Implicit Geometric Representation of Gas Turbine Blades For Optimal Shape Design

Tarek Mansour

M.A.Sc. Thesis

in

The Department

of

Mechanical and Industrial Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science (Mechanical Engineering) at

Concordia University

Montréal, Québec, Canada

April 2005

Your file Votre référence
ISBN: 0-494-04422-5
Our file Notre référence
ISBN: 0-494-04422-5

ABSTRACT

Implicit Geometric Representation of Gas Turbine Blades For Optimal Shape Design

Tarek Mansour, M.A.Sc. Student

Concordia Unviersity, 2005

Shape optimization requires proper geometric representation of the blade profile; the parameters of that representation are usually taken as design variables in the optimization process. Therefore the geometric model must be robust, flexible, efficient and accurate in representing both the global and local design spaces in order to obtain a successful optimization.

This work is concerned with the development and integration of two geometric representations of turbine blade profiles that are appropriate for aerodynamic optimization. The first model is the Modified Rapid Axial Turbine Design (MRATD) model where the blade is represented by five low-order curves that satisfy fifteen designer parameters; this model is suitable for a global search of the design space. The second model is based on a Non-Uniform Rational B-Spline (NURBS) parametrization that implicitly represents the MRATD profile and the implied designer parameters; this model can be used for a local shape refinement. The two models are presented and are assessed for flexibility, accuracy and curve smoothness when representing several typical turbine blade profiles. The models are also assessed in terms of their effect on the blades aerodynamic performance as measured by the pressure distribution along the blade surfaces. The usefulness of the MRATD model

is demonstrated in the global shape optimization of a subsonic cascade. The NURBS parametrization provides a means to control the blade profile locally, e.g., the blade curvature near a point smoothened by adjusting the NURBS control points and corresponding weights defining the given region. Finally, a 3D blade design model that uses the NURBS skinning technique is developed and assessed in terms of its precision and smoothness. This 3D model sets the foundation towards developing a robust scheme for 3D aerodynamic optimization.

ACKNOWLEDGEMENTS

I would like to sincerely thank my supervisor, Dr. Wahid Ghaly for his exceptional support, invaluable advice and his consistent efforts to push me to strive towards excellence. This work is a testament of my appreciation for all the energy and time he spent guiding and teaching me throughout this research.

I would also like to thank my colleagues who have provided excellent advice and support that have led to the completion of this work. In particular, Kasra Daneshkhah and Temesgen Mengistu who spent endless nights helping me implement their tools, the CFD code, the ANN and the optimization algorithms for all my problems. If not for them, this work would not have been completed.

Moreover, I would like to thank Dr. Chevy Chen, who was always available to answer all my questions with regards to computer aided design and NURBS in particular.

A special thanks must go to Giuseppe Pace, who assisted me in implementing the 3D NURBS parametrization presented in this work. His assistance was a valuable contribution to this work.

I would also like to thank Michel Dion of Pratt and Whitney Canada for the very useful discussions on the different aspects of gas turbine design.

Last but not least, a very special thank you to my wonderful parents and brother, and my loving wife for their endless support and encouragement. This work is dedicated to them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a, b	Ellipse constants
β	Blade angle or tangency angle
C	Chord length
ε	Objective variable or wedge angle
N_B	Number of blades
R	Radius
τ	Throat area
T	Thickness at a point on the blade pressure side
w	Weight in the objective function or weight in the NURBS function
ζ	Unguided turning angle

Subscripts

LE	Leading edge
IN	Inlet
OUT	Exit
PS	Pressure side
SS	Suction side
T	Tangential
TE	Trailing edge
X	Axial

Superscripts

w	4-Dimensional mapping

Acronyms

CFD	Computational fluid dynamics
GA	Genetic algorithm

LE	Leading edge
MRATD	Modified rapid axial turbine
NURBS	Non-uniform rational B-spline
RANS	Reynolds-averaged Navier-Stokes
RATD	Rapid axial turbine design
TE	Trailing edge

Chapter 1

INTRODUCTION

Progress in computer technology, as well as maturity of analysis and optimization methods have made the optimization design process much more practical and readily available for a growing number of industries. It is becoming an attractive alternative to traditional design methodologies that are expensive, time consuming, and are strongly dependent on the designer experience and knowledge. Aerodynamic optimization for turbomachinery applications, such as gas turbines, is of particular importance for the aerospace industry since small improvements in aerodynamic efficiency and performance can translate into significant savings. An imperative step in developing a practical and robust optimization procedure consists in developing a suitable geometric representation of the blade profile.

The development and application of a geometric representation for turbine blades is not a trivial task since turbine blades can have very diverse shapes depending on the design requirements. For example, the turbine blade of a gas turbine engine can have very high turning angles of up to 135^o, the leading edge (LE) and trailing edge (TE) can be either sharp (e.g., impulse blade) or round (e.g., inlet

guide vane); the profiles can also vary drastically in the spanwise direction due to loss considerations as well as spanwise work distribution. Furthermore due to the very high stresses that they must endure and the extremely high temperatures exiting from the combustor, designers sometimes end up with very convoluted blade shapes. It is a challenge to successfully formulate a representation that is simple, yet highly flexible and robust so as to allow an aerodynamic shape optimizer to automatically explore all possible regions of the design space and avoid unrealistic blade shapes.

1.1. Aerodynamic Optimization Problems in Turbomachinery

Generally, for aerodynamic optimization of turbine or compressor blades, the designer may seek to maximize efficiency and/or minimize pressure ratio while meeting the structural as well as the aerodynamic constraints (maximize aerodynamic loading and minimize total engine weight). In order to accomplish this task, one must have an optimization routine that is built on three critical elements: the geometric model that approximates the blade profile and whose parameters are the design variables, the analysis tool based on which the objective function is evaluated, and the optimization tool that drives the optimization process. Each element has a specific task to accomplish and all three work in unison to reach the design objectives. If one of the three elements fails in its task or is not suited for the specified optimization problem, it is unlikely that the design objectives will be reached.

The optimization process consists of a loop starting from the original geometry and iterating between different geometries in search of the optimal shape. It is the

task of the geometric model to generate all these geometries. Depending on the type of optimization (global or local), the design model has to be simple, flexible, efficient, robust and accurate in order to be suitable for aerodynamic optimization.

Once a geometry is generated, the process control goes to the analysis tool to compute the objective function. This can be a numerical solver that simulates the flow in the blade passage and then computes the objective function based on the flow field. However, it can also be an approximation model such as a Response Surface Method (RSM), e.g., Artificial Neural Network (ANN), that would take a specific geometry and compute the objective function. Usually, for an aerodynamic optimization, what is sought from the analysis tool are the efficiency, the total pressure loss, the mass flow rate, the pressure ratio and the inlet and exit flow angles. It is to be noted that this is usually the most costly computation in the optimization process.

Finally, process control goes to the optimization tool which is required in order to search and locate the optimal design. It does so by modifying the geometric model design parameters to generate different geometries. The optimization algorithms that can be used are very diverse, although some are more appropriate than others depending on the task. For example, global search techniques such as Genetic Algorithm (GA) and Simulated Annealing (SA) are remarkably well suited in locating regions in the design space.where an optimum would be found, and this without getting trapped in some local minimum. On the other hand, they perform poorly when they must locate the exact point of optimal design. In contrast, local search techniques such as gradient search methods, Newton's method and Sequential Quadratic Programming are very capable in locating local optima but usually fail when searching for the global optimum. Usually, a combination of a global

search method followed by a local search method is the ideal strategy in order to successfully and accurately locate the global optimum design in a complex design space.

1.2. Defining the Optimization Problem

In order to determine if a design is optimal, the optimization scheme computes an objective function that must be minimized. The function is usually of the following form:

$$F = w_1\varepsilon_1 + w_2\varepsilon_2 + ... + w_n\varepsilon_n, \tag{1.1}$$

where F is the objective function to be minimized, w_i is a weight coefficient and ε_i is the design objective. The optimization scheme can be single-objective ($n = 1$) or multi-objective ($n > 1$). If it is single-objective, the weight coefficient can be removed but when it is multi-objective, the weight coefficients need to be chosen very carefully since this will have a significant impact on the optimal shape.

Further the optimization schemes can be single-point or multi-point. A single-point optimization is required when the geometry is expected to perform optimally at a single design condition. Conversely, a multi-point optimization is required when the geometry needs to perform adequately over a range of design and off-design conditions. A multi-point optimization adds some complexity to the problem since the geometry has to be tested at multiple design conditions, increasing the number of iterations required and expanding the design space to be explored.

Finally the optimization can be multi-disciplinary, that is, it seeks to optimize a geometry based on two or more engineering disciplines. In turbine blade design

optimization, the engineering disciplines that are most likely to be optimized concurrently are aerodynamics, thermodynamics, heat transfer and structural dynamics. This significantly increases the complexity of the optimization problem.

1.3. Turbine Blade Shape Representation

This work focuses on developing turbine blade geometric models suitable for aerodynamic optimization, that is, the models must be simple, efficient, robust, accurate and flexible (both in a local and global sense). These requirements are not easily met by the current models used in industry and quoted in the literature. Furthermore, it is impractical if not impossible to develop one model that can be used for both global and local optimization problems. Therefore, two turbine blade geometric models will be developed in this work.

