Department of Mechanical Engineering

University of Cape Town

COMPUTATIONAL FLUID DYNAMIC BASED OPTIMISATION OF AN INDUSTRIAL AXIAL FAN FOR RAPID PROTOTYPING

J.A. van Rooyen

Supervisor: Prof. A.G. Malan

Thesis submitted to the University of Cape Town in candidature for the degree of Doctor in Philosophy

June 2017

Declaration

I, J.A. Van Rooyen, hereby declare that the work on which this thesis is based is my original work (except where acknowledgements indicate otherwise) and that neither the whole work, nor any part of it, has been, is being, or is to be submitted for another degree in this, or any other university. I authorise the University of Cape Town to reproduce for the purpose of research, either the whole, or any portion of the contents in any manner whatsoever.

Signed .....(candidate)

Date .....

This PhD research thesis is dedicated to Karien and Ziara.

Summary

Axial air flow fans are widely used for air movement. In an increasingly international and competitive market, smaller fan companies find themselves in need of rapid preliminary design. This need is addressed in this study through the development of a first-revision, Computational Fluid Dynamics (CFD) based, optimisation tool which allows for rapid prototyping of a ducted axial fan. The result is an Elemental^{TM}-based multi-disciplinary software tool, comprising 2D CFD, mesh movement, and constrained geometric optimisation. The analytical equation employed to represent the aerofoil significantly reduces the cost of the optimisation. A pseudo-3D fan model is generated by superimposing 2D CFD results. This is done without the general assumption of the free-vortex method, which is not a necessity for fan design and other velocity distributions may be used. For this purpose, an enhanced finite volume discretisation method was developed. A penalty function minimisation, by means of an unconstrained optimisation algorithm, is implemented thereafter. The primary objective is to deliver a specific fan static pressure rise, while optimising for fan static efficiency by means of altering the rotor blade geometry. The spherical quadratic steepest descent method is employed, which does not rely on any explicit line searches, as required by traditional steepest descent techniques. The rapid prototyping tool is finally applied to an under-performing base fan (Fan-D) which cannot meet a specified duty point. The resulting optimised fan (Fan-Optim) is manufactured and experimentally tested, in accordance with the ISO 5801 standard. The pseudo-3D model is proven to predict fan performance accurately at the target duty point, while capturing fan behaviour over a range of volumetric flow rates. The former is to within 13% of the fan static pressure rise and within 2.3% of fan static efficiency. While Fan-Optim meets the desired duty point within 2%, it offers a considerable improvement in fan static efficiency over Fan-D. Furthermore, an approximate 38% reduction in blade material is achieved as a secondary effect.

Acknowledgements

I would like to acknowledge and thank my supervisor Prof. Arnaud Malan for his support and for giving me the opportunity to conduct this research work. Without his ingenuity it would not have been possible for me to be a full time student and a member of the InCFD research group. He has been an inspiration to me and I value his input to this work. I would also like to thank Eddie Raath (CFW Fans) and his team for their partial funding, valuable expertise and experimental work. I am also grateful for the assistance from Andrew Mowat and Nathan Zhou who helped me, as a "newbie", wherever they can to bring me up to speed with the coding environment in Elemental. Finally I would like to thank my wife, daughter, family and friends for their continual support and encouragement.

The resources provided by the University of Cape Town's ICTS High Performance Computing team (http://hpc.uct.ac.za) is acknowledged and appreciated. Funding obtained from the South African National Research Foundation, the South African Research Chair in Industrial CFD and the Department of Science and Technology is acknowledged and appreciated.

Contents

Nomenclature

A	Area (m^2)
$\mathbf{C}$	Bézier curve nodal coordinates
d	Distance (m)
F	Force (N)
F_r	Force on profile at radius r (N/m)
H	Total pressure (Pa)
ℓ	Length (m)
m	Quantity
N	Rotational speed (rpm)
$\mathbf{n}$	Face unit normal vector
$\mathbf{N}$	Node unit normal vector
$\dot{m}$	Mass flow rate (kg/s)
p	Static pressure (Pa)
p_{fsp}	Fan static pressure rise also known as total to static pressure (Pa)
P	Power (W)
$\mathbf{P}$	Nodal coordinates (m)
Q	Volumetric flow rate $\left(m^3/s\right)$
r	Radius (m)
t	Bézier parametric parameter varies between zero and one
T	Torque $(N.m.)$
u	Velocity (m/s)
v	Relative velocity (m/s)
$\mathbf{t}$	Edge unit tangent vector
x	x coordinate (m)
y	y coordinate (m)

Greek symbols

ϕ	Diameter (m)
Φ	Flow field variable
ω	Rotational speed (radians per second))
μ	Shear viscosity (kg/ms)
ν	Kinematic viscosity $\left(m^2/s\right)$
$\tilde{\nu}$	Turbulent viscosity variable: $\left(m^2/s\right)$
λ	Design variable
η	Total efficiency
η_s	Static efficiency

Subscripts

c	Casing of an axial fan.
F	Face.
h	Hub of an axial fan.
L	Left.
R	Right.
t	Tip of an axial fan.

x	x direction
y	y direction
$turb$	Turbulent

Mathematical operators

$\cdot$	Vector inner product
$\|variable\|$	Norm of variable (absolute value in case of a scalar)
∂	Partial derivative of variable
∇	Gradient operator of variable
Δ	Increment in variable
$\int_{r_t}^{r_h} ....dr$	Radial integral of variable from the hub to tip radius

Dimensionless Numbers

Re	Reynolds number: $\rho u L/\mu$

Notes on Notation

Both vector and index notations are used in this thesis. Vectors and nodal coordinates are denoted in bold. Where index notation is used, component subscripts may appear as super or subscripts and are typically denoted by x and y.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

An axial flow fan is classified as a turbo machine that generates airflow (axial direction) and a pressure rise between two points. Typical applications range from small diameter, computer cooling/ventilation fans, to large diameter, mine/commercial building ventilation fans ($\sim 1000mm$). The latter are large energy consumers due to high quantity of installations, electrical motor size, and long operating hours. When tendering, it is therefore critical for a supplier to quote on the most energy efficient fan for the required operating specification. A broad range of fan suppliers are typically invited to submit a tender. As compared to multi-national companies, local enterprises do not always have available a cost-effective, energy-efficient fan. This necessitates the development of a rapid prototyping tool for preliminary design support. A good example is that of CFW Fans (Pty) Ltd., a local fan design, development and manufacturing company. CFW solicited this research for the purpose of developing a CFD based preliminary design tool. The tool will tailor an existing fan such that the required fan operating specification is met in an energy efficient manner. This must be achieved by geometrically optimising the mathematically defined fan blade geometry by focussing on fan efficiency. The more detailed design following the preliminary design will only occur after the tender has been awarded and falls outside the scope of this work.

Historically, a fan was modelled using an analytical free-vortex method as employed by Wallis [1] and Lewis [2]. The 2D cascade at the mean radius is analysed to determine inlet and outlet velocity vectors required for a specific average pressure rise. It is further assumed that the total pressure rise and axial velocity remains constant along the blade span with no radial flow (radial equilibrium). By satisfying these assumptions the velocity vectors at the blade hub and tip can be calculated. Specifying the blade profile geometry to match the calculated required velocity vectors can be done by a direct or inverse method, as stated by Lewis [2]. In the direct method approach a series of cascades are analysed (experimentally or analytically) and the aerofoil that best meets the flow condition with reasonable efficiency is chosen. The inverse method allows the designer to specify the surface velocity or pressure profile distribution along the profile surface. Wallis [1], Lewis [2] and Bruneau and von Backström [3] successfully employed the direct method based on empirical and analytical 2D cascade analysis for specific aerofoils, whereas Kokturk [4] generated the required aerodynamic properties by utilising computational fluid dynamics (CFD) to analyse 2D cascades. Pascu [5] successfully designed a non-free-vortex axial

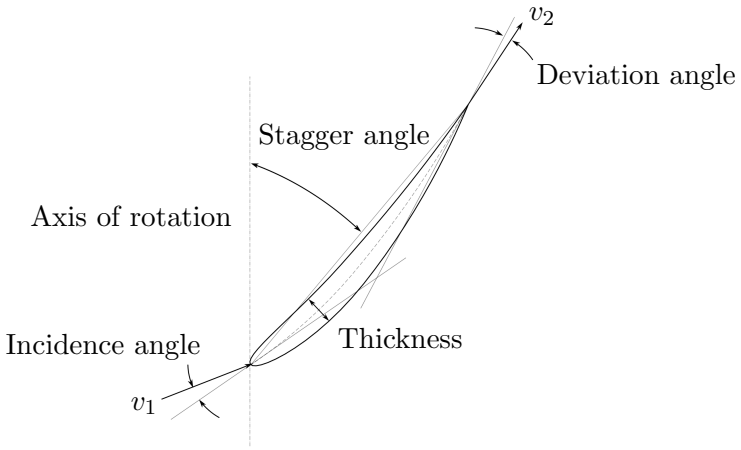

Figure 1.1: Example of an axial fan rotor blade 2D aerofoil[1]inlet (v_1) and outlet (v_2) relative velocity vectors

fan (thin and constant thickness rotor blade aerofoil) by employing an inverse analytical model. The latter assumes inviscid flow and a zero incidence angle (Figure 1.1). Analytical methods do, however, have limitations. This is due to their heavy reliance on empirical data for losses, inlet incidence angle and outlet deviation angle (Figure 1.1). The addition of 2D CFD improves the analytical method, but remains limited due to tailoring of data for specific flows only. A more comprehensive 3D CFD axial fan model has also been employed by numerous authors (Borges [6], Guedel et al. [7], Elhadi and Keqi [8], le Roux et al. [9], Jang et al. [10], and Augustyn et al. [11]). This is, however, at the expense of a dramatic increase in computational cost viz., 4.7×10^6 to 15×10^6 mesh elements used by le Roux et al. [9] and Guedel et al. [7], as compared to circa 5×10^4 for 2D. The relatively low Reynolds numbers of 300×10^3 to 400×10^3, as is the case in this work, is a further source of hindrance [9]. In the latter cited recent work, the use of circa 4.7×10^6 mesh elements still resulted in errors (modelled compared to experimental data) of up to 25% and 20% in predicting fan static pressure rise and fan power.

The majority of the above-mentioned authors who employed CFD utilised commercial software (ANSYS CFX, NUMECA, Fluent, and STAR-CCM+). The software uses finite volume method based solvers to solve the flow field on a discretised spatial domain (mesh). Structured meshes were employed by le Roux et al. [9] and Augustyn et al. [11] and unstructured meshes were used by Kokturk [4] and Jang et al. [10]. A combination of structured and unstructured meshes (hybrid) were employed by Pascu [5], Borges [6], Guedel et al. [7] and Elhadi and Keqi [8]. The authors solved the viscous Reynolds-averaged Navier-Stokes (RANS) set of equations on a mesh with an appropriate turbulence model instead of solving the instantaneous equations. The latter method is known as the Direct Numerical Simulation (DNS) method and is computationally very expensive, as noted by Augustyn et al. [11].

Various turbulence models have been employed by the cited authors which includes one-equation and two-equation models. The one-equation model of Spalart and Allmaras [12] was employed by [4, 8, 9, 11]. Kokturk [4] (2D cascade analysis) and Augustyn et al. [11] specifically selected this turbulence model due it being developed for aerospace applications that involves wall-bounded flows, while simultaneously being able to account for boundary layers subjected to ad-

[1]NACA6409 9% from www.http: //airfoiltools.com

verse pressure gradients. NUMECA also utilised this as their default turbulence model as it provides good convergence for typical turbomachinery flow cases and is more robust and computationally less expensive then the two-equation $k-\epsilon$ model (le Roux et al. [9]). Crivellini and D'Allesandro [13] successfully employed the standard Spalart-Allmaras one-equation turbulence model (without any trip term) for the simulation of laminar separation bubbles on aerofoils at Reynolds numbers $\geq 150\times10^3$. The turbulence model's capability was proven via the evaluation of several aerofoil characteristics at flow conditions similar to this work. This included the pressure coefficient as a function of aerofoil chord length, the prediction of flow re-attachment and the prediction of lift and drag coefficients at various angles of attacks. The two-equation model types that have been successfully employed for fan modelling by [4–7,10,11] are the $k-\epsilon$ model (Launder and Spalding [14]), the RNG (Renormalisation Group) $k-\epsilon$ model (Shih et al. [15]) and the $k-\omega$ Shear Stress Turbulence (SST) model (Menter [16]).

Various CFD based optimisation methodologies have been developed for aerodynamic shape and fan blade geometry optimisation. The work where 3D CFD has been employed to optimise fans includes Huang and Gau [17], Lee et al. [18], Egorov et al [19], Lin et al. [20] and Kim et al. [21–23]. Lee et al. [18], Kim et al. [21–23], and Lin et al. [20] employed CFD to generate an analytical representation of the objective functions with respect to the design variables. Lin et al. [20] used CFD to train a back-propagation artificial neural network (gradient steepest descent based) whereas the former two authors generated a second order polynomial response surface. A gradient-based optimisation scheme was employed by Lee et al. [18], whereas Kim et al. [21–23] employed a Non-dominated Sorting of Genetic Algorithm (NSGA-II, Deb et al. [24]) for multi-objective function optimisation. Lin et al. [20] employed a complex optimisation scheme (constraint simplex method). Huang and Gau [17] employed a Levenberg-Marquardt algorithm (LMA), also known as the Damper Least-Squares (DLS), to solve a set of non-linear least squares problems in an inverse design problem. The algorithm is effectively a steepest-descent method for the initial optimisation iterations and a Newton's method for the remaining iterations. Egorov et al. [19] employed an Indirect Optimisation on the basis of Self-Organization (IOSO) scheme for the purpose of optimising multi-objective functions with a large number of design variables.

A popular trend in aerodynamic 2D and 3D shape optimisation is the application of adjoint optimisation. It has been employed by numerous authors for 2D up to full 3D aerodynamic shape optimisation problems. Xiong et al. [25], Jameson [26] and Arens et al. [27] employed it for 2D aerofoil optimisation. Jameson et al. [28] and Straathof [29] employed it for 3D wing design. Wang [30], Neittaanmäki et al. [31] and Papadimitriou and Giannakoglou [32] employed it for turbomachinery optimisation (2D and 3D). Brezillon and Gauger [33] and Choi et al. [34] utilised it for 3D aircraft and helicopter rotor design respectively. The method is shown as particularly valuable where large number of design variables are present, as eloquently described by Jameson et al. [28]. The authors explains that the simplest way to conduct shape optimisation is to define the geometry by a number of design variables. An objective function is then selected, for example drag coefficient. In a gradient-based optimisation method the objective function gradients with respect to the design variables are required. This makes the process dependent on the number of design variables and can be computationally costly. As described by the authors the problem can be changed into an inverse problem. In this case, the search is carried out for the design variables

which will lead to the optimal geometry necessary to deliver a desired known objective function value instead. If the adjoint formulation is then applied, the adjoint governing equations (adjoint solver required), as well as an additional adjoint equation (computational cost is equivalent to solving one flow solution) is solved. It is then possible to calculate the objective function gradients without calculating the sensitivity to the flow variables. This implies no additional flow solutions being required to calculate the latter, which in turn reduces the computational cost dramatically. It is thus clear that the major advantage of employing an adjoint formulation is the elimination of the dependency on the number of design variables. The major drawback of this method, however, as is mentioned by the cited authors, is that unfortunately a physical realisable shape may not exist and that the problem must be carefully formulated.

1.2 Objectives

Having reviewed the cited different levels of modelling techniques it is clear that analytical methods are limited. Full 3D CFD modelling techniques are computationally still demanding for a preliminary design through geometric optimisation. The relatively low Reynolds numbers of 300×10^3 to 400×10^3, as is the case in this work, is a further source of hindrance. An improved preliminary optimisation based blade design tool therefore needs to strike a balance between accuracy and computational cost. This work thus proposes the use of 2D CFD to represent a 3D fan. Accordingly, the fan pressure and torque are computed via a collection of 2D CFD calculations at various radial (blade span) positions. The 2D cascades will be located so as to allow linear integration between them to enable performance calculation, instead of using only the mean radius cascade. The work will further determine the minimum number of 2D cascades required to most accurately predict fan performance. The latter will be rigorously assessed via experimental measurements performed according to international standards by CFW. Furthermore, the fan will be manufactured without any blade root clearance and with a blade tip clearance less than 1%. According to Wallis [1], no pressure or efficiency adjustments are required as long as this value is not exceeded. Therefore, the tip clearance is excluded in the fan modelling strategy.

The flow field in the 2D cascades will be determined by solving the 2D RANS equations combined with a Spalart-Allamaras turbulence model via the Elemental$^{\text{TM}}$ software. An implicit form of the selected turbulence model, successfully employed by numerous authors, will be implemented. The partial implicit solution procedure employed by Elemental$^{\text{TM}}$ will also be modified to full implicit. This will be achieved by casting the convective and viscous terms in the momentum equation into an implicit form. The governing equations will be discretised on anisotropic meshes with high element growth rates. A novel finite volume discretisation method, designed to improve accuracy on such meshes, named Enhanced Taylor Advection-Diffusion (ETAD) scheme (Merrick et al. [35]) will be implemented for this purpose. The method will be further developed into a novel ETAD method, namely, ETAD$_{\text{p}}$, which will include the discretisation of pressure related terms. Both ETAD and ETAD$_{\text{p}}$ will, for the first time, be applied to an industrial flow problem for the purpose of validation.

The cited CFD-based aerodynamic and geometric shape optimisation work shows that the trend is to focus on developing technology that caters for multi-objective functions, high numbers of design variables (up to hundreds) or inverse design optimisation methods. This, however, is not the focus of this work.

Rather, a direct optimisation approach will be undertaken with only one objective function and two design variables per 2D aerofoil. The 3D fan rotor blade aerofoil geometry will be represented by an analytical equation that is a function of five variables of which two parameters will be varied in this work. A unique algorithm was developed as part of a separate proprietary project, to cater for the mathematical representation of the blade. During the separate project, the aerofoil selection, as well as the selection of the two most significant design variables, were concluded. It should be noted that the use of this mathematical representation is relevant to this work only as far as the definition of the two selected variables is concerned. After reviewing the vast range of sophisticated optimisation techniques employed by the various authors cited, it was decided to take a pragmatic route by choosing gradient-based optimisation. Such a method was successfully employed by Lee et al. [18]. Further, Snyman [36] is of the opinion that non-gradient-based methods, such as genetic algorithms, simulated annealing, particle swarm optimisation and other evolutionary methods, are in many cases computationally too expensive to be viable. The gradient-based method selected is the unconstrained spherical quadratic steepest descent method (SQSD) developed by Snyman and Hay [37]. It is fairly simple, computational efficient and stable according to the cited authors. It also circumvents the standard line search requirement. The SQSD will be applied to a penalty function formulation of a constrained problem which will comprise of the objective function and constraints (equality and in-equality).

In this work a novel method is proposed to optimise a 3D fan's rotor blade via the use of two 2D cascades, one close to the hub (inner cascade) and one close to the tip (outer cascade). Each cascade will be individually utilised to geometrically optimise the blade aerofoil at its representative radius. The inner cascade is typically not capable of delivering the required static pressure rise, and therefore this profile will be optimised for peak efficiency without specifying the required static pressure rise. In contrast the outer profile will be optimised to a peak efficiency at a specified static pressure rise. In this procedure it is assumed that the radial flow component is small. The effect of this assumption will also be investigated in greater detail.

Rigorous experimental validation will be done to determine the accuracy of the fan modelling technique as well as to assess the optimised design. Fan static pressure rise, shaft power and fan static efficiency values will be compared to the experimental data for this purpose. The predicted fan performance trend over a range of volumetric flow rates will also be evaluated. Finally the improvement in fan performance will be assessed.

1.3 Publications

Forthcoming publications from this work:

- Van Rooyen, J.A. and Malan, A.G; Industrial Fan Modelling Technology. In proceedings: Fourth African Conference on Computational Mechanics (AfriCOMP), page 88-89, Marrakech, Morocco (2015).

- Van Rooyen, J.A., Malan, A.G and Raath, E.; High Performance Industrial Fan Optimisation. In proceedings: VII European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS), paper ID 6442, Crete, Greece (2016).

- Merrick, D., Malan, AG and van Rooyen, J.A.; A Novel Finite Volume Discretisation Method for Advection-Diffusion Systems on Stretched Meshes. Submitted for review: Journal of Computational Physics (2016).
- Van Rooyen, J.A., Malan A.G. and Raath E.; CFD Based Optimisation of an Industrial Axial Fan. Draft version: International Journal of Numerical Methods for Heat and Fluid Flow (2017).

1.4 Outline of Thesis

The thesis contains 9 chapters and a description of each follows:

- *Chapter 1. Introduction:* This chapter contains an overview regarding the background and objectives of this work.
- *Chapter 2. Axial Flow Fans:* Axial fan general categories, components, duty point as well as general terminology are presented in this chapter. The axial fan unit used in this work is discussed as well as the experimental test facility and procedure.
- *Chapter 3. Modelling Strategy:* In this chapter the pseudo-3D fan model is discussed as well as a boundary condition and convergence algorithm that was implemented for this work.
- *Chapter 4. Governing Equations:* The Reynolds-averaged Navier-Stokes (RANS) and Spalart-Allmaras turbulence model are presented.
- *Chapter 5. Flow Discretisation and Flow Solution:* In this chapter the numerical solution of the governing equations are discussed. This includes spatial discretisation and the solution procedure employed in Elemental$^{\text{TM}}$. Furthermore the full implicit implementation and further development of a higher order discretisation scheme is presented.
- *Chapter 6. Constrained Optimisation:* The implemented optimisation methodology with an automated efficient mesh movement scheme is presented.
- *Chapter 7. Validation:* In this chapter the various modelling technologies developed and implemented are validated. This is done through applications to relevant test cases. Finally, the pseudo-3D fan modelling technique employed in this work is experimentally validated.
- *Chapter 8. Application:* A fan duty point and base fan is prescribed by CFW. The predicted performance of this fan is evaluated before it is optimised by the CFD based optimisation methodology. Finally, the predicted performance of the optimised fan is compared to the requirement, base fan and experimental results.
- *Chapter 9. Conclusions and Potential Future Work:* The main outcomes of this work are summarised and recommendations are made for further improvements in future research.

Chapter 2

Axial Flow Fans

An axial flow fan is a turbomachine that generates a pressure increase between an inlet and outlet while the airflow is in the direction of the rotor axis. According to Wallis [1] there are three main axial flow fan categories:

- *Free fan:* Rotor rotates open in a space with no inlet or outlet ducting i.e. typical desk, room or ceiling fan.
- *Diaphragm-mounted fan:* Fan that moves air from one large space to another.
- *Ducted fan:* Rotor is mounted inside a duct with the air entering and leaving the blades in an axial direction.

The objective of this work is to develop a rapid prototyping tool for a ducted axial flow fan, as required by a Cape Town based fan design, development and manufacturing company, CFW Fans. This fan system is described in greater detail below.

2.1 Ducted Axial Fan Systems

A ducted axial flow fan may consist of several elements as shown in Figure (2.1). The inlet ducting guides the airflow from an open area to the stationary inlet guide vanes, where the latter gives the air a pre-swirl before entering the rotating blades of the fan. The blades are generally made up of a series of aerofoils, a typical section of which is illustrated in Figure 2.2. The air exits the rotating blade zone towards the stationary outlet guide vanes (main purpose to convert the rotational velocity to static pressure, termed dynamic pressure recovery) and tail fairing (assists in dynamic pressure recovery by preventing a sudden cross sectional area change), after which it exits the outlet duct section.

According to Wallis [1] there are five main design possibilities for axial ducted fan units:

- *Rotor unit:* Consists only of a rotor with no pre-swirl and the momentum in the rotating flow at the outlet is lost.
- *Rotor-straightener unit:* Consists of a rotor and outlet guide vanes.
- *Pre-rotator-rotor unit:* Consists of an inlet guide vane and rotor where the former gives the air entering a pre-swirl in the opposite direction to the rotor motion and the rotor then removes the pre-swirl.

- *Pre-rotator-rotor-straightener unit:* Consists of an inlet guide vane, rotor and outlet guide vane (combination of two previous configurations).
- *Contra-rotating rotors unit:* Consists of two rotors, where the second rotor removes the swirl created by the first rotor.

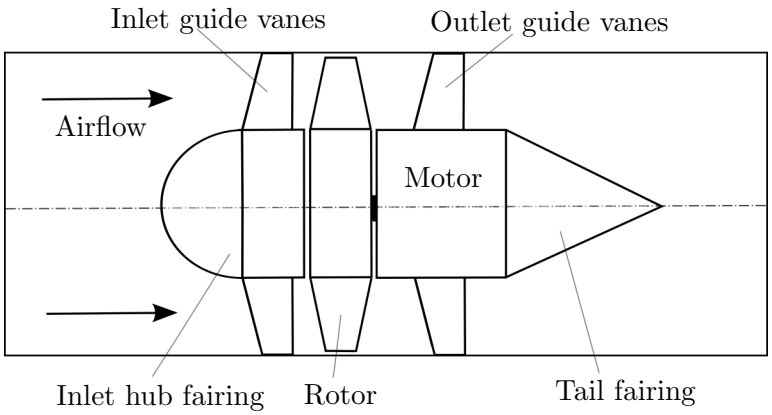

Figure 2.1: Ducted axial fan

Figure 2.2: Typical aerofoil section

This research work is focused on an exhaust fan which has a rotor-straightener unit configuration. For the purpose of this work the model has been simplified to minimise the outlet pressure losses, as shown in Figure 2.3 (excludes straightener). The tail fairing is completely excluded and the air-cooled electrical motor is covered with a smooth cylinder (same diameter as hub) which continues up to the end of the outlet duct. This eliminates both the sudden expansion loss behind the motor and the losses due to the fins of the air-cooled motor.

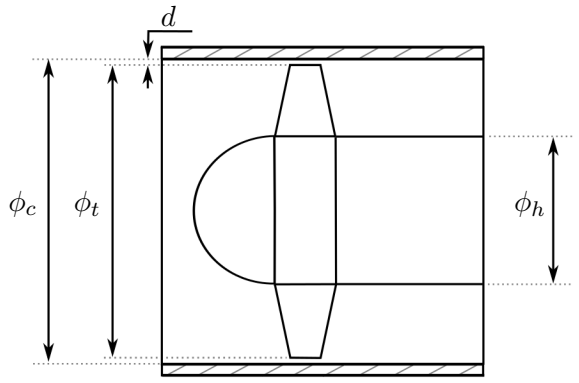

Figure 2.3: Schematic of an axial flow fan

2.2 Fan Duty

When ducted axial flow fans are used in industry they need to operate at a specific volumetric air through a duct system. The latter incurs flow-induced

pressure losses. This specified operating point is referred to as the duty point. The losses in the system are expressed in terms of total pressure loss while the usable total pressure is generally the difference between the fan inlet and outlet total pressure (gauge):

$$\Delta H_{useful} = H_{outlet} - H_{inlet} \tag{2.1}$$

When an exhaust fan (as is the case in this work) is used to move air from a chamber/room to the atmosphere, the useful total pressure is:

$$\Delta H_{useful} = p_{outlet} - H_{inlet} \tag{2.2}$$

where p_{outlet} denotes atmospheric static pressure. Fan total pressure is computed by Equation (2.1), when the useful work is downstream; otherwise, Equation (2.2) is used if the useful work is upstream (exhaust fans). For exhaust fans, where the air is moved from a chamber/room (at a total pressure relative to atmosphere) to atmosphere at the outlet, $p_{outlet} = H_{outlet} = 0.0$. This occurs due the dynamic pressure being dissipated into the atmosphere and the outlet static pressure is then equal to the atmospheric total pressure. Fan manufacturers, however, specify fans according to fan static pressure (p_{fsp}) and total pressure rise (ΔH):

$$\Delta p_{fsp} = p_{outlet} - H_{inlet} \tag{2.3}$$

$$\Delta H = \Delta p_{fsp} + \frac{1}{2}\rho {u_{axial}}^2 \tag{2.4}$$

where the axial velocity is defined as:

$$u_{axial} = \frac{Q}{A_{outlet}} \tag{2.5}$$

where Q and A_{outlet} are respectively the volumetric flow rate and cross sectional area (based on duct diameter) at the fan outlet. However, in this work the axial velocity, used in the modelling strategy, is calculated by using the annulus cross sectional area.

