης ‘std’).

Παίρνοντας έμπνευση από τις με-το-χέρι-διαφορισμένες διακριτές συζυγείς εξισώσεις, προτείνεται η ακόλουθη έκφραση για τη διακριτοποίηση των συζυγών διανυσμάτων ατριβούς ροής

$$\Phi_m^{\text{inv,adj},PQ}=-\frac{1}{2}\left(\Psi_n^P+\Psi_n^Q\right)A_{nmk}^P\mathsf{n}_k^{PQ}-\frac{1}{2}\left[\left(\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\Psi_n\right)^{R,\text{adj}}-\left(\left|\tilde{\mathcal{A}}_{n\ell k}^{LR}\mathsf{n}_k\right|\Psi_n\right)^{L,\text{adj}}\right]\frac{\partial V_\ell^P}{\partial U_m^P} \tag{4.4}$$

Στην Εξ. 4.4, δεύτερης τάξης ακρίβεια χωρίς περιοριστή, οι συζυγείς καταστάσεις “L”eft και “R”ight δίνονται ως

$$\phi^{L,\text{adj}}=\phi^P+\frac{1}{2}\left.\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\right|^{P,\text{adj}},\quad \left.\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\right|^{P,\text{adj}}=\mathsf{t}_r^{PQ}D_r^P\phi^P+\sum_{\Lambda\in\mathcal{N}(P)}Z_r^{\Lambda P}\sum_{M\in\mathcal{N}(\Lambda)}\mathsf{t}_r^{\Lambda M}\phi^\Lambda \tag{4.5α′}$$

$$\phi^{R,\text{adj}} = \phi^{Q} + \frac{1}{2}\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\bigg|^{Q,\text{adj}}, \frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\bigg|^{Q,\text{adj}} = \mathsf{t}_r^{PQ} D_r^P \phi^Q + \sum_{\Lambda\in\mathcal{N}(P)} Z_r^{\Lambda P} \sum_{M\in\mathcal{N}(\Lambda)} \mathsf{t}_r^{\Lambda M}\phi^M \quad (4.5\beta')$$

όπου ο M κόμβος που συνδέεται με ακμή με κάποιον εκ των $\Lambda\in\mathcal{N}(P)$. Ακόμα, εισάγεται η τροποποιημένη Ιακωβιανή ορίζουσα $\left|\tilde{\mathcal{A}}^{LR}_{n\ell k}\mathsf{n}_k\right|$ που ορίζεται ως $\left|\tilde{\mathcal{A}}^{LR}_{n\ell k}\mathsf{n}_k\right| = \left|\tilde{A}^{LR}_{nqk}\mathsf{n}_k\right|\frac{\partial U_q^L}{\partial V_\ell^L} - \left(U_q^R - U_q^L\right)\frac{\partial\left|\tilde{A}^{LR}_{nqk}\mathsf{n}_k\right|}{\partial V_\ell^L}$. Η έκφραση αυτή περιέχει την παράγωγο της Ιακωβιανής ορίζουσας ως προς τις πρωτεύουσες ροϊκές μεταβλητές και παραπέμπει σε ένα 'συζυγές' ανάπτυγμα Taylor της Ιακωβιανής ορίζουσας στο $\mathbf{V}$ χώρο.

Πέρα από τη χρήση της $\left|\tilde{\mathcal{A}}^{LR}_{n\ell k}\mathsf{n}_k\right|$, αντί της τυπικής, οι συζυγείς L και R καταστάσεις ("L,adj", "R,adj") ορίζονται διαφορετικά από ότι στο πρωτεύον πρόβλημα. Στη διακριτοποίηση του πρωτεύοντος προβλήματος, Εξ. 2.17, η κατάσταση L μιας μεταβλητής ϕ, σε μια ακμή PQ, απορρέει από ένα ανάπτυγμα Taylor πρώτης τάξης ακρίβειας και είναι το άθροισμα του ϕ^P και του ήμισυ του εσωτερικού γινομένου του $\overrightarrow{PQ}$ και της χωρικής παραγώγου του ϕ στον κόμβο P. Ο ανωτέρω υπολογισμός εξαρτάται αποκλειστικά από την κομβική τιμή του ϕ στον P και τους γείτονές του Λ_i (Σχ. 2.1). Στην Εξ. 4.5, η συζυγής κατάσταση L ("L,adj") της μεταβλητής ϕ στην ακμή PQ είναι το άθροισμα της τιμής του ϕ στον κόμβο P και του ήμισυ της απόκλισης του $\overrightarrow{PQ}$ σταθμισμένο με το ϕ, $\frac{\partial(\mathsf{t}_r\phi)}{\partial x_r}\Big|^{P,\text{adj}}$. Όμοια για τις καταστάσεις R και "R,adj".

Συγκρίνοντας τα 'κλασικά', Εξ. 4.3, και τα *TDDC*, Εξ. 4.4, συζυγή σχήματα διακριτοποίησης παρατηρείται πως το κεντρικό τμήμα είναι κοινό. Αυτό που αλλάζει (που είναι, πρακτικά, η πρωτοτυπία της *TDDC* συζυγούς μεθόδου) είναι η διακριτοποίηση του όρου της τεχνητής συνεκτικότητας με την εισαγωγή της τροποποιημένης Ιακωβιανής ορίζουσας και των συζυγών καταστάσεων "L,adj" και "R,adj", Εξ. 4.5.

Για τα στερεά όρια $\mathcal{f}\in\mathcal{B}^{\mathcal{W}}(P)$, προτείνεται η κάτωθι διακριτοποίηση για τα συζυγή διανύσματα ατριβών ροών σύμφωνα με τη *TDDC* συζυγή μέθοδο

$$\Phi_m^{\text{inv,adj},\mathcal{f}} = \begin{bmatrix} v_k^{A,P}\mathsf{n}_k^{\mathcal{f}}\left[\Psi_{q+1}v_q^A - \Psi_5\left(-h_t + \frac{\gamma-1}{2}v_k^A v_k^A\right)\right]^P \\ -\left[\Psi_1+\Psi_{q+1}v_q^A+\Psi_5 h_t\right]^P\mathsf{n}_1^{\mathcal{f}} + v_k^{A,P}\mathsf{n}_k^{\mathcal{f}}\left[-\Psi_2+(\gamma-1)\Psi_5 v_1^A\right]^P \\ -\left[\Psi_1+\Psi_{q+1}v_q^A+\Psi_5 h_t\right]^P\mathsf{n}_2^{\mathcal{f}} + v_k^{A,P}\mathsf{n}_k^{\mathcal{f}}\left[-\Psi_3+(\gamma-1)\Psi_5 v_2^A\right]^P \\ -\left[\Psi_1+\Psi_{q+1}v_q^A+\Psi_5 h_t\right]^P\mathsf{n}_3^{\mathcal{f}} + v_k^{A,P}\mathsf{n}_k^{\mathcal{f}}\left[-\Psi_4+(\gamma-1)\Psi_5 v_3^A\right]^P \\ -\gamma\Psi_5^P v_k^{A,P}\mathsf{n}_k^{\mathcal{f}} \end{bmatrix} + \frac{\partial J}{\partial p}\bigg|^P \frac{\partial p}{\partial U_m}\bigg|^P \quad (4.6)$$

Για το επ' άπειρο όριο $\mathcal{f}\in\mathcal{B}^{\mathcal{F}}(P)$, τα συζυγή διανύσματα ατριβών ροών διακριτοποιούνται ως

$$\Phi_m^{\text{inv,adj},\mathcal{f}} = -\Psi_n^P\mathcal{A}^{-P}_{nmk}\mathsf{n}_k^{\mathcal{f}} - \Psi_n^Q\mathcal{A}^{-P}_{nmk}\mathsf{n}_k^{\mathcal{f}} \quad (4.7)$$

όπου, στον ψευδόκομβο Q, $\Psi_n^Q = 0$ και $\mathcal{A}^{-P}_{nmk}\mathsf{n}^f_k = \frac{1}{2}\left(A^P_{nmk}\mathsf{n}^f_k - \left|\mathcal{A}^P_{nmk}\mathsf{n}^f_k\right|\right)$, $\left|\mathcal{A}^P_{nmk}\mathsf{n}^f_k\right| = \left|A^P_{nmk}\mathsf{n}^f_k\right| - \left(U^Q_q - U^P_q\right)\frac{\partial\left|A^P_{nqk}\mathsf{n}^f_k\right|}{\partial U^P_m}$ και $\left|\mathcal{A}^P_{nmk}\mathsf{n}^f_k\right|$ η τροποποιημένη Ιακωβιανή ορίζουσα. Η Εξ. 4.7 εκφράζει το προτεινόμενο, συμβατό, μη-συντηρητικό σχήμα διακριτοποίησης σύμφωνα με τη *TDDC* συζυγή μέθοδο.

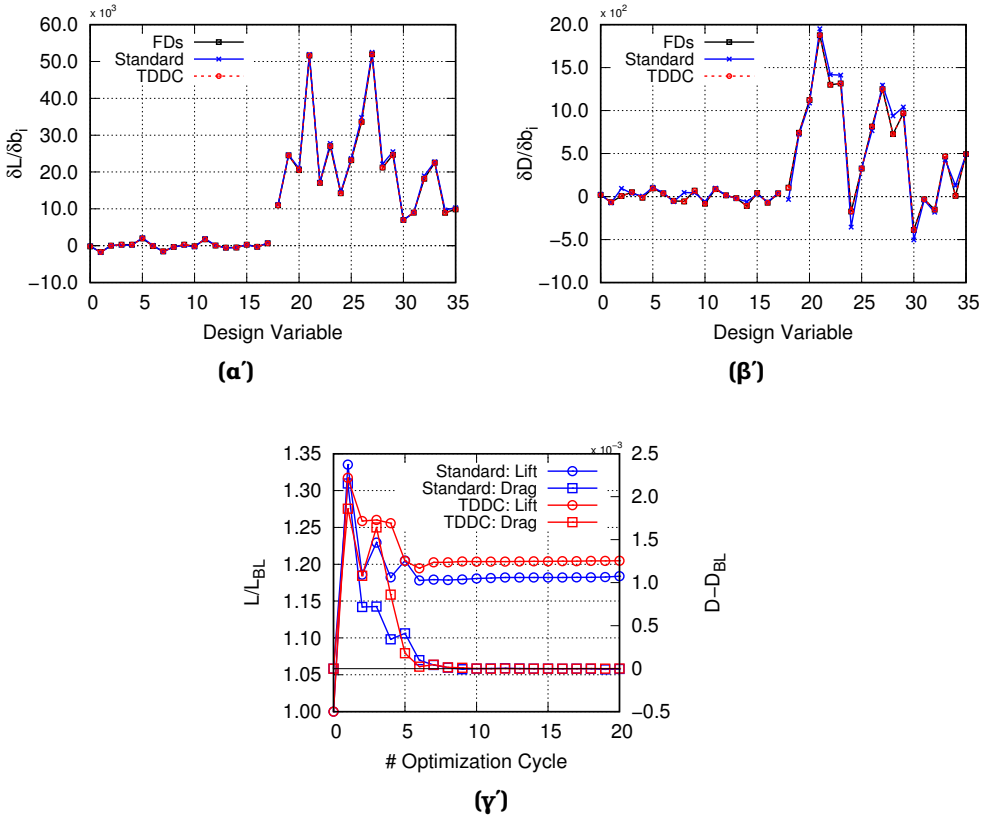
Στο πλήρες κείμενο της Διατριβής παρουσιάζονται τα προτεινόμενα *TDDC* σχήματα διακριτοποίησης και για τους υπόλοιπους τύπους ορίων.