One model, the so-called *Modified Rapid Axial Turbine Design* (MRATD) model, is developed for use in the context of global optimization problems which is essential in order to efficiently, and accurately explore the entirety of the feasible design space. The MRATD model starts with the RATD model, originally developed by Pritchard [1], and modifies it to better represent turbine blade profiles and to suit current design practices as well as automatic shape optimization. These modifications allowed for a more flexible and reliable geometric model that would enhance the quality of the optimization results.

The second model, which uses NURBS, is developed to allow for further exploration of the design space locally. This step in the optimization process is essential in order to accurately target the optimum design point for a given global blade geometry.

Both models have been developed, implemented and assessed for their smoothness, flexibility, efficiency, robustness and accuracy, which are all necessary properties when dealing with optimization problems.

Chapter 2

THE MODIFIED RAPID AXIAL TURBINE DESIGN MODEL

In this chapter, we explore the means by which a low-fidelity, low order model that uses designer parameters, is generated. When creating a new turbine blade design, the designer starts with only a few blade parameters, which are also used as design parameters that describe the geometry globally (e.g., inlet and exit blade angles, stagger angle, minimum trailing edge radius, maximum thickness, throat area). Therefore, the design space to be explored can be significantly large, and it is critical that the model being used is able to entirely cover that space with accuracy and flexibility. However, when dealing with automated optimization, other important criteria to consider are the number of design parameters in the model and the model robustness in terms of generating feasible geometries. Incorporating many parameters might increase the flexibility of a model, but to the detriment of the optimizer performance. Generally, a larger number of design parameters translates into a more complex design space, hence a more difficult optimization problem that might also have many infeasible regions. These conditions led to the development of a low-fidelity model based on the RATD model, originally developed by Pritchard [1], which was modified to suit current design practices as well as automatic shape

optimization.

2.1. RATD Model Design Methodology

Numerous techniques have been developed throughout the years in order to describe airfoil shapes. These include the NACA airfoil profiles (where a thickness distribution is wrapped around a camber line), the Joukowski airfoil, Bézier representation [2], B-splines [3], etc. These representations were originally used in external flow applications and some were lately used in internal flow. Although they are effective in certain aspects, most of these techniques are not convenient for representing a turbine blade, particularly for optimization purposes. Some of these models lack flexibility, while others incorporate too many design parameters adding excessive complexity to the model.

Pritchard [1] was successful in the development of a relatively simple but realistic and practical model for turbine blade geometry representation, and this appears as a promising alternative to other low fidelity blade shape representations that are available in the literature. The model, identified as the *Rapid Axial Turbine Design* (RATD) model, is not only very simple and straightforward to use, it also provides a minimum set of design parameters that can be used to obtain an extensive family of turbine blade profiles. In his work, Pritchard [1] identified eleven geometric parameters that are necessary and sufficient in order to model an axial turbine blade using circular arcs and cubic polynomials, and developed the basic methodology to implement it. These eleven parameters include: the airfoil radial location, axial and tangential chords, inlet blade and wedge angles, exit blade angle, LE and TE radii, unguided turning, number of blades and throat area (see Fig. 2.1).

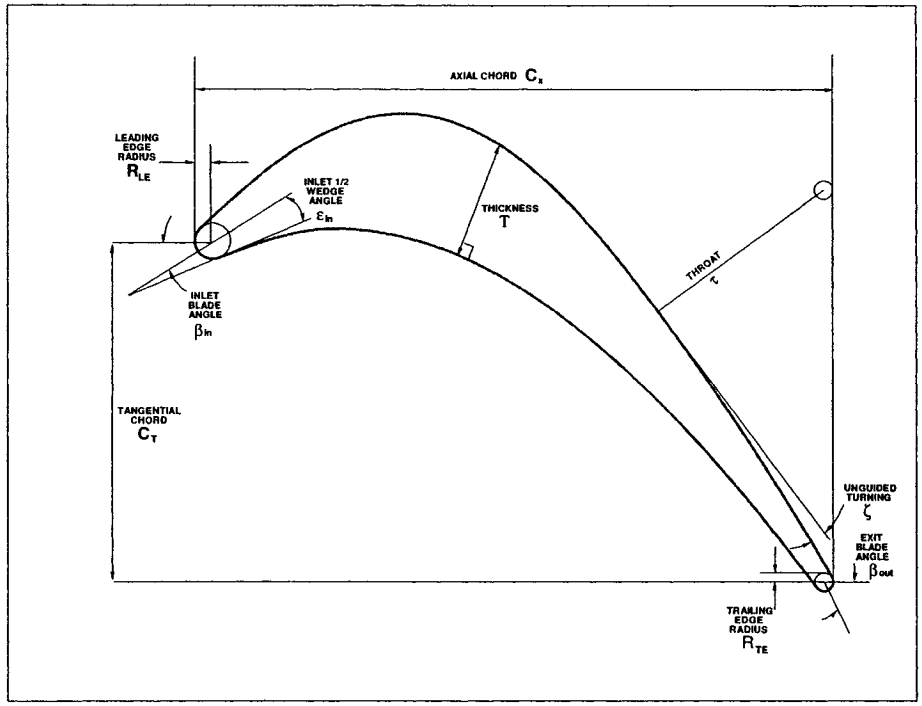

Figure 2.1: Design Parameters for the RATD Model.

The methodology behind the RATD model is simple and very efficient, which makes it an attractive tool for aerodynamic optimization. The model basically breaks down the blade into five distinct regions composed of the leading and trailing edge arcs, the suction and pressure side surfaces both modeled with a cubic polynomial and a circular arc for the uncovered part of the blade suction surface. With the eleven parameters mentioned above, it is possible to determine the locations of the five key points shown in Fig. 2.2, which represent intersection points between the five regions of the blade. Moreover, by imposing C^1 continuity of the blade profile at the intersection points, it is possible to determine uniquely the blade shape.

The design procedure begins by the sequential calculation of the five key points across the blade profile. The RATD model uses an iterative procedure to compute

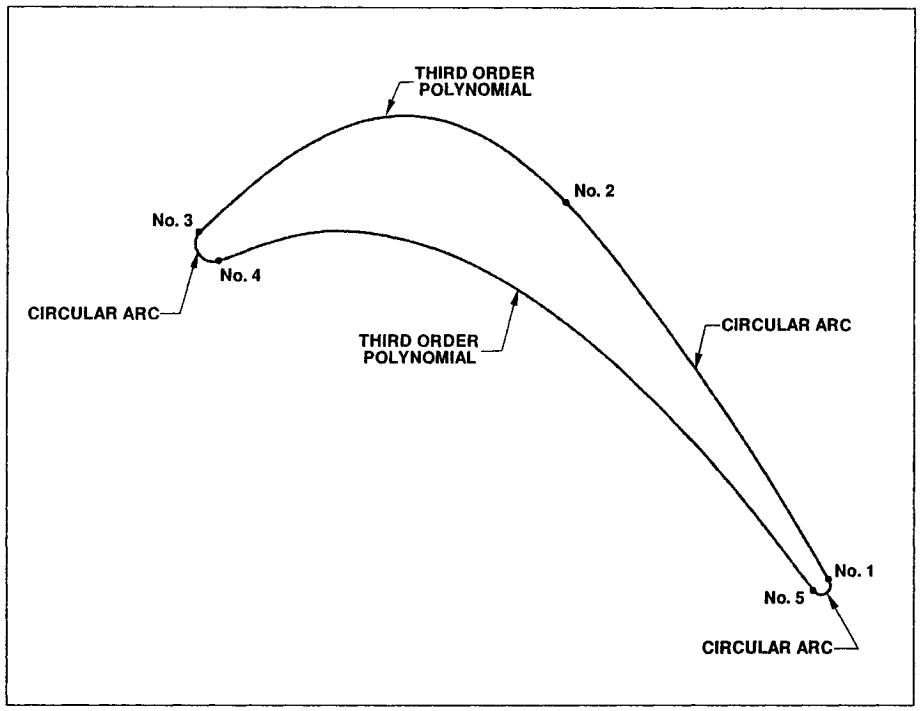

Figure 2.2: The five key points and section curves for the RATD model.

the blade geometry, which is necessary in order to remove the discontinuity at the throat (point 2). This will be explained shortly.