The term p_{fsp} is misleading because it suggests that it is a static pressure rise, whereas it is in fact a total-to-static pressure rise. Wallis [1] specifically eliminates the use of this term because of the serious interpretation difficulties (remembering that for exhaust fans the velocity downstream the outlet is zero) associated with its use. In this work the fan industry terminology will be used because p_{fsp} makes sense for exhaust fans where the inlet total pressure (measured upstream as per Figure 2.5) consists only of a static pressure, with no dynamic pressure present. This assumption can be made due the inlet dynamic pressure being less then $2Pa$ in the experimental set-up, as specified by the relevant ISO 5801 standard. It must, however, be noted that the fan total pressure ΔH does not take into account any rotational velocity.

To evaluate how effectively shaft power is converted into the air flow, fan static and total efficiencies are calculated as follows:

$$\eta_s = \frac{Q\Delta p_{fsp}}{P} \tag{2.6}$$

$$\eta = \frac{Q\Delta H}{P} \tag{2.7}$$

where P denotes the shaft power. The torque and rotational speed is used to calculate the power (P):

$$P = T\omega \tag{2.8}$$

$$\omega = \frac{2\pi N}{60} \tag{2.9}$$

where ω and N respectively denote rotational speed in terms of radians per second and revolutions per minute.

2.3 Experimental Facility

CFW have world class experimental facilities, as shown in Figure 2.4. These are used by the company for fan research and development work, as well as to generate fan performance curves for their fan catalogues. There are three sets of graphs that are generated, namely, the pressure rise (Δp_{fsp} and ΔH), power, and efficiency (η_s and η) against volumetric air flow rate. CFW conducts the experimental testing in accordance with ISO 5801. The experimental test configuration is of type A (see ISO 5801) i.e. installation with a free inlet and free outlet, that is typically used to test exhaust fans, as shown in Figure 2.5.

Figure 2.4: CFW experimental test facility

2.3.1 Experimental Configuration

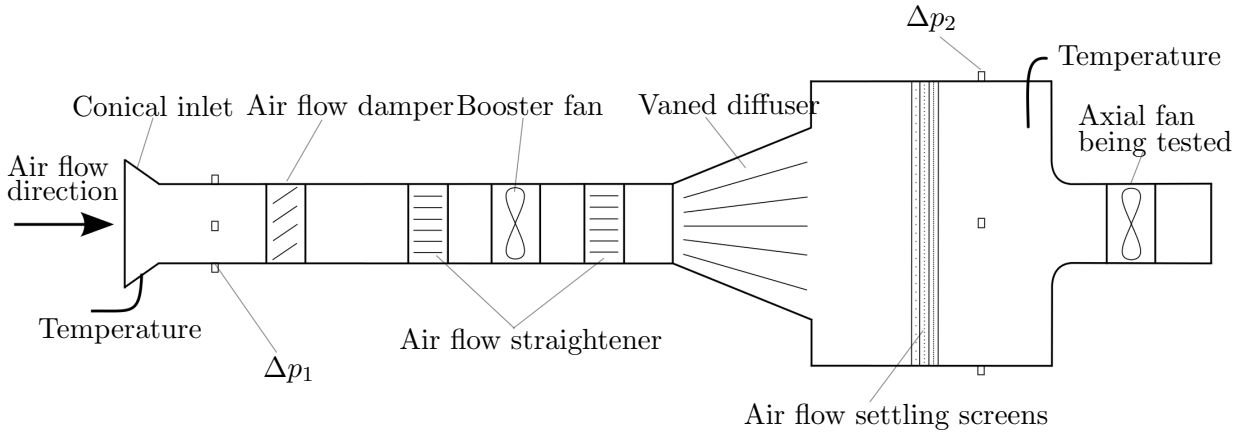

Figure 2.5: Schematic of experimental set-up

The experimental facility is composed of the following main components:

- *Static pressure and temperature points:* The static pressure and temperature (dry bulb and wet bulb) values are used to calculated air properties at specific points in the system, and static pressure differences over components.

- *Conical inlet:* This is a pressure differential device (Figure 2.4) that is used to determine the air flow rate by using the Δp_1 and air properties as per ISO 5801. Here Δp_1 denotes the static pressure difference between the air flow and the atmosphere.

- *Air flow damper:* Used to manually set the air flow rate in the duct system.

- *Air flow straighteners:* Placed at the inlet and outlet of the booster fan. The former straightens air flow due to disturbance caused by the damper, and the latter straightens air exiting the booster fan.

- *Booster fan:* This fan is controlled by a variable speed drive and is used to overcome the resistance in the system up to the settling chamber.

- *Vaned diffuser:* Diffuses the air from the booster fan into a large settling chamber within a relatively short distance.

- *Settling chamber and screens:* The settling chamber creates a disturbance-free and uniform air flow. Three air flow settling screens ensure this by being placed in the order of coarse to fine ($60\%, 50\%$ and 40% free area).

- *Axial fan:* The fan under investigation is connected to the settling chamber and withdraws air from an undisturbed condition and exhausts into the atmosphere. The motor used to drive the axial fan is a calibrated electrical motor, which makes it possible to determine the shaft power by calculating the electrical load.

To generate the performance curves the Q, Δp_{fsp}, P and N, readings are required. The experimental procedure followed by CFW will next be described.

2.3.2 Experimental Procedure

The first objective of the experimental procedure is to achieve a free air condition (axial fan has no pressure rise, $\Delta p_2 = 0.0$) in the settling chamber, with both the booster fan and axial fan running. This is achieved by throttling the air flow damper and altering the speed of the booster fan. Once achieved, the first data point of the performance curves can be generated. The volume flow rate is calculated by means of the conical inlet measurements and $\Delta p_{fsp} = \Delta p_2$. The axial fan shaft power is read off the electrical motor calibration graph by using the electrical load (kW) of the motor. A calibrated electrical motor was specifically sourced for this purpose, and its calibration also verified by CFW. The electrical load is calculated by using the measured voltage, current and power factor. The subsequent performance curve data point is determined by altering the volumetric flow rate by throttling the air flow damper, and repeating the procedure above. This is done until the flow has been throttled to practically zero.

Chapter 3

Modelling Strategy

Traditionally, axial flow fans were designed by means of a 2D analytical method, based on an inlet and outlet velocity vector diagram at the mean radius, as described by Wallis [1]. The velocity vector diagrams are determined by utilising the mean radius, volumetric air flow rate, fan system geometry, and rotor rotational speed. In addition, the outlet vector diagram is dependent upon the pressure rise. By employing the conservation of momentum theory, Bernoulli relationship, and a free-vortex assumption, the fan performance can be calculated. Free-vortex flow assumes that the total pressure rise remains constant in the radial direction over the blade span and that the rotational velocity is inversely proportional to the radius. This assumption is utilised to calculate the velocity vectors at several radiuses, other than the mean radius.

This method is, however, highly dependent on empirical data and analytical estimations. These are then typically specific to an aerofoil geometry and does not lend itself well to generic automated modelling. The above limitations can be overcome by utilising 2D CFD (incompressible, RANS solver with an applicable turbulence model) to resolve the flow field at various radial sections, as was done by Kokturk [4] for a reversible axial fan. The 2D aerofoil lift and drag coefficients, as well as the velocity vectors relative to the aerofoil, are therefore no longer dependent on empirical data any more. The modelling process is still an iterative one (perturb aerofoil) to ensure that the vector diagrams in the analytical calculations are the same as the vector diagrams resulting from the CFD analysis. An alternative to a 2D CFD based model is a more comprehensive 3D CFD model, as employed by numerous authors. Here the majority of the losses in the system are captured at the expense of a dramatic increase in computational cost due to the size of the model (5×10^6 to 15×10^6 mesh elements used by le Roux et al. [9] and Guedel et al. [7] respectively). General 3D CFD axial fan modelling was also successfully employed by Borges [6], Guedel et al. [7], Elhadi and Keqi [8], le Roux et al. [9], and Augustyn et al. [11].

3.1 Fan Model

This work constitutes the first effort to develop a rapid preliminary design tool. As such, not all aspects were optimally efficient. The tool should be geometrically general (allowing application to any aerofoil or application) while circumventing the computational cost of full 3D models. Accordingly, a pseudo-3D model (based on 2D calculations) will be developed and it will be made up of several sectional cuts as shown in Figure (3.1). By unrolling one such cut, a 2D plane with a linear series of aerofoils (also known as a cascade), is generated as shown

in Figure (3.2).

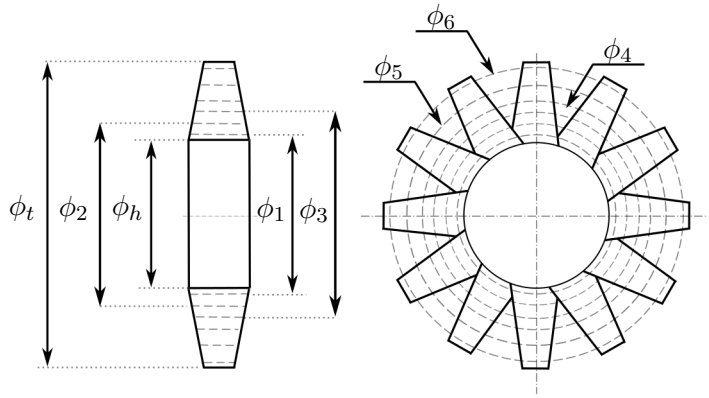

Figure 3.1: Schematic of fan rotor to be modelled

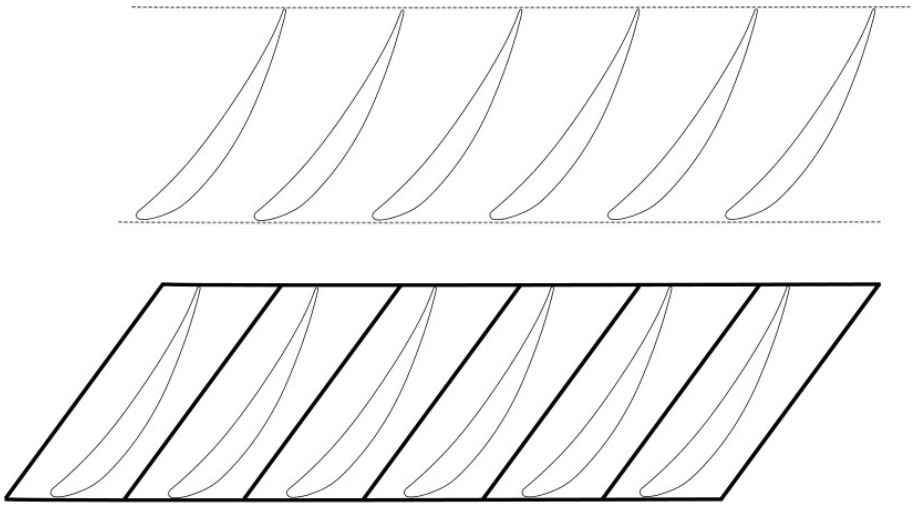

Figure 3.2: 2D cascades

An appropriate 2D CFD model will be generated to resolve the flow field over this cascade, from which pressure rise over the aerofoils as well as the aerofoil forces can be calculated. These 2D CFD results are then utilised to calculate the pseudo-3D fan's static pressure rise as follows:

$$\Delta p_{fsp} = \frac{2\pi \int_{r_h}^{r_t} \rho \Delta p_{fsp_r} u_{axial} r dr}{\dot{m}} + \Delta p_{loss} \tag{3.1}$$

where ρ, p_{fsp_r}, u_{axial}, Δp_{loss} and $\dot{m}$ are respectively the fluid density, fan static pressure rise over 2D aerofoil at radius r, axial velocity, pressure loss and mass flow rate. This equation can be simplified with a constant axial velocity assumption along the blade span:

$$\Delta p_{fsp} = \frac{2 \int_{r_h}^{r_t} \Delta p_{fsp_r} r dr}{r_t^2 - r_h^2} + \Delta p_{loss} \tag{3.2}$$

where r_t and r_h are respectively the fan rotor tip and hub radiuses. It should be noted that only a constant axial velocity is assumed. The free-vortex assumption of a constant total pressure rise over the blade span, with a specified velocity vector distribution, is not made in this work. Torque of the pseudo-3D fan is calculated as follows:

$$T = m_{blades} \times \int_{r_h}^{r_t} F_{x_r} dr \tag{3.3}$$

where m_{blades} and F_{x_r} are respectively the number of fan rotor blades and the tangential force on the 2D aerofoil at radius r. The variables p_{fsp_r} and F_{x_r} are represented by a linear equation between each pair of sectional cuts (ϕ_1 to ϕ_2, ϕ_2 to ϕ_3, ϕ_3 to ϕ_4, ϕ_4 to ϕ_5 and ϕ_5 to ϕ_6). The values at ϕ_h and ϕ_t are calculated by linear extrapolation from ϕ_1 to ϕ_2 and ϕ_5 to ϕ_6 respectively. The duct pressure loss estimation, as described by Stoecker and Jones [38], consists of an inlet and outlet component, and is calculated as follows:

$$\Delta p_{loss} = \rho \left[\left(\frac{f L u_{axial}^2}{2D} \right)_{inlet} + \left(\frac{f L u_{axial}^2}{2D} \right)_{outlet} \right] \tag{3.4}$$

where f, L, and D are respectively the friction factor, length and hydraulic diameter. The friction factor is calculated with the Colebrook formulation described by Stoecker and Jones [38]:

$$f = \left(\frac{1}{1.14 + 2\log\left(D/\epsilon\right) - 2\log\left[1 + \left(9.3D\right) / \left(\epsilon Re\sqrt{f}\right)\right]} \right) \tag{3.5}$$

where ϵ is the absolute roughness coefficient (selected sheet metal value, $\epsilon = 0.00015$). Finally the pseudo-3D fan's static efficiency is calculated utilising Equation (2.6).

Considering the 2D CFD simulation, it would be costly to model the entire series of 2D aerofoils (Figure 3.2). Instead, the domain is reduced to a single aerofoil, while accounting for aerofoil-aerofoil interaction via the employment of a linear periodic boundary condition (Figure 3.3). The width of the domain originates from the linear repeat in 2D cascade shown in Figure 3.2. The length upstream and downstream of the the aerofoil was selected such that the inlet and outlet boundary conditions did not affect the flow field around the aerofoil, similar to Kokturk [4]. The angle of the periodic boundaries is selected to be parallel with the aerofoil chord. It simplifies mesh generation, while allowing space for the aerofoil geometry to be altered during optimisation. It was found that if the positioning of the aerofoil inside the flow domain is not done carefully it could result in mesh movement limitations, which would require mesh re-generation during optimisation. The aerofoil geometry is represented by an analytical equation which the author developed as part of a separate project. This equation, which cannot be disclosed due to confidentiality, allows for altering the aerofoil geometry (thickness, droop, camber etc.) via approximately four variables. The geometric optimisation (detailed in Chapter (6)) will operate on these variables.

ICEM was used to generate an anisotropic hybrid FIDAP Neutral mesh that contains structured quadrilateral elements in the boundary layer and unstructured triangular elements growing in size as one moves away from the boundary (Figures 3.4 and 3.5). The size of the various 2D meshes over the blade span varies between 44×10^3 and 65×10^3 elements. Further mesh refinement, to double the number of mesh elements, had a negligible effect on the applicable values of interest in this work. The CFD software employed to solve the steady state, incompressible, and viscous air flow is an in-house code developed at the University of Cape Town's Mechanical Engineering department viz. Elemental$^{\text{TM}}$. It is an efficient, edge-based, vertex centred finite volume code that is second order

accurate and applicable to arbitrary elements. A Spalart-Allmaras turbulence model was also employed due to the transitional flow regime present in this study, with a Reynolds number varying between 300×10^3 to 400×10^3 over the blade span. The reader is also reminded that this turbulence model was successfully employed by Crivellini and D'Allesandro [13] to predict key aerofoil characteristics at various angles of attack at similar flow conditions as encountered in this work.

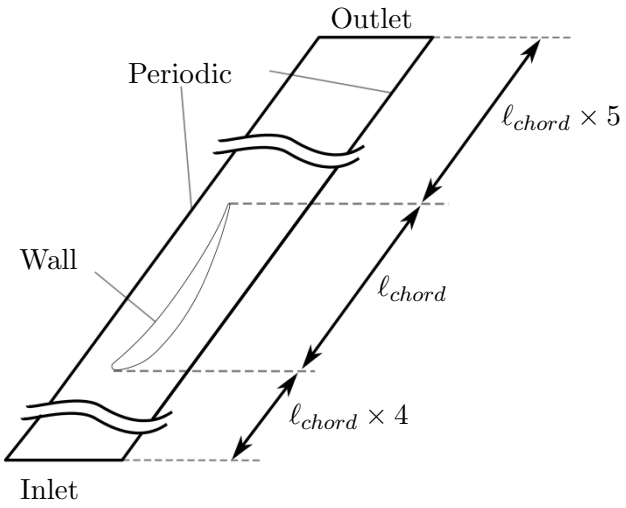

Figure 3.3: 2D flow domain

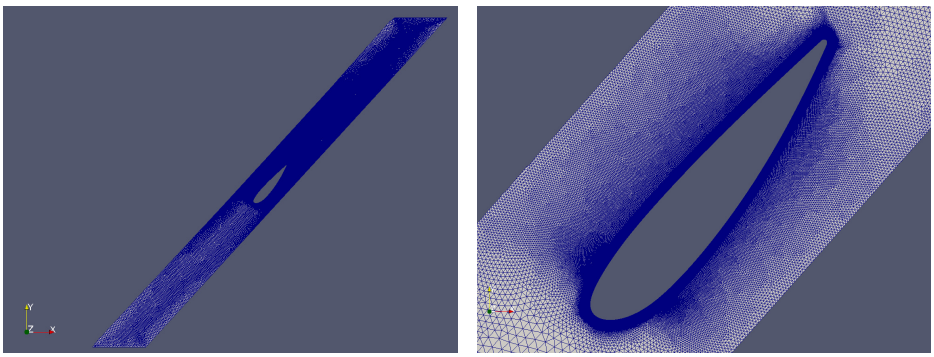

Figure 3.4: Mesh of 2D flow domain

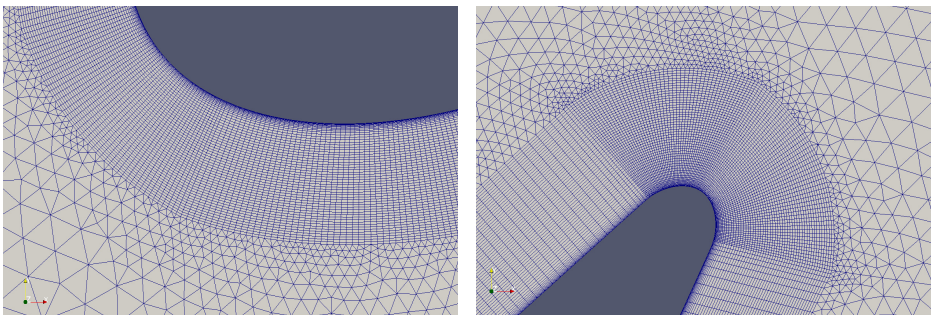

Figure 3.5: Mesh of 2D flow domain: zoom of aerofoil leading (left) and trailing (right) edges

3.2 Linear Periodic Boundary Condition

Linear periodic boundaries are two parallel boundaries that are the same length and have a matching mesh node spacing. This is enforced in the preprocessor during the flow domain geometry and mesh generation process. This boundary condition ensures invariance with respect to the flow field variables at two periodic nodes, $\mathbf{P}_0$ and $\mathbf{P}_1$ (co-linear in the direction of the linear repetition, Figure 3.6). This implies that flow field information has to be shared between these two periodic nodes. This information is not available from the mesh generator output file (FIDAP Neutral), and therefore must be determined.

Prior to describing the methodology chosen, a brief digression is required for the purpose of context. This Ph.D. study started out as mesh generation specific. Due however to the funder pulling out after circa 12 months, the work had to be altered to this project. During the mesh generation work a quadtree/octree search engine was developed. It was consequently available for use here. End of digression.

Before flow field information can be shared over a periodic boundary, one first has to determine the nodes to which this applies. This is achieved in the following manner:

- Generate a list of coordinates of boundary nodes on the left periodic boundary.
- Calculate the periodicity vector. This vector will point in the direction of the linear repetition and the size will be the width of the flow domain.
- Project right periodic boundary nodes by using the periodicity vector. This will create duplication of nodes (a list of duplicated nodes representing the left periodic boundary) which are added to the nodal list.
- Use a fast search engine to identify and link nodes.

For explanation purposes, the 2D version (quadtree) of the search engine is discussed in more detail. The fast search engine is a quadtree data structure where a square bounding box is used to enclose a specific set of nodes. The bounding box is then continuously split into four equal boxes. This is done until there is one unique node in a bounding box. A node is identified as unique if its distance to a neighbouring node is less than a specified tolerance. In other words, if there are two nodes with the same coordinates in a quadtree the sub dividing of bounding boxes will stop once the box contains only these two nodes (distance between them will be less then specified tolerance value). An example of the bounding boxes and tree structure is shown in Figures (3.7) and (3.8) for a set of nodes $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{P}_3$. Here the first two nodes are within the specified tolerance. There are a total of 9 bounding boxes created where nodes $\mathbf{P}_1$ and $\mathbf{P}_2$ end in bounding box number 7 (tree level 1); node $\mathbf{P}_3$ ends up in bounding box 2 (tree level 0).

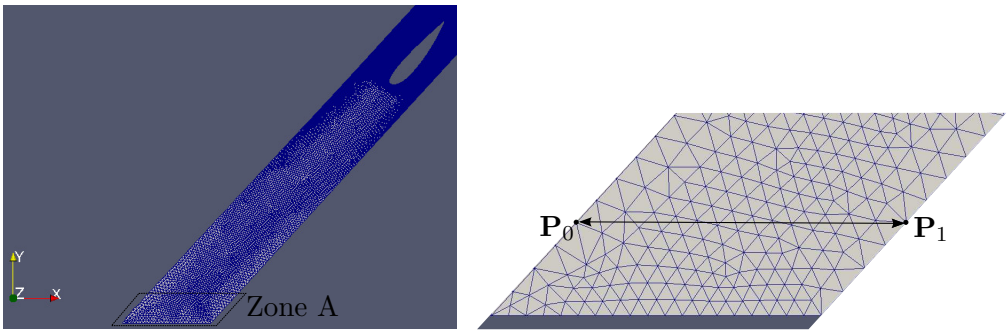

Figure 3.6: Bottom section of flow domain (left) and exploded view of Zone A (right)

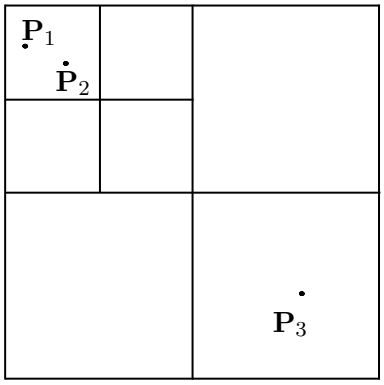

Figure 3.7: Quadtree bounding boxes example

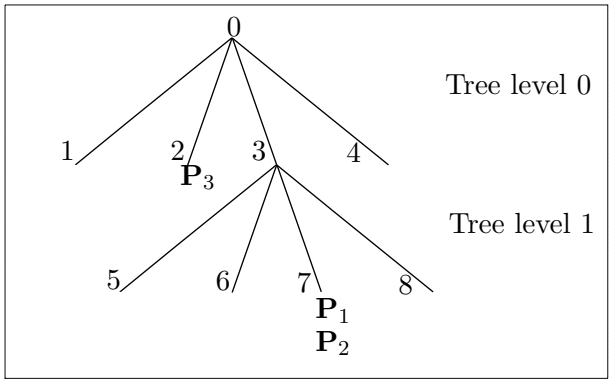

Figure 3.8: Quadtree tree structure example

3.3 Convergence

Determining when a 2D CFD solution is converged is important from a modelling point of view, but even more so for automated optimisation i.e. the solution needs to automatically sense convergence before moving on to the next optimisation functional evaluation. This is done in a CPU efficient manner. The naive approach would be to decide on a specific solution residual tolerance (imbalance in the discretised equations). The exact relation of this on the actual required simulation outputs (for optimisation purposes) is, however, not known a priori, and may result in excessive optimisation times. The specific values of importance in this work are tangential force on the aerofoil and static pressure rise

over the fan. It is, however, more prudent to specifically monitor the variation of these and terminate the solution once this is reached. It has therefore been decided to rather monitor one variable, namely F_{x_r} (Equation (3.3)). This tangential force value is dependent on a converged pressure field around the aerofoil which should also imply that the domain inlet and outlet pressures have been converged. This will be validated in Chapter (7). In this work, the tangential force value is defined as converged when it is within a certain % of an averaged value. To calculate this, the values at every 2000 solver iterations are stored until a history of 5 data points (continuously updated every 2000 iterations) is achieved. The latter is utilised to calculate the average value. Convergence is achieved once the minimum and maximum value of data points stored are within 0.25% from the average value. The % difference from the average value is calculated relative to the average value.

The monitor function developed, stops the solver once convergence of a specified variable is achieved. The following modular algorithm details the steps followed:

- *Variables:*
  - Float moniValue: variable passed to monitoring function on which convergences should be checked.
  - Float averMoni(0.0): average monitor value initialised to 0.0.
  - Bool isconv(false): has monitored value reached convergence level.
  - int mdataPoints(5): number of data points utilised in convergence calculation.
  - int icount(-1): number of times the convergences functions have been employed.
  - **D**: vector of data points being utilised in the convergence calculation.
  - Float $R\,(0.5)$: % band around averMoni.
- *If*$(icount < mdataPoints)$

  $icount+ = 1;$
  $D[icount] = moniValue;$

  else if$(icount \geq mdataPoints)$

  If$(icount > (mdataPoints - 1))$
  $D[ii] = D[ii + 1];$ where $ii = 0, 1, 2..\,(mdataPoints - 1)$
  $D[mdataPoints - 1] = moniValue;$
  $averMoni = D[ii];$ *where* $ii = 0, 1, 2..mdataPoints$
  $averMoni/ = mdataPoints;$
  Float $minValue\left(1 \times 10^{200}\right);$
  Float $maxValue\left(1 \times 10^{-200}\right);$
  minValue = Minimum $(minValue, D[ii]);$ where $ii = 0, 1, 2..mdataPoints$
  maxValue = Maximum $(maxValue, D[ii]);$ where $ii = 0, 1, 2..mdataPoints$
  If $\left(\left(|maxValue - averMoni| < \left|\frac{averMoni \times R}{200}\right|\right)\right.$ and
  $\left.\left(|minValue - averMoni| < \left|\frac{averMoni \times R}{200}\right|\right)\right)$
  $isconv = true;$
  $icount+ = 1;$

Chapter 4

Governing Equations

In this work a 2D, steady state, incompressible, viscous Reynolds-averaged Navier-Stokes (RANS) set of equations are solved. The mass and momentum conservation solved are as follows:

$$\nabla \cdot (\mathbf{u}) = 0 \tag{4.1}$$

where $\mathbf{u}$ is a velocity vector.

$$\begin{array}{rcl} \rho\nabla \cdot (u_x\mathbf{u}) + \frac{\partial p}{\partial x} - \nabla \cdot [(\mu + \mu_{turb})\nabla \cdot u_x] & = & 0 \\ \rho\nabla \cdot (u_y\mathbf{u}) + \frac{\partial p}{\partial y} - \nabla \cdot [(\mu + \mu_{turb})\nabla \cdot u_y] & = & 0 \end{array} \tag{4.2}$$

where u_x, u_y, p, μ (constant) and μ_{turb} (varying) are fluid speed in the x and y directions, pressure, shear viscosity and turbulent viscosity respectively. The latter variable is approximated by applying an appropriate model, as detailed in the following section.