4.2 Πιστοποίηση

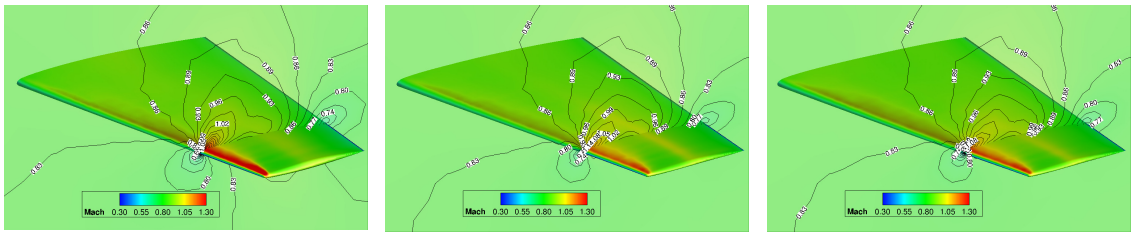
Στην ενότητα αυτή πιστοποιείται η *TDDC* συζυγής μέθοδος σε 2Δ και 3Δ εφαρμογές. Στόχος είναι η αποτίμηση της ακρίβειας υπολογισμού των παραγώγων ευαισθησίας με τη *TDDC* συζυγή μέθοδο και η σύγκρισή τους με εκείνες που υπολογίζονται από την 'κλασική' συζυγή μέθοδο (στην οποία χρησιμοποιούνται σχήματα διακριτοποίησης ευρέως γνωστά στη βιβλιογραφία) και με ΠΔ. Στο πλήρες κείμενο μελετώνται τέσσερις εφαρμογές, εδώ παρουσιάζονται δύο εξ αυτών.

Στην πρώτη εφαρμογή μελετάται η ροή γύρω από τη διηχητική πτέρυγα ONERA M6, [48]. Η ροή θεωρείται ατριβής με επ' άπειρο αριθμό Mach $M_\infty = 0.8395$, γωνία πρόσπτωσης AoA $= 3.06^\circ$ και γωνία εκτροπής $\alpha_{yaw} = 0.0^\circ$. Στόχος της βελτιστοποίησης είναι η μεγιστοποίηση της άνωσης (L) υπό τον περιορισμό ότι η οπισθέλκουσα δεν θα αυξηθεί συγκριτικά με αυτή της αρχικής γεωμετρίας. Οι παράγωγοι ευαισθησίας ως προς τις μεταβλητές σχεδιασμού συγκρίνονται με ΠΔ, Σχ. 4.1α΄ και 4.1β΄. Η *TDDC* συζυγής μέθοδος υπολογίζει παραγώγους με ακρίβεια τουλάχιστον έξι σημαντικών ψηφίων, αντίθετα, στην 'κλασική' συζυγή μέθοδο υπάρχουν παράγωγοι που υπολογίζονται ακόμα και με αντίθετο πρόσημο (μεταβλητή σχεδιασμό με ID 8 και 18). Η διαφορά στην ακρίβεια υπολογισμού των παραγώγων ευαισθησίας αντικατοπτρίζεται και στην πορεία σύγκλισης της βελτιστοποίησης, Σχ. 4.1γ΄, όπου για τον ίδιο αριθμό κύκλων βελτιστοποίησης η 'κλασική' και η *TDDC* συζυγής μέθοδος οδηγούν σε αύξηση της άνωσης κατά $\sim 19.2\%$ και $\sim 21.5\%$, αντίστοιχα, ικανοποιώντας τον περιορισμό. Το πεδίο του αριθμού Mach για την αρχική και τις βελτιστοποιημένες πτέρυγες παρουσιάζεται στο Σχ. 4.2.

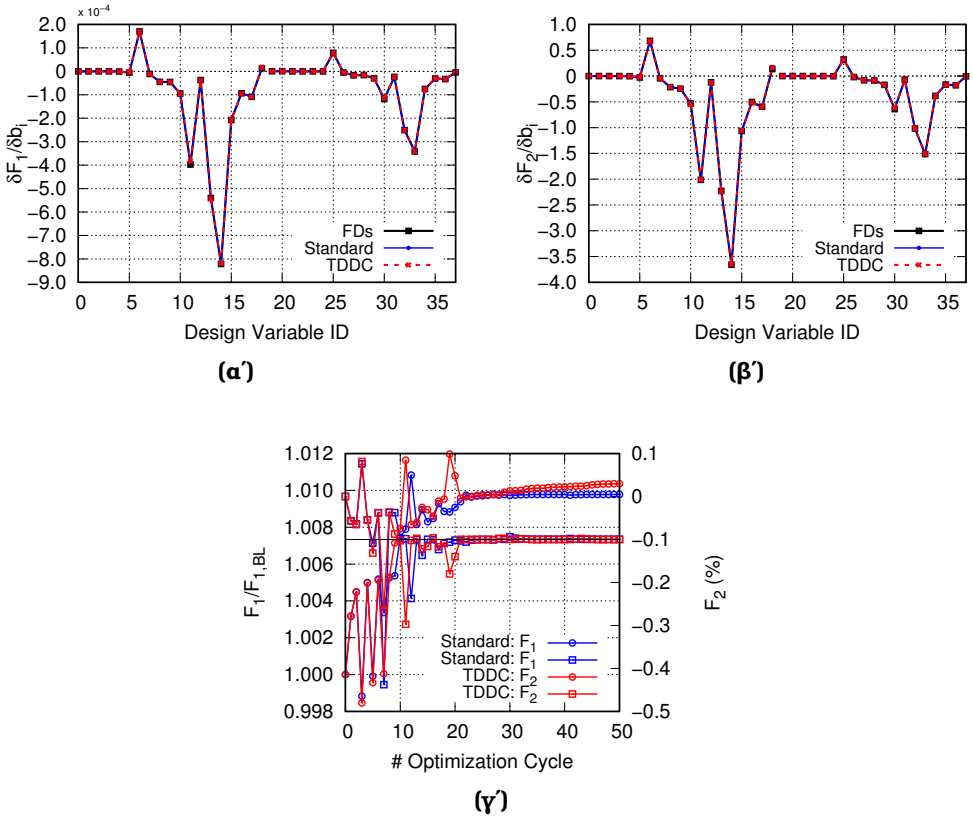
Η δεύτερη περίπτωση αφορά τη βελτιστοποίηση μορφής ενός 2Δ στροβίλου υψηλής πίεσης, [17], για μέγιστη ανηγμένη παροχή εισόδου (F_1) υπό τον περιορισμό η στροφή της ροής (F_2) να μην μεταβάλλεται περισσότερο από το $\pm 1‰$ της στροφής στην αρχική γεωμετρία. Εδώ, χρησιμοποιείται το μοντέλο των Spalart–Allmaras. Οι παράγωγοι ευαισθησίας των F_1 και F_2 καθώς και η πορεία σύγκλισης της βελτιστοποίησης παρουσιάζεται στο Σχ. 4.3. Στην περίπτωση αυτή, η ακρίβεια υπολογισμού τους είναι τουλάχιστον έξι σημαντικά ψηφία για την *TDDC* και τρία για την 'κλασική' συνεχή συζυγή μέθοδο. Ως αποτέλεσμα, η βελτιστοποίηση με τη *TDDC* συζυγή μέθοδο οδηγεί σε αύξηση της ανηγμένης παροχής κατά 1.197%, σε αντίθεση με την 'κλασική' συζυγή όπου η βελτίωση είναι 0.979%.



Σχήμα 4.1: *Βελτιστοποίηση Διηχητικής Πτέρυγας*: Παράγωγοι ευαισθησίας ως προς την (α′) την άνωση (L) και (β′) την οπισθέλκουσα (D) όπως υπολογίστηκαν από την 'κλασική' συνεχή και τη *TDDC* συζυγή μέθοδο αλλά και με ΠΔ. (γ′) Πορεία σύγκλισης των βελτιστοποιήσεων με την 'κλασική' συνεχή και τη *TDDC* συζυγή μέθοδο.



Σχήμα 4.2: *Βελτιστοποίηση Διηχητικής Πτέρυγας*: Πεδίο αριθμού Mach γύρω από (α′) την αρχική και τη βελτιστοποιημένη πτέρυγα χρησιμοποιώντας (β′) την 'κλασική' συνεχή και (γ′) τη *TDDC* συζυγή μέθοδο. Τομή στο 75% του εκπετάσματος της πτέρυγας.



Σχήμα 4.3: *Βελτιστοποίηση Στροβίλου*: Παράγωγοι ευαισθησίας ως προς (α′) την ανηγμένη παροχή (F_1) και (β′) τη στροφή της ροής (F_2) όπως υπολογίστηκαν από την 'κλασική' συνεχή, τη *TDDC* συζυγή μέθοδο και από ΠΔ. Πορεία σύγκλισης της βελτιστοποίησης για τις δύο συζυγείς μεθόδους.

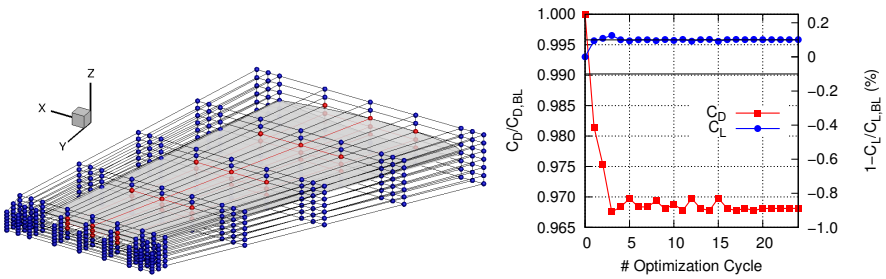
Κεφάλαιο 5

Βελτιστοποίηση Μορφής με χρήση της Συνεχούς Συζυγούς Μεθόδου για Ροές με Μετάβαση

Η συνεχής συζυγής μέθοδος για ροές με μετάβαση όπως αναπτύχθηκε στο Κεφάλαιο 3 που κάνει χρήση των *TDDC* σχημάτων διακριτοποίησης, Κεφάλαιο 4, χρησιμοποιείται για τη βελτιστοποίηση μορφής δύο εφαρμογών βιομηχανικού ενδιαφέροντος.