To compute x_1, y_1 and β_1 at key-point 1, the required design parameters are C_x, R_{TE}, β_{OUT}, and ε_{OUT}. It should be noted that ε_{OUT} is not part of the design parameter list. In fact, this is the variable that must be iterated upon in order to remove the throat discontinuity. On the first iteration ε_{OUT} is initialized as

$$\varepsilon_{OUT} = \frac{\zeta}{2}, \tag{2.1}$$

which is a good initial guess. The required values at each key point are the x and y-coordinates and the angle of tangency. For point one, these values are computed using laws of geometry as follows:

$$\beta_1 = \beta_{OUT} - \varepsilon_{OUT} \tag{2.2}$$

$$x_1 = C_x - R_{TE}(1 + \sin\beta_1) \tag{2.3}$$

$$y_1 = R_{TE}\cos\beta_1 \tag{2.4}$$

For point 2, the required parameters are β_{OUT}, ε_{OUT}, ζ, R, N_B and τ. Again, using laws of geometry, the tangency angle and x and y-coordinates are computed as follows:

$$\beta_2 = \beta_{OUT} - \varepsilon_{OUT} + \zeta \tag{2.5}$$

$$x_2 = C_x - R_{TE} + (\tau + R_{TE})\sin\beta_2 \tag{2.6}$$

$$y_2 = \frac{2\pi R}{N_B} - (\tau + R_{TE})\cos\beta_2 \tag{2.7}$$

The same procedure is repeated for the three remaining key points:

$$\beta_3 = \beta_{IN} - \varepsilon_{IN} \tag{2.8}$$

$$x_3 = R_{LE}(1 - \sin\beta_3) \tag{2.9}$$

$$y_3 = C_t + R_{LE}\cos\beta_3 \tag{2.10}$$

$$\beta_4 = \beta_{IN} - \varepsilon_{IN} \tag{2.11}$$

$$x_4 = R_{LE}(1 + \sin\beta_4) \tag{2.12}$$

$$y_4 = C_t - R_{LE}\cos\beta_4 \tag{2.13}$$

$$\beta_5 = \beta_{OUT} - \varepsilon_{OUT} \tag{2.14}$$

$$x_5 = C_x - R_{TE}(1 - \sin\beta_5) \tag{2.15}$$

$$y_5 = -R_{TE}\cos\beta_5 \tag{2.16}$$

After this is complete, it is necessary to check whether there is a throat discontinuity. Therefore we locate the intersection point (x_{01}, y_{01}) between the lines departing orthogonally from points 1 and 2 using the following relation:

$$x_{01} = \frac{(y_1 - y_2)\tan\beta_1\tan\beta_2 + x_1\tan\beta_1 + x_2\tan\beta_2}{\tan\beta_2 - \tan\beta_1} \tag{2.17}$$

$$y_{01} = y_1 - \frac{x_1 - x_{01}}{\tan\beta_1} \tag{2.18}$$

$$R_{01} = \sqrt{(x_1 - x_{01})^2 + (y_1 - y_{01})^2} \tag{2.19}$$

The y-coordinate at point 2, yy_2, is recalculated, this time using the geometrical relations for circles:

$$yy_2 = y_{01} + \sqrt{R_{01}^2 - (x_2 - x_{01})^2)} \tag{2.20}$$

If yy_2 is not equal to y_2, then there is a throat discontinuity and ε_{OUT} must be corrected using the following approximation in order to close the throat:

$$\varepsilon_{OUT} = \varepsilon_{OUT}(\frac{y_2}{yy_2}) \tag{2.21}$$

With ε_{OUT} adjusted, the procedure is repeated once again starting from Eq.(2.2), until the throat discontinuity is removed.

At this point, the five key points are joined using the curves shown in Fig. 2.2. The circular arcs are defined using the parametric form:

$$C(u) = (x(u), y(u)) \tag{2.22}$$

$$x(u) = R\cos u \tag{2.23}$$

$$y(u) = R\sin u \tag{2.24}$$

On the other hand, for the suction and pressure side surfaces, the coefficients of the 3^{rd} order polynomial equation must be computed. The mathematical relations are presented, using the pressure side surface as an example:

$$C(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \tag{2.25}$$

$$a_3 = \frac{\tan\beta_5 + \tan\beta_4}{(x_5 - x_4)^2} - \frac{2(y_5 - y_4)}{(x_5 - x_4)^3} \tag{2.26}$$

$$a_2 = \frac{y_5 - y_4}{(x5 - x4)^2} - \frac{\tan\beta_4}{x5 - x4} - a_3(x_5 + 2x_4) \tag{2.27}$$

$$a_1 = \tan\beta_4 - 2a_2 x_4 - 3a_3 x_4^2 \tag{2.28}$$

$$a_0 = y_4 - a_1 x_4 - a_2 x_4^2 - a_3 x_4^3 \tag{2.29}$$

With this, the blade geometry is completely modeled and the x and y-data-points along the profile can be output for analysis. Finally, the following section

will explain why this method of blade generation is not adequate for an optimization task, and how a modified version of the RATD model can be a much superior design tool.

2.2. The Designer Needs and the Modified RATD Model (MRATD)

The RATD model was intended as a low fidelity model for the preliminary design of turbine blades however, it lacks flexibility, smoothness, reliability and current design needs. The goal of the RATD model was to have a tool that is simple and flexible enough to readily model a variety of turbine blade families using only a minimum number of design parameters so that the designer can easily generate and analyze an extensive variety of practical turbine blade shapes with a minimum number of parameters and minimum effort. Even though it was not intended for the purpose of optimization, it has many qualities that make it an attractive blade design model that can be used, after proper modification that are described later in the section, in an automated optimization system. These modifications are aimed to better represent current turbine geometry, and to allow for elliptic leading edges; they are also aimed to eliminate infeasible shapes (when used in automatic shape optimization) by adding suitable parameters to the model. The RATD model was modified to address these needs and the resulting model is referred to hereafter as the MRATD model. All the modifications to the RATD model will be explained in detail next.

Testing of the RATD model was done by trying to approximate a variety of existing turbine blade geometries given in [4]. The exercise revealed that the RATD model lacks some flexibility and that it is very prone to providing infeasible blade

profiles. The lack of flexibility was due to three factors that have been addressed in the MRATD model.

2.2.1 Increasing the model flexibility

First, the suction and pressure side regions are linked by the common variable ε_{IN} so that any change in that variable will affect the totality of the suction and pressure side regions. In order to have more independent control over the pressure and suction side regions, ε_{IN} is split into two parts: $\varepsilon_{IN,PS}$ and $\varepsilon_{IN,SS}$. The procedure for calculating the parameters at the five key points remains the same , except that for points 3 and 4, $\varepsilon_{IN,SS}$ and $\varepsilon_{IN,PS}$ are used respectively instead of ε_{IN}.

Second, the uncovered and pressure side surfaces are linked by the ε_{OUT} variable, which again, provides limited control of the pressure side region. To remedy this, ε_{OUT} is also split into $\varepsilon_{OUT,SS}$ and $\varepsilon_{OUT,PS}$. The new variables are used instead of ε_{OUT} in Eq.(2.1) and for the computation of parameters at points 1 and 5.

Finally, it is observed that the values for $\varepsilon_{IN,SS}$ are very sensitive to and severely constrained by the behavior of the suction side surface that has the tendency to either overshoot resulting in an almost infinite thickness, or to cross the pressure side surface which results in a negative thickness. This problem is resolved by adding the thickness parameter T located at a point $x = x_T$ on the pressure side (see Fig. 2.1). The thickness is defined as the length of a line departing orthogonally from the pressure side and ending on the suction side. The methodology used to implement this design parameter is presented next.

Once the pressure side surface is defined, it is easy to determine the corresponding y-coordinate at x_T on the pressure side. From that point, we use a central

difference scheme in order to compute the line that runs orthogonally from the pressure side to the suction side. The length of the line is equal to T, the thickness at that point. Therefore, we can determine the x and y-coordinates of a point located along the suction surface. With this additional information, we have five boundary conditions that can be used to determine the curve defining the suction surface.

With these modifications in place, the model's flexibility is dramatically improved. Furthermore, only three designer parameters are added to the model, thus the simplicity of the model is not compromised. In fact, the parameters that are added provide the designer with more control over the geometry, thus allowing a much larger span for the upper and lower boundaries of $\varepsilon_{IN,SS}$ while simultaneously constraining the suction side surface using the thickness. These changes result in expanding the feasible design space and simultaneously reducing the infeasible one to improve the reliability of the MRATD model. Furthermore, the pressure side surface can be modified independently of the suction side and uncovered surfaces in order to adjust the cross-sectional area of the airfoil, which is critical when considering the structural integrity of a turbine blade.

2.2.2 The suction and pressure surface representation

The next issue with respect to the RATD model is the choice of a better geometric representation to model the suction and pressure side surfaces. To design and model smooth geometries, the use of high order polynomials is certainly not a good choice. First, the curvature of high order polynomials can be relatively wavy and, depending on its severity, might have a negative impact on the pressure distribution. Furthermore, high order polynomials have a tendency of generating wavy geometry, resulting in noticeably infeasible blade profiles (see Fig. 2.3). It becomes awkward

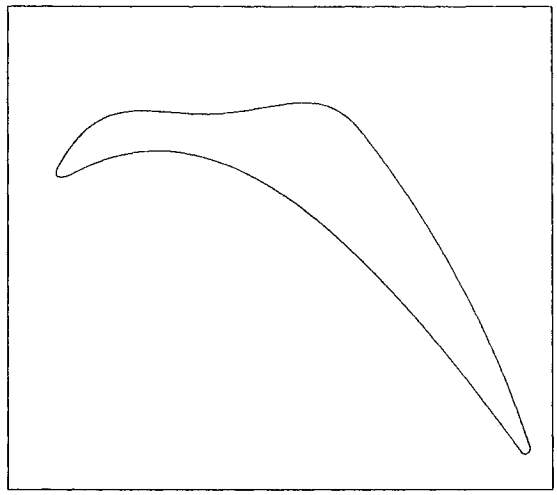

Figure 2.3: An infeasible geometry obtained by assigning an unrealistic combination of design parameters and polynomial segments for the suction and pressure sides.

to devise a strategy that will smoothen these functions and keep them well-behaved. This problem is resolved by replacing them with conic profiles.