4.1 Turbulence Modelling

At a constant volumetric flow rate the fan rotor blade experiences a circa constant axial inlet velocity over the blade span. In the frozen rotor model, the air rotational speed, relative to the blade, varies linearly over the blade span as follows:

$$v = \omega r \tag{4.3}$$

where v denotes the rotational speed in meter per second at a radius r, and ω denotes the rotational speed in radians per second. This variation over the blade span results in a varying Reynolds number in the order of 300×10^3 to 400×10^3, which falls within the transitional regime for external flow. This implies that turbulent effects will have to be taken into account by a suitable model. An additional item to be considered in selecting a turbulence model, is large separation bubbles that can form at the aerofoil trailing edge, once the aerofoil stagger is perturbed in the optimisation process that follows. Crivellini and D'Allesandro [13] investigated the reliability of the standard Spalart-Allmaras one-equation turbulence model (without any trip term) for the simulation of laminar separation bubbles on aerofoils at similar Reynolds numbers. The authors found that this model captured the laminar bubble separation behaviour well at Reynolds numbers $\geq 150 \times 10^3$, on aerofoils with very different geometric characteristics. The prediction of the lift and drag coefficients compared to experimental results were found to be reasonable. The authors specifically

found a good comparison between experimental and predicted lift coefficient for a NACA0018 aerofoil, which was experimentally proven to be greatly effected by the presence of a separation bubble. For the purpose of this work, it was therefore decided to implement the standard Spalart-Allmaras one-equation turbulence model without any trip term. The turbulent viscosity in Equation (4.2) is accordingly quantified as follows:

$$\mu_{turb} = \rho f_{v1} max\left(0, \widetilde{\nu}\right) \tag{4.4}$$

where $\widetilde{\nu}$ is the turbulent viscosity variable and f_{v1} is defined as:

$$\begin{aligned} f_{v1} &= \chi^3 / \left(\chi^3 + c_{v1}^3\right) \\ c_{v1} &= 7.1 \\ \chi &= \frac{\widetilde{\nu}}{\nu} \end{aligned} \tag{4.5}$$

where ν is the kinematic viscosity. The turbulent viscosity variable is computed by solving the following Spalart-Allmaras governing equation:

$$\begin{aligned} \nabla \cdot \left(\widetilde{\nu}\mathbf{u}\right) = \frac{1}{\sigma}\left(\nabla \cdot \left[\left(\nu + \widetilde{\nu}\right)\left(\nabla \cdot \widetilde{\nu}\right)\right] + C_{b2}\left(\nabla \cdot \widetilde{\nu}\right)^2\right) + \\ C_{b1}\widetilde{S}\widetilde{\nu} - C_{w1} f_w \left(\frac{\widetilde{\nu}}{d}\right)^2 \end{aligned} \tag{4.6}$$

where the following constants were used:

$$\begin{aligned} c_{b1} &= 0.1355 \\ c_{b2} &= 0.622 \\ \sigma &= \tfrac{2}{3} \\ c_{w2} &= 0.3 \\ c_{w3} &= 2 \\ c_{w1} &= \tfrac{c_{b1}}{\kappa^2} + \tfrac{1+c_{b2}}{\sigma} \\ \kappa &= 0.41 \end{aligned} \tag{4.7}$$

The remaining relations are:

$$\begin{aligned} \widetilde{S} &= max\left(0,\, S + \tfrac{\widetilde{\nu} f_{v2}}{\kappa^2 d^2}\right) \\ S &= \sqrt{2\mathbf{S}_{ij}\mathbf{S}_{ij}} \\ \mathbf{S}_{ij} &= \tfrac{1}{2}\left(\tfrac{\partial u_i}{\partial x_j} - \tfrac{\partial u_j}{\partial x_i}\right) \\ f_{v2} &= 1 - \tfrac{\chi}{1+\chi f_{v1}} \\ f_w &= g\left(\tfrac{1+c_{w3}^6}{g^6+c_{w3}^6}\right)^{\frac{1}{6}} \\ g &= r + c_{w2}\left(r^6 - r\right) \\ r &= max\left(0,\, \tfrac{\widetilde{\nu}}{\widetilde{S}\kappa^2 d^2}\right) \end{aligned} \tag{4.8}$$

where d and S are the shortest distance from the field point to the nearest wall and magnitude of the vorticity. The former variable can be determined by simply calculating the distance from the field point to the closest mesh node on the aerofoil. However, it was rather decided to employ a more accurate method as detailed in the following section. The method was developed during the mesh generation phase as mentioned in Chapter 3 (brief digression). The possible advantage of its use was also assessed. The viscosity variable initial and boundary value have been set to 1.8×10^{-7} and 0, respectively.

4.1.1 Shortest Distance Calculation

Consider mesh nodes and boundary depicted in Figure 4.1. The closest distance to the boundary is denoted d. In the interest of accuracy, the boundary is described via a third order Bézier curve. The latter is defined between two boundary nodes via their nodal coordinates and boundary normal vectors. A Bézier curve is generated between two nodes $\mathbf{P}_0$ and $\mathbf{P}_3$, where the parametric curve coordinate $\mathbf{C}_i(t)$ is a function of the parameter t in the following equation:

$$\mathbf{C}_i(t) = (1-t)^3\mathbf{P}_0 + 3(1-t)^2 t\mathbf{P}_1 + 3(1-t)t^2\mathbf{P}_2 + t^3\mathbf{P}_3; \quad t = [0:1] \tag{4.9}$$

where $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{P}_3$ are defined as Bézier control points. $\mathbf{P}_1$ and $\mathbf{P}_2$ are calculated according to the formulation of Walton and Meek [39] with the following equations:

$$\mathbf{P}_1 = \mathbf{P}_0 + \frac{\ell}{18}(6\mathbf{\Gamma}_i - 2\varrho_i\mathbf{N}_0 + \sigma_i\mathbf{N}_3) \tag{4.10}$$

$$\mathbf{P}_2 = \mathbf{P}_3 - \frac{\ell}{18}(6\mathbf{\Gamma} + \varrho\mathbf{N}_0 - 2\sigma\mathbf{N}_3) \tag{4.11}$$

where ℓ, $\mathbf{\Gamma}$, ϱ and σ are defined as follows:

$$\begin{array}{rcl} \ell & = & \|\mathbf{P}_3 - \mathbf{P}_0\| \\ \mathbf{\Gamma} & = & \frac{\mathbf{P}_3 - \mathbf{P}_0}{\ell} \\ \varrho & = & \frac{6(2a_0 + aa_1)}{4-a^2} \\ \sigma & = & \frac{6(2a_1 + aa_0)}{4-a^2} \end{array} \tag{4.12}$$

where a, a_0 and a_1 are defined as follows:

$$\begin{array}{rcl} a & = & \mathbf{N}_0 \cdot \mathbf{N}_1 \\ a_0 & = & \mathbf{N}_0 \cdot \mathbf{\Gamma} \\ a_1 & = & \mathbf{N}_1 \cdot \mathbf{\Gamma} \end{array} \tag{4.13}$$

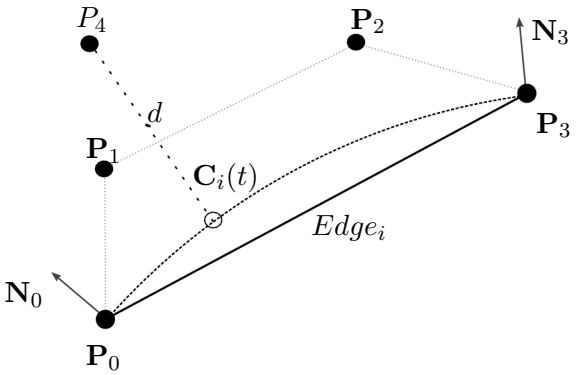

Figure 4.1: Edge represented by Bézier curve

The actual calculation of the closest distance to the boundary (d) is then solved numerically (owing the third order nature of the Bézier curve). In the interest of computational efficiency, the Bézier containing the closest boundary

point to a specific internal node is first sought. The egde's nodal coordinates are then utilised to generate a curve for the specific aerofoil section. The node $\mathbf{C}_i$ which would lead to the shortest distance:

$$d = \sqrt{(P_{4x} - C_{ix})^2 + (P_{4y} - C_{iy})^2} \tag{4.14}$$

is then calculated in an iterative manner by applying a Newton method to $f(t)$ to determine the optimal parametric value as follows:

$$t_{new} = t_{cur} - \frac{f(t_{cur})}{\frac{d}{dt} f(t_{cur})} \tag{4.15}$$

where t_{new} and t_{cur} are the newly calculated and current parametric parameter values respectively and $f(t)$ and $\frac{d}{dt} f(t)$ are defined as follows:

$$\begin{aligned} f(t) &= \tfrac{d}{dt}\left[(P_{4x} - C_{ix})^2 + (P_{4y} - C_{iy})^2\right] \\ &= -2(P_{4x} - C_{ix})\tfrac{d}{dt}C_{ix} - 2(P_{4y} - C_{iy})\tfrac{d}{dt}C_{iy} \end{aligned} \tag{4.16}$$

$$\begin{aligned} \frac{d}{dt} f(t) = 2\left(\frac{d}{dt}C_{ix}\right)^2 - 2(P_{4x} - C_{ix})\frac{d^2}{dt^2}C_{ix} + \\ 2\left(\frac{d}{dt}C_{iy}\right)^2 - 2(P_{4y} - C_{iy})\frac{d^2}{dt^2}C_{iy} \end{aligned} \tag{4.17}$$

The Newton method requires a reasonable initial guess. A linear interpolation is used for this purpose (Equation (4.18)) and it was found that the closest distance was typically computed in three or less iterations when the parametric value change is less then 5%.

$$t_{cur} = \frac{\sqrt{(P_{4x} - P_{0x})^2 + (P_{4y} - P_{0y})^2}}{\sqrt{(P_{4x} - P_{0x})^2 + (P_{4y} - P_{0y})^2} + \sqrt{(P_{4x} - P_{3x})^2 + (P_{4y} - P_{3y})^2}} \tag{4.18}$$

where the nomenclature is as previously defined. It is worth noting that, in this work, the above did not result in large differences in computed μ_{turb} as compared to simply using the closest node distance. The latter being the distance between the internal node and the closest edge node on the aerofoil.

Chapter 5

Flow Discretisation and Flow Solution

In this section the numerical solution of the governing equations in Chapter 4 is described. This includes the spacial finite volume discretisation of the flow domain. The governing Equations (4.1) and (4.2) can be rewritten in a weak form:

$$\int_S \left(\mathbf{F}^j + \mathbf{H}^j - \mathbf{G}^j\right) n_j dS = \int_V \mathbf{Q} dV \tag{5.1}$$

where V and S are the arbitrary volume and surface with outward pointing unit normal vector $\mathbf{n}$. The source terms are represented by $\mathbf{Q}$. The remaining variables are defined as follows:

$$\mathbf{F}^j = \mathbf{W}u_j \tag{5.2}$$

$$\mathbf{W} = \begin{pmatrix} W_0 \\ W_1 \\ W_2 \end{pmatrix} = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \end{pmatrix} \tag{5.3}$$

$$\mathbf{H}^j = \begin{pmatrix} 0 \\ p\delta_{1j} \\ p\delta_{2j} \end{pmatrix} \tag{5.4}$$

$$\mathbf{G}^j = \begin{pmatrix} 0 \\ \sigma_{1j} \\ \sigma_{2j} \end{pmatrix} \tag{5.5}$$

$$\sigma_{ij} = (\mu + \mu_{turb}) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{5.6}$$

where δ_{ij}, σ_{ij} and x_i are the Kronecker delta, stress and co-ordinates respectively. The spatial discretisation algorithm will next be described.

5.1 Spatial Discretisation

The spatial finite volume discretisation employed by Elemental$^{\text{TM}}$ is a vertex-centred edge-based algorithm (Malan et al. [40]). The method is applicable to both structured and unstructured (complex geometries) meshes. A finite volume is constructed by connecting edge midpoints to element centroids with only one node in the control volume as per Vahdati et al. [41] (Figure 5.1).

The surface integrals in the governing Equation (5.1) are calulated in an edge-wise manner. The bounding surface information is stored per edge and dubbed edge-coefficient. For internal edge Υ_{mn}, the latter is defined as:

$$\mathbf{C} = \mathbf{n}^{mn_1} S_{mn_1} + \mathbf{n}^{mn_2} S_{mn_2} \tag{5.7}$$

where S_{mn_1} is a bounding surface-segment that intersects Υ_{mn} (Figure 5.1). The surface integral in Equation (5.1) for V_m (at node m) follows:

$$\int_S \left(\mathbf{F}^j + \mathbf{H}^j - \mathbf{G}^j\right) n_j dS \approx \sum_{\Upsilon_{mn} \cap V_m} \left(\overline{\mathbf{F}^j}_{mn} + \overline{\mathbf{H}^j}_{mn} - \overline{\mathbf{G}^j}_{mn}\right) C^j_{mn} \tag{5.8}$$

where $\bar{F}$ terms denote edge-averaged values which are calculated such that second-order accuracy of the overall scheme is ensured (Malan and Lewis [42]). $\bar{\mathbf{G}}^j_{mn}$ is calculated as follows:

$$\overline{\mathbf{G}^j}_{mn} = \overline{\mathbf{G}^j}_{mn}|_{tang} + \overline{\mathbf{G}^j}_{mn}|_{norm} \tag{5.9}$$

where the $|_{tang}$ and $|_{norm}$ are the tangent and normal contributions respectively. The former is calculated with a directional derivative, while the latter is approximated with a finite volume first derivative.

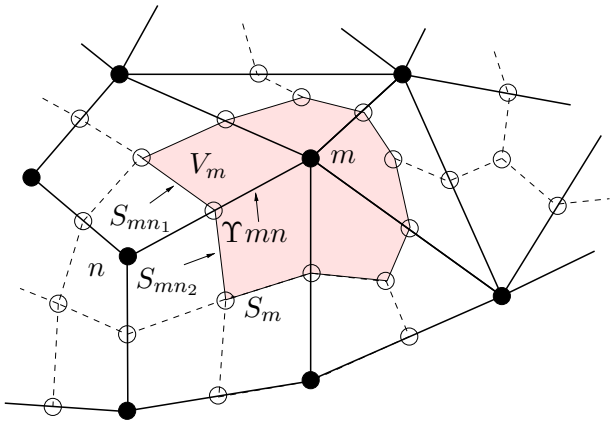

Figure 5.1: Schematic diagram of the construction of the dual-mesh [1]

5.2 Solution Procedure

In solving the governing equation there are two numerical difficulties: Firstly, odd-even decoupling is encountered due to the finite volume discretisation of the convective terms (linear interpolation). Secondly, there is a need (due to incompressibility of fluid) for a pressure field change so that the continuity equation is satisfied. For this purpose an Upwind Pressure-Projection Artificial Compressibility (UP-AC) algorithm (Oxtoby and Malan [44]) is employed by Elemental^{TM}. This is described below.

5.2.1 UP-AC Algorithm

The first step in solving Equation (5.1) follows in a semi-discrete form:

[1] Here, Υ_{mn} depicts the edge connecting the nodes m and n (Mowat et al. [43])

$$\frac{W_i^*}{\Delta\tau}V^\tau = -\int_S \left(F_i^j - G_i^j\right) n_j dS|^\tau + Q_i V|^\tau, \text{ for } i = 1,\, 2. \tag{5.10}$$

where the τ superscript denotes the previous solution or pseudo time-step and $\Delta\tau$ is calculated with Equation (5.12). Convective odd-even decoupling is prevented by upwinding the momentum in the flux term $\mathbf{F}$ via third order interpolation viz. the Monotonic Upstream-Centered Scheme for Conservation Laws (MUSCL) (van Leer [45]). Similar upwinding can be used to construct left and right states. However, the current scheme employs a one-sided extrapolation from the upwind node. W_i^* is an auxiliary variable which is utilised in the second step as follows:

$$\frac{1}{c_\tau^2}\frac{\left(p^{\tau+\Delta\tau_2} - p^\tau\right)}{\Delta\tau_2}V^\tau = -\int_S \left[\rho u_k|^\tau + \Delta\tau\left(\frac{W_k^*}{\Delta\tau} - \frac{\partial H_k^j}{\partial x_j}\bigg|^{\tau+\alpha\Delta\tau_2}\right)\right] n_k dS \tag{5.11}$$

where c_τ is the pseudo-acoustic velocity (Equation (5.13)). $\Delta\tau$ is calculated as follows:

$$\Delta\tau = CFL\left[\frac{(|u_i| + c_\tau)}{\Delta x_i} + \frac{2\mu}{\rho\Delta x_i^2}\right]^{-1} \tag{5.12}$$

where CFL and Δx_i are the Courant-Friedrichs-Lewy number and the effective mesh spacing in the direction i respectively. Further, c_τ is the pseudo-acoustic velocity and is defined as:

$$c_\tau^2 = max\left(\epsilon^2,\, 1.2u_j u_j\right) \tag{5.13}$$

The value of ϵ is typically chosen as $0.1u_{max}$, where u_{max} is the peak flow velocity in the domain (Malan et al. [40]). A variable α is introduced to allow for switching between an explicit and implicit method. The former corresponds to $\alpha = 0$ and the latter $\alpha = 1$. The equation is solved in implicit form in a matrix-free manner. This is done by using a preconditioned Generalised Minimum Residual (GMRES) method for $\alpha = 1$ and $\Delta\tau_2 = \Delta\tau$.

The third (final) step of the solution procedure in a semi-discrete form follows as:

$$\frac{\left(W_i^{\tau+\Delta\tau} - W_i^\tau\right)}{\Delta\tau}V^\tau = \frac{W_i^*}{\Delta\tau}V^\tau - \int_S H_i^j n_j dS\bigg|^{\tau+\Delta\tau_2} \equiv R_i(\mathbf{W}) \text{ for } i = 1,\, 2. \tag{5.14}$$

where R_i is defined as the residual.

5.2.2 Full Implicit Implementation

Currently, in Elemental$^{\text{TM}}$, the residual in Equation (5.14) is partially implicit (pressure only). In an effort to reduce the computational cost, the UP-AC algorithm is modified to the full implicit version [40] for this research work:

$$R_i(\mathbf{W})|^{\tau+\Delta\tau} \approx R_i(\mathbf{W})|^\tau + \frac{\partial R_i(\mathbf{W})}{\partial W_i}\bigg|^\tau \Delta W_i \tag{5.15}$$

where $R_i(\mathbf{W})|^\tau$ is given by Equation (5.14). The convective contribution to the Jacobian is computed as:

$$\frac{\partial R_i(\mathbf{W})}{\partial W_i}\bigg|_{conv}^{\tau} \approx \frac{1}{V}\frac{\partial}{\partial W_i}\sum_{\Upsilon_{mn}\cap V_m}\overline{W_i}u_j C_{mn}^j\bigg|^{\tau} \tag{5.16}$$

where u_j is approximated by an edge average value. An interpolation variable β is introduced in the approximation of $\overline{W_i}$:

$$\overline{W_i} \approx \beta W_{ni} + (1-\beta)W_{mi} \tag{5.17}$$

where $\beta = 0.5$ is utilised in this work.

The viscous contribution to the Jacobian is computed as:

$$\frac{\partial R_i(\mathbf{W})}{\partial W_i}\bigg|_{visc}^{\tau} \approx \frac{1}{V}\frac{\partial}{\partial W_i}\left((\mu+\mu_{turb})\sum_{\Upsilon_{mn}\cap V_m}\nabla W_j C_{mn}^j\bigg|^{\tau}\right) \tag{5.18}$$

Finally, Equation (5.15) is updated with the convective and viscous derivatives:

$$\begin{aligned} R_i(\mathbf{W})|^{\tau+\Delta\tau} \approx\ & R_i(\mathbf{W})|_{conv}^{\tau} + R_i(\mathbf{W})|_{visc}^{\tau} + \\ & \frac{\partial}{\partial W_i}\left(\sum_{\Upsilon_{mn}\cap V_m}\left[\beta W_{ni} + (1-\beta)W_{mi}\right]\frac{(u_{jn}+u_{jm})}{2}C_{mn}^j\right)\bigg|^{\tau}\Delta W_i + \\ & \frac{\partial}{\partial W_i}\left((\mu+\mu_{turb})\sum_{\Upsilon_{mn}\cap V_m}\frac{(W_{jn}-W_{jm})}{\ell_{mn}}C_{mn}^j\right)\bigg|^{\tau}\Delta W_i \end{aligned} \tag{5.19}$$

where the nomenclature is as previously defined.

5.2.3 Turbulence

The solution procedure followed in solving the turbulent viscosity variable in Equation (5.20) is similar to the pseudo time stepping approach taken in the UP-AC algorithm. The additional step in a semi-discrete form (with re-arranged terms in the governing) follows:

$$\begin{aligned} \frac{\widetilde{\nu}^{\tau+\Delta\tau}}{\Delta\tau}V^{\tau} =& -\int_S \widetilde{\nu}u_j n_j dS|^{\tau} + \frac{(1+C_{b2})}{\sigma}\int_S\left[(\nu+\widetilde{\nu})\left(\nabla\cdot\widetilde{\nu}\right)\right]n_j dS|^{\tau} - \\ & \frac{C_{b2}}{\sigma}(\nu+\widetilde{\nu})\int_S(\nabla\cdot\widetilde{\nu})\,n_j dS|^{\tau} + \left[C_{b1}\widetilde{S}\widetilde{\nu} - C_{w1}f_w\left(\frac{\widetilde{\nu}}{d}\right)^2\right]_m \\ \equiv & R_{turb}(\widetilde{\nu}) \end{aligned} \tag{5.20}$$

The first integral term is approximated as follows:

$$-\int_S \widetilde{\nu}u_j n_j dS|^{\tau} \approx -\sum_{\Upsilon_{mn}\cap V_m}\widetilde{\nu}_{up}u_j C_{mn}^j \tag{5.21}$$

where $\widetilde{\nu}_{up}$ and u_j are approximated by an upwind node value and a linearly interpolated face value respectively. The second integral is approximated as follows:

$$\begin{aligned} &\frac{(1+C_{b2})}{\sigma}\int_S (\nu+\widetilde{\nu}_{av})\,(\nabla\cdot\widetilde{\nu})\,n_j dS|^\tau \\ &\approx \frac{(1+C_{b2})}{\sigma}\sum_{\Upsilon_{mn}\cap V_m}(\nu+\widetilde{\nu}_{av})\,(\nabla\cdot\widetilde{\nu})\,C^j_{mn} \end{aligned} \tag{5.22}$$

where $\widetilde{\nu}_{av}$ and $\nabla\cdot\widetilde{\nu}$ are approximated by edge average value and component derivative over the edge (Equation 5.9).

The third integral is approximated as follows:

$$\begin{aligned} &-\frac{C_{b2}}{\sigma}(\nu+\widetilde{\nu})\int_S(\nabla\cdot\widetilde{\nu})\,n_j dS|^\tau \\ &\approx -\frac{C_{b2}}{\sigma}(\nu+\widetilde{\nu}_m)\sum_{\Upsilon_{mn}\cap V_m}(\nabla\cdot\widetilde{\nu})\,C^j_{mn} \end{aligned} \tag{5.23}$$

where $\nabla\cdot\widetilde{\nu}$ is approximated by a component derivative over the edge.

The turbulence implicit implementation is done in a similar fashion to the convective term by:

$$R_{turb}(\widetilde{\nu})|^{\tau+\Delta\tau} \approx R_{turb}(\widetilde{\nu})|^\tau + \left.\frac{\partial R_{turb}(\widetilde{\nu})}{\partial\widetilde{\nu}}\right|^\tau \Delta\widetilde{\nu} \tag{5.24}$$

where the residual derivative is computed as:

$$\begin{aligned} \left.\frac{\partial R_{turb}(\widetilde{\nu})}{\partial\widetilde{\nu}_m}\right|^\tau \approx &-\frac{\partial}{\partial\widetilde{\nu}_m}\left(\sum_{\Upsilon_{mn}\cap V_m}\widetilde{\nu}_{up}u_{j_{face}}C^j_{mn}\right)\Bigg|^\tau + \\ &\frac{(1+C_{b2})}{\sigma}\frac{\partial}{\partial\widetilde{\nu}_m}\left(\sum_{\Upsilon_{mn}\cap V_m}(\nu+\widetilde{\nu}_{av})\frac{(\widetilde{\nu}_n-\widetilde{\nu}_m)}{\ell_{mn}}C^j_{mn}\right)\Bigg|^\tau - \\ &\frac{C_{b2}}{\sigma}(\nu+\widetilde{\nu}_m)\frac{\partial}{\partial\widetilde{\nu}_m}\left(\sum_{\Upsilon_{mn}\cap V_m}\frac{(\widetilde{\nu}_n-\widetilde{\nu}_m)}{\ell_{mn}}C^j_{mn}\right)\Bigg|^\tau + \\ &\frac{\partial}{\partial\widetilde{\nu}_m}\left[C_{b1}\widetilde{S}\widetilde{\nu}-C_{w1}f_w\left(\frac{\widetilde{\nu}}{d}\right)^2\right]\Bigg|^\tau_m \end{aligned} \tag{5.25}$$

The above implicit implementation was found to result in a five times computational speed-up, as compared to the explicit version. Further improvements affecting the UP-AC algorithm included the co-development (with Merrick et al. [35]) and implementation of an improved (more accurate) computation of velocity and pressure face values. This is detailed below.

5.3 Higher Order Discretisation: ETAD

The meshes utilised in this work are unstructured, with high element growth rates (stretched meshes) to save on the total amount of elements in the mesh. Merrick et al. [35] developed a novel finite volume disretisation method for advection-diffusion systems, named the Enhanced Taylor Advection-Diffusion

(ETAD) scheme. The method extends on state-of-the-art upwind schemes in that it (a) formally accounts for mesh stretching, as well as (b) treats advection and diffusion terms in a more holistic manner. The cited work included demonstrating the method via application to incompressible flow modelling, where only certain terms in the governing equation are treated. These exclude the discretisation of pressure related terms. In this work the blended ETAD scheme was implemented, further developed into ETAD_p scheme, and applied to an industrial turbulent flow problem for the first time.

For illustrative purposes, a 1D advection-diffusion equation is used to describe the basic ETAD scheme:

$$\frac{d(u_x\Phi)}{dx} = k\frac{d^2\Phi}{dx^2} \tag{5.26}$$

where Φ, u_x, x and k are respectively, the scalar field being solved for, flow speed, x-coordinate, and the diffusion constant.

Consider a 1D mesh as shown in Figure 5.2 that consists of four nodes and a control volume (height of unity and two side faces) at node c. The ETAD scheme utilises only data from the two nodes connected to the face's edge, as shown for the left face in Figure 5.2. This implies that only the following data from the nodes u and c is used:

- Function values.
- Computed gradients.
- Up- and downwind mesh stretching factors.
- Edge length.

The generic left face approximation follows:

$$\Phi_{f_L} = A\Phi_u + B\Phi_c + C\left.\frac{d\Phi}{dx}\right|_u \Delta x + D\left.\frac{d\Phi}{dx}\right|_c \Delta x \tag{5.27}$$

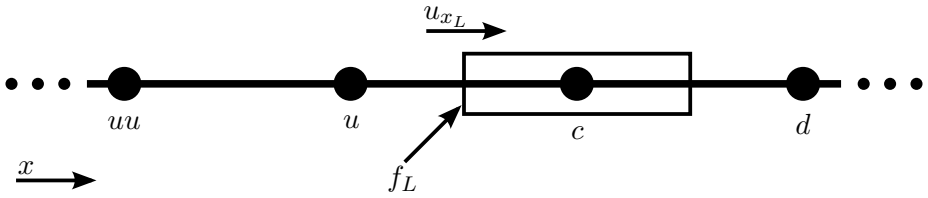

Figure 5.2: An ETAD scheme internal face stencil.

where the coefficients A, B, C and D are the coefficients to be determined for three mesh type locations viz. internal face shown in Figure 5.2, upwind-boundary-adjacent face shown in Figure 5.3, and the downwind-boundary-adjacent face shown in Figure 5.4. The ETAD scheme utilises a Taylor Series Expansion (TSE) to calculate these coefficients, in the interest of accuracy:

$$\begin{aligned}\Phi(x) = \Phi(x_0) + \left.\frac{\Phi}{x_0}\right|_{x_0}(x-x_0) + \left.\frac{d^2\Phi}{dx^2}\right|_{x_0}\frac{(x-x_0)^2}{2!} + \\ \left.\frac{d^3\Phi}{dx^3}\right|_{x_0}\frac{(x-x_0)^3}{3!} + \ \ldots\ + \left.\frac{d^n\Phi}{dx^n}\right|_{x_0}\frac{(x-x_0)^n}{n!}\end{aligned} \tag{5.28}$$

With reference to the figures it is convenient to define the following notation:

$$f = \frac{x_u - x_{uu}}{x_c - x_{uu}} \tag{5.29}$$

$$g = \frac{x_c - x_u}{x_d - x_u} \tag{5.30}$$

$$h = \frac{x_d - x_c}{x_{dd} - x_c} \tag{5.31}$$

$$\Delta x = x_c - x_u \tag{5.32}$$

where f, g and h are the mesh stretching factors associated with the respective edges, and Δx is the edge length over the upwind control volume face. Node position x_{dd} (not shown in Figure 5.2) is the node downstream of position x_d. By implication, the following holds:

$$x_u - x_{uu} = \left(\frac{f}{1-f}\right)\Delta x \tag{5.33}$$

$$x_d - x_c = \left(\frac{1-g}{g}\right)\Delta x \tag{5.34}$$

$$x_{dd} - x_d = \left(\frac{1-h}{h}\right)\left(\frac{1-g}{g}\right)\Delta x \tag{5.35}$$

$$V_c = \frac{A\Delta x}{2g} \tag{5.36}$$

$$V_u = \frac{A\Delta x}{2(1-f)} \tag{5.37}$$

where A and V are respectively cell face area and volume.