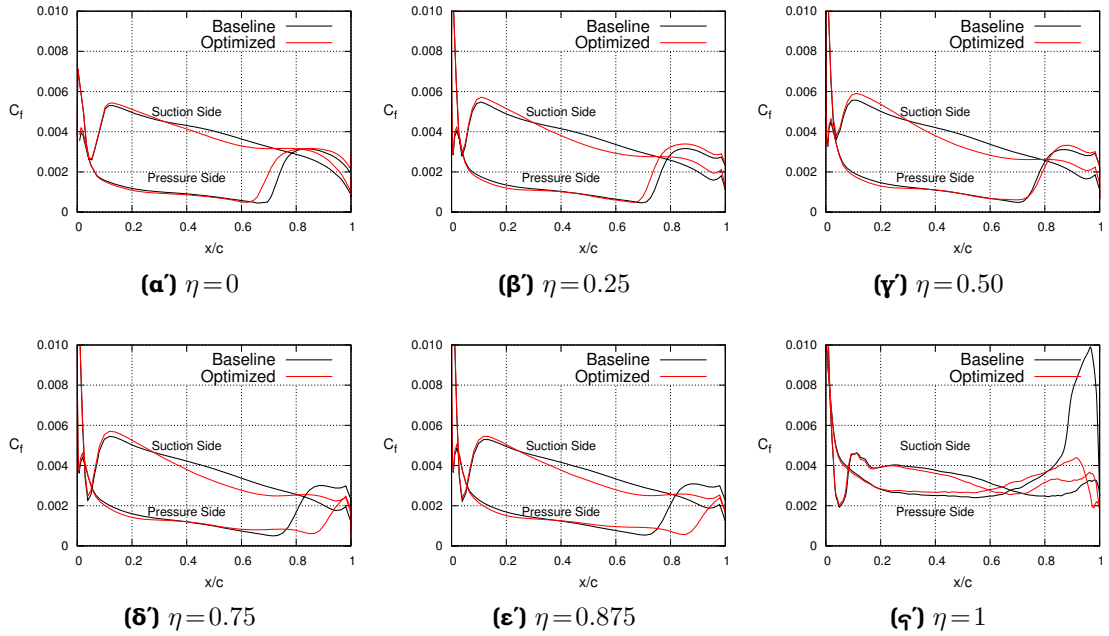
5.1 Βελτιστοποίηση Μορφής Μεμονωμένης Πτέρυγας

Στην ενότητα αυτή, βελτιστοποιείται η πτέρυγα ONERA M6, [48], σε ροή με μετάβαση με το μοντέλο SA-sLM2015. Στόχος είναι η ελαχιστοποίηση του συντελεστή οπισθέλκουσας (C_D) υπό τον περιορισμό πως ο συντελεστής άνωσης (C_L) δεν μεταβάλλεται περισσότερο από $\pm 1‰$ συγκριτικά με την αρχική γεωμετρία. Ο επ' άπειρον αριθμός Mach, ο αριθμός Reynolds, οι γωνίες πρόσπτωσης και εκτροπής, η ισχύς της τύρβης και η τραχύτητα της επιφάνειας είναι $M_\infty = 0.262$, $Re = 3.5 \cdot 10^6$, AoA $= 5.0°$, $\alpha_{yaw} = 0.0°$, $Tu = 0.2\,\%$ και $h_{rms} = 5.0\,\mu m$. Η παραμετροποίηση της πτέρυγας και η πορεία σύγκλισης της βελτιστοποίησης παρουσιάζονται στα Σχ. 5.1α΄ και 5.1β΄. Στο Σχ. 5.2 παρουσιάζονται κατανομές του συντελεστή τριβής για τομές σε διάφορες θέσεις κατά το εκπέτασμα της πτέρυγας. Συνολικά, η μετατόπιση της γραμμής της μετάβασης οδηγεί σε $\sim 3.3\%$ μείωση του C_D.



(α΄) (β΄)

Σχήμα 5.1: *Βελτιστοποίηση Μορφής Μεμονωμένης Πτέρυγας*: (α΄) Παραμετροποίηση. (β΄) Πορεία σύγκλισης της βελτιστοποίησης με το μοντέλο μετάβασης SA-sLM2015.



(α′) $\eta = 0$ **(β′)** $\eta = 0.25$ **(γ′)** $\eta = 0.50$

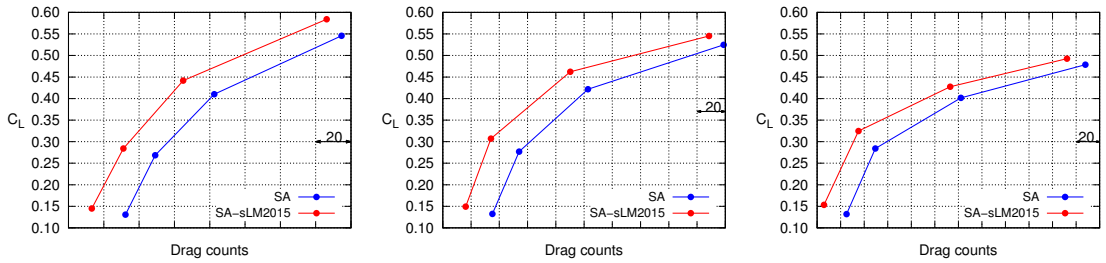
(δ′) $\eta = 0.75$ **(ε′)** $\eta = 0.875$ **(ϛ′)** $\eta = 1$

Σχήμα 5.2: *Βελτιστοποίηση Μορφής Μεμονωμένης Πτέρυγας*: Κατανομή του συντελεστή τριβής (C_f) στην πλευρά υπερπίεσης και υποπίεσης για την αρχική και τη βελτιστοποιημένη πτέρυγα σε διάφορες τομές (η) κατά το εκπέτασμά της (α′) $\eta = 0$, (β′) $\eta = 0.25$, (γ′) $\eta = 0.50$, (δ′) $\eta = 0.75$, (ε′) $\eta = 0.875$ και (ϛ′) $\eta = 1$.

5.2 Βελτιστοποίηση Μορφής Πτέρυγας Αεροσκάφους Υψηλού Λόγου Επιμήκους

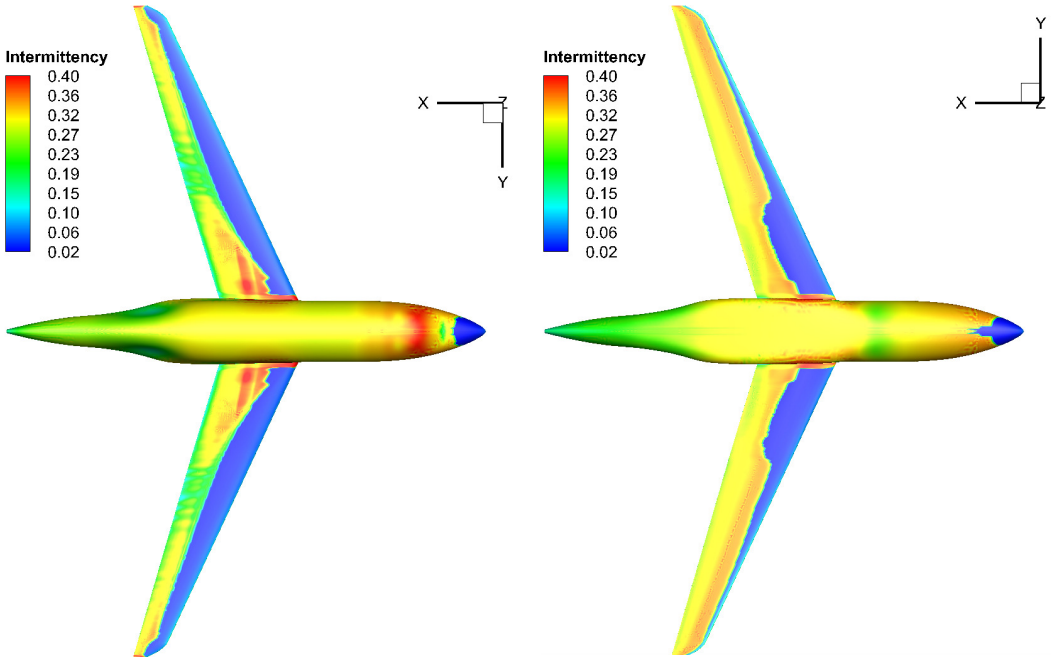
Η εφαρμογή αυτή εστιάζει στη βελτιστοποίηση μορφής πτέρυγας αεροσκάφους υψηλού λόγου επιμήκους (High Aspect-Ratio Wing, HARW). Στόχος της βελτιστοποίησης είναι η ελαχιστοποίηση του συντελεστή οπισθέλκουσας (C_D) υπό τον περιορισμό ο συντελεστής άνωσης της πλήρους γεωμετρίας (C_L) ισούται με 0.5 με ακρίβεια $\pm 10^{-4}$. Η ανάλυση της ροής πραγματοποιείται για τρεις αριθμούς Mach και τρεις γωνίες πρόσπτωσης, $M_\infty = 0.76,\ 0.78,\ 0.80$ και $\text{AoA} = 0.0^\circ,\ 1.0^\circ,\ 2.0^\circ,\ 3.0^\circ$. Οι πολικές με το μοντέλο τύρβης των Spalart–Allmaras, με και χωρίς το μοντέλο μετάβασης SA-sLM2015, παρουσιάζονται στο σχήμα Σχ. 5.3, ενώ στο Σχ. 5.4 φαίνεται το πεδίο του γ στην άνω και στην κάτω πλευρά του αεροσκάφους.

Η πτέρυγα του αεροσκάφους παραμετροποιείται όπως φαίνεται στο Σχ. 5.5α′. Η πορεία σύγκλισης της βελτιστοποίησης (όπου επετεύχθη $\sim 17\%$ μείωση του C_D) και το πεδίο του συντελεστή τριβής για την αρχική και βελτιστοποιημένη γεωμετρία παρουσιάζονται στα Σχ. 5.5β′ και 5.6.



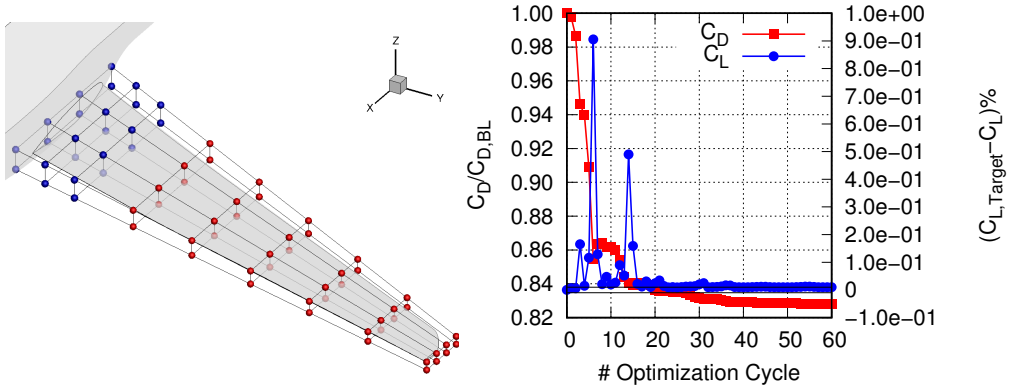
(α′) $M_\infty = 0.76$ **(β′)** $M_\infty = 0.78$ **(γ′)** $M_\infty = 0.80$

Σχήμα 5.3: *Βελτιστοποίηση Αεροσκάφους HARW*: Πολικές C_L-C_D όπως υπολογίστηκαν από προσομοιώσεις με το μοντέλο τύρβης Spalart–Allmaras (χωρίς μετάβαση) και με το μοντέλο μετάβασης SA-sLM2015 για (α′) $M_\infty = 0.76$, (β′) $M_\infty = 0.78$ και (γ′) $M_\infty = 0.80$.