Conics are functions that are defined by the general relation:

$$ax^2 + by^2 + 2hxy + 2fx + 2gy + c = 0 \tag{2.30}$$

They can be classified according to the coefficients of the above equation as follows (assuming non-degenerate conics):

$$\alpha = ab - h^2 \tag{2.31}$$

$\alpha = 0$	parabola	(2.32)
$\alpha > 0$	ellipse	(2.33)
$\alpha > 0$ and $a = b$	circle	(2.34)
$\alpha < 0$	hyperbola	(2.35)

To implement the conics into the MRATD model for the pressure and suction side regions, five boundary conditions are needed. For the suction side, these boundary conditions are the two end points and their respective slopes, as-well as the point defined by T. On the other hand, the pressure side is defined by only four boundary conditions: the two endpoints and their respective slopes. In order to fit a conic curve on the pressure side through the given boundary conditions, two options are plausible. The first alternative would be to add an extra design parameter that defines a point on the pressure side curve. This method has two disadvantages. First, it will increase the number of design parameters which must be kept to a minimum. Also, since the pressure side geometry is not critical, as long as it stays relatively smooth, it is impractical to have more constraints define the pressure side curve. A more convenient method is to use an approximate curve defined by the 3^{rd} order polynomial used in the RATD model, compute the y-position with x_T and use the given coordinates as the fifth boundary condition.

Several advantages result from the use of conics on the pressure and suction sides. Other than the obvious improvements in curve smoothness, the model will have a self-termination switch that is activated when infeasible values are assigned to the design parameters. With the RATD model, these infeasible set of design parameters resulted in wavy profiles, and detecting them proves to be quite challenging. Since conics cannot fit through a wavy set of boundary conditions, it is easy to devise an algorithm that will detect such conditions and terminate the program immediately. Moreover, a NURBS parametrization is used to model the conics due to a simpler implementation of the above strategy. More detail is provided in Sec. 3.6.

2.2.3 Requirements for current designs

A final modification to the original RATD model is to replace the leading edge circular arc with an elliptical arc. Current blade designs use elliptical arcs for the increased smoothness obtained at key-points 3 and 4, as well as the improved flow behavior over a wide range of incidence angles. The elliptical arc is described in terms of two new design parameters (that will replace the radius of a circular LE in the previous list), namely the minor and major axes of the elliptical arc. The equation of an ellipse is:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \tag{2.36}$$

where a and b define the size and location of the minor and major axis. In order to determine the x and y-coordinates of key-points 3 and 4, the derivative of Eq.(2.36) must be computed:

$$y = b\sqrt{1 - \frac{x^2}{a^2}} \tag{2.37}$$

$$y' = \frac{-bx}{a^2(1 - \frac{x^2}{a^2})^{0.5}} \tag{2.38}$$

Noting that ε_{IN} can be related to Eq.(2.38) as follows:

$$y' = \tan \varepsilon_{IN}, \tag{2.39}$$

we end up with:

$$x = \frac{a^2 \tan \varepsilon_{IN}}{\sqrt{a^2 \tan^2 \varepsilon_{IN} + b^2}} \tag{2.40}$$

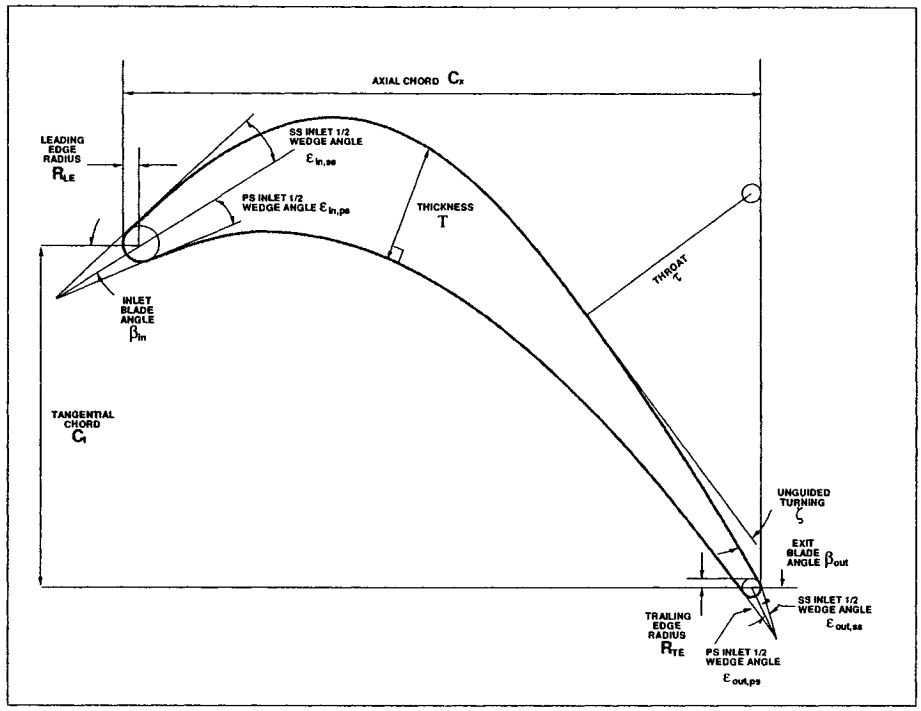

Figure 2.4: Design Parameters for the MRATD Model.

$$y = b\sqrt{1 - \frac{x^2}{a^2}} \tag{2.41}$$

Using Eqs.(2.40, 2.41) with β_{IN}, $\varepsilon_{IN,SS}$ and $\varepsilon_{IN,PS}$, it becomes possible to locate key-points (x_3, y_3) and (x_4, y_4).

With the implementation of the LE elliptical arc, we have completed our discussion pertaining to all the modifications performed on the RATD model. Figures 2.4 and 2.5 show schematically the added parameters and the modified curves of the MRATD model.

2.3. Assessment of the MRATD Model

The modifications and the new features added to the RATD model, which are described in Sec. 2.2, were intended to increase the flexibility and to improve the

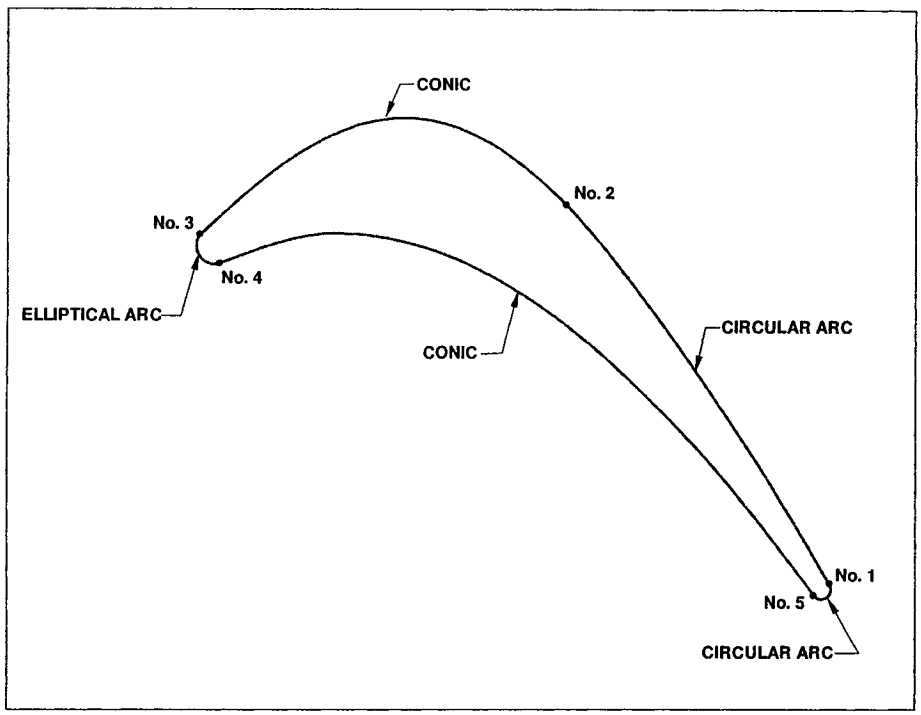

Figure 2.5: The five key points and section curves for the MRATD model.

smoothness of the MRATD model. At this point, it is important to determine if the changes made to the original model are valid and will in fact improve the applicability of the model in design optimization.

First, in order to assess the flexibility of the model, eight existing blade geometries are approximated. These geometries are shown in Figs. 2.6 to 2.13 along with their respective geometric error in units of axial chord. The flexibility of the MRATD model is clearly demonstrated from these examples. It is capable of representing blades with high turning angles (Figs. 2.6, 2.7, 2.8), low turning angles (Figs. 2.11, 2.12, 2.13), high stagger (Figs. 2.7, 2.11, 2.12, 2.13), low stagger (Figs.2.8, 2.9) and with elliptical or circular leading edges. Further since the functions used to join the five key points are naturally smooth and well-behaved curves, rarely does this model generate infeasible geometries. With the proper constraints set by the designer, the optimizer can freely investigate a very wide design space without wasting any time or resources on infeasible geometries such as the one in Fig. 2.3. This important aspect of the new model allows the optimization to skip any infeasible set of design parameters, unlike with the old representation, where the infeasible geometries cannot be detected and the CFD flow simulation tool would run, only to realize that the solution does not converge, wasting valuable hours of simulation time.

Finally, to demonstrate the smoothness of the new model compared with the original one, the curvature distributions on the suction side of the DFVLR blade are evaluated using the MRATD and RATD models (see Fig. 2.14). The plots clearly demonstrate the improved behavior of the conics representation with respect to the polynomial functions.