Considering Equation (5.27) to calculate the field variable at the volume's face. The gradients are approximated by a finite difference scheme as follows:

$$\left.\frac{d\Phi}{dx}\right|_u \approx \frac{1-f}{\Delta x}\left(\Phi_c - \Phi_{uu}\right) \tag{5.38}$$

$$\left.\frac{d\Phi}{dx}\right|_c \approx \frac{g}{\Delta x}\left(\Phi_d - \Phi_u\right) \tag{5.39}$$

The flux resulting from the advection-diffusion equation at the internal face is discretised next. This is achieved by employing the internal face field variable approximation (Equation (5.27)) and gradient approximations (Equations (5.38) and (5.39)), which results in the following expression:

$$\begin{aligned} u_x\Phi_{f_L} - k\left.\frac{d\Phi}{dx}\right|_{f_L} \approx {} & \left[-u_xC(1-f)\right]\Phi_{uu} + \left[u_xA - u_xDg + \frac{k}{\Delta x}\right]\Phi_u \\ & + \left[u_xB + u_xC(1-f) - \frac{k}{\Delta x}\right]\Phi_c + \left[u_xDg\right]\Phi_d \end{aligned} \tag{5.40}$$

The right hand side of Equation (5.40) is now expanded by applying the Taylor Series expansion to Φ_{uu}, Φ_u, Φ_c, and Φ_d which leads to Equation (5.41). The terms are arranged such that the left hand side terms match the left hand side terms of Equation (5.40), with the exception of the additional coefficients.

$$
\begin{aligned}
&u_x(A+B)\Phi_{f_L} + \left(\frac{1}{2}\right)\left[u_x(-A+B+2C+2D) - \frac{2k}{\Delta x}\right]\Delta x\frac{d\Phi}{dx}|_{f_L} \\
&= [-u_xC(1-f)]\Phi_{uu} + \left[u_xA - u_xDg + \frac{k}{\Delta x}\right]\Phi_u + \\
&\quad \left[u_xB + u_xC(1-f) - \frac{k}{\Delta x}\right]\Phi_c + [u_xDg]\Phi_d - \\
&\quad u_x\left(\frac{1}{8}\right)\left(A+B+\left(\frac{4f}{f-1}\right)C + \left[\frac{4(1-g)}{g}\right]D\right)\Delta x^2\frac{d^2\Phi}{dx^2}|_{f_L} - \\
&\quad u_x\left(\frac{1}{48}\right)\left(-A+B+\left[\frac{6f^2+2}{(f-1)^2}\right]C + \left[\frac{2(3g^2-6g+4)}{g^2}\right]D\right)\Delta x^3\frac{d^3\Phi}{dx^3}|_{f_L} + \\
&\quad \left(\frac{1}{48}\right)\left(\frac{2k}{\Delta x}\right)\Delta x^3\frac{d^3\Phi}{dx^3}|_{f_L} - \dots
\end{aligned}
\tag{5.41}
$$

The coefficients A,B, C and D can now be solved in such a manner that would guarantee 4^{th}-order accuracy at the face. This is achieved by establishing four linearly independent equations. The first two are found by equating the Φ_{fL} and $k\frac{d\Phi}{dx}$ terms in Equations (5.40) and (5.41). The remaining two equations result by equating the two lowest error terms Δx^2 and Δx^3 in Equation (5.41) to zero. The coefficients A,B, C and D, determined by simultaneously solving the four equations, are expressed as follows:

$$A = 1 - B \tag{5.42}$$

$$B = \frac{1}{2} - C - D \tag{5.43}$$

$$C = \left(\frac{(1-f)(1-g)}{fg}\right)D + \frac{1-f}{4f} \tag{5.44}$$

$$D = \frac{g^2(2fk + fu_x\Delta x - 2k + u_x\Delta x)}{8u_x\Delta x(g-1)(fg-f+1)} \tag{5.45}$$

where $u_x \neq 0$.

The upwind-boundary-adjacent face (Figure 5.3) is treated in a similar manner to above, with the following exceptions:

- The boundary node gradient discretisation:

$$\left.\frac{d\Phi}{dx}\right|_u \approx \frac{1}{\Delta x}\left(\Phi_c - \Phi_u\right) \tag{5.46}$$

- The system of four equations that requires simultaneous solution has a determinant of zero. Therefore, the highest order error term and the gradient at the boundary node are both discarded ($C = 0$).

The coefficients A,B and D, similarly determined by simultaneously solving three equations, as follows:

$$A = 1 - B \tag{5.47}$$

$$B = \frac{1}{2} - D \tag{5.48}$$

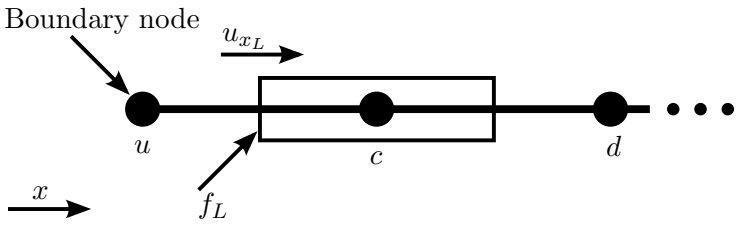

Figure 5.3: An ETAD scheme upwind-boundary-adjacent face stencil.

$$D = \frac{g}{4(g-1)} \tag{5.49}$$

The downwind-boundary-adjacent face is considered next (Figure 5.4). Here a similar procedure to the upwind-boundary-adjacent boundary face is followed. Only, in this case $D = 0$.

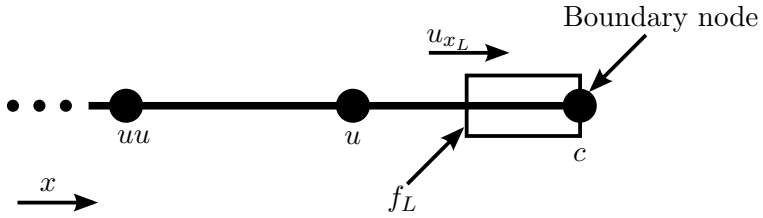

Figure 5.4: An ETAD scheme downwind-boundary-adjacent face stencil.

The coefficients A,B and C, similarly determined by simultaneously solving three equations, are expressed as follows:

$$A = 1 - B \tag{5.50}$$

$$B = \frac{1}{2} - C \tag{5.51}$$

$$C = \frac{1-f}{4f} \tag{5.52}$$

The convection at a face ($u_{x_{f_L}}$) is also discretised to higher order. The approximation for the internal face is represented by a similar equation as for the field variable Φ_{fL}, as shown below:

$$u_{x_{f_L}} = Au_{x_u} + Bu_{x_c} + C \left.\frac{du_x}{dx}\right|_u \Delta x + D \left.\frac{du_x}{dx}\right|_c \Delta x \tag{5.53}$$

The coefficients A,B and C are solved in a similar fashion to the above and leads to:

$$A = 1 - B \tag{5.54}$$

$$B = \frac{1}{2} - C - D \tag{5.55}$$

$$C = \left(\frac{(1-f)(1-g)}{fg}\right) D + \frac{1-f}{4f} \tag{5.56}$$

$$D = \frac{g^2(f+1)}{8(g-1)(fg-f+1)} \tag{5.57}$$

The ETAD scheme methodology has been described in a 1D case and is now extended to 2D and 3D unstructured meshes. The alterations required are as follows:

- $\left.\frac{du_x}{dx}\right|_u$ and $\left.\frac{du_x}{dx}\right|_u$ are replaced by the derivative in the edge direction $\mathbf{t}\cdot\nabla\Phi_u$ and $\mathbf{t}\cdot\nabla\Phi_c$.
- Δx is replaced by edge length.
- u_x is replaced by $\mathbf{u}_f \cdot \mathbf{n}$ (advection value at the face, f, in the direction of the face's unit normal vector, $\mathbf{n}$).
- The stretching factors are now to be computed via an algorithm which is applicable to unstructured meshes (Merrick et al. [35]).

5.4 Blended ETAD

Merrick et al. [35] conducted a critical analysis of Order-of-Accuracy (OoA) and odd-even decoupling. For the OoA accuracy analysis on equispaced, constantly stretched and non-equispaced meshes, the ETAD scheme is shown to offer at least an order of magnitude improvement in computing a face value, as compared to popular upwind methods. The latter includes Cubic-Upwind-Interpolation (CUI) (Agarwal [46]) and Quadratic- Upwind-Interpolation (QUICK) (Leonard [47]). However, due to more balanced interpolation, the ETAD method results in odd-even decoupling as the mesh tends to be equispaced for advection-dominant flows. To address this, an algorithm was proposed as part of this research work to blend the ETAD scheme with a κ-upwind scheme (Normalised Variable and Space Formulation (NVSF)), in order to achieve an accurate and stable numerical method as follows:

$$\Phi_{Blend} = (\Pi)\Phi_{NVSF} + (1-\Pi)\Phi_{ETAD} \tag{5.58}$$

where:

$$\Pi = \Pi_s + (1-\Pi_s)\Pi_r \tag{5.59}$$

and Π_s and Π_r are calculated to respectively account for stretching and advection dominance as follows:

$$\Pi_s = \begin{cases} 1 - \left|\dfrac{f-0.5}{0.0455}\right|, & \text{if } 0.455 < f < 0.545. \\ 0, & \text{otherwise.} \end{cases} \tag{5.60}$$

$$\Pi_r = \begin{cases} \left|\dfrac{u_x}{k}\right|, & \text{if } \left|\dfrac{u_x}{k}\right| \leq 1.5\left|\dfrac{u_x}{k}\right|_{Domain\ Max}. \\ 1, & \text{otherwise.} \end{cases} \tag{5.61}$$

The blend scheme reverts to NVSF on equispaced grids while tending to ETAD once sufficient stretching of 1.2 is present. Additionally, the scheme reverts to NVSF once a specific advection-diffusion ratio (50% higher than domain maximum advection-diffusion ratio) has been reached.

5.5 ETAD Further Development: ETAD$_\mathbf{p}$

This research work expands on the above method by further considering the pressure and pressure gradient at finite volume faces (Figure 5.5). This transforms ETAD to ETAD$_\mathrm{p}$. In ETAD$_\mathrm{p}$, the pressure at a face (p_{f_L}) is approximated

in a similar fashion to convection at a face (u_{f_L}), by utilising Equation (5.53) and the respective expressions for the coefficients A, B, C and D in Equations (5.54) to (5.57). The gradient approximation, however, differs from the flow variable approximation at a face and is developed next.

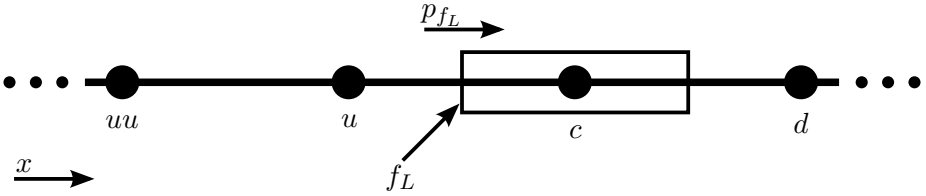

Figure 5.5: An ETAD_p scheme internal face stencil.

The generic left face pressure gradient (in the direction of the edge) approximation follows:

$$\left.\frac{dp}{dx}\right|_{f_L} = Ap_u + Bp_c + C\left.\frac{dp}{dx}\right|_u \Delta x + D\left.\frac{dp}{dx}\right|_c \Delta x \tag{5.62}$$

where p is the pressure. The coefficients A, B, C and D are determined for the internal face only, as shown in Figure 5.5. This is because coefficients C and D both equated to zero for the other mesh locations (upwind-boundary-adjacent and downwind-boundary-adjacent). This leads to the elimination of the gradient terms in the higher order approximation.

Consider the pressure gradient approximation in Equation (5.62). The gradients at nodes u and c are approximated in a similar fashion to the field variable gradients (Equations (5.38) and (5.39)). By substituting the discrete gradients into Equation (5.62) the pressure gradient approximation reduces to:

$$\begin{aligned}\left.\frac{dp}{dx}\right|_{f_L} &\approx Ap_u + Bp_c + C\frac{(1-f)}{\Delta x}(p_c - p_{uu})\Delta x + \frac{Dg}{\Delta x}(p_d - p_u)\Delta x \\ &= [-C(1-f)]p_{uu} + [A - Dg]p_u + [B + C(1-f)]p_c + [Dg]p_d\end{aligned} \tag{5.63}$$

A Taylor Series Expansion (Equation (5.28)) is now applied to the pressure terms p_{uu}, p_u, p_c and p_d and leads to:

$$\begin{aligned}p_{uu} = p_{f_L} - \left[\frac{f+1}{2(1-f)}\right]\left.\frac{dp}{dx}\right|_{f_L}\Delta x + \left[\frac{f+1}{2(1-f)}\right]^2\left.\frac{d^2p}{dx^2}\right|_{f_L}\frac{\Delta x^2}{2} - \\ \left[\frac{f+1}{2(1-f)}\right]^3\left.\frac{d^3p}{dx^3}\right|_{f_L}\frac{\Delta x^3}{6} + ...\end{aligned} \tag{5.64}$$

$$\begin{aligned}p_u = p_{f_L} - \left(\frac{1}{2}\right)\left.\frac{dp}{dx}\right|_{f_L}\Delta x + \left(\frac{1}{2}\right)^2\left.\frac{d^2p}{dx^2}\right|_{f_L}\frac{\Delta x^2}{2} - \\ \left(\frac{1}{2}\right)^3\left.\frac{d^3p}{dx^3}\right|_{f_L}\frac{\Delta x^3}{6} + ...\end{aligned} \tag{5.65}$$

$$\begin{aligned}p_c = p_{f_L} + \left(\frac{1}{2}\right)\left.\frac{dp}{dx}\right|_{f_L}\Delta x + \left(\frac{1}{2}\right)^2\left.\frac{d^2p}{dx^2}\right|_{f_L}\frac{\Delta x^2}{2} + \\ \left(\frac{1}{2}\right)^3\left.\frac{d^3p}{dx^3}\right|_{f_L}\frac{\Delta x^3}{6} + ...\end{aligned} \tag{5.66}$$

$$p_d = p_{f_L} + \left(\frac{2-g}{2g}\right)\left.\frac{dp}{dx}\right|_{f_L}\Delta x + \left(\frac{2-g}{2g}\right)^2\left.\frac{d^2p}{dx^2}\right|_{f_L}\frac{\Delta x^2}{2} + \left(\frac{2-g}{2g}\right)^3\left.\frac{d^3p}{dx^3}\right|_{f_L}\frac{\Delta x^3}{6} + ... \tag{5.67}$$

Equations (5.64) - (5.67) are now substituted into Equation (5.63). The terms are arranged such that the left hand side terms match the left hand side terms in Equation (5.63), with the exception of the additional coefficients.

$$\begin{aligned}\left.\frac{dp}{dx}\right|_{f_L} \approx\ & [-C(1-f)+A-Dg+B+C(1-f)+Dg]p_{f_L} + \\ & \left([-C(1-f)]\left[-\frac{f+1}{2(1-f)}\right]+[A-Dg]\left(\frac{1}{2}\right)+[B+C(1-f)]\left(\frac{1}{2}\right)+\right. \\ & \left.[Dg]\left(\frac{2-g}{2g}\right)\right)\left.\frac{dp}{dx}\right|_{f_L}\Delta x + \left(-C(1-f)\left[\frac{f+1}{2(1-f)}\right]^2+\right. \\ & (A-Dg)\left(\frac{1}{2}\right)^2+[B+C(1-f)]\left(\frac{1}{2}\right)^2+Dg\left(\frac{2-g}{2g}\right)^2\Bigg)\left.\frac{d^2p}{dx^2}\right|_{f_L}\frac{\Delta x^2}{2} + \\ & \left(C(1-f)\left[\frac{f+1}{2(1-f)}\right]^3-(A-Dg)\left(\frac{1}{2}\right)^3+[B+C(1-f)]\left(\frac{1}{2}\right)^3+\right. \\ & \left.Dg\left(\frac{2-g}{2g}\right)^3\right)\left.\frac{d^3p}{dx^3}\right|_{f_L}\frac{\Delta x^3}{6}\end{aligned} \tag{5.68}$$

Coefficients A, B, C and D can now be solved in such a manner that would guarantee 4^{th}-order accuracy at the face. This is achieved by establishing four linear independent equations. The first equation is found by equating the p_{f_L} term to zero:

$$\begin{aligned}0 &= [-C(1-f)+A-Dg+B+C(1-f)+Dg]p_{f_L} \\ 0 &= A+B\end{aligned} \tag{5.69}$$

The second equation is found by ensuring the $\left.\frac{dp}{dx}\right|_{f_L}$ term is present:

$$\begin{aligned}\left.\frac{dp}{dx}\right|_{f_L} &= \left(-C(1-f)\left[-\frac{f+1}{2(1-f)}\right]+(A-Dg)\left(\frac{1}{2}\right)+[B+C(1-f)]\left(\frac{1}{2}\right)+\right. \\ &\quad \left.Dg\left(\frac{2-g}{2g}\right)\right)\left.\frac{dp}{dx}\right|_{f_L}\Delta x \\ 1 &= \left(-\frac{A}{2}+\frac{B}{2}+C+D\right)\Delta x\end{aligned} \tag{5.70}$$

The remaining two equations are found by equating the two lowest error terms Δx^2 and Δx^3 to zero.

$$
\begin{aligned}
0 = &\left(-C(1-f)\left[\frac{f+1}{2(1-f)}\right]^2 + (A-Dg)\left(\frac{1}{2}\right)^2 + [B+C(1-f)]\left(\frac{1}{2}\right)^2 + \right. \\
&\left. Dg\left(\frac{2-g}{2g}\right)^2 \right) \left.\frac{d^2p}{dx^2}\right|_{f_L} \frac{\Delta x^2}{2} \\
&= \frac{-4Cf}{1-f} + A + B + \frac{4D(1-g)}{g}
\end{aligned} \tag{5.71}
$$

$$
\begin{aligned}
0 = &\left(C(1-f)\left[\frac{f+1}{2(1-f)}\right]^3 - (A-Dg)\left(\frac{1}{2}\right)^3 + [B+C(1-f)]\left(\frac{1}{2}\right)^3 + \right. \\
&\left. Dg\left(\frac{2-g}{2g}\right)^3 \right) \left.\frac{d^3p}{dx^3}\right|_{f_L} \frac{\Delta x^3}{6} \\
&= \frac{C(6f^2+2)}{(1-f)^2} - A + B + \frac{D(6g^2-12g+8)}{g^2}
\end{aligned} \tag{5.72}
$$

The coefficients A, B, C and D, determined by simultaneously solving the four equations above, are expressed as follows:

$$A = \frac{-(6f^2-3f+1)g^2 - (-9f^2+6f-1)g - 4f^2 + 4f}{[fg^2+(1-2f)g+f-1]4f\Delta x} \tag{5.73}$$

$$B = -A \tag{5.74}$$

$$C = \frac{(-f^2+2f-1)g}{(4f^2g-4f^2+4f)\Delta x} \tag{5.75}$$

$$D = \frac{(1-f)g^2}{[4fg^2+(1-4f)2g+4(f-1)]\Delta x} \tag{5.76}$$

where the nomenclature is as defined previously.

Chapter 6

Constrained Optimisation

The objective of this chapter is to develop a constrained optimisation methodology that will be applied to an under-performing fan. The method must alter the geometry of the rotor blade by perturbing the aerofoil geometry, such that the specified duty point is met with an improved static efficiency. The efficiency will be calculated with an assumption of 100% rotational velocity recovery into static pressure, by the outlet guide vanes. This is done as the modelling and optimisation of these outlet guide vanes do not form part of this study. As stated previously, various sophisticated optimisation schemes are in existence. Most of these are today specifically implemented to cater for a large number of design variables. In this work, only two design variables (most significant variables concluded in a separate project) are utilised (due to the analytical function used to describe the aerofoil geometry) and therefore an unconstrained steepest gradient algorithm has been selected viz. the spherical quadratic steepest descent method (SQSD), developed by Snyman and Hay [37]. It is a fairly simple, computational efficient, and stable method. It also circumvents the standard line search requirement. A constrained gradient-based method, which is seen as the extension of the SQSD, namely the Dynamic-Q optimisation method (Snyman and Hay [48]) does however exist. The latter has successfully been employed by Craig et al. [49] and it incorporates the gradient-based leap-frog dynamic trajectory method (LFOPC) of Snyman [50]. LFOPC has successfully been implemented in the successive response surface method (SRSM) of Stander and Craig [51], which in turn has been implemented in LS-OPT software (Stander et al. [52]) and successfully employed by Craig and Kingsley [53] and Stander et al. [54]. In this work, it has rather been decided to apply the unconstrained SQSD method to a penalty function formulation of a constrained problem, which is comprised of the objective function and constraints (equality and inequality). A finite difference gradient calculation is employed.

6.1 Optimisation Methodology

It is proposed to represent the 3D fan by two aerofoils, ϕ_1 and ϕ_6 (Figure 3.1), in the optimisation phase. The optimisation methodology followed consists of two primary steps. The first involves the optimisation of the aerofoil geometry at ϕ_1 to a point of maximum efficiency. This is done without prescribing the static pressure required, as it is known that the profile at this specific diameter cannot generate the required pressure rise. Once optimised, the pressure rise of the pseudo-3D model is known at ϕ_1. This information, in conjunction with the required fan static pressure rise, is used to calculate the pressure rise requirement

for the profile at ϕ_6. Step two of the optimisation then commences by prescribing the required static pressure rise and optimising for maximum efficiency.

6.2 Optimisation Scheme

The generic optimisation problem is defined as the constrained minimisation of the following objective function for a specific section (ϕ_1 or ϕ_6):

$$f(\boldsymbol{\lambda}) = -\frac{\Delta p_{fsp_r}}{F_{x_r}} \tag{6.1}$$

The methodology involves solving for the design variables with the following inequality and equality constraints:

$$g_i(\boldsymbol{\lambda}) \leq 0; \quad i = 1, 2, 3....m_g \tag{6.2}$$
$$h_j(\boldsymbol{\lambda}) = 0; \quad j = 1, 2, 3....m_h \tag{6.3}$$

where $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3 \ \ ... \ \lambda_m]$ is the design variable vector. A pragmatic approach is taken in solving this constrained minimisation problem by constructing a penalty function (Snyman [36] and Vanderplaats [55]) formulation of the problem as follows:

$$F(\boldsymbol{\lambda}, \rho, \beta) = f(\boldsymbol{\lambda}) + \sum_{i=1}^{m_g} \beta[g_i(\boldsymbol{\lambda})]^2 + \sum_{j=1}^{m_h} \rho[h_j(\boldsymbol{\lambda})]^2 \tag{6.4}$$

where the penalty parameters are defined as:

$$\rho > 0 \tag{6.5}$$
$$\beta = \begin{cases} 0, & \text{if } g_i(\boldsymbol{\lambda}) \leq 0 \\ \varrho > 0, & \text{otherwise} \end{cases} \tag{6.6}$$

and ϱ is a constant value. The solution to the problem is represented by $\boldsymbol{\lambda}^{Optim}(\rho, \beta)$. The penalty function value is equal to the objective function as long as the design variables are within the feasible region ($g_i(\boldsymbol{\lambda}) \leq 0$ and $h_j(\boldsymbol{\lambda}) = 0$). As soon as this condition is not met, the penalty function value increases due to the presence of the penalty term for the specific constraint condition that is violated.

A gradient-based unconstrained optimisation algorithm is next applied to the constrained problem (Equation (6.4)). The classically steepest descent method perturbs each variable in the direction of the gradient $-\boldsymbol{\nabla} F(\boldsymbol{\lambda}^{k-1})$. The new design variable is then calculated by:

$$\boldsymbol{\lambda}^k = \boldsymbol{\lambda}^{k-1} - \lambda_k \boldsymbol{\nabla} F(\boldsymbol{\lambda}^{k-1}) \tag{6.7}$$

where λ_k is such that

$$F(\boldsymbol{\lambda}^{k-1} - \lambda_k \boldsymbol{\nabla} F(\boldsymbol{\lambda}^{k-1})) = min_\lambda F(\boldsymbol{\lambda}^{k-1} - \lambda \boldsymbol{\nabla} F(\boldsymbol{\lambda}^{k-1})) \tag{6.8}$$

and λ (≥ 0) is determined by a line search method. Snyman [36] improved the overall performance of the standard steepest descent method and removed the required line search by applying it successively, to a sequence of quadratic

approximations of $F(\boldsymbol{\lambda})$. The improved method, dubbed the spherical quadratic steepest descent (SQSD) method, only utilises gradient information (no second derivatives). This also reduces storage requirements. Snyman [36] found the SQSD method to be reliable, stable and competitive, when compared to well established conjugate gradient methods. The author also found the method performs well when applied to ill-conditioned problems.