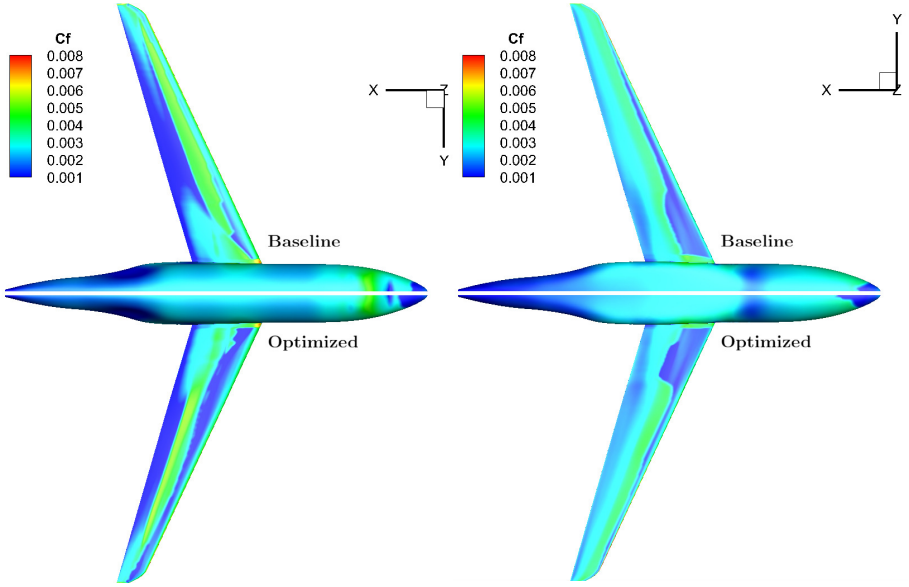
(α′) Άνω Όψη **(β′)** Κάτω Όψη

Σχήμα 5.4: *Βελτιστοποίηση Αεροσκάφους HARW*: Πεδίο του γ για την άνω και κάτω πλευρά του αεροσκάφους για $M_\infty = 0.78$ και AoA $= 2°$.



(α′) **(β′)**

Σχήμα 5.5: *Βελτιστοποίηση Αεροσκάφους HARW*: (α′) Παραμετροποίηση της πτέρυγας. (β′) Πορεία σύγκλισης της βελτιστοποίησης για $M_\infty = 0.78$.



(α′) Άνω Όψη **(β′)** Κάτω Όψη

Σχήμα 5.6: *Βελτιστοποίηση Αεροσκάφους HARW*: Πεδίο του συντελεστή τριβής (C_f) στην επιφάνεια του αεροσκάφους για την αρχική και τη βελτιστοποιημένη γεωμετρία για $M = 0.78$ και $C_L = 0.50$.

Κεφάλαιο 6

ΒΝΔ ως Υποκατάστατα Μοντέλων Τύρβης & Μετάβασης για Βελτιστοποίηση Μορφής με ΕΑ

Η χρήση Βαθέων Νευρωνικών Δικτύων (ΒΝΔ) στον τομέα της υπολογιστικής ρευστοδυναμικής μπορεί να λειτουργήσει ως αρωγός στην κατεύθυνση μείωσης του υπολογιστικού κόστους των προσομοιώσεων, εξασφαλίζοντας ταυτόχρονα υψηλή ακρίβεια προβλέψεων. Εδώ, προτείνεται η χρήση ΒΝΔ ως υποκατάστατα των μοντέλων τύρβης και μετάβασης κατά την αριθμητική επίλυση των εξισώσεων RANS. Ο ρόλος του ΒΝΔ είναι η πρόβλεψη του πεδίου της τυρβώδους συνεκτικότητας σε κάθε ψευδο-χρονικό βήμα της επίλυσης των εξισώσεων μέσης ροής αντικαθιστώντας τα μοντέλα τύρβης και μετάβασης (Spalart–Allmaras, $k-\omega$ SST και $\gamma-\tilde{R}e_{\theta t}$), οδηγώντας σε μείωση του υπολογιστικού κόστους της προσομοίωσης. Το νέο λογισμικό χρησιμοποιείται για τη βελτιστοποίηση μορφής μέσω ΜΑΕΑ. Οι βασικές αρχές των ΒΝΔ περιγράφονται στο πλήρες κείμενο της Διατριβής.

6.1 ΒΝΔ ως Υποκατάστατα Μοντέλων Τύρβης

Στην ενότητα αυτή προτείνεται η χρήση ΒΝΔ ως υποκατάστατο των μοντέλων τύρβης και μετάβασης, στοχεύοντας στη μείωση του υπολογιστικού κόστους του λογισμικού αξιολόγησης και στην εισαγωγή αυτού σε βελτιστοποίηση μορφής με ΜΑΕΑ.

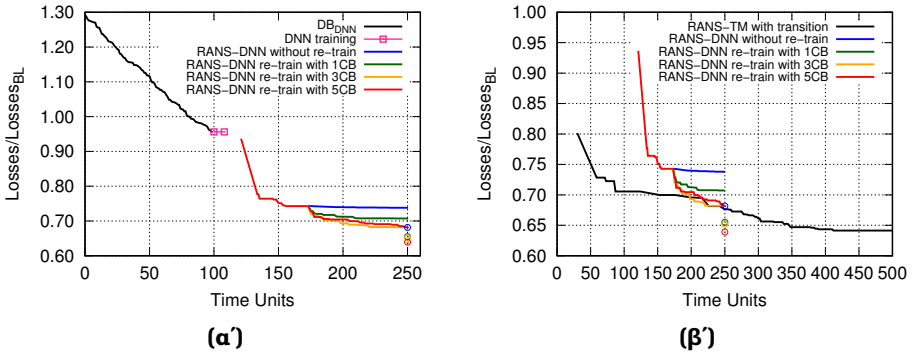
Πρώτο βήμα αποτελεί η γένεση της βάσης δεδομένων πάνω στην οποία θα εκπαιδευθεί το ΒΝΔ. Σε όλες τις εφαρμογές χρησιμοποιείται η τεχνική δειγματοληψίας Latin hypercube, κάθε νέα γεωμετρία προσομοιώνεται με το ακριβές λογισμικό και αποθηκεύεται στη βάση δεδομένων. Επόμενο βήμα αποτελεί η εύρεση του βέλτιστου ΒΝΔ, το οποίο αναφέρεται τόσο στις υπερπαραμέτρους (αριθμός των επιπέδων, αριθμός νευρώνων ανά επίπεδο, τύπος συναρτήσεων ενεργοποίησης) όσο και στα (γεωμετρικά και ροϊκά) δεδομένα εισόδου του δικτύου. Εδώ, το βέλτιστο ΒΝΔ απορρέει από μια βελτιστοποίηση με ΜΑΕΑ της πλατφόρμας EASY, [2], που αναπτύχθηκε στην ΜΠΥΡ&Β και στοχεύει στη μείωση του σφάλματος πρόβλεψης του δικτύου. Για κάθε επίπεδο, ο αριθμός των νευρώνων ορίζεται ως δύναμη του 2, ενώ οι διαθέσιμες συναρτήσεις ενεργοποίησης είναι οι Rectified Linear Unit (ReLU), hyperbolic tangent (*tanh*), sigmoid και Gaussian Error Linear Unit (GELU). Για τα δεδομένα εισόδου, ο αλγόριθμος μπορεί να επιλέξει ανάμεσα σε γεωμετρικές (χωρικές συντε-

ταγμένες και απόσταση από το στερεό τοίχωμα) και ροϊκές ποσότητες (πυκνότητα, διάνυσμα της ταχύτητας, πίεση, στροβιλισμός και μέτρο των διατμητικών τάσεων). Μετά την εκπαίδευση, το ΒΝΔ εισάγεται στο οικείο λογισμικό PUMA και παρέχει το πεδίο της τυρβώδους συνεκτικότητας σε κάθε ψευδο–χρονικό βήμα της επίλυσης των εξισώσεων μέσης ροής. Το νέο λογισμικό (που θα συμβολίζεται ως RANS-DNN) είναι οικονομικά αποδοτικότερο του ακριβούς επιλύτη (που θα συμβολίζεται ως RANS-TM) και χρησιμοποιείται στη βελτιστοποίηση μορφής με ΜΑΕΑ. Επομένως, ο ρόλος του ΜΑΕΑ είναι διττός καθώς φέρει εις πέρας τόσο την εύρεση του βέλτιστου ΒΝΔ όσο και τη βελτιστοποίηση μορφής. Ανάλογα με τη φύση της κάθε εφαρμογής, ενδέχεται να είναι απαραίτητη η επανεκπαίδευση/ανανέωση του ΒΝΔ κατά τη διαδικασία της βελτιστοποίησης, Η ανάπτυξη και εκπαίδευση του δικτύου πραγματοποιείται στην πλατφόρμα TensorFlow, ενώ για τη σύζευξή του με το οικείο λογισμικό PUMA γίνεται χρήση του TensorFlow C++ API.

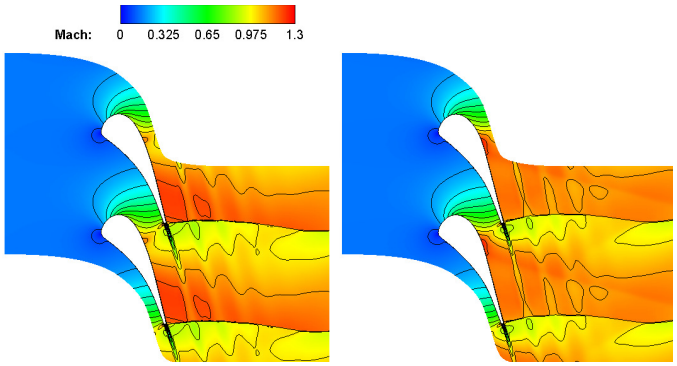
Στο πλήρες κείμενο της Διατριβής παρουσιάζονται τρεις εφαρμογές, εδώ αναφέρονται ενδεικτικά οι δύο εξ αυτών.