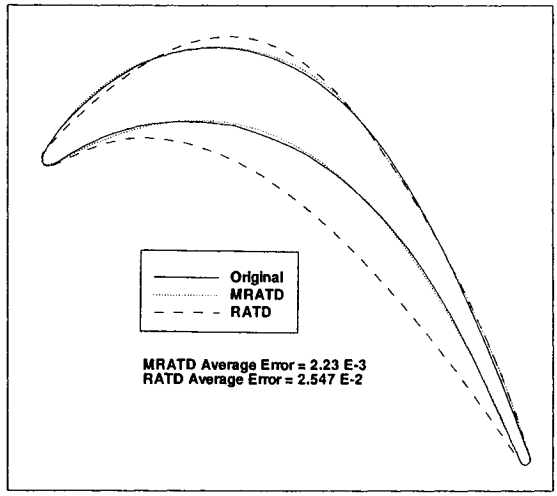

Figure 2.6: The DFVLR blade geometry approximated using the MRATD model [4].

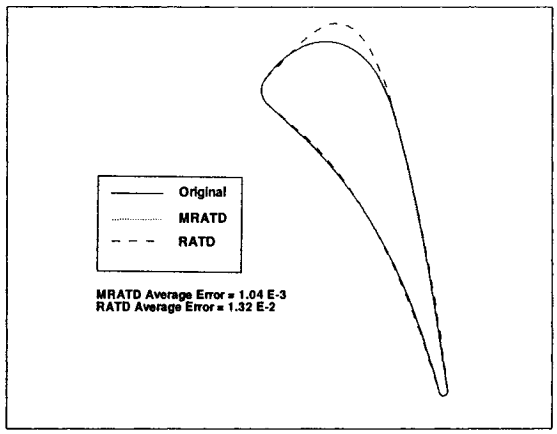

Figure 2.7: The VKI blade geometry approximated using the MRATD model [4].

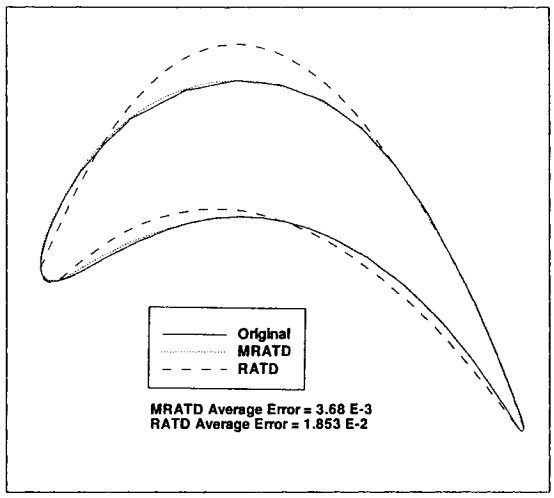

Figure 2.8: The ETU-4 section 1 (hub) blade geometry approximated using the MRATD model [4].

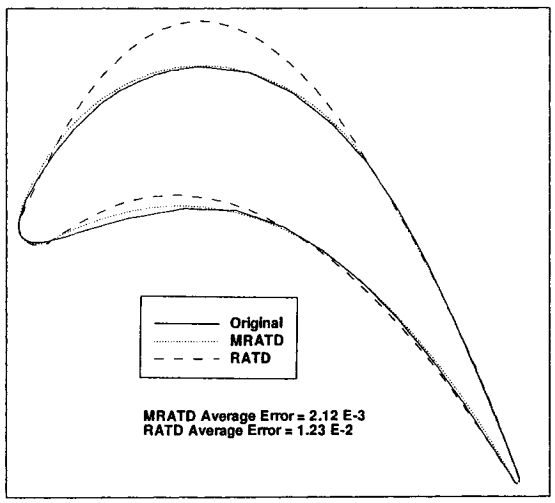

Figure 2.9: The ETU-4 section 2 (8% span) blade geometry approximated using the MRATD model [4].

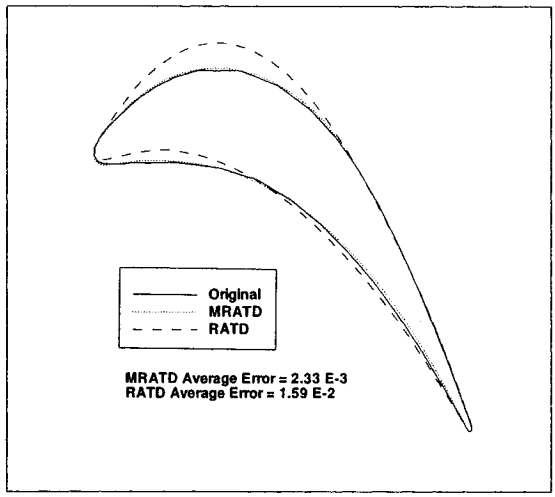

Figure 2.10: The ETU-4 section 3 (22% span) blade geometry approximated using the MRATD model [4].

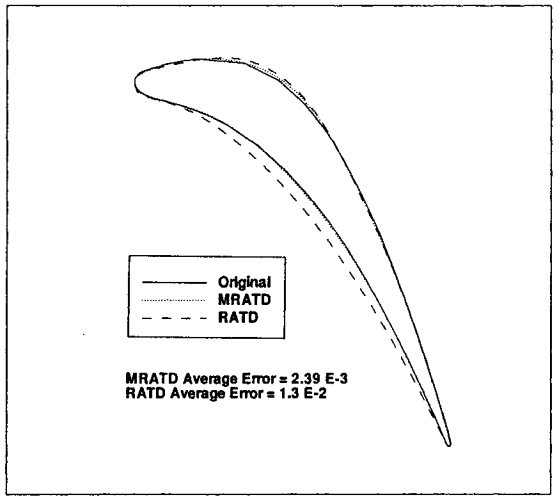

Figure 2.11: The ETU-4 section 4 (48% span) blade geometry approximated using the MRATD model [4].

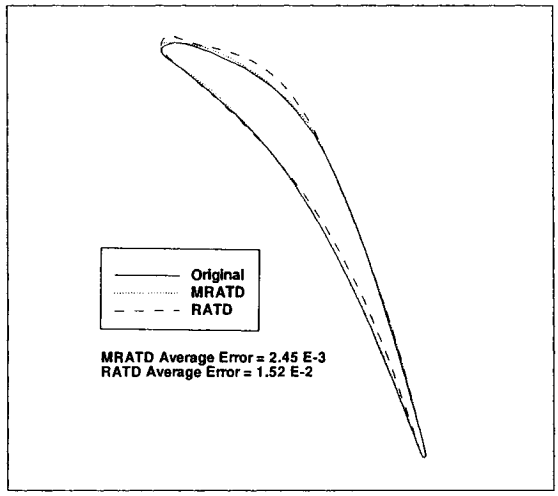

Figure 2.12: The ETU-4 section 5 (74% span) blade geometry approximated using the MRATD model [4].

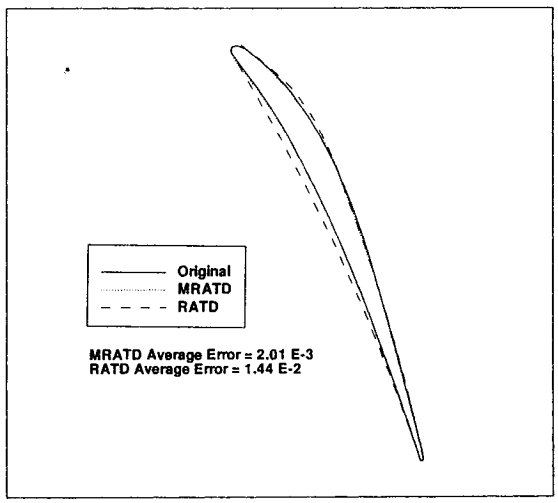

Figure 2.13: The ETU-4 section 6 (tip) blade geometry approximated using the MRATD model [4].

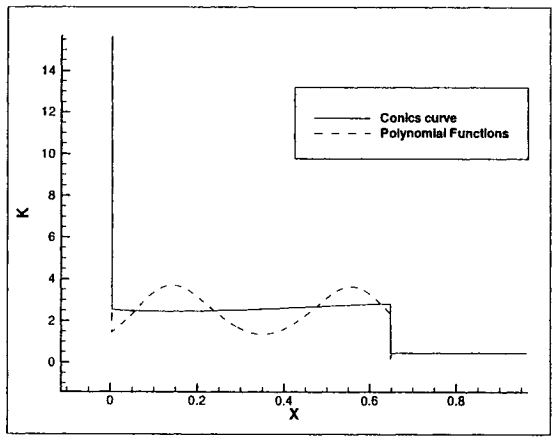

Figure 2.14: Comparison of the RATD vs MRATD suction side curvatures for the DFVLR geometry approximation.

2.4. MRATD Model and Automatic Optimization

In the previous sections, the MRATD model was developed and it was assessed in terms of flexibility and profile smoothness. Although the model is very capable of coarsely approximating general turbine blade profiles, as well as providing for a relatively smooth profile, it also has some limitations when used in an automatic optimization scheme; since the model is a low order representation of the geometry, it can only be used in sweeping the design space for global design parameters given by the designer's parameters. This section will discuss the advantages and limitations of the MRATD model with respect to aerodynamic optimization.