The implemented SQSD algorithm is as follows:

- *Initialisation:*
  - Specify constants for penalty parameters $\rho = 1 \times 10^2$ and $\varrho = 1 \times 10^4$.
  - Specify convergence tolerances $\epsilon_g = 0.01$ and $\epsilon_x = 5 \times 10^{-4}$.
  - Specify step size in direction of steepest descent $\rho_{step} = 0.05$.
  - Specify design variable perturbation limits $\mathbf{L}$.
  - Set starting point $\boldsymbol{\lambda}^0$.
  - Calculate initial curvature estimate $c_0 = \frac{\nabla F(\boldsymbol{\lambda}^0)}{\rho_{step}}$.
  - Set iteration number $k = 1$.
  - Calculate the design variable (S_d), inequality (S_g) and equality (S_h) constraints scaling factors such that:

$$1.0 = S_{d_0}\lambda_0 \tag{6.9}$$

$$\left|\frac{\partial f(\boldsymbol{\lambda}^0)}{\partial \lambda_1}\right| = S_{d_n}\left|\frac{\partial f(\boldsymbol{\lambda}^0)}{\partial \lambda_2}\right|;\ \text{where } n = 2, 3..m_d \tag{6.10}$$

$$\left\|\boldsymbol{\nabla} f(\boldsymbol{\lambda}^0)\right\| = S_{g_i}\left\|\boldsymbol{\nabla} g_i(\boldsymbol{\lambda}^0)\right\|;\ \text{where } i = 1, 2..m_g \tag{6.11}$$

$$\left\|\nabla f(\boldsymbol{\lambda}^0)\right\| = S_{h_j}\left\|\nabla h_j(\boldsymbol{\lambda}^0)\right\|;\ \text{where } j = 1, 2..m_h \tag{6.12}$$

$$\tag{6.13}$$

- *Step 1:*

$$\text{If } \left\|\nabla F(\boldsymbol{\lambda}^{k-1})\right\| < \epsilon_g \text{ then } \boldsymbol{\lambda}^{Optim} = \boldsymbol{\lambda}^{k-1} \text{ and stop procedure.} \tag{6.14}$$

$$\text{else } \boldsymbol{\lambda}^k = \boldsymbol{\lambda}^{k-1} - \frac{\nabla F(\boldsymbol{\lambda}^{k-1})}{c_{k-1}} \tag{6.15}$$

- *Step 2:*

$$\text{If } \left\|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}\right\| > \rho_{step} \text{ then} \tag{6.16}$$

$$\boldsymbol{\lambda}^k = \boldsymbol{\lambda}^{k-1} - \rho_{step}\frac{\nabla F(\boldsymbol{\lambda}^{k-1})}{\left\|\nabla F(\boldsymbol{\lambda}^{k-1})\right\|} \tag{6.17}$$

- *Step 3:*

$$\text{If } \left({\lambda_n}^k - {\lambda_n}^{k-1}\right) > L_n \text{ then } {\lambda_n}^k = {\lambda_n}^{k-1} + L_n \tag{6.18}$$

$$\text{else if } \left({\lambda_n}^k - {\lambda_n}^{k-1}\right) < -L_n \text{ then } {\lambda_n}^k = {\lambda_n}^{k-1} - L_n \tag{6.19}$$

$$\text{where } n = 1, 2..m_d \tag{6.20}$$

- *Step 4:*

$$\text{If } \left\|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^k\right\| < \epsilon_\lambda \text{ then } \boldsymbol{\lambda}^{Optim} = \boldsymbol{\lambda}^{k-1} \text{ and stop procedure.} \tag{6.21}$$

- *Step 5:*

$$c_k = \frac{2\left[F(\boldsymbol{\lambda}^{k-1}) - F(\boldsymbol{\lambda}^k) - \nabla^T F(\boldsymbol{\lambda}^k)(\boldsymbol{\lambda}^{k-1} - \boldsymbol{\lambda}^k)\right]}{\left\|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}\right\|^2} \tag{6.22}$$

- *Step 6:*

$$c_k = \begin{cases} c_k, & \text{if } c_k \geq 0 \\ 0.01, & \text{otherwise} \end{cases} \tag{6.23}$$

- *Step 7:*

$$k = k + 1 \text{ and go to step 1.} \tag{6.24}$$

Note that the perturbation limitation of each design variable in Step 3 is an addition to the standard SQSD method. In this study it was found that limiting each design variable individually was required, in addition to the step size limitation of Step 2. The value of c_k in Step 6 for the case where the calculated value is less then zero is set to 0.01 instead of 10^{-60}. This is done, in an effort, to prevent any extreme changes in design variables.

6.3 Gradient Calculation

The gradients of various functions (penalty, objective, equality and inequality constraints) with respect to each design variable are utilised in the SQSD scheme. A forward/backward difference method was employed to compute this (a higher order method was not employed to reduce computational cost):

$$\frac{\partial f(\boldsymbol{\lambda})}{\partial \lambda_i} \approx \frac{\Delta f(\boldsymbol{\lambda})}{\Delta \lambda_i}; \text{ where } i = 1, 2.. \text{ number of design variables} \tag{6.25}$$

$$\frac{\Delta f(\boldsymbol{\lambda})}{\Delta \lambda_i} = \frac{f(\boldsymbol{\lambda} + \Delta\lambda_i) - f(\boldsymbol{\lambda})}{\Delta \lambda_i} \tag{6.26}$$

where $\Delta f(\boldsymbol{\lambda})$ is the change in objective function due to variable perturbation of $\Delta\lambda_i$.

The method implemented alternates between a forward and backward difference method, depending on the last perturbation (value increased or decreased) of the specific design variable. There are thus two items to resolve: (a) the direction of the finite difference method and (b) the size of the variable perturbation in the gradient calculation.

The design variable perturbation $\Delta\lambda_i$ for design variable i is selected to be in the opposite direction of the design variable perturbation resulting from the optimisation scheme. In other words, if the design variable is reduced, a forward difference gradient scheme is employed. If the design variable increases, a backward difference gradient scheme is employed. This is done to assist in preventing the design variable from moving past the optimal functional value, resulting in an incorrect gradient calculation (incorrect sign). This is illustrated in the example below.

Consider an objective function $y(x) = x^2$, that is a function of one design variable x (Figure 6.1). There are three data points x_1, x_2 and x_3. Lets assume the gradient at x_1 is required and the optimisation scheme has been decreasing

design variable x. In a forward difference gradient scheme, the design variable perturbation will be positive. In other words, $\Delta x = x_3 - x_1$ will lead to a positive gradient. In a backward difference gradient scheme, the design variable perturbation will be negative. In other words, $\Delta x = x_2 - x_1$ will lead to a negative gradient. The optimisation scheme will reduce (closer to the optimum point) the design variable if a forward difference gradient scheme is employed. The design variable will, however, increase (move further away from the optimum point) if a backward difference gradient scheme is employed. In this case it is clear that the forward difference scheme results in a design variable perturbation in the expected direction, and the backward difference does not. It is therefore proposed to base the type of finite difference gradient scheme (forward/backward) on the direction of the design variable perturbation in the optimisation scheme.

The design variable perturbation size in the finite difference gradient scheme is typically chosen to be very small (1×10^{-6}), except in the case where numerical noise is present, as is the case in this work. The perturbation value should be selected carefully by employing a sensitivity graph for each λ_i, as recommended by Snyman [36]. The sensitivity graph of a design variable considered in this work is shown in Figure 6.2. Here the perturbation value was selected as 1.041667×10^{-2} which coincides with the graph gradient tending to zero.

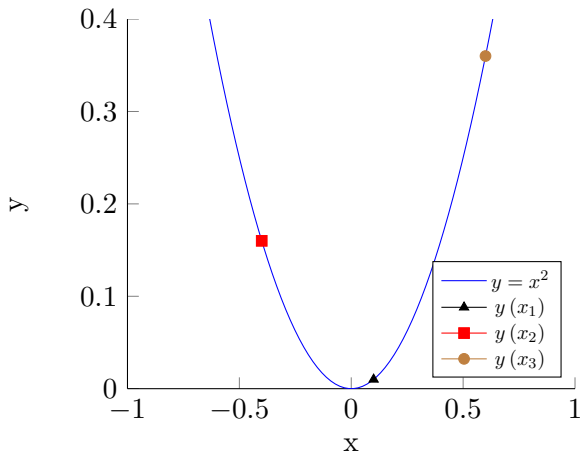

Figure 6.1: Illustrative example of backward and forward finite difference gradient schemes

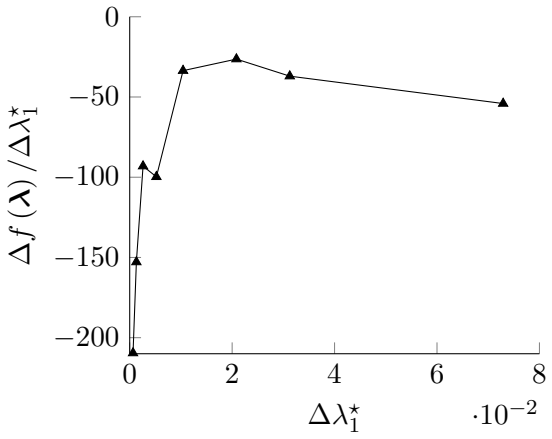

Figure 6.2: Sensitivity graph of finite difference approximation to $\lambda_1^\star$

6.4 Mesh Movement

Due to perturbations of the aerofoil geometry in the optimisation procedure, a new mesh is required after each variable perturbation. An automatic and efficient radial basis function (Mowat et al. [43] and Oxtoby et al. [56]) interpolation scheme was implemented to cater for this. This would allow for only one mesh per cascade to be generated. The movement of an internal mesh node $\mathbf{P}_i$ is determined as follows:

$$\Delta\mathbf{P}_i = r_b\Delta\mathbf{P}_1 + (1 - r_b)\Delta\mathbf{P}_2 \tag{6.27}$$

where $\mathbf{P}_1$ and $\mathbf{P}_2$ are the nodal coordinate perturbations of the closest aerofoil boundary node and the collinear external boundary node respectively. The variable r_b is defined as:

$$r_b = \frac{d_2^{\frac{3}{2}}}{d_1^{\frac{3}{2}} + d_2^{\frac{3}{2}}} \tag{6.28}$$

where d_1 and d_2 are the distances from the internal mesh node to the closest aerofoil boundary node $\mathbf{P}_1$ and the collinear boundary node $\mathbf{P}_2$, respectively. The collinear node ($\mathbf{P}_2$) to $\mathbf{P}_1$ and $\mathbf{P}_i$ is determined by creating a vector $\mathbf{A}_{1i}$ pointing from $\mathbf{P}_1$ and $\mathbf{P}_i$. An external boundary node is then identified such that the vector $\mathbf{A}_{i2}$ is in the same direction as $\mathbf{A}_{1i}$ and the angle difference between the two vectors is minimal (smallest angle possible, considering all the external boundary nodes).

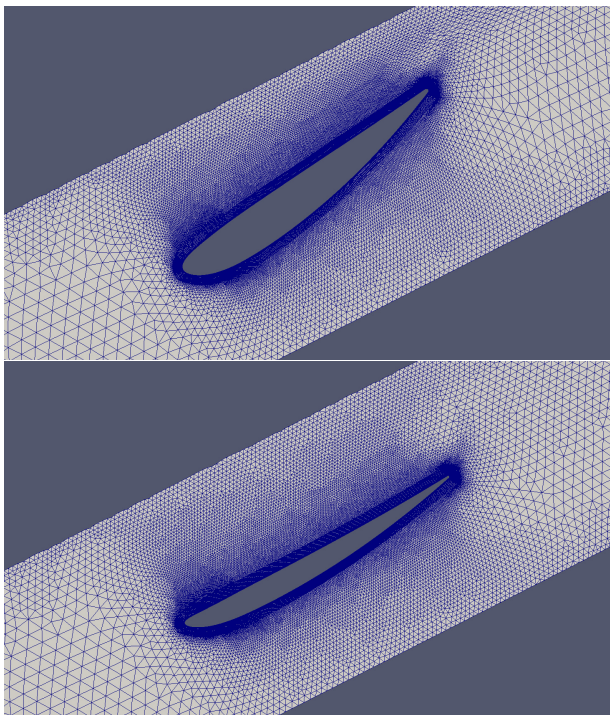

Figure 6.3: Cascade mesh at ϕ_6 before (top) and after (bottom) aerofoil perturbation

6.5 Geometric Optimisation of Profile Section: ϕ_1

The following penalty function is defined for the geometric profile optimisation of section ϕ_1:

$$F_{\phi_1}(\boldsymbol{\lambda}, \rho, \beta) = f_{\phi_1}(\boldsymbol{\lambda}) + \sum_{i=1}^{4} \beta[g_i(\boldsymbol{\lambda})]^2 \tag{6.29}$$

where aerofoil stagger angle (λ_1) and aerofoil percentage thickness (λ_2) are the design variables (Figure 1.1). Their respective inequality constraints are defined as:

$$g_1(\boldsymbol{\lambda}) = -\lambda_1 + \lambda_{1_{LowerLimit}} \tag{6.30}$$
$$g_2(\boldsymbol{\lambda}) = \lambda_1 - \lambda_{1_{UpperLimit}} \tag{6.31}$$
$$g_3(\boldsymbol{\lambda}) = -\lambda_2 + \lambda_{2_{UpperLimit}} \tag{6.32}$$
$$g_4(\boldsymbol{\lambda}) = \lambda_2 - \lambda_{2_{LowerLimit}} \tag{6.33}$$

As the blade profile at ϕ_1 cannot reach the required fan static pressure rise requirement, the equality constraint is removed.

6.6 Geometric Optimisation of Profile Section: ϕ_6

The following penalty function is defined for the geometric profile optimisation of section ϕ_6:

$$F_{\phi_6}(\boldsymbol{\lambda}, \rho, \beta) = f_{\phi_6}(\boldsymbol{\lambda}) + \sum_{i=1}^{4} \beta[g_i(\boldsymbol{\lambda})]^2 + \rho[h_1(\boldsymbol{\lambda})]^2 \tag{6.34}$$

where aerofoil stagger angle (λ_1) and aerofoil percentage thickness (λ_2) are the design variables taken into account with their respective inequality constraints (g_1 to g_4). The latter is similar to Equations (6.31) to (6.33). The equality constraint is defined as:

$$h_1(\boldsymbol{\lambda}) = (\Delta p - \Delta p_{req})_{fsp_r} \tag{6.35}$$

where $(\Delta p_{req})_{fsp_r}$ is the required fan static pressure such that the pseudo-3D fan can meet the requirement.

6.7 CFD Based Constrained Optimisation

The optimisation methodology was integrated into the Elemental$^{\text{TM}}$ CFD code with the pseudo 3D optimisation process starting with section ϕ_1 as follows:

1. Set initial values for the 2D aerofoil design variables.
2. The optimisation module employs the CFD module to generate the aerofoil geometry.
3. A parallel CFD run (base run) is initiated with the current design variables.

4. In parallel with item 3, multiple parallel CFD runs are initiated for the purpose of the gradient calculations (Equation (6.26)). These runs employ a base run's flow field as the initial flow field if available (this to speed up convergence).

5. The converged flow field of the base run is stored.

6. Once all the CFD runs have completed, the optimisation scheme is employed to calculate the potential optimal design variables.

7. The process is stopped if the optimal penalty function value is reached; else steps 3 to 7 are iteratively repeated.

Once an optimised aerofoil geometry at ϕ_1 is achieved, a similar process is initiated to determine the optimised aerofoil geometry at ϕ_6.

Chapter 7

Validation

This chapter validates the various modelling technologies developed and implemented in this work. The main objective of the latter being geometric optimisation of an axial fan to an optimal static efficiency. For this purpose, hybrid meshes with high growth rates are employed. The higher order discretisation scheme is specifically implemented and improved to reduce errors on such meshes. The performance of this scheme is then evaluated on a laminar flow case, as well as a representative turbulent aerofoil case for the first time. The turbulence model is itself evaluated, via comparison to experimental data. The last technology validation is that of the pseudo-3D fan modelling technique employed. This is done via experimental comparison. In addition, the number of 2D blade sections required to yield optimal accuracy is assessed.

7.1 Lid-Driven Cavity

For the 2D validation of the ETAD and ETAD_p schemes a lid-driven cavity laminar flow example with steady recirculation flow in a square (with dimensions 1×1) cavity is used (Figure 7.1). It is a classic benchmark test case (with available benchmark data) for such schemes.

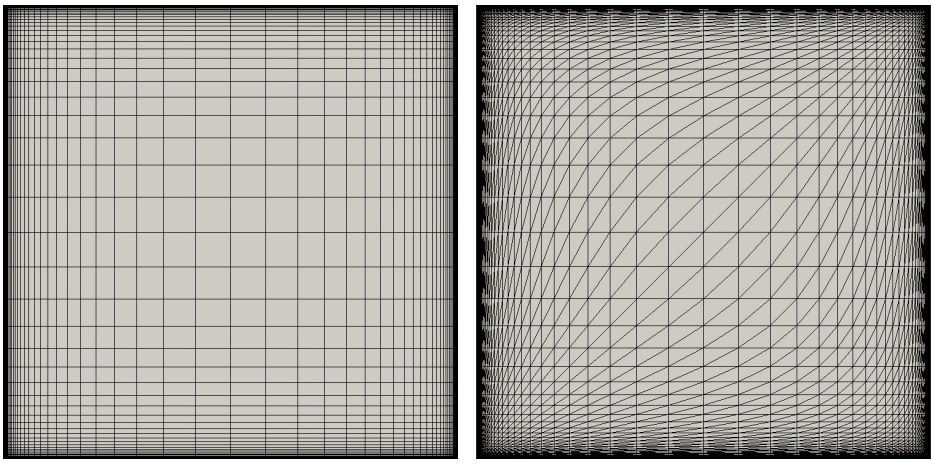

Figure 7.1: Lid-Driven Cavity structured (left) and unstructured (right) mesh 81×81 nodes

The uniform motion of the upper boundary (lid) creates the recirculated flow. The boundaries are prescribed as no-slip and the cavity is considered as a closed system with no flow entering or exiting. For the purpose of error estimation

a grid converged solution is utilised (Merrick et al. [35]). The L2-norm error calculations are done for the x-direction velocity through a vertical slice through the centre of the cavity. Validation runs were conducted with the following combinations:

- Reynolds numbers of 1×10^3, 5×10^3 and 10×10^3.
- Mesh sizes of 81×81, 95×95 and 111×111, with a constant stretching factor of 0.45 (equivalent to growth rate of 1.2).
- Anisotropic structured and unstructured meshes (Figure (7.1)).

ETAD and ETAD_p's error reductions are compared to three established schemes, namely the Cubic-Upwind-Interpolation (CUI) (Agarwal [46]), the Quadratic-Upwind-Interpolation (QUICK) and the Normalised Variable and Space Formulation (NVSF) scheme (Darwish [57], Darwish and Moukalled [58]). The first two schemes are selected due their popularity and preference for their accuracy. Both schemes, however, have a reduced order-of-accuracy on stretched meshes and therefore, the selection of the NVSF scheme which is designed to maintain order-of-accuracy on such meshes (Merrick et .al [35]).

$Re = 1 \times 10^3$					
Structured	**Scheme**				
Mesh Size	**ETAD_p**	**ETAD**	**CUI**	**QUICK**	**NVSF**
81×81 nodes	**55.99**	**62.29**	96.31	100.00	74.20
95×95 nodes	**49.06**	**63.46**	96.11	100.00	77.06
111×111 nodes	NA	**62.01**	96.32	100.00	74.15
Averaged Indices	**52.52**	**62.59**	96.25	100.00	75.14
Maximum Variation	**-3.47**	**0.87**	0.14	0.00	-1.92

$Re = 5 \times 10^3$					
Structured	**Scheme**				
Mesh Size	**ETAD_p**	**ETAD**	**CUI**	**QUICK**	**NVSF**
81×81 nodes	**49.01**	**64.61**	92.16	100.00	76.56
95×95 nodes	**48.63**	**69.41**	93.98	100.00	81.96
111×111 nodes	NA	**68.55**	92.13	100.00	76.69
Averaged Indices	**48.82**	**67.52**	92.75	100.00	78.41
Maximum Variation	**-0.19**	**-2.91**	-1.23	0.00	-3.56

$Re = 10 \times 10^3$					
Structured	**Scheme**				
Mesh Size	**ETAD_p**	**ETAD**	**CUI**	**QUICK**	**NVSF**
81×81 nodes	**57.77**	**82.40**	97.70	100.00	79.05
95×95 nodes	**60.68**	**80.41**	97.70	100.00	80.55
111×111 nodes	NA	**78.45**	97.41	100.00	79.02
Averaged Indices	**59.23**	**80.42**	97.60	100.00	79.54
Maximum Variation	**-1.46**	**1.98**	0.20	0.00	-1.01

Table 7.1: 2D Lid-Driven Cavity L2-norm % error (w.r.t. QUICK) comparison on non-equispaced structured grids.

For each Reynolds number, the schemes were tested on the three different mesh sizes and the two different mesh types. For the 111×111 node mesh

(structured and unstructured), ETAD_p was unable to converge at the various Reynolds numbers. Therefore, no data is presented for this mesh size. The L2-norm % error for each scheme is normalised to that of the QUICK scheme for comparison purposes. The relation used is:

$$\% \text{ error} = \frac{\text{error}}{\text{error}_{QUICK}} \times 100.00 \tag{7.1}$$

First consider the structured mesh results in Table 7.1. On average, the performance of CUI and QUICK is similar, with a maximum difference of 7%. Overall, NVSF performed better with an averaged error reduction between 20% and 25%, relative to QUICK. The error reduction does, however, decrease as the Reynolds number is increased. ETAD performs even better with a maximum and minimum averaged error reduction of 37% and 20% (relative to QUICK) at the lowest and highest Reynolds number respectively. The error reduction trend, as a function of the Reynolds number, is similar to that of NVSF. This can be expected due to the incorporation of the latter in the ETAD's blending function. The most superior scheme is clearly ETAD_p. The increase in the Reynolds numbers also has a smaller effect on the error reduction capability when compared to ETAD. The averaged error reduction achieved varies between 41% and 51%. The latter value is more than double that of that achieved by ETAD and NVSF.

$\boldsymbol{Re = 1 \times 10^3}$					
Unstructured Mesh Size	**Scheme**				
	ETAD_p	**ETAD**	**CUI**	**QUICK**	**NVSF**
81×81 nodes	**59.22**	**74.49**	108.26	100.00	78.20
95×95 nodes	**49.04**	**69.29**	107.14	100.00	81.10
111×111 nodes	NA	**71.65**	107.56	100.00	79.86
Averaged Indices	**54.13**	**71.81**	107.65	100.00	79.72
Maximum Variation	**5.09**	**2.68**	-0.61	0.00	1.52

$\boldsymbol{Re = 5 \times 10^3}$					
Unstructured Mesh Size	**Scheme**				
	ETAD_p	**ETAD**	**CUI**	**QUICK**	**NVSF**
81×81 nodes	**37.34**	**45.07**	112.57	100.00	93.13
95×95 nodes	**61.61**	**51.94**	114.53	100.00	105.23
111×111 nodes	NA	**44.51**	112.12	100.00	92.85
Averaged Indices	**49.47**	**47.17**	113.08	100.00	97.07
Maximum Variation	**-12.13**	**4.77**	-1.46	0.00	-8.16

$\boldsymbol{Re = 10 \times 10^3}$					
Unstructured Mesh Size	**Scheme**				
	ETAD_p	**ETAD**	**CUI**	**QUICK**	**NVSF**
81×81 nodes	**49.25**	**49.20**	106.37	100.00	94.31
95×95 nodes	**52.24**	**50.97**	99.40	100.00	97.37
111×111 nodes	NA	**43.67**	107.18	100.00	94.30
Averaged Indices	**50.74**	**47.95**	104.23	100.00	95.32
Maximum Variation	**-1.49**	**-4.28**	4.92	0.00	-2.04

Table 7.2: 2D Lid-Driven Cavity L2-norm % error (w.r.t. QUICK) comparison on non-equispaced unstructured grids.

Next consider the unstructured mesh results in Table 7.2. The difference between the averaged results of CUI and QUICK vary between 4% and 13%. Compared to the structured results these values have doubled. At a Reynolds number of 1×10^3, NVSF achieved a similar averaged error reduction compared to error reduction achieved on the structured mesh. This is, however, not the case as the Reynolds number is increased. On the unstructured mesh NVSF performed similar to QUICK and it was not able to achieve similar error reductions to that achieved on the structured mesh ($Re = 5 \times 10^3$ and $Re = 10 \times 10^3$). ETAD performs better than NVSF with a maximum and minimum error reduction of 53% and 28% respectively, relative to QUICK. The averaged error reduction achieved is 9% less at the lowest Reynolds number and more then double at the highest Reynolds number when compared to the error reductions achieved on the structured mesh. The superior scheme resulting in significant error reductions at all the Reynolds numbers is again ETAD_p. The averaged error reductions achieved are similar to the reductions achieved on the structured meshes ($Re = 1 \times 10^3$ and $Re = 5 \times 10^3$) with an error reduction improvement of 8% at the highest Reynolds. ETAD_p is thus clearly the all-round superior method.

7.2 NACA0018 Aerofoil

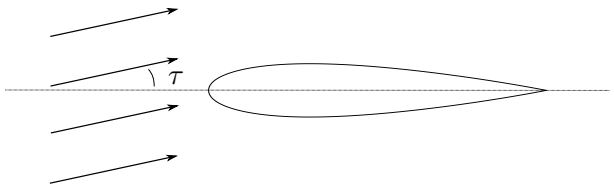

Figure 7.2: NACA0018 aerofoil

In this section, a 2D NACA0018 aerofoil case (Figure 7.2) is employed for the purpose of validating the implemented Spalart-Allmaras turbulence model, as well as the performance of the higher order discretisation schemes. The angle of attack and Reynolds number is chosen as $\tau = 10^\circ$ and 500×10^3, respectively. The latter is similar to the Reynolds number encountered in the 2D cascades over the blade span. For the fan under consideration in this work, the Reynolds number varies from 300×10^3 to 400×10^3 (root to tip respectively). The angle of attack was selected as it results in a trailing edge separation bubble, which is also present on fan rotor blades.

The flow domain consists of a far field boundary (Figure 7.3) as well as no-slip condition on the aerofoil surface. Figure 7.4 shows the employed anisotropic hybrid (quadrilateral and triangular elements) 2D mesh. Three mesh sizes are utilised (54×10^3, 78×10^3 and 106×10^3 elements) with aerofoil average mesh spacings of 8.65×10^{-3}, 4.32×10^{-3} and 3.45×10^{-3} respectively. The steady state, incompressible, and turbulent flow field is solved with the Elemental^{TM} CFD software.

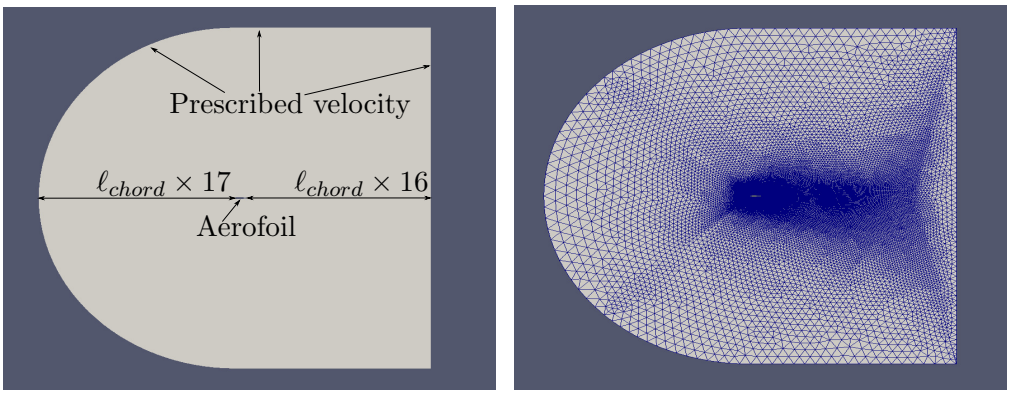

Figure 7.3: NACA0018 aerofoil flow domain (left) and global mesh with 106×10^3 elements (right)

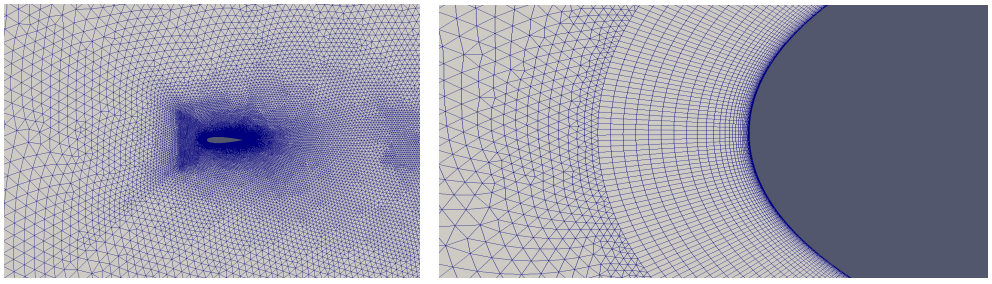

Figure 7.4: NACA0018 aerofoil flow domain (106×10^3 elements mesh) zoomed in (left) and leading edge boundary layer (right)

7.2.1 Turbulence model

In this section, extrapolated values of the predicted drag and lift coefficients will be compared to experimental results (Timmer [59]). All three meshes are employed for extrapolation purposes to ensure mesh independent data are used in the comparison. The drag and lift coefficients at a zero mesh spacing is predicted, using the Richardson Extrapolation [60] as follows:

$$f_{exrapolated} = f_1 + \frac{(f_1 - f_2)}{(r^p - 1)} \tag{7.2}$$

where f_1 and f_2 are the solutions of the fine and coarse mesh respectively. Further r and subscript p are the grid refinement ratio and order of solution. The grid refinement ratio is defined as:

$$r = \frac{m_{nodes_1}}{m_{nodes_2}} \tag{7.3}$$

where m_{nodes_1} and m_{nodes_2} are the number of nodes on the aerofoil in the fine and coarse mesh respectively. These variables are represented by the inverse of the aerofoil averaged mesh spacing. The order of the solution is defined as:

$$p^{i+1} = wp^i + (1 - w)\frac{\ln \theta}{\ln r} \tag{7.4}$$

where $w = 0.5$ and i are the relaxation factor and the iteration number respectively. The initial order of the solution at $i = 0$ is set to $p^0 = 2$. The variable θ is defined as:

$$\theta = \frac{\left(r_{12}^{p^i} - 1\right)\epsilon_{23}}{\left(r_{23}^{p^i} - 1\right)\epsilon_{12}} \tag{7.5}$$

where

$$\epsilon_{12} = f_1 - f_2 \tag{7.6}$$

$$\epsilon_{23} = f_2 - f_3 \tag{7.7}$$

The certainty level of the predicted value is quantified with a grid-convergence-index (GCI) which is defined as:

$$GCI = F_s \left| \frac{f_2 - f_1}{f_1} \right| \frac{1}{(r^p - 1)} \tag{7.8}$$

where F_s is the safety factor and is set to 1.25.