6.2 Βελτιστοποίηση Μορφής Διηχητικού Πτερυγίου Στροβίλου

Η ενότητα αυτή επικεντρώνεται στη βελτιστοποίηση μορφής του διηχητικού πτερυγίου στροβίλου LS89, [4] (Περίπτωση 116), για ελάχιστες απώλειες ολικής πίεσης σταθμισμένες με τη μάζα υπό τον περιορισμό η γωνία εξόδου να παραμένει κοντά σε αυτή της αρχικής γεωμετρίας σε ένα εύρος $\pm 0.5^\circ$. Οι συνθήκες ροής είναι $T_t^{\text{in}} = 418.90\,\text{K}$, $p_t^{\text{in}} = 3.269\,\text{bar}$, $Tu = 0.8\%$, $\alpha^{\text{in}} = 0.0^\circ$ και $p_t^{\text{out}} = 1.550\,\text{bar}$. Στο στέρεο τοίχωμα $T^{\text{w}} = 297.55\,\text{K}$. Ο ρόλος του ΒΝΔ είναι η αντικατάσταση τεσσάρων εξισώσεων, δύο του μοντέλου τύρβης $k-\omega$ SST και δύο του μοντέλου μετάβασης $\gamma - \tilde{Re}_{\theta t}$, και η πρόβλεψη του πεδίου της τυρβώδους συνεκτικότητας. Παραμετροποιώντας το πτερύγιο στροβίλου, δημιουργούνται και αξιολογούνται με το ακριβές λογισμικό 100 νέες γεωμετρίες για τη βάση δεδομένων εκπαίδευσης και εντοπίζεται το βέλτιστο ΒΝΔ (ως προς την αρχιτεκτονική και τα δεδομένα εισόδου) μέσω ΜΑΕΑ. Ο νέος RANS-DNN επιλύτης είναι κατά $\sim 38\%$ ταχύτερος του ακριβούς RANS-TM επιλύτη. Για δεδομένο υπολογιστικό κόστος (στο οποία περιλαμβάνεται ο χρόνος δόμησης του ΒΝΔ) πραγματοποιούνται δύο βελτιστοποιήσεις βασιζόμενες στον RANS-TM και στον RANS-DNN επιλύτη, Σχ. 6.1α′. Για τη βελτιστοποίηση με τον RANS-DNN επιλύτη απαιτείται επανεκπαίδευση του ΒΝΔ κατά τη διάρκεια της βελτιστοποίησης προκειμένου να αυξηθεί η ακρίβειά του. Για δεδομένο υπολογιστικό κόστος, ο RANS-TM καταλήγει σε πτερύγιο στροβίλου με 0.6762 απώλειες πίεσης συγκριτικά με την αρχική γεωμετρία, ενώ ο RANS-DNN με 0.6389. Μάλιστα ο RANS-TM επιλύτης θα χρειαζόταν $\sim 40\%$ περισσότερο υπολογιστικό χρόνο για να συγκλίνει σε λύση ίδιας ποιότητας με εκείνης του RANS-DNN, Σχ. 6.1β′. Τα πεδία του αριθμού Mach για το αρχικό και το βελτιστοποιημένο πτερύγιο με τον RANS-DNN επιλύτη παρουσιάζονται στο Σχ. 6.2. Η ένταση του κύματος κρούσης έχει μειωθεί οδηγώντας και στη μείωση των απωλειών πίεσης.



Σχήμα 6.1: *Βελτιστοποίηση Πτερυγίου Στροβίλου LS89*: Πορείες σύγκλισης των βελτιστοποιήσεων, με ΜΑΕΑ, με τον RANS-DNN επιλύτη. Παραμετρική μελέτη ως προς των αριθμό των ατόμων που χρησιμοποιούνται για την επανεκπαίδευση του ΒΝΔ. (α′): Η βάση δεδομένων σε φθίνουσα σειρά και το κόστος εκπαίδευσης του ΒΝΔ. (β′): Πορεία σύγκλισης της βελτιστοποίησης με τον RANS-TM επιλύτη.

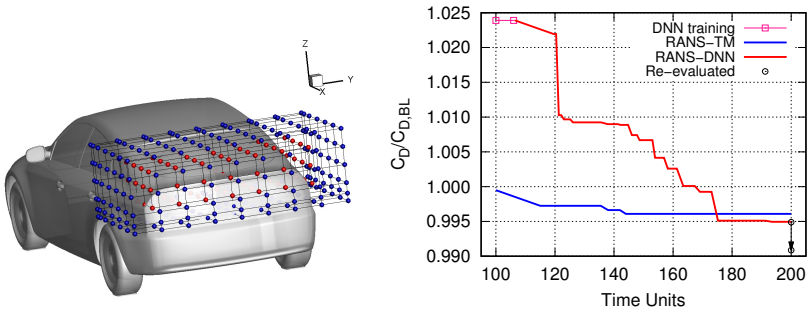


Σχήμα 6.2: *Βελτιστοποίηση Πτερυγίου Στροβίλου LS89*: Πεδίο αριθμού Mach για το αρχικό (αριστερά) και το βελτιστοποιημένο πτερύγιο με τον RANS-DNN επιλύτη.

6.3 Βελτιστοποίηση Μορφής Μοντέλου Αυτοκινήτου DrivAer

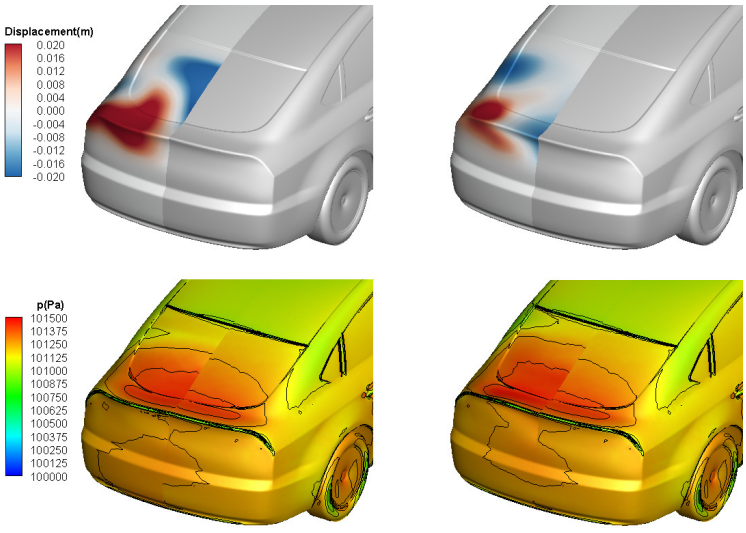
Στην ενότητα αυτή πραγματοποιείται βελτιστοποίηση μορφής μοντέλου αυτοκινήτου τύπου DrivAer, [16], για ελάχιστο συντελεστή οπισθέλκουσας (C_D). Το ρευστό θεωρείται ασυμπίεστο με $U_\infty = 38.85\,\mathrm{m/s}$, $\nu = 1.5\,\mathrm{m^2/s}$ και $\mathrm{AoA} = 0.0^\circ$. Στόχος του ΒΝΔ είναι η αντικατάσταση της αριθμητικής επίλυσης του μοντέλου τύρβης Spalart–Allmaras παρέχοντας το πεδίο της τυρβώδους συνεκτικότητας. Γύρω από το μοντέλο αυτοκινήτου δημιουργείται ένα πλέγμα $\sim 1.4M$ κόμβων και παραμετροποιείται το επάνω-πίσω μέρος του, Σχ. 6.3α′. Δημιουργείται μια βάση δεδομένων με 100 γεωμετρίες αυτοκινήτου και προσδιορίζεται το βέλτιστο ΒΝΔ (αρχιτεκτονική και ροϊκά/γεωμετρικά δεδομένα εισόδου) μέσω βελτιστοποίησης βασιζόμενη σε ΜΑΕΑ. Ο νέος επιλύτης RANS-DNN είναι κατά $\sim 20\%$ οικονομικά αποδοτικότερος συγκριτικά με τον ακριβή RANS-TM. Η πορεία σύγκλισης των δύο βελτιστοποιήσεων παρουσιάζεται

στο Σχ. 6.3β′. Για το ίδιο υπολογιστικό κόστος, ο RANS-TM επιλύτης συνέκλινε σε μια γεωμετρία αυτοκινήτου με 0.39% μείωση στο C_D, ενώ ο RANS-DNN επιλύτης σε 0.91% μείωση. Η μετατόπιση κάθετα στην επιφάνεια του αυτοκινήτου και η κατανομή της πίεσης για τις δύο βελτιστοποιημένες λύσεις παρουσιάζονται στο Σχ. 6.4.



(α′) **(β′)**

Σχήμα 6.3: *Βελτιστοποίηση Μοντέλου Αυτοκινήτου DrivAer*: (α′) Παραμετροποίηση του επάνω-πίσω μέρους του αυτοκινήτου. (β′) Πορείες σύγκλισης των βελτιστοποιήσεων με ΜΑΕΑ για τον RANS-TM και τον RANS-TM επιλύτη. Η βελτιστοποιημένη λύση του RANS-DNN επιλύτη επαναξιολογείται με τον RANS-TM.



Σχήμα 6.4: *Βελτιστοποίηση Μοντέλου Αυτοκινήτου DrivAer*: Μετατόπιση κάθετα στην επιφάνεια του αυτοκινήτου και πεδίο πίεσης στις δύο βελτιστοποιημένες γεωμετρίες αυτοκινήτου με τον RANS-TM (αριστερά) και τον RANS-DNN (δεξιά) επιλύτη. Σύγκριση με την αρχική γεωμετρία (δεξί μισό τμήμα του αυτοκινήτου).

Κεφάλαιο 7

ΒΝΔ σε Πολυπεδιακά Προβλήματα και Προβλήματα Βελτιστοποίησης Στροβιλομηχανών με Περιστρεφόμενες και Σταθερές Πτερυγώσεις

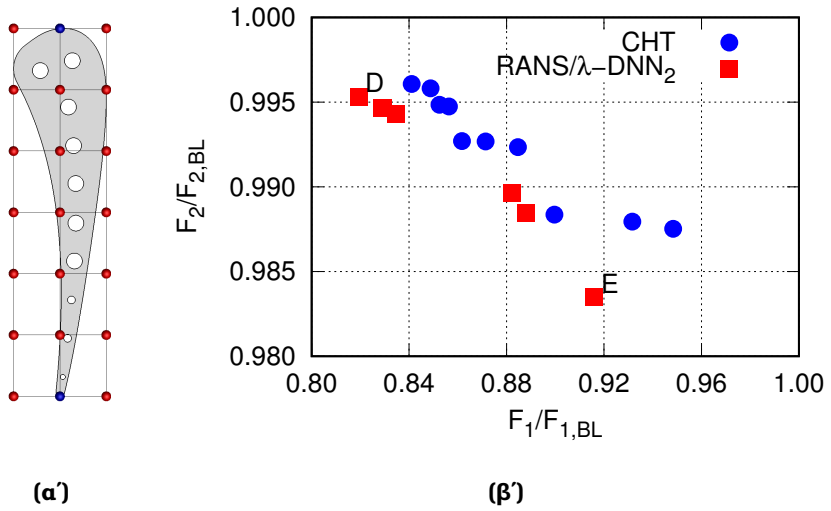
Στο κεφάλαιο αυτό τα ΒΝΔ εφαρμόζονται σε πολυπεδιακά προβλήματα βελτιστοποίησης και σε προβλήματα βελτιστοποίησης πτερυγώσεων στροβιλομηχανών που περιλαμβάνουν αλληλεπίδραση μεταξύ περιστρεφόμενων και σταθερών μερών. Για τα πρώτα προτείνεται η χρήση της δίκλαδης αρχιτεκτονικής $\lambda-$DNN για την αντικατάσταση του επιλύτη ενός πεδίου, ενώ για τα προβλήματα στροβιλομηχανών τα ΒΝΔ χρησιμοποιούνται ως τρόπος μείωσης των μερών που απαιτείται να επιλυθούν αριθμητικά. Και στις δύο περιπτώσεις ο κύριος στόχος είναι η μείωση του υπολογιστικού κόστους του λογισμικού αξιολόγησης. Όλες οι βελτιστοποιήσεις πραγματοποιούνται με την πλατφόρμα EASY [2], που έχει αναπτυχθεί στην ΜΠΥΡ&Β.