The MRATD model is ideal for a global inspection of the design space due to its flexibility; this was demonstrated in the previous section by approximating a number of turbine blade geometry. Further, an optimum balance is reached between the number of design parameters that were used for maximum flexibility and minimum model complexity. Finally, most of the infeasible design space can be eliminated using proper constraints, but without sacrificing any of the model's flexibility.

Furthermore, the model parameters are precisely the designer's parameters that are used in a typical design cycle. They refer to circular arc radii, profile thickness, metal angles, wedge angles, throat area, number of blades, profile length, profile height, and so on. This provides the designer with a very intuitive means of varying the blade geometry, where the outcome of any modification can be predicted quite easily. It also provides the optimizer with a model that best describes the design space with a small number of design parameters, which results in an easier task for the optimization process.

Finally, the MRATD design parameters are common ones regularly used by the turbine blade designers. Designers can interpret flow conditions across a turbine nozzle based on the values of the MRATD parameters. Even after completion of an aerodynamic optimization, where the blade geometry was modified to improve the flow conditions, an experienced designer can understand and interpret the solution, based on the MRATD parameters.

Nonetheless, the MRATD model has its inherent limitations that stem from the fact that the blade shape is approximated with five low order curves. Therefore, the model can only be used in a global search of the design space since the modification of one of its design parameters will result in a global change in the geometry, thus preventing the optimizer from exploring the effect of modifying local regions of the blade profile. In fact, the capability of local shape control is necessary in flow situations where local flow conditions must be improved, e.g., when removing or weakening a shock in transonic flow cases; this capability will be developed in the next chapter.

The other disadvantage is that the model has only C^1 continuity at the junction points (five key points), i.e., there is a curvature discontinuity at these five junction

points. This characteristic can have a negative effect on the suction side region, especially at the throat location. Due to the sudden change in curvature at the throat, and the diffusion taking place at the throat exit, a sharp rise in static pressure may occur, increasing the chances for separation and higher aerodynamic losses.

To allow for more control over local regions of the blade shape without losing the nice features of the MRATD model (in particular the designer's parameters that determine the global blade profile and are used by designers to assess any given design), a second geometric representation is superposed onto the MRATD model. That second representation allows for local control of the blade shape representation and also allows for higher order continuity to be imposed at junction points. This is presented, discussed and implemented in the next chapters.

Chapter 3

NURBS PARAMETRIZATION OF THE MRATD MODEL

The MRATD model provides for a good baseline blade geometry given the small number of design parameters required to define it. However, the conic arcs used to describe the suction side and the uncovered surface cannot inherently allow enough flexibility for local blade shape refinement. Therefore an implicit parametric representation of the MRATD model using a minimum number of design parameters becomes essential in order to reduce aerodynamic optimization effort while maintaining a high level of accuracy and flexibility and expressing the design variables in terms of the designer's parameters. From these conditions, two known alternatives are available, Non-Uniform Rational B-Splines (NURBS) and B-Splines. The latter are a subset of NURBS but due to their distinct properties they are often considered as separate entities.

3.1. Why use NURBS?

Due to a number of reasons, the more general representation given by NURBS would be a better choice for the parametrization than the B-spline function. First,

NURBS can represent conic shapes exactly, which is not the case for B-splines. When conducting aerodynamic analysis on the blade profile, this level of accuracy at the leading and trailing edges is crucial. Second, with the weights as additional parameters, we can have higher flexibility, which allows for a better inspection of the design space during optimization. Ghaly and Mengitsu [5] demonstrated the advantages of using NURBS, as opposed to using B-splines, for the representation of gas turbine blades when dealing with shape optimization. In their work, they were able to obtain results that are accurate to within machining tolerance when approximating a target geometry using NURBS with as low as nineteen control points for the DFVLR turbine cascade [4]. Furthermore, the aerodynamic performance obtained on the original profile was almost identical with that obtained on the NURBS approximated profile thus corroborating for the goodness of the approximation.

3.2. NURBS Functions

Before explaining the method used to develop the model, a brief definition of NURBS functions will be provided. For a more detailed account of NURBS, Piegel [6] provides an interesting and simplified approach to this sometimes-confusing topic. NURBS, or Non-Uniform Rational B-Splines, are defined as:

$$\mathbf{C}(u) = \frac{\sum_{i=0}^{n} N_{i,p}(u)\mathbf{P}_i w_i}{\sum_{j=0}^{n} N_{j,p}(u) w_j}, \tag{3.1}$$

where $\mathbf{C}(u)$ are the x and y-coordinates of the curve being generated, n is the number of control points $\mathbf{P}_i$, w_i the weights and $N_{i,p}(u)$ are the basis functions defined on the nonperiodic and nonuniform knot vector [6]:

$$\mathbf{U} = \{\underbrace{a, \ldots, a}_{p+1}, u_{p+1}, \ldots, u_{m-p-1}, \underbrace{b, \ldots, b}_{p+1}\}$$

The value of $N_{i,p}(u)$ is based on two variables, the knot value u, as well as the degree p of the function and is defined as [6]:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u \leq u_{i+1}, \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{3.3}$$

An example of a curve generated by a NURBS function is shown in Fig. 3.1. This example demonstrates a few of the important qualities of NURBS functions. First, the end points of a NURBS curve will always coincide with the first and last control points. Second, a NURBS function will always lie within its corresponding control polygon. Finally changing the position of any control point will have the effect of stretching the affected region of the NURBS curve in that same direction [6].

Three steps are required to compute the x and y-coordinates of a point on a NURBS curve [6]:

1. Find the knot span in which u lies
2. Compute the nonzero basis functions $N_{i,p}(u)$ using Eqs. (3.2 ,3.3)
3. Substitute for u, $N_{i,p}$, $\mathbf{P}_i$ and w_i into Eq. (3.1)

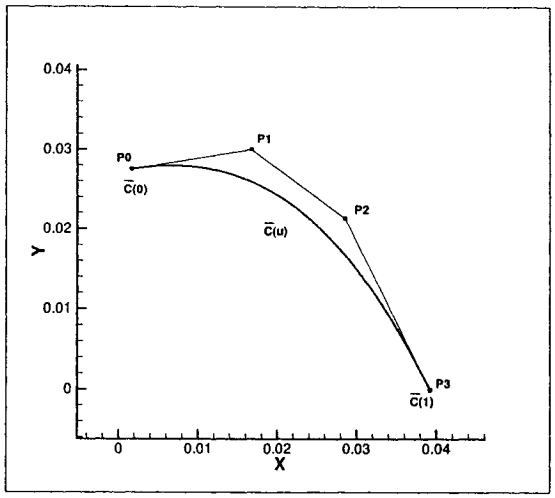

Figure 3.1: NURBS function and its corresponding control polygon.

3.3. Derivatives of NURBS

Derivatives of a NURBS curve have to be computed in order to impose higher order continuity across the 5 key points of the blade profile. The continuity of a NURBS curve at a point is determined by the degree p of the function and its multiplicity k at the given point; mainly the function is $p - k$ times continuously differentiable at any point. Before proceeding, a clarification is to be made pertaining to the four-dimensional representation of a NURBS function. In essence, a NURBS function is a four-dimensional curve projected on a three dimensional space, by mapping $C^w(u)$ from the origin to the hyperplane $W = 1$ [6].

$$\mathbf{C} = H\{\mathbf{C}^w\} = H\{X, Y, Z, W\} = \begin{cases} \left(\frac{X}{W}, \frac{Y}{W}, \frac{Z}{W}\right) & \text{if } W \neq 0, \\ \text{direction } (X, Y, Z) & \text{if } W = 0 \end{cases} \tag{3.4}$$

Let us now develop a formula that will express the derivatives of $\mathbf{C}(u)$ in terms of the derivatives of the non-rational curve $\mathbf{C}^w(u)$ [6].

Let

$$\mathbf{C}(u) = \frac{w(u)\mathbf{C}(u)}{w(u)} = \frac{\mathbf{A}(u)}{w(u)} \tag{3.5}$$

Then

$$\mathbf{C}'(u) = \frac{w(u)\mathbf{A}'(u) - w'(u)\mathbf{A}(u)}{w(u)^2}$$

$$= \frac{w(u)\mathbf{A}'(u) - w'(u)w(u)\mathbf{C}(u)}{w(u)^2}$$

$$= \frac{\mathbf{A}'(u) - w'(u)\mathbf{C}(u)}{w(u)},$$

and using Leibnitz' rule [6]

$$\mathbf{A}^{(k)}(u) = (w(u)\mathbf{C}(u))^{(k)} = \sum_{i=0}^{k} \binom{k}{i} w^{(i)}(u)\mathbf{C}^{(k-i)}(u)$$

$$= w(u)\mathbf{C}^{(k)}(u) + \sum_{i=1}^{k} \binom{k}{i} w^{(i)}(u)\mathbf{C}^{(k-i)}(u)$$

from which we obtain

$$\mathbf{C}^{(k)}(u) = \frac{\mathbf{A}^{(k)}(u) - \sum_{i=1}^{k} \binom{k}{i} w^{(i)}(u)\mathbf{C}^{(k-i)}(u)}{w(u)} \tag{3.6}$$

where the derivatives of $\mathbf{A}$ and w are computed as follows [6]:

$$\mathbf{A}^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k}(u)\mathbf{P}_i^{(k)} w_i^{(k)} \tag{3.7}$$

$$\mathbf{A}_i^{(k)} = \begin{cases} \mathbf{P}_i w_i & \text{if } k = 0, \\ \frac{p-k-1}{u_{i+p+1}-u_{i+k}}(\mathbf{P}_{i+1}^{(k-1)} w_{(i+1)}^{(k-1)} - \mathbf{P}_i^{(k-1)} w_{(i)}^{(k-1)}) & \text{if } k > 0 \end{cases} \tag{3.8}$$

and

$$w^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k}(u) w_i^{(k)} \tag{3.9}$$

$$w_i^{(k)} = \begin{cases} w_i & \text{if } k = 0, \\ \frac{p-k-1}{u_{i+p+1}-u_{i+k}}(w_{(i+1)}^{(k-1)} - w_{(i)}^{(k-1)}) & \text{if } k > 0 \end{cases} \tag{3.10}$$