The predicted drag and lift coefficients (via Richardson Extrapolation) are now compared to the experimental results of Timmer [59]. The percentage difference (error) between these values is shown in Table 7.3 with each respective GCI. The CUI discretisation scheme is employed in the CFD. The error in the predicted drag and lift coefficients is found to be -32.78% and -0.41% respectively. Compared to the extensive numerical results in the drag and lift coefficient predictions by Hassan et al. [61], the drag coefficient error found in this work should not be surprising. The author employs various, more complex, two-equation turbulence models. The net result is an error in drag prediction of up to 436%. In addition, it is also well known that RANS, as used in this work, is typically less accurate in predicting drag coefficients.

Coefficient	$\boldsymbol{GCI_{23}}$	$\boldsymbol{GCI_{12}}$	**Error**
C_d	0.022%	0.0053%	-32.78%
C_L	0.162%	0.116%	-0.41%

Table 7.3: Summary of the GCI and error

7.2.2 Higher Order Discretisation

Having validated ETAD_p for a laminar flow case, both ETAD and ETAD_p are now applied to modelling turbulent flow. The predicted drag and lift coefficients are compared to the experimental results of Timmer [59]. The predicted data obtained with the finest mesh (Figures 7.5) consisting of 106×10^3 was used.

It must be noted that the standard ETAD and ETAD_p were not stable for the viscous turbulent cases. It was, however, remedied when using standard face interpolation values for velocity and pressure instead of Equation (5.53). When reference is made to the ETAD and ETAD_p scheme in this section, it is thus without these higher order face approximation.

The resolved velocity magnitude and pressure contours are shown in Figure 7.6. The air approaching the aerofoil slows down and speeds up at the bottom and top of the leading edge. The static pressure follows suit by increasing and decreasing at the bottom and top of the leading edge.

The errors in the predicted drag and lift coefficients are calculated for ETAD_p, ETAD, CUI, NVSF, QUICK and schemes and presented in Table 7.4. QUICK is used as the benchmark error. ETAD performed well by out-performing all

Coefficient	Scheme				
	ETAD$_p$	**ET**	**CUI**	**NVSF**	**QUICK**
C_d	99.64%	99.74%	98.63%	99.85%	100.00%
C_L	97.69%	85.53%	137.02%	119.52%	100.00%

Table 7.4: NACA0018: error comparison of drag and lift coefficient.

the other schemes in the prediction of the lift coefficient. An error reduction of 14.47% is achieved. ETAD$_p$, however, did not perform as well as expected, resulting in an error reduction of 2.31% in the lift coefficient prediction. It is also noted that the error improvement of the ETAD scheme for the viscous turbulent case is not in the same order as the error improvements in the lid-driven cavity cases. Neither ETAD nor ETAD$_p$ significantly improved the drag coefficient prediction, which is probably due to the method not being applied to the turbulent viscosity equation. First order was employed for the latter in all cases. All the schemes therefore performed similarly in predicting the drag coefficient.

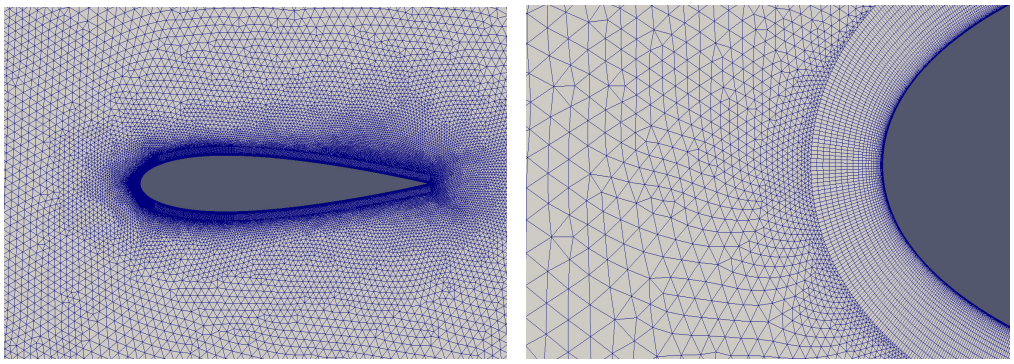

Figure 7.5: NACA0018 aerofoil (left) and leading edge (right) mesh with 106×10^3 elements.

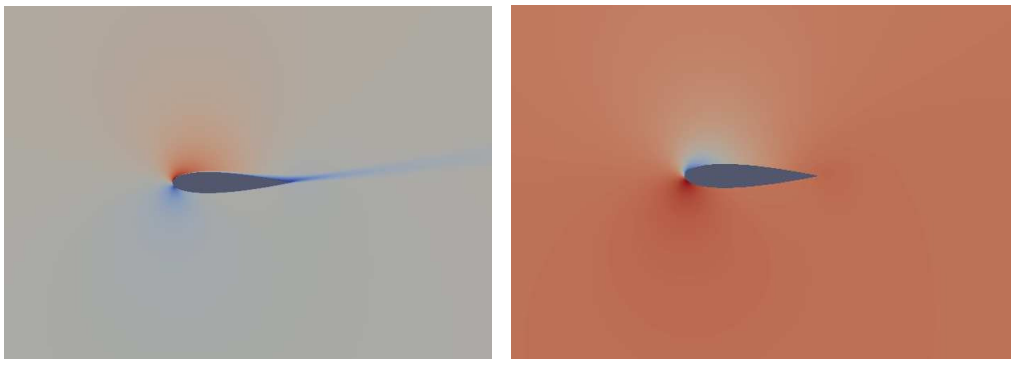

Figure 7.6: NACA0018 aerofoil velocity magnitude(left) and static pressure (right) contours for 106×10^3 element mesh.

7.3 Fan Modelling Technique

The purpose of this section is to first determine the minimum number of 2D cascades required in a pseudo-3D fan model. This is done with four pseudo-3D models, consisting of six, four, three and two 2D cascades, respectively. The

predicted fan static pressure and shaft power is then first compared to a 3D CFD fan model and experimental results at the required volumetric flow rate. Secondly the predicted (via minimum 2D cascades) fan static pressure rise, shaft power and static efficiency is compared to experimental results over a range of volumetric flows. Fan-Optim (design to be discussed in Chapter 8) was utilised for this purpose and was manufactured and tested in accordance to the ISO 5801 standard by CFW.

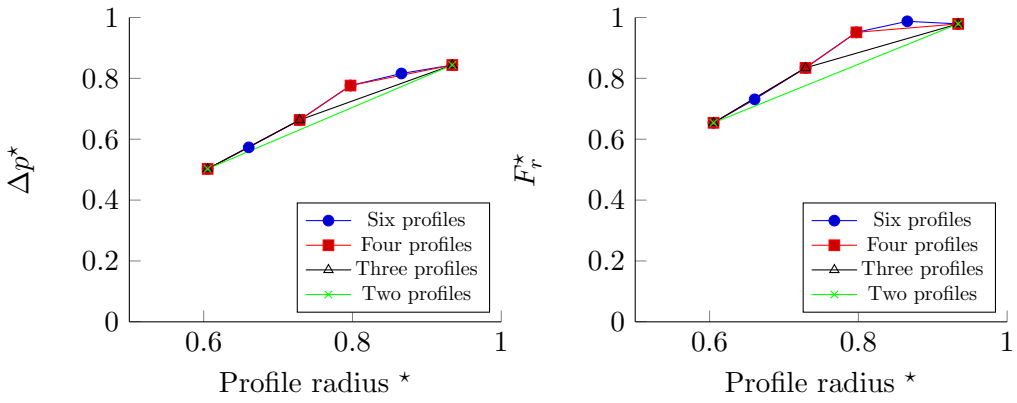

Figure 7.7: Fan-Optim: static pressure rise (left) and aerofoil force (right) vs profile radius

Figure 7.7 shows that two cascades lead to a pseudo-3D model with a linear static pressure and tangential force distribution over the blade span. The distribution is shown to become non-linear as the number of cascades is increased to six. Clearly four profiles yield results similar to that if using six cascades (within 0.1%), and are employed for the remainder of the study. This is also reflected in Figure 7.8, which shows the predicted fan static pressure rise and shaft power as a function of number of profiles.

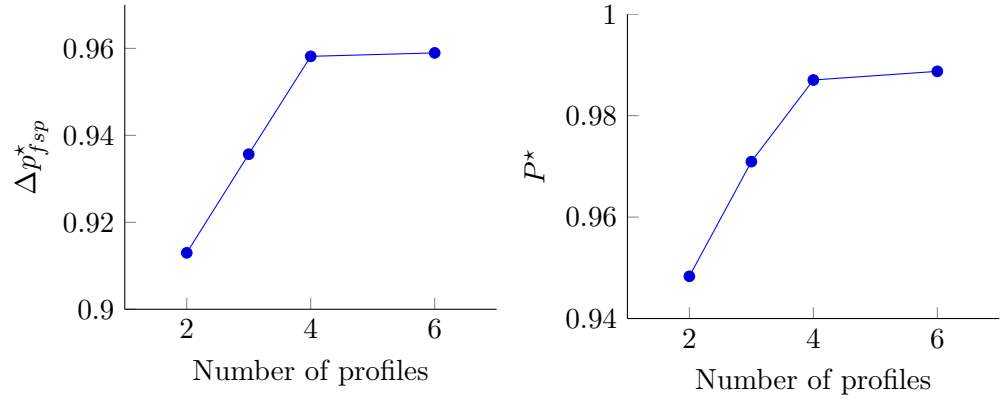

Figure 7.8: Fan-Optim: fan static pressure rise (left) and shaft power (right) vs number of profiles

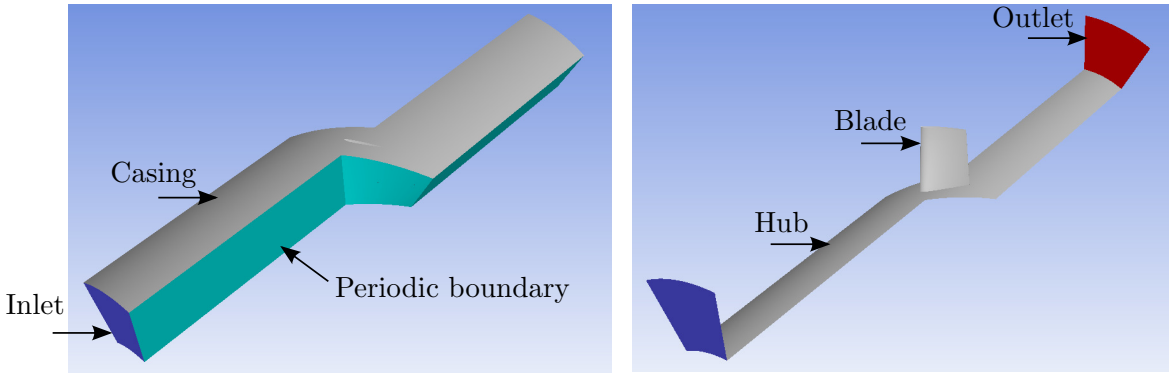

Figure 7.9: 3D flow domain

The pseudo-3D predicted values (using four cascades) are next compared to a 3D CFD model's predicted values, as well as experimental results. The 3D model (Figure 7.9) is based on the 3D domain that is used to generate the pseudo-3D model. This consists of a 3D rotor (Figure 3.1) and an annular inlet and outlet flow domain, similar to le Roux et al. [9] and Augustyn et al. [11]. The upstream and downstream section lengths are similar to the pseudo-3D models lengths. Considering the 3D CFD simulation, it would be costly to model the entire 3D rotor. Instead, the domain is reduced to a single rotor blade, while accounting for the blade-blade interaction via the employment of a rotational periodic boundary condition (Figure 7.9). The width of the domain at a specific radius originates from the rotational repeat. The periodic boundary faces are generated in a similar fashion to the method used in the 2D CFD modelling. A no-slip boundary condition is prescribed to the rotor blade surface, with slip conditions prescribed to the remaining boundaries. A mass flow inlet is prescribed along with a static pressure outlet with radial equilibrium (le Roux et al. [9]). The 3D CFD model consists of a single rotating reference (Augustyn et al. [11]), where the fluid domain and blade rotational speed are prescribed and calculated with Equation (2.9).

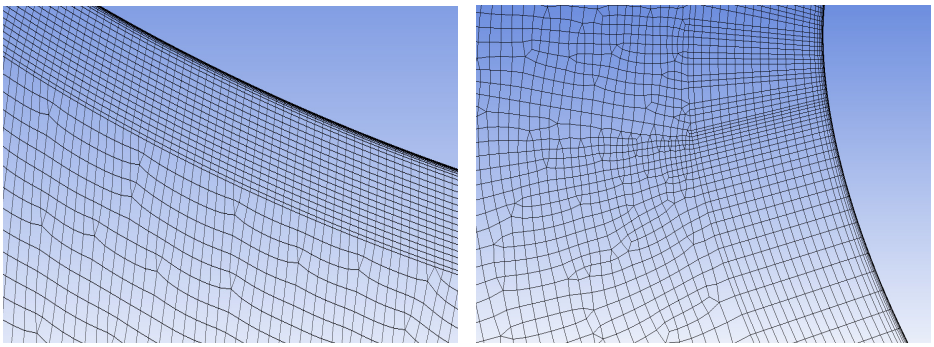

Figure 7.10: 3D flow domain boundary layer mesh

ICEM was used to generate an anisotropic hybrid mesh, comprising of structured hexahedron elements in the boundary layer around the rotor blade with unstructured pyramid, tetrahedron and hexahedron elements in the rest of the domain (Figure 7.10). There are, in total, 3×10^6 elements in the mesh, which is similar to the mesh size used by le Roux et al. [9] and Augustyn et al. [11].

Commercial CFD software is employed to solve the steady state, incompressible and viscous air flow in the absolute reference frame. Instead of employing the Spalart-Allmaras turbulence model, the $k-\epsilon$ turbulence model with an enhanced

wall treatment is employed to cater for the 3D transitional flow (Augustyn et al. [11]). In addition, Augustyn et al. [11] also found that Spalart-Allmaras requires more mesh elements for convergences of the CFD model. The Simple pressure-velocity coupling, combined with QUICK and 2^{nd} order spatial discretisation schemes for momentum and pressure, were selected.

In the above analysis, pressure loss due to boundary layer effects on the hub and casing surfaces were not accounted for at neither the inlet nor the outlet sections. Consequently, an additional analysis was conducted only of these sections of the domain. The geometry is depicted in Figure 7.11, where pressure losses due to the inlet and outlet sections are modelled separately. For the inlet CFD model, the casing and hub boundaries are prescribed as no-slip and slip walls respectively. Mass flow rate is prescribed at the inlet boundary, with static pressure (radial equilibrium) being specified at the outlet. For the outlet CFD model, a no-slip wall boundary condition is prescribed on the casing and hub boundaries. The flow at the inlet is set to rotate at a rotational speed originating from the resulting flow field of the 3D CFD fan model. The ICEM generated meshes, containing boundary layers at the hub and casing, consists of 1.0×10^6 elements each (Figure 7.12).

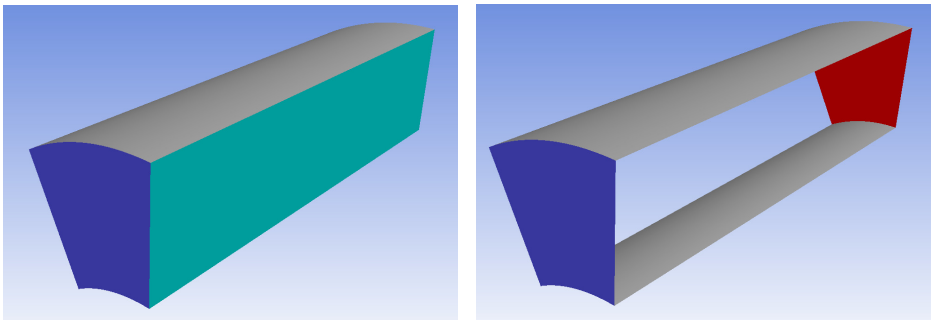

Figure 7.11: 3D annular flow domain

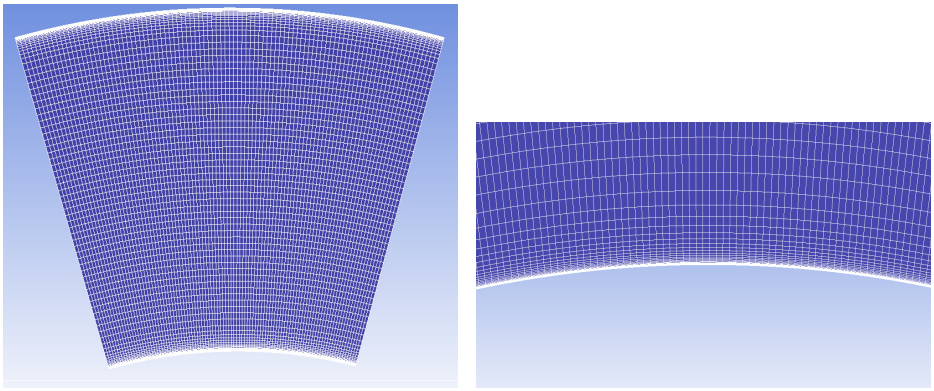

Figure 7.12: Mesh of 3D annular flow domain

The fan static pressure rise and shaft power at the duty point of interest are next compared. That is, we compare the values as obtained via the three methods viz. pseudo-3D (using four cascades) and 3D CFD (including inlet and outlet sections) are compared to experimental data. The findings are summarised in Table 7.5. Note that the pseudo-3D calculated static pressure rise and power are to within 11.1% and 1.7% respectively, of the 3D model. The proposed pseudo-3D model is similarly within 13% and 15% respectively, of the experimental

data. The over-prediction of the fan static pressure rise and power compared to experimental data can be expected. The reason being that the CFD models do not account for all the losses in the system. Blade tip clearance, the effect of the blade root interface with the hub and the electrical motor brackets are for example, not included in the CFD. Furthermore, the computational cost of the 3D CFD model is of the order of 10 times more expensive then the pseudo-3D model (using four cascades). It is also 20 times more expensive then the pseudo-3D model (using two cascades) which is used in the optimisation methodology. These values are based on CFD models being solved on an Intel Core $i7 - 2760QM$ (CPU @ $2.40GHz \times 8$ with $8GB$ of memory) computer. These results clearly indicate the use of the new proposed model as an efficient new alternative for fan optimisation.

Description	**Performance indicators**		**Model Variation,** %	
	$p^{\star}_{fsp}$	$P^{\star}$	p_{fsp}	P
Experimental	1.0	1.0	NA	NA
3D CFD model	1.02	1.14	0.0	0.0
Pseudo-3D model	1.13	1.15	11.1	1.7

Table 7.5: Fan Optim: comparison of predicted fan static pressure rise and shaft power at the duty point

The experimental data are now compared to the predicted results of a pseudo-3D fan comprising of four cascades over a range of volumetric flow rates. The comparison starts out at a volumetric flow rate higher than the required duty point, thereafter the flow rate is reduced until stall or an unsteady state flow regime is reached. The latter is determined at the point at which the 2D steady state CFD runs no longer converge.

The predicted fan static pressure rise, power and static efficiency compare well to the experimental data (Figure 7.13). The fan's performance is predicted up to a volumetric flow rate that is relatively close to the maximum fan static pressure rise. The performance at lower volumetric flow rates could not be predicted due to the steady state solver not being able to solve the 2D cascades flow field. This is the region in which much of the fan blade goes into stall and occurs due to a large difference between the air flow inlet angle and the blade stagger angle. At the duty point ($Q^{\star} = 0.75$), the predicted fan static pressure rise, shaft power and static efficiency differ by 13%, 15.5% and 2.2% respectively, from the experimental values. The difference in fan static pressure rise and shaft power are relatively similar, effectively cancelling each other out in the efficiency calculation.

In all of the CFD solutions in this work the convergence algorithm (Section 3.3) is employed to stop the iterative computational solution process. For the purpose of validating this algorithm, the flow field around an aerofoil at $\phi^{\star} = 0.61$ of Fan-D is considered at a normalised stagger angle of 0.875. The tangential force $F^{\star}_{x_r}$ value is selected as the monitor variable. The monitoring function retrieves data every 2×10^3 solver iterations and the history of five data points is employed to assess convergence. The convergence tolerance is set to 0.5%.

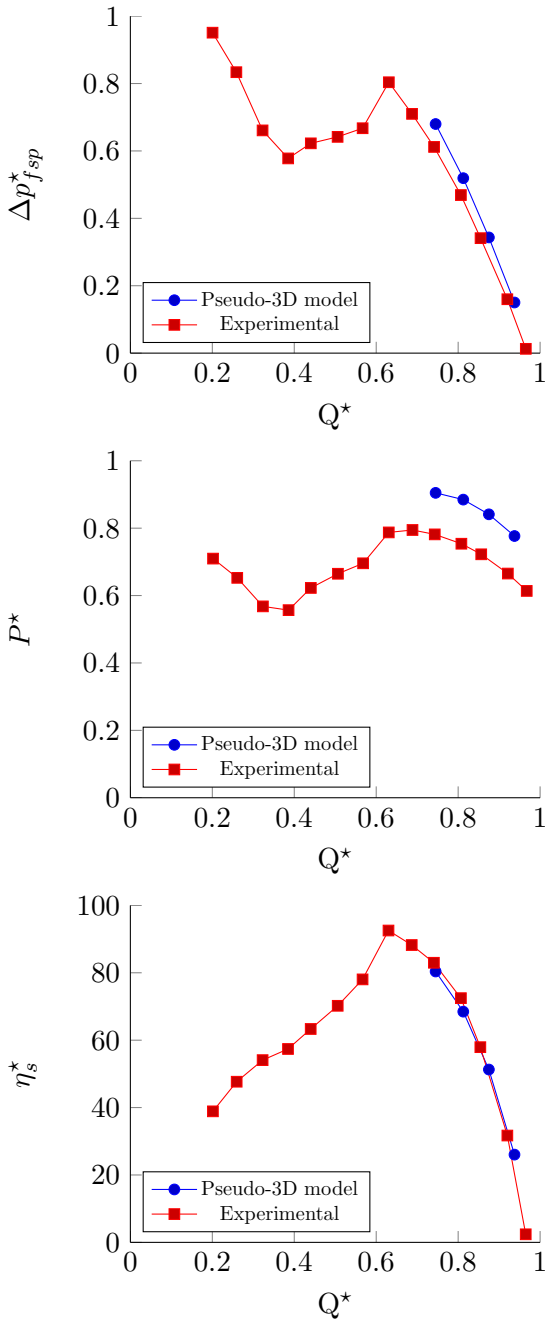

Figure 7.13: Fan-Optim: fan static pressure (top), shaft power (middle) and fan static efficiency (bottom) vs volumetric flow rate

Figure 7.14 shows the convergence history with metrics of interest. Initially, the values are very far from the specified tolerance. After 34×10^3 iterations only the minimum value is within the tolerance. It is only at 36×10^3 iterations that both the minimum and maximum values are within the specified tolerance. At this point the CFD solver will be terminated. From iteration number 34×10^3 to 36×10^3 the value of the tangential force changed by 0.003%. This indicates a converged solution has been reached at 36×10^3 iterations and proves that the monitoring function does stop the solver at the correct point. Upon investigation it was found that the static pressure rise was shown to have a similar trend to the tangential force. Further solver iterations up to 150×10^3 (not shown in figures) resulted in a negligible change in the tangential force and static pressure

rise.

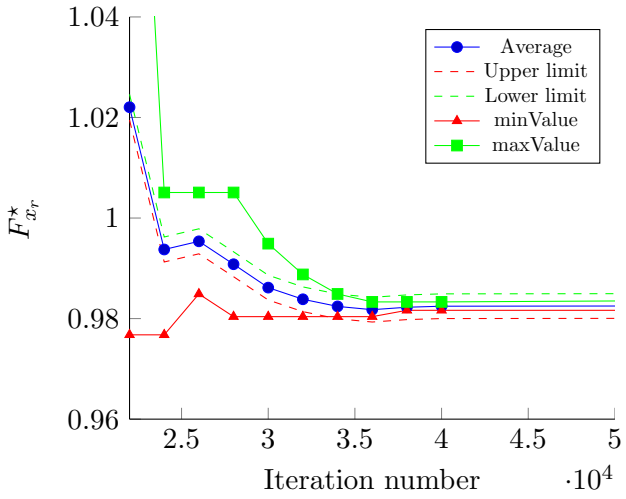

Figure 7.14: Average $F^{\star}_{x_r}$ of Fan-D at $\phi^{\star} = 0.61$ vs iteration number

Chapter 8

Application

Having developed the several technologies required for fan aerodynamic optimisation, this chapter investigates its application and evaluation. The fan duty point is as prescribed by CFW, while a base fan (Fan-D) is selected as starting point. Fan-D's aerofoil geometry is based on a design carried out for CFW for another application, while the stagger angle is specifically ill-chosen to assess the robustness of the developed optimisation technology. The fan will first be modelled to demonstrate its inadequate fan static pressure rise capability at the required volumetric flow rate. This will include the value at the default blade stagger angle (used as starting point in the optimisation process), as well as the maximum value that the fan is capable of producing. The latter will be determined by manual blade stagger angle perturbations (no blade geometric twist or 2D aerofoil alterations) in the fan model. This process will emulate the typical resource-intensive process followed in an experimental set-up with the aim of increasing fan static pressure rise (the blade angle is adjusted manually until the desired performance is achieved). Thereafter the developed modelling technology is applied and the optimisation process conducted to create Fan-Optim using Fan-D as starting point (with original ill-chosen stagger angle). Fan-Optim is finally built and experimentally tested. The final evaluation involves comparison of Fan-Optim's predicted performance data to the requirement, Fan-D and the experimentally measured performance data.

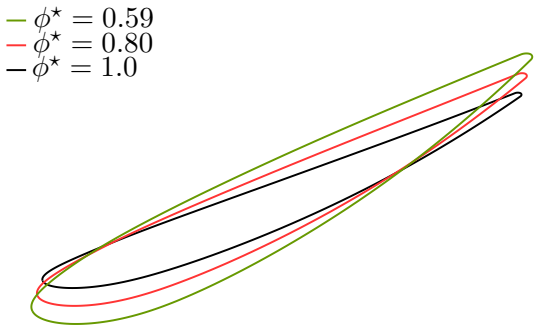

Figure 8.1: Fan-D aerofoils

8.1 Base Fan Evaluation

Fan-D's rotor blade geometry is defined by a mathematically defined 2D aerofoil. The latter varies linearly as a function of the radius from the hub to the tip. Three 2D sections are presented in Figure 8.1, where the section number corresponds to the normalised diameter (normalised to blade diameter). The performance of this fan is now predicted by a pseudo-3D fan model comprising of four profiles.

The predicted fan static pressure rise at the specified volumetric flow rate is presented in Figure 8.2. At the default normalised stagger angle ($\lambda^\star = 0$) the predicted fan static pressure rise is 13.8% below the required specification.