7.1 Η Αρχιτεκτονική $\lambda-$DNN σε Πολυπεδιακά Προβλήματα Βελτιστοποίησης Μορφής

Σε προβλήματα συζευγμένης μεταφοράς θερμότητας, οι εξισώσεις ροής επιλύονται μαζί με την εξίσωση αγωγής θερμότητας στο στερεό σώμα. Στο πλήρες κείμενο της Διατριβής παρουσιάζεται αναλυτικά η εξίσωση αγωγής θερμότητας καθώς και η σύζευξή της με τις ροϊκές εξισώσεις. Προτείνεται η χρήση της αρχιτεκτονικής $\lambda-$DNN σε πολυπεδιακά προβλήματα βελτιστοποίησης. Πρόκειται για μια δίκλαδη αρχιτεκτονική με χαρακτηριστικό γνώρισμά της τη διακριτή επεξεργασία δεδομένων εισόδου διαφορετικής φύσεως. Εδώ χρησιμοποιείται σε προβλήματα βελτιστοποίησης συζευγμένης μεταφοράς θερμότητας αντικαθιστώντας τον επιλύτη της εξίσωσης αγωγής θερμότητας με στόχο τη μείωση του υπολογιστικού κόστους του λογισμικού αξιολόγησης. Στο πλήρες κείμενο εφαρμόζεται σε δύο προβλήματα, ενδεικτικά εδώ παρουσιάζεται το ένα.

Η εφαρμογή αφορά την αεροθερμική βελτιστοποίηση του εσωτερικά ψυχόμενου πτερυγίου στροβίλου C3X, [17], στοχεύοντας στην ελαχιστοποίηση τόσο των απωλειών ολικής πίεσης του θερμού αερίου όσο και της μέγιστης θερμοκρασίας του μετάλλου. Οι συνθήκες ροής για το θερμό αέριο, το μέταλλο και τους 10 αγωγούς ψύξης, όπως

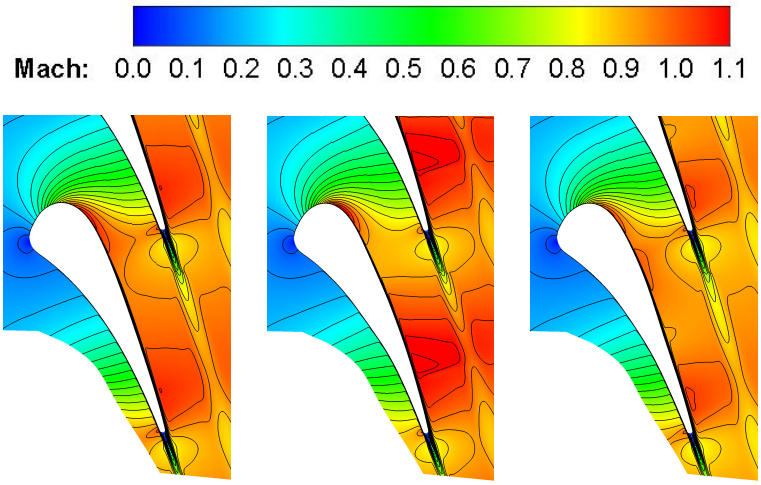
και η επαλήθευση με τα διαθέσιμα πειραματικά δεδομένα παρουσιάζεται με λεπτομέρεια στο πλήρες κείμενο. Για τις ανάγκες της βελτιστοποίησης το πτερύγιο παραμετροποιείται όπως στο Σχ. 7.1α΄. Στην εφαρμογή αυτή, η αρχιτεκτονική $\lambda-$DNN χρησιμοποιείται με δύο τρόπους. Στον πρώτο, το $\lambda-$DNN αντικαθιστά τον επιλύτη της μεταφοράς θερμότητας του μετάλλου προβλέποντας τη θερμοκρασιακή κατανομή στη διεπιφάνεια στερεού και αερίου σε κάθε αεροθερμικό κύκλο, ενώ στο δεύτερο εκπαιδεύεται για να προβλέπει τη θερμοκρασιακή κατανομή του τελευταίου (συγκλιμένου) κύκλου. Εδώ επιλέγεται να παρουσιαστεί η δεύτερη χρήση (RANS/$\lambda-$DNN$_2$). Στο Σχ. 7.1β΄ παρουσιάζεται το μέτωπο μη-κυριαρχούμενων λύσεων για τη ΜΑΕΑ βελτιστοποίηση με τον ακριβή (συμβολίζεται ως CHT) και τον RANS/$\lambda-$DNN$_2$ επιλύτη. Το μέτωπο που προκύπτει από τη βελτιστοποίηση με τον RANS/$\lambda-$DNN$_2$ επιλύτη κυριαρχεί έναντι αυτού με τον CHT. Μάλιστα, το πρώτο έχει συγκλίνει στο $\sim 73\%$ του διαθέσιμου υπολογιστικού κόστους κάτι που αναδεικνύει την αποτελεσματικότητα και ακρίβεια της προτεινόμενης αρχιτεκτονικής. Το πεδίο του αριθμού Mach για την αρχική γεωμετρία και για αυτές που αντιστοιχούν στα δύο άκρα του μετώπου παρουσιάζεται στο Σχ. 7.2.



Σχήμα 7.1: *Βελτιστοποίηση του Εσωτερικά Ψυχόμενου Πτερυγίου Στροβίλου C3X*: (α΄) Παραμετροποίηση. (β΄) Μέτωπο μη-κυριαρχούμενων λύσεων όπως προέκυψε από τις βελτιστοποιήσεις με τον CHT και τον RANS/$\lambda-$DNN$_2$ επιλύτη.

7.2 ΒΝΔ σε Προβλήματα Βελτιστοποίησης Στροβιλομηχανών με Περιστρεφόμενες και Σταθερές Πτερυγώσεις

Στην ενότητα αυτή, βελτιστοποιείται υδροστρόβιλος αποτελούμενος από 21 σταθερά, 21 οδηγά (δρομείς), 9 κινούμενα πτερύγια και τον αγωγό εξόδου, Σχ. 7.3α΄. Για την

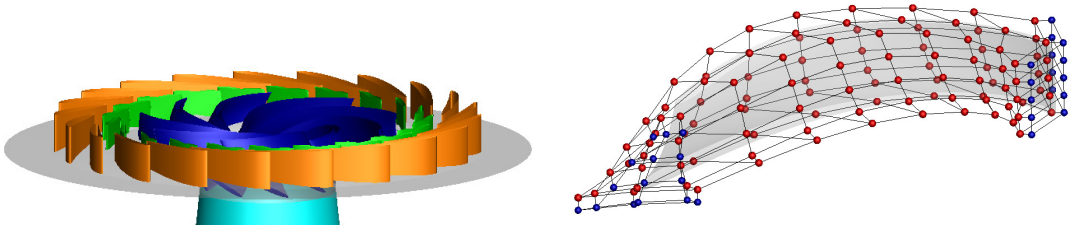


(α′) Αρχικό Πτερύγιο **(β′)** Πτερύγιο D **(γ′)** Πτερύγιο E

Σχήμα 7.2: *Βελτιστοποίηση του Εσωτερικά Ψυχόμενου Πτερυγίου Στροβίλου C3X*: Πεδίο αριθμού Mach για την αρχική γεωμετρία και για αυτές που αντιστοιχούν στα δύο άκρα του μετώπου μη-κυριαρχούμενων λύσεων του Σχ. 7.1β′, όπως προέκυψαν από τη βελτιστοποίηση με τον $\lambda-\text{DNN}_2$ επιλύτη.

επίλυση της ροής χρησιμοποιείται η τεχνική επιφάνειας ανάμειξης μεταξύ περιστρεφόμενων και σταθερών πτερυγώσεων. Η βελτιστοποίηση έχει ως στόχο τη μείωση των διακυμάνσεων της πίεσης ανάμεσα στα κινούμενα και σταθερά πτερύγια εξασφαλίζοντας πώς δεν θα δημιουργηθεί σπηλαίωση και πώς θα διατηρηθεί η παραγόμενη ισχύς. Καθώς οι ποσότητες ενδιαφέροντος ορίζονται αποκλειστικά στο χωρίο του δρομέα, η ιδέα είναι να δομηθούν δύο ΒΝΔ τα οποία θα αντικαθιστούν την τεχνική ανάμειξης παρέχοντας τις οριακές συνθήκες εισόδου (αντί για την επίλυση των σταθερών και οδηγών πτερυγίων) και εξόδου (αντί για την επίλυση του αγωγού εξόδου) στο δρομέα. Έτσι η αριθμητική επίλυση θα πραγματοποιείται μόνο για το χωρίο του δρομέα μειώνοντας το κόστος της κατά 40%. Το λογισμικό που χρησιμοποιεί την τεχνική ανάμειξης θα αναφέρεται ως CFD^{F}, ενώ εκείνο που κάνει χρήση των ΒΝΔ ως CFD^{R}. Για τις ανάγκες της βελτιστοποίησης τα κινούμενα πτερύγια του υδροστροβίλου παραμετροποιούνται όπως φαίνεται στο Σχ. 7.3β′.

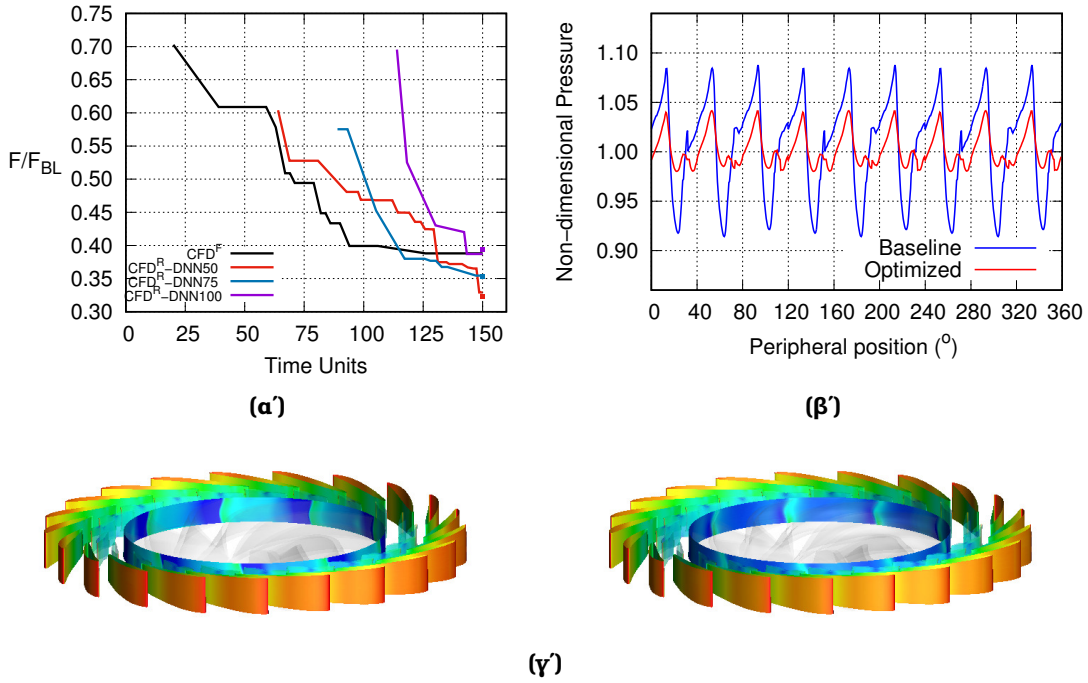
Στο πλήρες κείμενο πραγματοποιείται παραμετρική μελέτη αναφορικά με το μέγεθος της βάσης δεδομένων με την οποία γίνεται η εκπαίδευση των δικτύων ($\text{DB}^{50}_{\text{DNN}}$, $\text{DB}^{75}_{\text{DNN}}$,$\text{DB}^{100}_{\text{DNN}}$). Για κάθε βάση δεδομένων, οι αρχιτεκτονικές των ΒΝΔ απορρέουν από μια ΜΑΕΑ βελτιστοποίηση. Οι πορείες σύγκλισης των βελτιστοποιήσεων με τα CFD^{F} και CFD^{R} λογισμικά παρουσιάζονται στο Σχ. 7.4α′. Φαίνεται πως, για δεδομένο υπολογιστικό κόστος, οι βελτιστοποιήσεις που βασίζονται σε $\text{DB}^{50}_{\text{DNN}}$ και $\text{DB}^{75}_{\text{DNN}}$ υπερτερούν εκείνης με το ακριβές λογισμικό καθώς οδηγούν σε $\sim 68\%$ και $\sim 65\%$ μείωση των διακυμάνσεων της πίεσης, έναντι $\sim 61\%$. Για τα ΒΝΔ με $\text{DB}^{100}_{\text{DNN}}$



(α′) **(β′)**

Σχήμα 7.3: *Βελτιστοποίηση Υδροστροβίλου*: (α′) Ο υδροστρόβιλος αποτελούμενος από σταθερά πτερύγια (πορτοκαλί), οδηγά πτερύγια (πράσινα), κινούμενα πτερύγια (δρομείς) (μπλε) και (το αρχικό τμήμα) του αγωγού (μπλε του πάγου). (β′) Παραμετροποίηση.