3.4. Curve Interpolation

Another important topic to develop is curve interpolation to point data. In order to fit a B-spline curve through a set of data (NURBS functions cannot be interpolated since the problem is overdefined), we must solve for a set of linear equations obtained from [6]

$$\mathbf{Q}_k = C(\overline{u}_k) = \sum_{i=0}^{n} N_{i,p}(\overline{u}_k)\mathbf{P}_i \tag{3.11}$$

Suppose we are given a set of points $\mathbf{Q}_k$ where $k = \{0, 1, 2, \ldots, n\}$. We start by computing the $\overline{u}_k$ and u_j as follows [6]:

$$\overline{u}_0 = 0 \qquad \overline{u}_n = 1$$

and

$$d = \sum_{k=0}^{n} |\mathbf{Q}_k - \mathbf{Q}_{k-1}| \tag{3.12}$$

where d is the total chord length [6].

$$\overline{u}_k = \overline{u}_{k-1} + \frac{|\mathbf{Q}_k - \mathbf{Q}_{k-1}|}{d} \qquad k = 1, \ldots, n-1 \tag{3.13}$$

and

$$u_0 = \ldots = u_p = 0 \qquad u_{m-p} = \ldots = u_m = 1$$

$$u_{j+p} = \frac{1}{p} \sum_{i=j}^{j+p-1} \overline{u}_i \qquad j = 1, \ldots, n-p \tag{3.14}$$

Using Eq. 3.11, it is now possible to set up the $(n+1) \times (n+1)$ system of linear equations and solve for $\mathbf{P}_i$ [6].

In the case where the derivatives have to be imposed at specific knots on the curve, Eq. 3.6 provides the remaining set of linear equation that can be added to the original $(n+1) \times (n+1)$ matrix [6].

3.5. Knot Insertion, Knot Removal and Degree Elevation

The next topic to be discussed pertains to knot insertion (also known as knot refinement when referring to multiple knot insertions), knot removal, and degree elevation. Knot insertion is used in order to add more flexibility to a given NURBS curve. By adding knots, one increases the number of control points defining the curve, without modifying any of the curve characteristics (the curve and its derivatives are maintained). Further, it is also necessary when merging two or more knot vectors in

order to obtain a common knot vector for multiple curves, which is required when constructing NURBS surfaces. On the other hand, knot removal is required in order to remove unnecessary knots, for instance, after merging two NURBS curves to form one. Finally, degree elevation is necessary when merging two curves of different degree p, into one curve. The lower degree curve can have its order elevated without modifying any of the curve characteristics allowing for the merger of the knot vectors [6].

We begin our discussion with knot insertion. Inserting knots into the knot vector without disturbing any of the curve properties implies that [6]

$$\sum_{i=0}^{n} N_{i,p}(u)\mathbf{P}_i^w = \sum_{i=0}^{n+1} \overline{N}_{i,p}(u)\mathbf{Q}_i^w \tag{3.15}$$

where $\overline{N}_{i,p}(u)$ are the new set of basis functions after the knot insertion. Suppose the knot inserted is $\overline{u}$ where $\overline{u} \in [u_k, u_{k+1})$, then we can write the new knot vector $\overline{\mathbf{U}} = \{\overline{u}_0 = u_0, \ldots, \overline{u}_k = u_k, \overline{u}_{k+1} = \overline{u}, \ldots, \overline{u}_{m+1} = u_m\}$ and [6]

$$\sum_{i=k-p}^{k} N_{i,p}(u)\mathbf{P}_i^w = \sum_{i=k-p}^{k+1} \overline{N}_{i,p}(u)\mathbf{Q}_i^w \tag{3.16}$$

for all $u \in [u_k, u_{k+1})$, and

$$\begin{aligned} N_{i,p}(u) &= \overline{N}_{i,p}(u) \qquad i = 0, \ldots, k-p-1 \\ N_{i,p}(u) &= \overline{N}_{i+1,p}(u) \qquad i = k+1, \ldots, n \end{aligned} \tag{3.17}$$

Equations 3.16 and 3.17, as well as the linear independence of the basis functions, imply that [6]:

$$\mathbf{P}_i^w = \mathbf{Q}_i^w \qquad i = 0, \ldots, k-p-1 \tag{3.18}$$

$$\mathbf{P}_i^w = \mathbf{Q}_{i+1}^w \qquad i = k+1, \ldots, n \tag{3.19}$$

Now consider the $N_{i,p}(u)$ for $i = k-p, \ldots, k$. They can be expressed in terms of the $\overline{N}_{i,p}(u)$ when $i = k-p, \ldots, k+1$ by (the proof is omitted) [6]

$$N_{i,p}(u) = \frac{\overline{u} - \overline{u}_i}{\overline{u}_{i+p+1} - \overline{u}_i}\overline{N}_{i,p}(u) + \frac{\overline{u}_{i+p+2} - \overline{u}}{\overline{u}_{i+p+2} - \overline{u}_{i+1}}\overline{N}_{i+1,p}(u) \tag{3.20}$$

Substituting Eq. 3.20 into Eq. 3.16 to get [6]

$$\left(\frac{\overline{u} - \overline{u}_{k-p}}{\overline{u}_{k+1} - \overline{u}_{k-p}}\overline{N}_{k-p,p}(u) + \frac{\overline{u}_{k+2} - \overline{u}}{\overline{u}_{k+2} - \overline{u}_{k-p+1}}\overline{N}_{k-p+1,p}(u)\right)\mathbf{P}_{k-p}^w$$

$$+\left(\frac{\overline{u} - \overline{u}_{k-p+1}}{\overline{u}_{k+2} - \overline{u}_{k-p+1}}\overline{N}_{k-p+1,p}(u) + \frac{\overline{u}_{k+3} - \overline{u}}{\overline{u}_{k+3} - \overline{u}_{k-p+2}}\overline{N}_{k-p+2,p}(u)\right)\mathbf{P}_{k-p+1}^w$$

$$\vdots$$

$$+\left(\frac{\overline{u} - \overline{u}_k}{\overline{u}_{k+p+1} - \overline{u}_k}\overline{N}_{k,p}(u) + \frac{\overline{u}_{k+P+2} - \overline{u}}{\overline{u}_{k+p+2} - \overline{u}_{k+1}}\overline{N}_{k+1,p}(u)\right)\mathbf{P}_k^w$$

$$= \overline{N}_{k-p,p}(u)\mathbf{Q}_{k-p}^w + \ldots + \overline{N}_{k+1,p}(u)\mathbf{Q}_{k+1}^w$$

Rearranging and replacing the knot vector $\overline{\mathbf{U}}$ by $\mathbf{U}$ we get [6]

$$
\begin{aligned}
0 = {} & \overline{N}_{k-p}\left(\mathbf{Q}^w_{k-p} - \mathbf{P}^w_{k-p}\right) \\
& + \overline{N}_{k-p+1}\left(\mathbf{Q}^w_{k-p+1} - \frac{\overline{u} - u_{k-p+1}}{u_{k+1} - u_{k-p+1}}\mathbf{P}^w_{k-p+1} - \frac{u_{k+1} - \overline{u}}{u_{k+1} - u_{k-p+1}}\mathbf{P}^w_{k-p}\right) \\
& \vdots \\
& + \overline{N}_k\left(\mathbf{Q}^w_k - \frac{\overline{u} - u_k}{u_{k+p} - u_k}\mathbf{P}^w_k - \frac{u_{k+p} - \overline{u}}{u_{k+p} - u_k}\mathbf{P}^w_{k-1}\right) + \overline{N}_{k+1}\left(\mathbf{Q}^w_{k+1} - \mathbf{P}^w_k\right)
\end{aligned} \tag{3.21}
$$

For $i = k - p + 1, \ldots, k$, we set

$$
\alpha_i = \frac{\overline{u} - u_i}{u_{i+p} - u_i} \tag{3.22}
$$

and note that

$$
1 - \alpha_i = \frac{u_{i+p} - \overline{u}}{u_{i+p} - u_i} \tag{3.23}
$$

Using the linear independence of the basis functions, and substituting Eqs. 3.22 and Eqs. 3.23 into Eq. 3.21, yields [6]

$$
\begin{aligned}
& \mathbf{Q}^w_{k-p} = \mathbf{P}^w_{k-p} \\
& \mathbf{Q}^w_i = \alpha_i \mathbf{P}^w_i + (1 - \alpha_i)\mathbf{P}^w_{i-1} \qquad k - p + 1 \le i \le k \\
& \mathbf{Q}^w_{k+1} = \mathbf{P}^w_k
\end{aligned} \tag{3.24}
$$