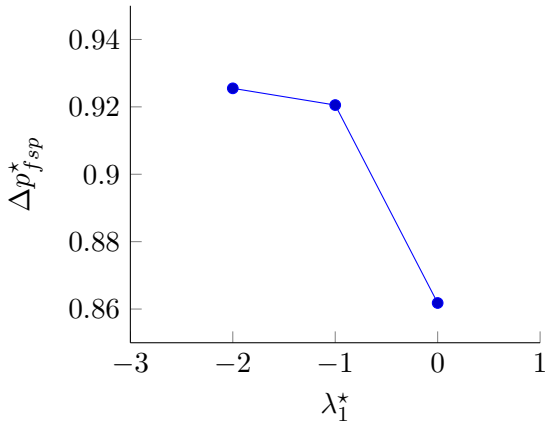

Figure 8.2: Fan-D fan static pressure rise vs profile stagger angle normalised

In an effort to increase the fan static pressure rise such that the requirement is met, the fan rotor blade stagger angle is manually adjusted. This is the only variable that can be altered without changing the aerofoil geometry (blade twist or 2D aerofoil). The analytical representation of static pressure rise in a 2D cascade (Figure 8.3) is utilised to determine if the angle should manually be increased or decreased, and is calculated as follows:

$$\Delta p_{r_{th}} = \frac{1}{2}\rho(v_1^2 - v_2^2) \tag{8.1}$$

where $\Delta p_{r_{th}}$, v_1 and v_2 represents the theoretical static pressure rise, the inlet and outlet relative velocity vectors (Figure 8.3).

Noting that the theoretical static pressure rise is positive and that the inlet relative velocity is specified and constant. The only remaining variable that can be adjusted is the outlet relative velocity. A decrease of the latter variable will result in an increase in the theoretical static pressure rise. This can be achieved by decreasing the angle between v_2 and v_{2_y} (Figure 8.3), which implies that a decrease in the fan rotor blade stagger angle (Figure 1.1) is required.

Figure 8.2 shows how the predicted fan static pressure rise increased as the blade stagger angle is decreased. The decrease does improve the fan static pressure rise, but it is still 7.5% below that required. In this process the stagger angle was decreased up to the point that the pseudo-3D model's CFD models failed to converge. The instability was found at the 2D cascade radially closest to the tip. At this diameter, the 2D CFD diverged when the stagger angle was decreased any further. To gain further insight, the velocity contours of Fan-D's aerofoils (default and minimum stagger angles) are presented in Figure 8.4. At the default stagger angle of 0.0° there is flow break away that starts to form at

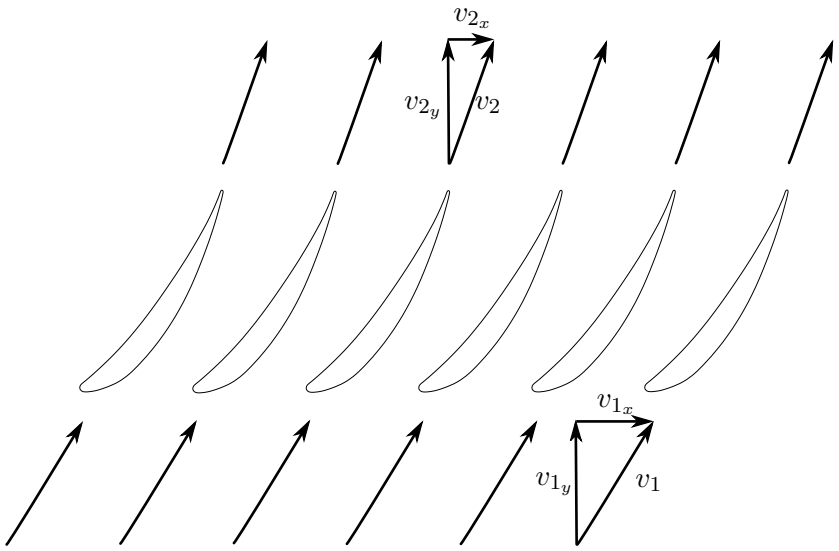

Figure 8.3: Relative velocity vectors in 2D flow domain

the trailing edge of the aerofoil. It increases in size in the direction towards the blade tip. This is an indication that Fan-D does not have enough blade twist in the radial direction and that a larger stagger angle is required as one moves radially outwards. In the process of attempting to increase the fan static pressure, the blade stagger angle was reduced up to 2.0°, which has the required effect. However, this unfortunately increases the flow separation region (as one moves towards the blade tip) at the trailing edge due to the increase in the relative angle between the inlet flow field and the aerofoil.

In summary, Fan-D's fan static efficiency is $\eta^\star = 62.3$, with a maximum fan static pressure rise value of $\Delta p^\star_{fsp} = 0.93$, which is 7.5% below that required.

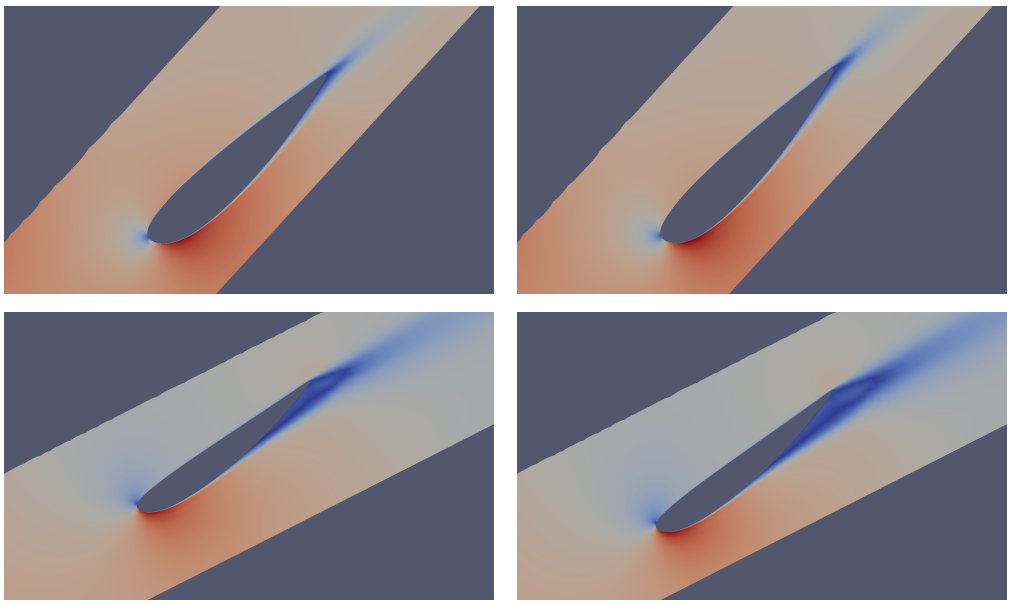

Figure 8.4: Fan-D 2D cascades at $\phi^\star = 0.610$ and 0.93 (from top to bottom respectively) relative velocity contours with stagger angles of 0.0° (left) and $-2.0°$(right)

8.2 Axial Fan Blade Geometric Optimisation

Fan-D is now geometrically optimised to ensure that the newly generated fan meets the required fan static pressure rise at the specified volumetric flow rate with a high efficiency. The procedure is started with the fan rotor blade set to the default stagger angle of 0.0°. As discussed in Chapter 6, the optimisation scheme is now applied to a pseudo-3D fan model consisting of two 2D aerofoils (ϕ_1 and ϕ_6).

Inner section ϕ_1

The aerofoil of Fan-D's rotor at ϕ_1 is now geometrically optimised by minimising the penalty function for the specific section. The penalty function contains the objective function and inequality constraints. The former is a function of the aerofoil stagger angle (λ_1) and thickness (λ_2); the latter contains the lower and upper limits for these design variables.

When the SQSD algorithm is applied to the penalty function the design variables are automatically perturbed, as presented in Figure 8.5. The penalty function is dramatically reduced in the first seven iterations. At this stage the stagger angle is close to the optimal value at the specific thickness. From iteration 8 to 10 the stagger angle is further refined while the thickness remains almost unchanged. The reason for this is illustrated in Figure 8.6, i.e. it is only when the stagger angle is close to the optimal value at iteration 11 that the reduction (iteration number 11 to 21) in thickness has a significant effect on the penalty function value. Further iterations result in refinement of both the design variables, with the thickness reaching the lower limit setting at iteration numbers 21, 28, 34 and 38. This can be seen from the jump in the penalty function due to the inequality penalty taking effect (Figure 8.5). The optimisation algorithm finally converges at iteration number 40, as shown by the objective function (Figure 8.6).

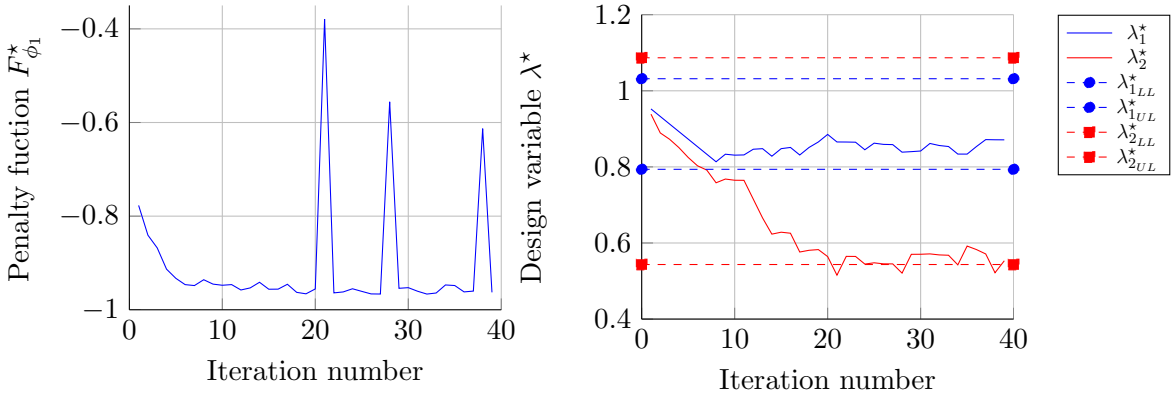

Figure 8.5: Fan-Optim diameter ϕ_1 penalty function (left) and design variables (right) vs iteration number

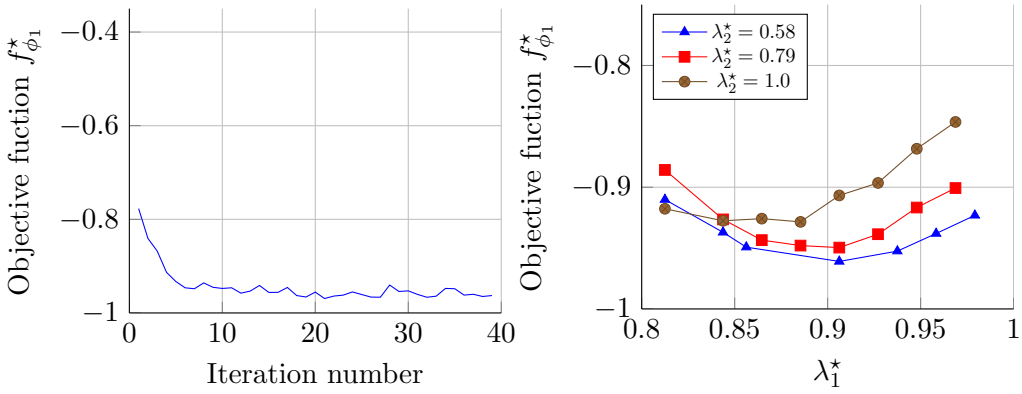

Figure 8.6: Fan-Optim diameter ϕ_1 objective function vs iteration number (left) and vs stagger angle and aerofoil thickness (right)

Outer section ϕ_6

The aerofoil of Fan-D at ϕ_6 is next geometrically optimised by minimising the penalty function for the specific section. The penalty function contains the objective function, inequality constraints and an equality constraint. Stagger angle and blade thickness are the design variables considered, each with their respective upper and lower limit constraints. The static pressure rise requirement is the equality constraint applied to the specific aerofoil. The value of which is governed by the static pressure rise achieved at ϕ_6 and the required fan static pressure rise of the pseudo-3D model.

When the SQSD algorithm is applied to the penalty function, the design variables are automatically perturbed, as presented in Figure 8.7. The penalty function converges after 22 iterations, primarily due to the increase in stagger angle, whereas the thickness reduced marginally. It should be noted that none of the design variables' upper and lower limits had an effect on the penalty function. After 22 iterations, the normalised objective function is improved by 12.6% from −0.867 to −0.976 (Figure 8.8). The equality constraint started 6.1% from that required and ended at 2.9% away. The improvement in the objective function is mainly due to a 9.5% change in the aerofoil stagger angle from 0.885 to 0.969. The −6% change in the aerofoil thickness from 0.933 to 0.877 has a smaller contribution to the objective function improvement. The reason for this is due to its small effect on the objective function compared to the aerofoil stagger angle, as presented in Figure 8.8.

To accentuate the contribution of the thickness, the optimisation process was resumed using iteration 22 as the initial condition. In addition, the process was commenced by applying the normalisation step which accentuated the effect of aerofoil thickness. It should be noted that this could be done here as the stagger angle was near optimal. A converged penalty function is finally achieved at iteration 86, resulting in an objective function change from −0.976 to −0.985 (Figure 8.8), equating to a 0.9% improvement in the fan static efficiency. The equality constraint started 2.9% from that required and ended up within 6%. The aerofoil stagger angle is marginally refined with a 0.4% change from 0.969 to 0.973. The aerofoil thickness changed by −28%, from 0.877 to 0.634. No further optimisation was carried out as the aerofoil thickness is already close to the lower limit and any further marginal reductions (from the current to the lower limit value) would not lead to a significant penalty function improvement.

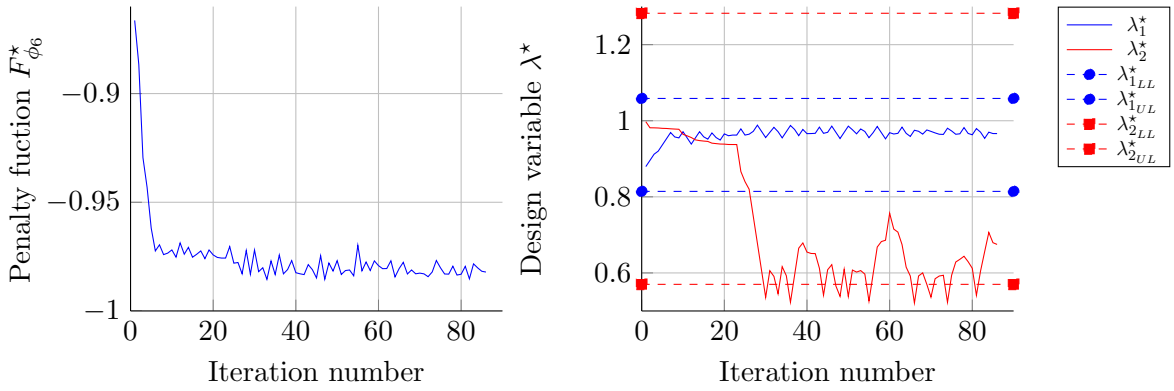

Figure 8.7: Fan-Optim diameter ϕ_6 penalty function (left) design variables (right) vs iteration number

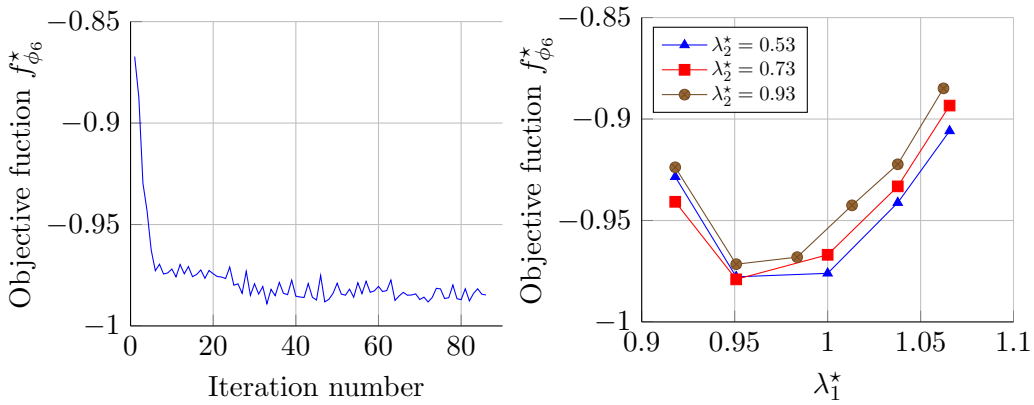

Figure 8.8: Fan-Optim diameter ϕ_6 normalised objective function vs stagger angle ($\lambda^\star_1$) and aerofoil thickness ($\lambda^\star_2$)

Optimised 3D Fan Performance

The optimised aerofoils at ϕ_1 and ϕ_6 are now employed to generate Fan-Optim's rotor blade 3D geometry. These two aerofoils are aligned by utilising their area centroids. The geometry between these two profiles is created via linear interpolation. Linear extrapolation is employed from ϕ_h to ϕ_1 and ϕ_6 to ϕ_t. 2D CFD is then run on these profiles to establish the pseudo-3D fan's performance.

Description	**Performance indicators**			**Variation, %**	
	$p^\star_{fsp}$	$P^\star$	η_s	p_{fsp}	P
Specification	1.0	1.0	100.0	0.0	0.0
Fan-D	0.86	0.95	90.4	-14.0	-5.0
Fan-D $(p_{fsp_{max}})$	0.93	1.01	92.9	-7.0	1.0
Fan-Optim	1.02	0.99	102.7	2.0	-0.8

Table 8.1: Pseudo-3D fan performance prediction of Fan-D and Fan-Optim

The pseudo-3D fan's (consisting of four cascades) normalised predicted performance at the specified volumetric flow rate is presented in Table 8.1. Fan-D's (default rotor blade stagger angle setting) fan static pressure rise prediction is

14% below that required, with a value of 0.86. A normalised static efficiency of 90.4% is achieved. The maximum predicted fan static pressure rise is 7% below, with a value of 0.93. A normalised static efficiency of 92.9% is achieved. The maximum fan static pressure rise is achieved via manual stagger angle perturbations, with the conclusion that the fan cannot meet the requirement. Fan-Optim's predicted fan static pressure rise is within 2% of the requirement, with a value of 1.02. This is 18.6% higher than the under-performing Fan-D pressure rise. A normalised static efficiency of 102.7% is achieved by Fan-Optim which is in turn, 13.6% higher than Fan-D's efficiency. The predicted absorbed fan power is 1% below that required. It is also interesting to note that Fan-Optim has an increased static efficiency and it meets the required fan static pressure with approximately 38% less material compared to Fan-D (starting point). This was achieved as a secondary effect in the optimisation process.

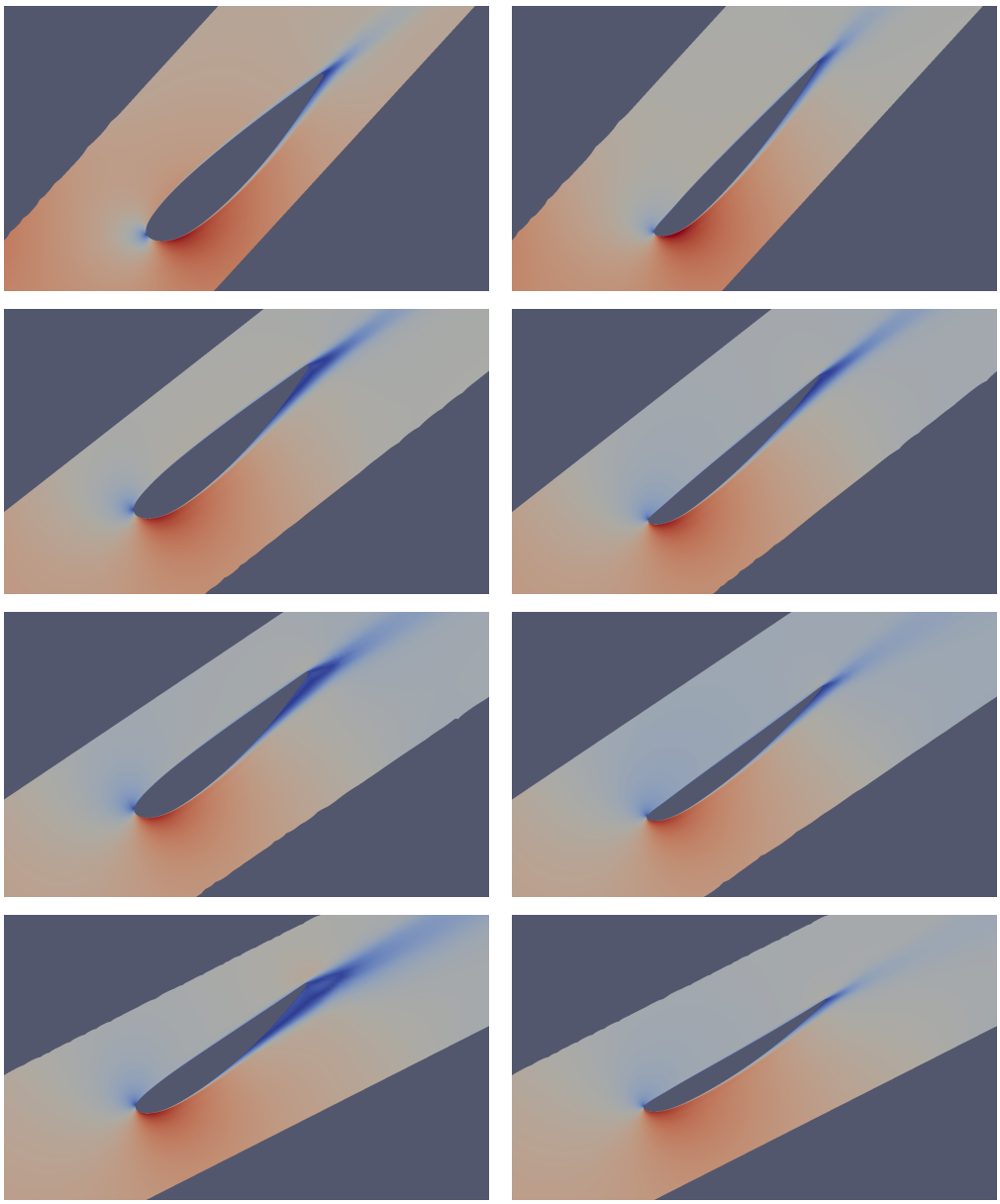

Figure 8.9: 2D cascades at $\phi^\star = 0.610$, 0.73, 0.80, and 0.93 (from top to bottom respectively) relative velocity contours of Fan-D (left) and Fan-Optim (right)

The following general observations are made by comparing the velocity (Figure 8.9) and static pressure (Figure 8.10) contour plots of Fan-D and Fan-Optim. From the velocity contour comparison it is clear that there is a high and low velocity on the suction and pressure sides respectively on each aerofoil. It is also shown that severe flow separation appears over the blade span of Fan-D as one moves from the hub to the tip. This is not present in Fan-Optim.

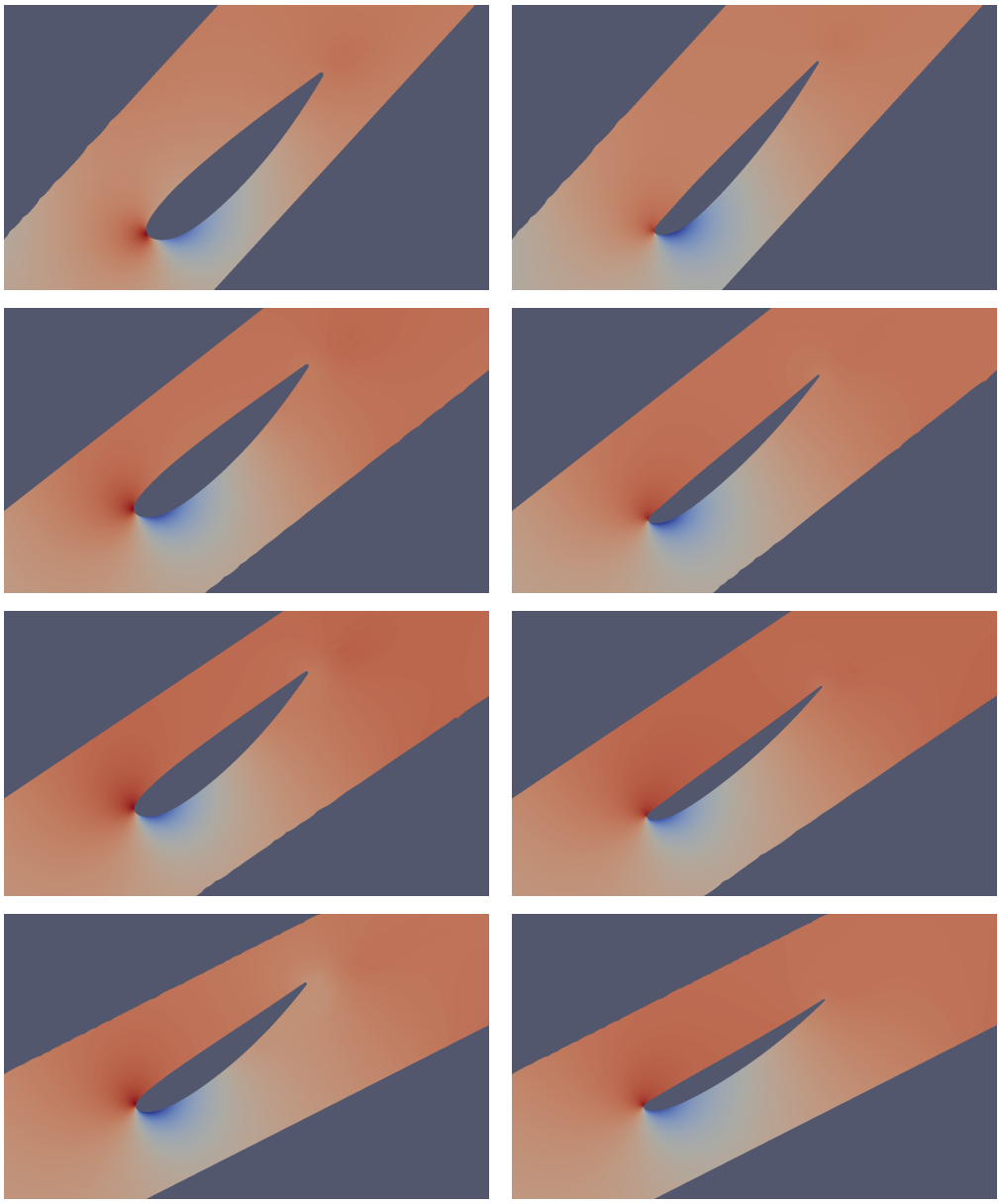

Figure 8.10: 2D cascades at $\phi^\star = 0.610$, 0.73, 0.80, and 0.93 (from top to bottom respectively) static pressure contours of Fan-D (left) and Fan-Optim (right)

In addition, there is also a downstream shift, from Fan-D to Fan-Optim, in the position of the stagnation point (point where the flow splits at the leading edge) that is present on the pressure side at the leading edge of each aerofoil. This is prevalent on sections closest to the hub and would probably be expected due to the aerofoil stagger angle decrease at ϕ_1 in the optimisation process. Towards the blade tip, the stagnation point appears to be in a similar position, despite the fact that the aerofoil stagger angle has been increased in the optimisation process.

There also appears to be a more gradual velocity gradient on the pressure side, beginning at the stagnation point on the leading edge, in Fan-Optim's profiles. On the aerofoil's suction side, Fan-Optim's leading edge velocity seems to be higher, closer to the hub, and lower, closer to the rotor blade tip, when compared to Fan-D.

From the static pressure contour comparison, the low pressure regions that are linked to the high velocities are clearly visible on the suction side of each aerofoil. The higher pressure and linked lower velocities are on the pressure side. The difference in Fan-D and Fan-Optim's static pressure corresponds to the differences described in the velocity contours. The low and high pressure regions at the leading edge also seem closer to each other in Fan-Optim, which can be expected to be due to the decrease in the aerofoil thickness in the optimisation process.