η μείωση είναι επίσης $\sim 61\%$. Στα Σχ. 7.4β′ και 7.4γ′, παρουσιάζεται η κατανομή της πίεσης στην περιφερειακή κατεύθυνση καθώς και το πεδίο αυτής σε σταθερή ακτινική θέση για την αρχική και τη βελτιστοποιημένη γεωμετρία ($\mathrm{DB}^{50}_{\mathrm{DNN}}$).



(α′) **(β′)**

(γ′)

Σχήμα 7.4: *Βελτιστοποίηση Υδροστροβίλου*: (α′) Πορείες σύγκλισης των βελτιστοποιήσεων με $\mathrm{CFD}^{\mathrm{F}}$, $\mathrm{CFD}^{\mathrm{R}}$ με $\mathrm{DB}^{50}_{\mathrm{DNN}}$, $\mathrm{DB}^{75}_{\mathrm{DNN}}$ και $\mathrm{DB}^{100}_{\mathrm{DNN}}$. (β′) Σύγκριση της κατανομής της πίεσης στην περιφερειακή κατεύθυνση και (γ′) του πεδίου της πίεσης για σταθερή ακτινική θέση για την αρχική και τη βελτιστοποιημένη με $\mathrm{CFD}^{\mathrm{R}}$ και $\mathrm{DB}^{50}_{\mathrm{DNN}}$ γεωμετρία.

Κεφάλαιο 8

Επίλογος

Στόχος της Διατριβής ήταν η περαιτέρω επέκταση και πιστοποίηση υπολογιστικών εργαλείων για προβλήματα αερο/υδροδυναμικής βελτιστοποίησης μορφής, με μεθόδους ΥΡΔ περιστασιακά συνεργαζόμενες με τεχνικές Μηχανικής Μάθησης. Στα εργαλεία αυτά περιλαμβάνονται (α) η ανάπτυξη της συνεχούς συζυγούς μεθόδου για ροές με μετάβαση σε συμπιεστό ρευστό, (β) η ανάπτυξη συμβατών σχημάτων διακριτοποίησης για τις συζυγείς εξισώσεις όπως αυτές απορρέουν από τη συνεχή συζυγή μέθοδο και (γ) η χρήση ΒΝΔ για προβλέψεις ροής. Τα παραπάνω χρησιμοποιήθηκαν σε προβλήματα αιτιοκρατικής βελτιστοποίησης μορφής και βελτιστοποίησης με εξελικτικούς αλγορίθμους. Σε όλες τις εφαρμογές που μελετήθηκαν, στόχος ήταν η πραγματοποίηση μιας οικονομικά αποδοτικής βελτιστοποίησης μορφής.

8.1 Στοιχεία Πρωτοτυπίας και Σχετικές Δημοσιεύσεις

Οι πρωτοτυπίες της Διατριβής συνοψίζονται στα επόμενα:

- Αναπτύχθηκε, προγραμματίστηκε και πιστοποιήθηκε η συνεχής συζυγής μέθοδος για ροές με μετάβαση για συμπιεστό ρευστό. Είναι η πρώτη φορά στη σχετική βιβλιογραφία που η ανάπτυξη της συνεχούς συζυγούς μεθόδου πραγματοποιείται λαμβάνοντας υπόψη τη μετάβαση από στρωτή σε τυρβώδη ροή.
- Αναπτύχθηκαν συμβατά σχήματα διακριτοποίησης για τις συνεχείς συζυγείς εξισώσεις σύμφωνα με τη *Think-Discrete Do-Continuous* (*TDDC*) συζυγή μέθοδο. Προτάθηκαν σχήματα διακριτοποίησης για τα διανύσματα ατριβών, συνεκτικών ροών, καθώς και για τα μοντέλα τύρβης/μετάβασης του συζυγούς προβλήματος. Πρόκειται για την πρώτη προσπάθεια γεφύρωσης του χάσματος μεταξύ της συνεχούς και της διακριτής συζυγούς μεθόδου, στη σχετική βιβλιογραφία.
- Αναβαθμίστηκαν, οι δυνατότητες του λογισμικού PUMA για τη μοντελοποίηση της τύρβης/μετάβασης καθώς, πλέον, μπορούν να χρησιμοποιηθούν εναλλακτικά ΒΝΔ.
- Αναπτύχθηκε μια πλήρως αυτοματοποιημένη διαδικασία βασιζόμενη σε εξελικτικούς αλγορίθμους υποβοηθούμενους με μεταπρότυπα για τον καθορισμό του βέλτιστου ΒΝΔ. Για τον καθορισμό αυτού απαιτείται η εύρεση τόσο της αρχιτεκτονικής του ΒΝΔ όσο και των δεδομένων εισόδου σε αυτό.

- Επεκτάθηκε η χρήση της αρχιτεκτονικής $\lambda-$DNN σε πολυπεδιακά προβλήματα βελτιστοποίησης μορφής.
- Χρησιμοποιήθηκαν ΒΝΔ ως υποκατάστατα της τεχνικής επιπέδου ανάμειξης ανάμεσα σε περιστρεφόμενες και σταθερές πτερυγώσεις σε πρόβλημα βελτιστοποίησης στροβιλομηχανών.

Κατά τη διάρκεια της Διατριβής πραγματοποιήθηκαν τέσσερις (4) δημοσιεύσεις σε επιστημονικά περιοδικά, έξι (6) δημοσιεύσεις σε επιστημονικά συνέδρια και ένα (1) κεφάλαιο σε βιβλίο. Μία εκ των δημοσιεύσεων σε συνέδρια, έλαβε το βραβείο καλύτερης δημοσίευσης. Ενδεικτικά αναφέρονται οι δημοσιεύσεις σε επιστημονικά περιοδικά και αυτή που από συνέδριο έλαβε το βραβείο, ενώ οι υπόλοιπες δημοσιεύσεις παρουσιάζονται αναλυτικά στο πλήρες κείμενο της Διατριβής.

- V. Asouti, M. Kontou, K. Giannakoglou, "Radial Basis Function surrogates for uncertainty quantification and aerodynamic shape optimization under uncertainties". Fluids 2023; 8(11).
- M.G. Kontou, V.G. Asouti, K.C. Giannakoglou, "DNN surrogates for turbulence closure in CFD-based shape optimization". Applied Soft Computing 2023; 143:110013.
- X. Trompoukis, K. Tsiakas, V. Asouti, M. Kontou, K. Giannakoglou, "Optimization of an internally cooled turbine blade - Mathematical development and application". International Journal of Turbomachinery, Propulsion and Power 2021; 6(2):20.
- M. Kontou, D. Kapsoulis, I. Baklagis, X. Trompoukis, K.C. Giannakoglou, "$\lambda-$DNNs and their implementation in conjugate heat transfer shape optimization". Neural Computing and Applications 2022; 34:843-854.
- **Best Student Paper Award:** M. Kontou, D. Kapsoulis, I. Baklagis and K. Giannakoglou, "$\lambda-$DNNs and their implementation in aerodynamic and conjugate heat transfer optimization". 21st Engineering Applications of Neural Networks (EANN) Conference, Halkidiki, Greece, June 5-7, 2020.

8.2 Μελλοντικές Εργασίες

- Επέκταση της συνεχούς συζυγούς μεθόδου για ροές με μετάβαση από στρωτή σε τυρβώδη για το μοντέλο μετάβασης SST-2003-LM2015. Διορθώσεις για τη συμπιεστότητα (compressibility corrections) στον πρωτεύοντα όσο και στον συζυγή επιλύτη μπορούν να προστεθούν στο λογισμικό PUMA για τη μοντελοποίηση της μετάβασης.
- Επέκταση της *TDDC* συζυγούς μεθόδου σε όλα τα σχήματα διακριτοποίησης και πρόσθετα χαρακτηριστικά που υπάρχουν στο λογισμικό PUMA και δεν έγινε χρήση τους στη Διατριβή αυτή. Αναφορικά με τα σχήματα διακριτοποίησης

για το διάνυσμα ατριβών ροών, η *TDDC* μέθοδος θα μπορούσε να αναπτυχθεί και να προγραμματιστεί για τους υπόλοιπους περιοριστές του PUMA. Για τη μοντελοποίηση της τύρβης, προτείνεται η επέκταση της *TDDC* συζυγούς μεθόδου στο μοντέλο τύρβης k–ω SST, επιπρόσθετα, μπορούν να αναπτυχθούν συμβατά σχήματα διακριτοποίησης για τις συναρτήσεις τοίχου (wall functions) οι οποίες εφαρμόζονται στα μοντέλα τύρβης Spalart–Allmaras και $k-\omega$ SST. Όσον αφορά τα πρόσθετα χαρακτηριστικά του λογισμικού PUMA, η *TDDC* μέθοδος μπορεί επίσης να διατυπωθεί για την αλληλεπίδραση περιστρεφόμενων και σταθερών πτερυγώσεων μέσω της τεχνικής επιπέδου ανάμειξης ή ολίσθησης.