Fan-D and Fan-Optim were manufactured and tested in accordance to the ISO 5801 standard by CFW. A comparison of the pseudo-3D predicted performance at the specified volumetric flow rate of both fans is shown in Table 8.2. The predicted fan static pressure rise is within 13% of the experimental results for both fans. The predicted absorbed power of Fan-D and Fan-Optim are within 8% and 16% respectively, of the experimental results. It is worth noting that Fan-D's power prediction is more accurate than that of Fan-Optim. The reason for this is that the latter fan performs more optimally with no severe flow break away near the blade tip (Figure 8.9). This implies that the tip of the blade (which is not accounted for) has a larger contribution to the power measured, in the case of Fan-Optim. The effect of not accounting for the blade tip clearance loss in the case of Fan-D is thus less important as the tip also has a smaller effect in reality due to large flow break away in this region.

Fan-D and Fan-Optim's predicted fan static efficiencies compares well to the experimental results and are within 4.3% and 2.3% respectively, of the measured results.

Description	**Performance indicators**		
	$p^{\star}_{fsp}$	$P^{\star}$	$\eta^{\star}_{s}$
Experimental	1.0	1.0	100.0
Fan-D	1.13	1.08	104.3
Fan-Optim	1.13	1.16	97.7

Table 8.2: Pseudo-3D prediction and experimental fan performance of Fan-D and Fan-Optim

The pseudo-3D modelling technique employed assumes a constant axial outlet velocity over the fan rotor blade span. This implies that the radial velocity is negligibly small. This assumption is critically reviewed in the following section.

8.2.1 Radial Velocity Effect

In the pseudo-3D fan modelling strategy employed in this work it is assumed that the axial velocity is radially constant at the outlet. This allowed for the simplification of the mass flow rated averaged predicted fan static pressure rise in Equation (3.2). If there is radial flow present, mass conservation would not be satisfied, due to the assumed constant axial outlet mass flow rate. If radial flow is present, then the predicted fan static pressure rise may be effected in the mass

flow rated averaged calculation(Equation (3.2)) due to the non-uniform velocity at the outlet. This matter is next investigated at the duty point of Fan-Optim. Furthermore, it is noted that, due to the radial pressure variation at the outlets of the different 2D sections, radial flow is expected.

Consider a case where the flow is in radial equilibrium (static pressure force in a radial direction is equal to the inertia force). By simplifying the problem, a radial equilibrium equation can be derived, as shown by Saravanamuttoo and Rogers [62], Wallis [1] and Lewis [2]:

$$\frac{1}{\rho}\frac{dp}{dr} = \frac{u_{rot}^2}{r} \tag{8.2}$$

where u_{rot} is the rotational velocity at a specific radius r. If the assumption of a constant total pressure rise and outlet axial velocity is combined with Equation (8.2), the free-vortex condition can be derived:

$$u_{rot}r = constant \tag{8.3}$$

Fan-Optim is not designed by enforcing the free-vortex condition, as reflected by a non-constant $u_{rot}r$ over the blade span. The effect of this non-constant value will now be investigated by calculating the theoretical pressure rise for two pseudo-3D inviscid models, Fan-T_1 and Fan-T_2. Each model will consist of six cascades.

Fan-T_1 will employ the inlet and outlet relative velocity vectors of Fan-Optim, with a constant axial velocity over the fan rotor blade span. Fan-T_2 will employ refined outlet relative velocity vectors that take radial flow into account. These vectors are determined by solving the inviscid, 3D airflow, in an annulus domain that represents the outlet zone downstream of the axial fan rotor. Fan-Optim's inlet relative velocity vectors are prescribed at the inlet of Fan-T_2. At the outlet of Fan-T_2, the static pressure of Fan-Optim is prescribed. Slip walls are prescribed at the inner and outer cylindrical surfaces. The purpose of this model is to determine the induced radial velocity vectors due the static pressure distribution that is present at the outlet of Fan-Optim. The Fan-T_2 model did, however, reveal that there is a relatively small radial velocity that is a function of radius. This implies that the axial velocity is thus not constant in a radial direction over the blade span. These refined velocity vectors are now used to calculate the theoretical static pressure rise of Fan-T_2.

The theoretical fan static pressure rise of Fan-T_1 and Fan-T_2 is determined by integrating the theoretical static pressure rise of each 2D cascade in a radial direction with Equation (3.1). The theoretical static pressure rise in a 2D cascade is calculated with Equation (8.1).

In Figure 8.11, the normalised theoretically static pressure rise over the blade span, with and without rotational velocity recovery, is presented. In both figures the effect of radial velocity on the static pressure rise can be seen. Towards the inner radius (hub), the static pressure rise of Fan-T_2 is lower then Fan-T_1. This is due to the a higher axial velocity in Fan-T_2 due to radial flow. Towards the outer radius (tip), the static pressure rise of Fan-T_2 is higher than Fan-T_1. This is due to a lower axial velocity in Fan-T_2 as a result of the radial flow. By comparing the theoretical static pressure rise (with and without rotational velocity recovery) of Fan-T_1 and Fan-T_2, it is found that the addition of the radial velocity does result in a decrease. This value is -3.3% in the case where no outlet rotational velocity is recovered and -2.4% where rotational recovery is employed. This implies that

Fan-Optim's predicted static pressure will be marginally over-estimated due to the exclusion of the radial velocity in the modelling technique.

The reduction estimate of 3.3% is next applied to Fan-Optim's predicted static pressure rise. This results in a normalised fan predicted static pressure rise that is, indeed, closer to the measured data (within 10% of the experimental value).

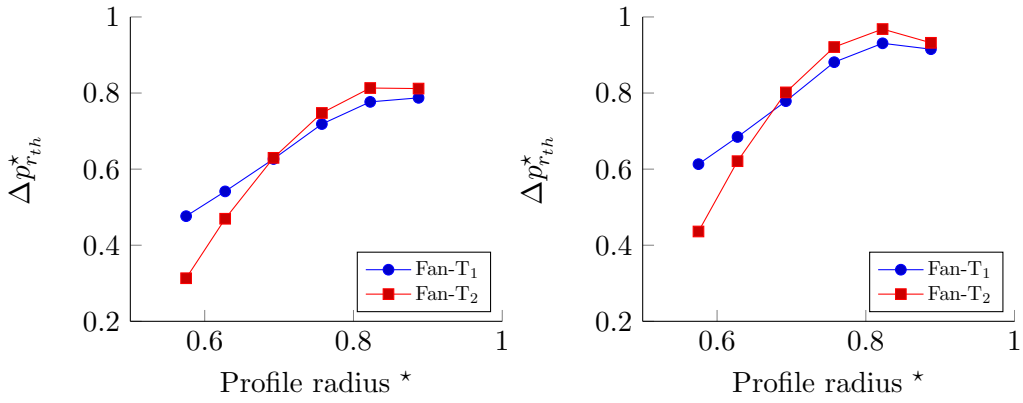

Figure 8.11: Theoretical static pressure with no rotational recovery (left) and with rotational recovery (right) vs profile radius

8.3 Parasitic Drag Prediction Error Effect

Considering the parasitic drag prediction error (due to the turbulence model employed), an estimation of the effect thereof on the optimisation process is evaluated. This is done by comparing objective function (Equation (6.1)) values of two cases at three aerofoil stagger angles (one either side of the optimal objective function value). The first case is that of an inviscid (no viscosity or turbulence) flow field in a 2D cascade at ϕ_1; the second case is a viscous flow field with the implemented Spalart-Allmaras turbulence model. Figure 8.12 shows the resulting normalised values. The objective function values differ by 0.5 and 4.8% at stagger angles of 0.92 and 0.99 respectively. At the optimum objective function value, the difference is 3.0%. Similarly, the static pressure rise differs by only 4%. The estimated effect of the drag prediction error on the objective function is therefore relatively small. The reason being that a -34.5% error is made in the portion that only contributes 3.0% to the net result. The implemented turbulence model is therefore deemed sufficient for the purpose of this work.

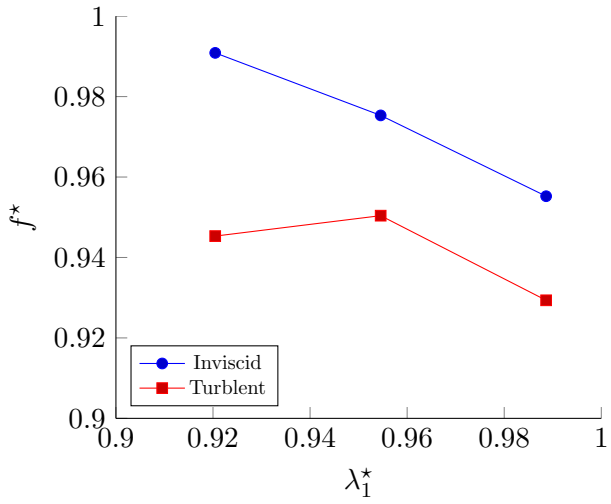

Figure 8.12: Inviscid and viscous turbulent comparison of the objective function on a 2D cascade

Chapter 9

Conclusions and Potential Future Work

9.1 Conclusions

A pseudo-3D fan modelling strategy to describe axial flow fans was developed and implemented. It consists of a collection of 2D cascades containing a series of 2D aerofoils, where the flow field in the latter is to be solved via 2D CFD. This was done in an effort to create a preliminary design tool that is generic, not highly dependent on empirical data like analytical models, and at a lower computational cost compared to full 3D CFD models. The series of aerofoils in the 2D cascade (at a specific radius) are reduced to one aerofoil in the CFD flow domain, by implementing a periodic boundary condition. The latter implementation utilised a quadtree data structure in a unique manner to identify duplicated nodes. Elemental^{TM} was used to solve the steady state, incompressible, and turbulent flow field in the 2D meshes, ranging from 44×10^3 to 65×10^3 elements. A Spalart-Allmaras turbulence model was implemented and employed due to the transitional flow regime present, with a Reynolds number varying between 300×10^3 and 400×10^3 over the fan blade span. In the interest of computational efficiency, a computed tangential force monitoring function was developed and implemented to assess solver convergence.

In this work, hybrid unstructured meshes were employed in the 2D computational models. Accordingly, a more accurate anisotropic discretisation scheme ETAD was implemented and further developed into ETAD$_p$. The former scheme allows for a more accurate approximation of a velocity field variable at a finite volume's face, in the solution procedure of Elemental^{TM}. The latter scheme includes higher order approximations of pressure, and pressure gradients, at finite volume faces. The higher order approximations are achieved by improving the accuracy of the advection-diffusion equation, via utilisation of node upstream and downstream values and gradients only. For ETAD$_p$, higher order approximations could only be achieved for internal faces. This is due to the elimination of the gradient contribution in the approximations for both the upwind-boundary-adjacent, and downwind-boundary-adjacent mesh locations.

A constrained optimisation methodology (with two design variables) was developed to optimise fan static efficiency by applying an optimisation algorithm to a penalty function formulation for 2D aerofoils. The aerofoils shapes were represented by analytical functions. Two 2D aerofoils were utilised to represent the 3D fan, one close to the hub, and one close to the tip, of the fan rotor blade. The aerofoil close to the hub did not include any pressure rise requirements whereas

at the outer aerofoil it was included in the penalty function formulation. A gradient-based optimisation algorithm (SQSD) was applied to the inner aerofoil first. The resulting 2D predicted static pressure rise, in conjunction with the static pressure requirement, were used to determine the required pressure rise at the outer aerofoil of the model. Aerofoil geometry perturbations, resulting in computational domain re-meshing, were catered for through an efficient automated mesh movement algorithm. This allowed for the re-use of a mesh in the optimisation process. The constrained optimisation methodology was implemented in Elemental^{TM} and the solutions were achieved by means of multiple parallel runs on the UCT hex cluster.

Two flow problems were used, namely, the viscous lid-driven cavity and turbulent NACA0018 aerofoil cases. Considering the ETAD and ETAD$_p$ schemes first, errors were compared to more traditional schemes (CUI, QUICK and NVSF). For the lid-driven cavity case, ETAD outperformed all the traditional schemes. It resulted in a maximum averaged L2-norm error reduction of 37% and 53% on the structured and unstructured meshes respectively, when compared to QUICK. However, ETAD was not able to maintain accuracy improvements over all of the Reynolds numbers. ETAD$_p$ resulted in even further error reductions. It achieved a maximum averaged L2-norm error reduction of 51% on both structured and unstructured meshes, when compared to QUICK. In addition, ETAD$_p$ retained accuracy improvements over all Reynolds numbers and was clearly the all-round superior method.

The NACA0018 aerofoil case was employed to validate the implemented Spalart-Allmaras turbulence model. The flow field was solved at a Reynolds number of 500×10^3, which is similar to the Reynolds number encountered in this work. Three meshes were employed to calculate the grid converged solution of the aerofoil's predicted drag and lift coefficients, via Richardson Extrapolation. The extrapolated values were compared to experimental results of Timmer [59]. A good comparison was found between the extrapolated and experimental results for the lift coefficient, with an error of -0.41%. The drag coefficient error was greater at a value of -32.78%. This value does seem large, but compared to the extensive numerical results in the drag and lift predictions by Hassan et al. [61], the drag coefficient error is not be surprising. The author employed various, more complex, two-equation turbulence models, with an error in drag prediction up to 436%.

In the application of the ETAD schemes in the NACA0018 aerofoil case, it was found that both schemes were unstable (noted for potential future work). This instability was remedied by using standard face interpolation values for velocity and pressure. Aerofoil predicted drag and lift coefficients were compared to experimental data. The novel schemes subsequently did not perform as well as the lid-driven cavity case. The drag prediction was found to be similar, with marginal differences between schemes when compared to QUICK. ETAD outperformed all the schemes in the prediction of the lift coefficient. An error reduction of 14.47% was achieved and ETAD$_p$ achieved an error reduction of just over 2.3%.

The final validation phase was concluded by evaluating the pseudo-3D fan modelling technique. This was done by comparing predicted fan static pressure rise and power, to experimental data and a full 3D CFD model at the specified volumetric flow rate. The former was found to be within 13%, and the latter 15%, of the experimental data for a model consisting of four cascades. The predicted values of the pseudo-3D model also compared well against the full 3D CFD. The

reduction in the number of cascades to two, resulted in a linear integration over the blade span. This indicates that more, rather then less, cascades should be employed. However, it was found that reducing the number of cascades from six to four did not result in a significant (within 0.1%) change in the predicted values. It was therefore decided to employ four cascades in the pseudo-3D model, instead of six. Further experimental validation showed a good comparison of the performance curves in the prediction of the fan static pressure rise, power, and static efficiency over a range of volumetric flow rates.

The optimisation methodology was next applied to Fan-D (base fan) to create an optimised fan (Fan-Optim). The 3D rotor blade geometry of the latter was created via linear interpolation and extrapolation. A pseudo-3D fan model of Fan-Optim (consisting of four cascades), was employed for the comparison of the predicted performance, to that required, and to Fan-D's predicted results. Fan-Optim's predicted static pressure rise, not only meets the requirement, but is also 18.6% higher than that of Fan-D. Fan-Optim similarly outperforms Fan-D with respect to static efficiency, which is 13.6% higher. As a secondary effect, the optimisation also resulted in a 38% saving in blade material. The effects of radial velocities that might be present, and the 2D aerofoil drag coefficient prediction error, were also evaluated for Fan-Optim. It was found that these have a marginal effect on the predicted static pressure rise.

For final experimental validation, both fans were manufactured and tested, by CFW, in accordance with ISO 5801. At the specified volumetric flow rate it was found that the predicted static pressure rise and static efficiencies compared well to experimental results. The former is within 13% of the experimental data, while the latter is within 2.3% to 4.3%.

9.2 Potential Future Work

The following are suggestions for potential future research work and improvements:

- *ETAD and ETAD$_p$*
  - The instability with the addition of higher order face convection and pressure approximations in a turbulent flow problem, should be investigated.
  - The blending function was developed by employing 1D advection-diffusion and 2D lid-driven cavity flow problems. Further refinement is required by adding a turbulent case.
  - Investigate possibilities of adding the gradient contributions for both the upwind-boundary-adjacent, and downwind-boundary-adjacent mesh locations.
- *Fan modelling*
  - Incorporate interaction between 2D cascades in the pseudo-3D model to allow for 3D flow effects.
  - The current axial fan model does not include a sudden expansion after the electrical motor. The 2D based pseudo-3D fan model can be improved by adding a 3D CFD model for the outlet that would automatically allow for expansion losses. This will also allow for the addition of outlet tail faring inclusion.

  - A similar modelling concept to the pseudo-3D model can be applied to account for the outlet guide vanes behind the fan rotor.
  - The psuedo-3D fan model should allow for the addition of blade tip clearance loss modelling.
  - Take into account blade root clearance.
  - The difference between the predicted and experimental fan absorbed power should be investigated in more depth.
  - The difference in the absorbed power predicted by Fan-D and Fan-Optim when compared to experimental data is not similar, and requires further investigation.
  - The generation of a 3D blade via a higher order function (instead of linear) should be investigated.

- *Optimisation methodology*
  - Implement a CFD based surrogate model to represent the objective function to be optimised. This will result in a reduction in computational cost in the derivative calculations.
  - The gradient calculation can be improved to a central difference method if the objective function is represented by an analytical equation.
  - Sound minimisation can be added as a second objective function. This might require a more specific multi-objective optimisation scheme.
  - If the number of objective functions are dramatically increased, another appropriate optimisation scheme must be selected.

References

[1] Wallis, R. A. (1983). *Axial Flow Fans and Ducts.* John Wiley and Sons, Inc.

[2] Lewis, R. (1996). *Turbomachinery performance analysis.* Butterworth-Heinemann.

[3] Bruneau, P. and von Backström, T. (1994). *The Design af a Single Axial Flow Fan for a Cooling Tower Application.* Master's thesis, Stellenbosch University.

[4] Kokturk, T. (2005). *Design and performance analysis of a reversible axial flow fan.* Master's thesis, MS Thesis, Middle East Technical University.

[5] Pascu, M. T. (2009). *Modern Layout and Design Strategy for Axial Fans.* Erlangen, Nürnberg, Univ., Diss., 2009.

[6] Borges, S. (2012). Cfd techniques applied to axial fans design of electric motors. In *Proceedings of the international conference on fan noise, technology and numerical methods (FAN2012).*

[7] Guedel, A., Robitu, M., and Chaulet, V. (2012). Cfd simulations to predict the energy efficiency of fan axial fan for various casing configurations. *Proceedings of FAN*, 18–20.

[8] Elhadi, E. E. and Keqi, W. (2003). Numerical simulation and modification of 3d flow phenomena in an axial flow fan. *Task quarterly*, 7(2), 199–213.

[9] le Roux, F. N., van der Spuy, S. J., and von Backström, T. W. (2013). *The CFD Simulation of an Axial Flow Fan.* Master's thesis, Stellenbosch University.

[10] Jang, C.-M., Choi, S.-M., and Kim, K.-Y. (2008). Effects of inflow distortion due to hub cap's shape on the performanceof axial flow fan. *Journal of Fluid Science and Technology*, 3(5).

[11] Augustyn, O. P., van der Spuy, S. J., and von Backström, T. W. (2013). *Experimental and Numerical Analysis of Axial Flow Fans.* Master's thesis, Stellenbosch University.

[12] Spalart, P. R. and Allmaras, S. R. (1992). A one equation turbulence model for aerodinamic flows. *AIAA journal*, 94.

[13] Crivellini, A. and D'Allesandro, V. (2014). Spalart-allmaras model apparent transition and rans simulations of laminar seperation bubles on airfoils. *International Journal of Heat and Fluid Flow*, 47, 70–83.

[14] Launder, B. E. and Spalding, D. (1974). The numerical computation of turbulent flows. *Computer methods in applied mechanics and engineering*, 3(2), 269–289.

[15] Shih, T.-H., Liou, W. W., Shabbir, A., et al. (1995). A new k- eddy viscosity model for high reynolds number turbulent flows. *Computers & Fluids*, 24(3), 227–238.

[16] Menter, F. R. (1993). Zonal two equation k-turbulence models for aerodynamic flows. *AIAA paper*, 2906, 1993.

[17] Huang, C. and Gau, C. (2012). An optimal design for axial-flow fan blade: Theoretical and experimental studies. *Journal of Mechanical Science and Technology*, 26, 427–436.

[18] Lee, K., Choi, Y., Kim, Y., et al. (2008). Design of axial fan using inverse design. *Journal of Mechanical Science and Technology*, 22, 1883–1888.

[19] Egorov, I., Shmotin, Y., and Fedechkin, K. (2005). Increasing of axial fan efficiency basing on optimization technology. *Proceedings of the 6thWorld Congresses of Structural and MDO, Rio de Janeiro, Brazil*, 30.

[20] Lin, B.-J., Hung, C.-I., and Tang, E. (2002). An optimal design of axial-flow fan blades by the machining method and an artificial neural network. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 216(3), 367–376.

[21] Kim, J.-H., Choi, J.-H., Husain, A., et al. (2010). Performance enhancement of axial fan blade through multi-objective optimization techniques. *Journal of Mechanical Science and Technology*, 24(10), 2059–2066.

[22] Kim, J.-H., Kim, J.-W., and Kim, K.-Y. (2011). Axial-flow ventilation fan design through multi-objective optimization to enhance aerodynamic performance. *Journal of Fluids Engineering*, 133(10), 101101.

[23] Kim, J.-H., Ovgor, B., Cha, K.-H., et al. (2014). Optimization of the aerodynamic and aeroacoustic performance of an axial-flow fan. *AIAA Journal*, 52(9), 2032–2044.

[24] Deb, K., Pratap, A., Agarwal, S., et al. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2), 182–197.

[25] Xiong, J., Liu, F., and Papamoschou, D. (2010). Aerodynamic shape optimization of fan-flow deflectors for noise reduction using adjoint method. In *28th AIAA Applied Aerodynamics Conference*, 2010-4675.

[26] Jameson, A. (1988). Aerodynamic design via control theory. *Journal of scientific computing*, 3(3), 233–260.

[27] Arens, K., Rentrop, P., Stoll, S., et al. (2005). An adjoint approach to optimal design of turbine blades. *Applied Numerical Mathematics*, 53(2), 93–105.

[28] Jameson, A., Martinelli, L., and Pierce, N. (1998). Optimum aerodynamic design using the navier–stokes equations. *Theoretical and computational fluid dynamics*, 10(1-4), 213–237.

[29] Straathof, M. H. (2012). *Shape parameterization in aircraft design: a novel method, based on b-splines*. Ph.D. thesis.

[30] Wang, D. (2008). *Turbomachinery aerodynamic and aeromechanic design optimization using the adjoint method*. Ph.D. thesis, Durham University.

[31] Neittaanmäki, P., Rossi, T., Korotov, S., et al. (2004). Design of gas turbine engines using cfd. In *European congress on computational methods in applied sciences and engineering (ECCOMAS)*.

[32] Papadimitriou, D. and Giannakoglou, K. (2007). Total pressure loss minimization in turbomachinery cascades using a new continuous adjoint formulation. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 221(6), 865–872.

[33] Brezillon, J. and Gauger, N. (2004). 2d and 3d aerodynamic shape optimisation using the adjoint approach. *Aerospace Science and Technology*, 8(8), 715–727.

[34] Choi, S., Lee, K., Potsdam, M. M., et al. (2014). Helicopter rotor design using a time-spectral and adjoint-based method. *Journal of Aircraft*, 51(2), 412–423.

[35] Merrick, D., Malan, A., and van Rooyen, J. (Submitted for review (2016)). A novel finite volume discretization method for advection-diffusion systems on stretched meshes. *Journal of Computational Physics*.

[36] Snyman, J. (2005). *Practical Mathematical Optimization: An introduction to basic Optimization theory and classical and new gradient-based algorithms*. Springer.

[37] Snyman, J. and Hay, A. (2001). The spherical quadratic steepest descent (sqsd) method for unconstrained minimization with no explicit line searches. *Computers & Mathematics with Applications*, 42(1), 169–178.

[38] Stoecker, W. F. and Jones, J. W. (1982). Refrigeration and air conditioning.

[39] Walton, D. and Meek, D. (1996). A triangular g1 patch from boundary curves. *Computer-Aided Design*, 28, 113–123.

[40] Malan, A., Lewis, R., and Nithiarasu, P. (2002). An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part *i*. theory and implementation. *International Journal for Numerical Methods in Engineering*, 54(5), 695–714.

[41] Vahdati, M., Morgan, K., Peraire, J., et al. (1989). A cell-vertex upwind unstructured grid solution procedure for high speed compressible viscous flow. In *IN: International Conference on Hypersonic Aerodynamics, Manchester, England, Sept. 4-6, 1989, Proceedings (A91-21176 07-02). London, Royal Aeronautical Society, 1989, p. 12.1-12.22. Research supported by British Aerospace, PLC.*, vol. 1. p. 12.

[42] Malan, A. and Lewis, R. (2003). Modelling coupled heat and mass transfer in drying non-hygroscopic capillary particulate materials. *Communications in Numerical Methods in Engineering*, 19(9), 669–677.

[43] Mowat, A., Malan, A., Van Zyl, L., et al. (2011). Hybrid finite-volume-rom approach to non-linear aerospace fluid-structure interaction modelling.

[44] Oxtoby, O. F. and Malan, A. G. (2012). A matrix-free, implicit, incompressible fractional-step algorithm for fluid–structure interaction applications. *Journal of Computational Physics*, 231(16), 5389–5405.

[45] Van Leer, B. (1977). Towards the ultimate conservative difference scheme. iv. a new approach to numerical convection. *Journal of Computational Physics*, 23(3), 276–299.

[46] Agarwal, R. (1981). A third-order-accurate upwind scheme for Navier-Stokes solutions at high reynolds numbers, AIAA paper 1981-112. In *19th AIAA Aerospace Sciences Meeting, St. Louis, MO, USA*.

[47] Leonard, B. (1979). The QUICK algorithm: a uniformly third-order finite difference method for highly convective flows. *Computer Methods in Applied Mechanics and Engineering*, 19, 59.

[48] Snyman, J. and Hay, A. (2002). The dynamic-q optimization method: An alternative to SQP ?. *Computers & Mathematics with Applications*, 44(12), 1589–1598.

[49] Craig, K., De Kock, D., and Snyman, J. (1999). Using cfd and mathematical optimization to investigate air pollution due to stacks. *International Journal for Numerical Methods in Engineering*, 44(4), 551–565.

[50] Snyman, J. (2000). The lfopc leap-frog algorithm for constrained optimization. *Computers & Mathematics with Applications*, 40(8), 1085–1096.

[51] Stander, N. and Craig, K. (2002). On the robustness of a simple domain reduction scheme for simulation-based optimization. *Engineering Computations*, 19(4), 431–450.

[52] Stander, N., Roux, W., Goel, T., et al. (2010). *LS-OPT v4.1 users manual*.

[53] Craig, K. and Kingsley, T. (2007). Design optimization of containers for sloshing and impact. *Structural and Multidisciplinary Optimization*, 33(1), 71–87.

[54] Stander, N., Craig, K., Müllerschön, H., et al. (2005). Material identification in structural optimization using response surfaces. *Structural and Multidisciplinary Optimization*, 29(2), 93–102.

[55] Vanderplaats, G. N. (2001). *Numerical Optimization Techniques for Engineering Design*. Vanderplaats Research and Development, Inc.

[56] Oxtoby, O., Malan, A., and Heyns, J. (2015). A computationally efficient 3d finite-volume scheme for violent liquid–gas sloshing. *International Journal for Numerical Methods in Fluids*, 79(6), 306–321.

[57] Darwish, M. (1993). A new high-resolution scheme based on the normalized variable formulation. *Numerical Heat Transfer, Part B*, 24, 353–371.

[58] Darwish, M. and Moukalled, F. (1994). Normalized variable and space formulation methodology for high-resolution schemes. *Numerical Heat Transfer, Part B*, 26, 79–96.

[59] Timmer (2008). Two-dimensional low-reynolds number wind tunnel results for airfoil naca 0018. *Wind engineering*, 32(6), 525–537.

[60] Roache, P. J. (1997). Quantification of uncertainty in computational fluid dynamics. *Annual review of fluid Mechanics*, 29(1), 123–160.

[61] Hassan, G. E., Hassan, A., and Youssef, M. E. (2015). Numerical investigation of medium range re numbers aerodynamics characteristics for NACA0018 airfoil. *CFD Letters*, 6(4), 175–187.

[62] Saravanamuttoo, H. I. H., Rogers, G. F. C., and Cohen, H. (2001). *Gas turbine theory*. Pearson Education.