- Η αποκτηθείσα εμπειρία για τις συμπιεστές ροές μπορεί να επεκταθεί και στις ασυμπίεστες. Αυτό μπορεί να επιτευχθεί είτε με την τεχνική προσταθεροποίησης χαμηλών αριθμών Mach για τις εξισώσεις επίλυσης συμπιεστών ρευστών, είτε με την παραλλαγή του λογισμικού PUMA για ασυμπίεστες ροές (με βάση την τεχνική της ψευδο-συμπιεστότητας). Ο απώτερος στόχος είναι η καθολική χρήση της *TDDC* συζυγούς μεθόδου στο λογισμικό PUMA.
- Σχετικά με τη μοντελοποίηση της τύρβης, τα ΒΝΔ θα μπορούσαν να εκπαιδευτούν σε δεδομένα από λογισμικό υψηλής πιστότητας και να ενσωματώσουν έπειτα την πληροφορία αυτή στις εξισώσεις RANS. Έτσι, η ακρίβεια προσομοιώσεων με λογισμικό υψηλής πιστότητας θα μπορούσε να επιτευχθεί με την επίλυση των υπολογιστικά οικονομικότερων εξισώσεων RANS.
- Μια διαφορετική προσέγγιση σχετικά με τα ΒΝΔ είναι η χρήση τους σε προβλήματα αιτιοκρατικής βελτιστοποίησης μορφής. Τα ΒΝΔ, εκπαιδευμένα για την πρόβλεψη των συναρτήσεων-στόχων ή/και περιορισμών της βελτιστοποίησης, μπορούν να διαφοριστούν έτσι ώστε να υπολογιστούν οι παράγωγοι ευαισθησίας ως προς τις μεταβλητές σχεδιασμού. Κατά συνέπεια, τα ΒΝΔ θα μπορούσαν να χρησιμοποιηθούν σε προβλήματα αιτιοκρατικής βελτιστοποίησης αντικαθιστώντας το συζυγή επιλύτη. Η χρήση τους για υπολογισμό παραγώγων υψηλότερης τάξης μπορεί, επίσης, να εξεταστεί. Υπό εξέλιξη έρευνα οδηγεί σε ενθαρρυντικά αποτελέσματα, [21].
- Η αρχιτεκτονική $\lambda-$DNN θα μπορούσε να χρησιμοποιηθεί για άλλα πολυπεδιακά προβλήματα βελτιστοποίησης, όπως αεροδομικά ή αεροελαστικά. Τα συμπεράσματα από τη χρήση της σε άλλα πεδία θα συνεισέφεραν στη γενίκευση αυτών που εξήχθησαν στην παρούσα Διατριβή σχετικά με την αεροθερμική βελτιστοποίηση.

Βιβλιογραφία

[1] AIAA Transition Modeling Workshop-I. https://transitionmodeling.larc.nasa.gov/workshop_i/?doing_wp_cron=1689003551.6611080169677734375000. Accessed: July 2021.

[2] The EASY (Evolutionary Algorithms SYstem) software. http://velos0.ltt.mech.ntua.gr/EASY, 2008.

[3] W. Anderson, J. Newman, D. Whitfield, and E. Nielsen. Sensitivity analysis for Navier-Stokes equations on unstructured meshes using complex variables. *AIAA Journal*, 39(1):56–63, 2001.

[4] T. Arts, M. Lambert de Rouvroit, and A. Rutherford. Aero-thermal investigation of a highly loaded transonic linear turbine guide vane cascade. a test case for inviscid and viscous flow computations. Technical Note 174, von Karman Institute for Fluid Dynamics, Belgium, 1990.

[5] V. Asouti. *Aerodynamic analysis and design methods at high and low speed flows, on multiprocessor platforms (in Greek)*. PhD thesis, National Technical University of Athens, 2009.

[6] J. Coder and M. Maughmer. Computational fluid dynamics compatible transition modeling using an amplification factor transport equation. *AIAA Journal*, 52:2506–2512, 2014.

[7] V. D'Alessandro, S. Montelpare, R. Ricci, and A. Zoppi. Numerical modeling of the flow over wind turbine airfoils by means of Spalart-Allmaras local correlation based transition model. *Energy*, 130:402–419, 2017.

[8] R. Dechter. Learning while searching in constraint-satisfaction-problems. In *Proceedings of the 5th National Conference on Artificial Intelligence. Philadelphia*, pages 178–185, 1986.

[9] S. Dhawan and R. Narasimha. Some properties of boundary layer flow during

the transition from laminar to turbulent motion. *Journal of Fluid Mechanics*, 3:418–436, 1958.

[10] K. Duraisamy, Z. Zhang, and A. Singh. New approaches in turbulence and transition modeling using data-driven techniques. In *53rd AIAA Aerospace Sciences Meeting*, 2015.

[11] M. Duta, S. Shahpar, and M. Giles. Turbomachinery design optimization using automatic differentiated adjoint code. In *Turbo Expo: Power for Land, Sea, and Air*, volume 6, 2007.

[12] H. Elbanna and L. Carlson. Determination of aerodynamic sensitivity coefficients in the transonic and supersonic regimes. In *27th Aerospace Sciences Meeting*, page 532, 1989.

[13] J. Elliott and J. Peraire. Practical three-dimensional aerodynamic design and optimization using unstructured meshes. *AIAA Journal*, 35(9):1479–1485, 1997.

[14] M. Giles and N. Pierce. Adjoint equations in CFD - duality, boundary conditions and solution behaviour. In *13th Computational Fluid Dynamics Conference*, 1997.

[15] G. Halila, J. Martins, and K. Fidkowski. Adjoint-based aerodynamic shape optimization including transition to turbulence effects. *Aerospace Science and Technology*, 107:106243, 2020.

[16] A. Heft, Th. Indinger, and N. Adams. Experimental and numerical investigation of the DrivAer model. In *ASME 2012 Fluids Engineering Summer Meeting*, Rio Grande, Puerto Rico, 2012.

[17] L. Hylton, M. Mihelc, E. Turner, D. Nealy, and R. York. Analytical and experimental evaluation of the heat transfer distribution over the surfaces of turbine vanes. *NASA-CR-168015*, 1983.

[18] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3, 1988.

[19] I. Kampolis and K. Giannakoglou. Synergetic use of different evaluation, parameterization and search tools within a multilevel optimization platform. *Applied Soft Computing*, 11(1):645–651, 2011.

[20] D. Kapsoulis. *Low-Cost Metamodel-Assisted Evolutionary Algorithms with applications in shape optimization in fluid dynamics*. PhD thesis, National Technical University of Athens, 2019.

[21] K. Kovani, M. Kontou, V. Asouti, and K. Giannakoglou. DNN-driven gradient-based shape optimization in fluid mechanics. In L. Iliadis, I. Maglogiannis, S. Alonso, C. Jayne, and E. Pimenidis, editors, *Engineering Applications of Neural Networks*, pages 379–390, Cham, 2023. Springer Nature Switzerland.

[22] R. Langtry. Extending the Gamma-Rethetat correlation based transition model for crossflow effects (invited). In *45th AIAA Fluid Dynamics Conference*, 2015.

[23] R. Langtry and F. Menter. Transition modeling for general CFD applications in aeronautics. In *43rd AIAA Aerospace Sciences Meeting and Exhibit - Meeting Papers*, 2005.

[24] R. Langtry and F. Menter. Correlation-based transition modeling for unstructured parallelized computational fluid dynamics codes. *AIAA Journal*, 47:2894–2906, 2009.

[25] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin. Exploring strategies for training deep neural networks. *Journal of Machine Learning Research*, 10(1), 2009.

[26] B. Lee and C. Kim. Automated design methodology of turbulent internal flow using discrete adjoint formulation. *Aerospace Science and Technology*, 11(2):163–173, 2007.

[27] J. Lions. *Optimal Control of Systems Governed by Partial Differentiable Equations*. Springer-Verlang, New York, NY, USA, 1971.

[28] C. Lozano, E. Andrés, M. Martín, and P. Bitrián. Domain versus bound-

ary computation of flow sensitivities with the continuous adjoint method for aerodynamic shape optimization problems. *International Journal for Numerical Methods in Fluids*, 70(10):1305–1323, 2012.

[29] A. Lozano-Durán, M. J. P. Hack, and P. Moin. Modeling boundary-layer transition in direct and large-eddy simulations using parabolized stability equations. *Phys. Rev. Fluids*, 3:023901, 2018.

[30] M. Lynde, R. Campbell, M. Rivers, S. Viken, D. Chan, A. Watkins, and S. Goodliff. Preliminary results from an experimental assessment of a natural laminar flow design method. *AIAA Scitech 2019 Forum*, 2019.

[31] E. Malkiel and R. Mayle. Transition in a separation bubble. *Journal of Turbomachinery*, 118(4):752–759, 1996.

[32] F. Menter, M. Kuntz, and R. Langtry. Ten years of industrial experience with the SST turbulence model. *Turbulence, Heat and Mass Transfer*, 4, 2003.

[33] M. Monfaredi. *Continuous Adjoint-based aeroacoustic shape optimization by coupling the Ffowcs-Williams Hawkings analogy with URANS equations*. PhD thesis, National Technical University of Athens, 2021.

[34] M. Morkovin. On the many faces of transition. In S. Wells, editor, *Viscous Drag Reduction*, pages 1–31, Boston, MA, 1969. Springer US.

[35] E. Nielsen and W. Anderson. Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. *AIAA Journal*, 37(11):1411–1419, 1999.

[36] D. Papadimitriou and K. Giannakoglou. A continuous adjoint method with objective function derivatives based on boundary integrals for inviscid and viscous flows. *Computers & Fluids*, 36(2):325–341, 2007.

[37] E. Papoutsis-Kiachagias. *Adjoint methods for turbulent flows, applied to shape or topology optimization and robust design*. PhD thesis, National Technical University of Athens, 2013.

[38] E. Papoutsis-Kiachagias, V. Asouti, K. Giannakoglou, K. Gkagkas,

S. Shimokawa, and E. Itakura. Multi-point aerodynamic shape optimization of cars based on continuous adjoint. *Structural and Multidisciplinary Optimization*, 59(2):675–694, 2019.

[39] M. Piotrowski and D. Zingg. Smooth local correlation-based transition model for the Spalart-Allmaras turbulence model. *AIAA Journal*, 2020.

[40] O. Pironneau. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64(1):97–110, 1974.

[41] P. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2):357–372, 1981.

[42] H. Schlichting. *Boundary-Layer Theory*. Springer Berlin, Heidelberg, 1979.

[43] D. Somers. Design and Experimental Results for a Natural-Laminar-Flow Airfoil for General Aviation Applications. *NASA Technical Paper 1861*, 1981.

[44] P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. *La Recherche Aérospatiale*, 1:5–21, 1994.

[45] A. Taebi. Deep learning for computational hemodynamics: A brief review of recent advances. *Fluids*, 7(6):197, 2022.

[46] X. Trompoukis. *Numerical solution of aerodynamic-aeroelastic problems on Graphics Processing Units (in Greek)*. PhD thesis, National Technical University of Athens, 2012.

[47] K. Tsiakas. *Development of shape parameterization techniques, a flow solver and its adjoint, for optimization on GPUs. Turbomachinery and external aerodynamics applications*. PhD thesis, National Technical University of Athens, 2019.

[48] F. Charpin V. Schmitt. Pressure distributions on the ONERA M6 wing at transonic Mach numbers, experimental data base for computer program assessment. *Technical report*, AGARD 138, 1979.

[49] J. van Ingen. *A Suggested Semi-empirical Method for the Calculation of the Boundary Layer Transition Region*. ARC-19337. TH Delft, Delft, 1956.

[50] B. van Leer. Flux-vector splitting for the Euler equations. In *Eighth International Conference on Numerical Methods in Fluid Dynamics*, pages 507–512, 1982.

[51] Y. Wu and et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *ArXiv*, abs/1609.08144, 2016.

[52] T. Zervogiannis. *Optimization methods in aerodynamics and turbomachinery based on the adjoint technique, hybrid grids and the exact Hessian matrix (in Greek)*. PhD thesis, National Technical University of Athens, 2011.

[53] A. Zymaris, D. Papadimitriou, K. Giannakoglou, and C. Othmer. Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. *Computers & Fluids*, 38(8):1528–1538, 2009.