Using Eqs. 3.18 and 3.24, we get the following relation for computing the new control points [6]

$$
\mathbf{Q}^w_i = \alpha_i \mathbf{P}^w_i + (1 - \alpha_i)\mathbf{P}^w_{i-1} \tag{3.25}
$$

where

$$\alpha_i = \begin{cases} 1 & \text{if } i \leq k-p \\ \frac{\bar{u}-u_i}{\bar{u}_{i+p}-u_i} & \text{if } k-p+1 \leq i \leq k \\ 0 & \text{if } i \geq k+1 \end{cases} \tag{3.26}$$

We continue now with the discussion of knot removal, which uses the same basic ideas as knot insertion. A repeated knot is removable t times, if the curve C^w is C^{p-s+t} continuous at $u = u_r$, where s is the multiplicity. It is important to note that the continuity must be with respect to $C^w(u)$, not its projection $C(u)$ which can be continuous even though $C^w(u)$ is not. Let $u = u_r \neq u_{r+1}$ be a knot of multiplicity s, where $1 \leq s \leq p$. The equations for computing the new control points for one removal of u are [6]

$$\mathbf{P}_i^1 = \frac{\mathbf{P}_i^0 - (1-\alpha_i)\mathbf{P}_{i-1}^1}{\alpha_i} \qquad r-p \leq i \leq \frac{1}{2}(2r-p-s-1)$$
$$\mathbf{P}_j^1 = \frac{\mathbf{P}_j^0 - \alpha_j \mathbf{P}_{j+1}^1}{(1-\alpha_j)} \qquad \frac{1}{2}(2r-p-s-2) \leq j \leq r-s$$

with

$$\alpha_k = \frac{u-u_k}{u_{k+p+1}-u_k} \qquad k = i, j \tag{3.27}$$

If one wishes to remove $u = u_r$ multiple times, the following equations should be used [6]

$$\mathbf{P}_i^t = \frac{\mathbf{P}_i^{t-1} - (1-\alpha_i)\mathbf{P}_{i-1}^t}{\alpha_i} \qquad r-p-t+1 \leq i \leq \frac{1}{2}(2r-p-s-t)$$
$$\mathbf{P}_j^t = \frac{\mathbf{P}_j^{t-1} - \alpha_j \mathbf{P}_{j+1}^t}{(1-\alpha_j)} \qquad \frac{1}{2}(2r-p-s-t+1) \leq j \leq r-s+t-1$$

with

$$\alpha_k = \frac{u - u_k}{u_{k+p+1} - u_k} \qquad k = i, j \tag{3.28}$$

The final matter to be discussed is that of degree elevation. Let $\mathbf{C}_p^w(u) = \sum_{i=0}^{n} N_{i,p}(u)\mathbf{P}_i^w$ be a pth-degree NURBS curve on the knot vector $\mathbf{U}$. In order to elevate the degree of the curve, the following relation must hold [6]

$$\mathbf{C}_p^w(u) = \mathbf{C}_{p+1}^w(u) = \sum_{i=0}^{\hat{n}} N_{i,p+1}(u)\mathbf{Q}_i^w \tag{3.29}$$

where the $\mathbf{C}_{p+1}^w(u)$ is the new degree-elevated curve, $\mathbf{Q}_i^w$ the new set of control points and $\hat{n}$ is the new index number for the control points. $\mathbf{C}_p^w(u)$ and $\mathbf{C}_{p+1}^w(u)$ are the same curve both geometrically and parametrically. The three unknowns in Eq. 3.29 are $\mathbf{Q}_i^w$, $\hat{n}$ and $\hat{\mathbf{U}}$, the new knot vector [6].

Determining $\hat{n}$ and $\hat{\mathbf{U}}$ is a trivial matter. Assume that the original knot vector $\mathbf{U}$ has the form [6]

$$\mathbf{U} = \{u_0, \ldots, u_m\} = \{\underbrace{a, \ldots, a}_{p+1}, \underbrace{u_1, \ldots, u_1}_{m_1}, \ldots, \underbrace{u_s, \ldots, u_s}_{m_s}, \underbrace{b, \ldots, b}_{p+1}\}$$

in order to maintain the continuity conditions of the original curve, the same knot must have multiplicity $m_i + 1$ for $\mathbf{C}_{p+1}(u)$. This yields [6]

$$\hat{n} = n + s + 1$$

$$\hat{\mathbf{U}} = \{u_0, \ldots, u_{\hat{m}}\} = \{\underbrace{a, \ldots, a}_{p+2}, \underbrace{u_1, \ldots, u_1}_{m_1+1}, \ldots, \underbrace{u_s, \ldots, u_s}_{m_s+1}, \underbrace{b, \ldots, b}_{p+2}\}$$

where $\hat{m} = m + s + 2$. The only remaining problem is to compute the $\{\mathbf{Q}_i\}$.

Firstly, the NURBS function is broken down to its Bézier components by using knot insertion. The Bézier control points of the segments are obtained by inserting each interior knot until it has multiplicity p. Then the Bézier segment is degree elevated [6].

Let

$$\mathbf{C}_p(u) = \sum_{i=0}^{p} B_{i,p}(u)\mathbf{P}_i$$

be a pth-degree Bézier curve, where [6]

$$B_{i,p}(u) = \binom{p}{i} u^i (1-u)^{n-i} \tag{3.30}$$

Its representation as a $(p+1)$th-degree curve is

$$\mathbf{C}_{p+1}(u) = \sum_{i=0}^{p+1} B_{i,p+1}(u)\mathbf{Q}_i \tag{3.31}$$

Setting these equal and multiplying $\mathbf{C}_p(u)$ by $(u + (1-u))$ yields [6]

$$\begin{aligned} \sum_{i=0}^{p+1} B_{i,p+1}\mathbf{Q}_i &= (u + (1-u)) \sum_{i=0}^{p} B_{i,p}\mathbf{P}_i \\ &= \sum_{i=0}^{p} ((1-u)B_{i,p} + uB_{i,p}\mathbf{P}_i \end{aligned} \tag{3.32}$$

Applying Eq. 3.30, we obtain [6]

$$\sum_{i=0}^{p+1} \binom{p+1}{i} u^i (1-u)^{p+1-i} \mathbf{Q}_i$$

$$= \sum_{i=0}^{p} \binom{p}{i} (u^i (1-u)^{p+1-i} + u^{i+1}(1-u)^{p-i}) \mathbf{P}_i \tag{3.33}$$

$$= \sum_{i=0}^{p} \binom{p}{i} (u^i (1-u)^{p+1-i} \mathbf{P}_i + \sum_{i=0}^{p+1} \binom{p}{i-1} u^i (1-u)^{p+1-i}) \mathbf{P}_{i-1} \tag{3.34}$$

Equating coefficients of $u^i(1-u)^{p+1-i}$ yields

$$\binom{p+1}{i} \mathbf{Q}_i = \binom{p}{i} \mathbf{P}_i + \binom{p}{i-1} \mathbf{P}_{i-1}$$

and

$$\frac{\binom{p}{i}}{\binom{p+1}{i}} = 1 - \frac{i}{p+1}$$

$$\frac{\binom{p}{i-1}}{\binom{p+1}{i}} = \frac{i}{p+1}$$

it follows that [6]

$$\mathbf{Q}_i = (1-\alpha_i)\mathbf{P}_i + \alpha_i \mathbf{P}_{i-1} \tag{3.35}$$

where

$$\alpha_i = \frac{i}{p+1} \qquad i = 0, \ldots, p+1$$

Finally, all the excess knots should be removed using the knot removal technique described earlier [6].

3.6. Representation of Conics using NURBS

We end our discussion of NURBS with one of the most interesting and attractive properties of these functions: their ability to represent conics exactly. The general implicit equation of a conic curve is given by Eq. 2.30. Again, we note that five boundary conditions are needed in order to define a given conic curve. The boundary conditions used in this model are the coordinates of the start and end points ($\mathbf{P}_0$ and $\mathbf{P}_2$) and their tangent directions ($\mathbf{T}_0$ and $\mathbf{T}_2$), as well as the coordinates of one point located on the conic curve ($\mathbf{P}$) [6].

The conic curve can be defined by three control points and a second degree NURBS curve, thus the knot vector has the following format [6]

$$\mathbf{U} = \{0, 0, 0, 1, 1, 1\}$$

The coordinates of the two end points intersect with the coordinates of the first and last control points. The remaining control point location is determined by intersecting the two tangent lines $\mathbf{T}_0$ and $\mathbf{T}_2$. Setting $w_0 = w_2 = 1$ and the only missing item is w_1. Omitting the derivation, the remaining weight is computed as follows [6]

$$w_1 = \frac{(1-u)^2(\mathbf{P}-\mathbf{P}_0)\cdot(\mathbf{P}_1-\mathbf{P}) + u^2(\mathbf{P}-\mathbf{P}_2)\cdot(\mathbf{P}_1-\mathbf{P})}{2u(1-u)|\mathbf{P}_1-\mathbf{P}|^2} \tag{3.36}$$

where

$$u = \frac{a}{1+a} \qquad a = \sqrt{\frac{|\mathbf{P}_0\mathbf{Q}|}{\mathbf{Q}\mathbf{P}_2}} \tag{3.37}$$

and $\mathbf{Q}$ is defined as the point of intersection between the lines connecting points $\mathbf{P}_0$ with $\mathbf{P}_2$ and $\mathbf{P}_1$ with $\mathbf{P}$ (see Fig. 3.2). The following facts pertaining to the

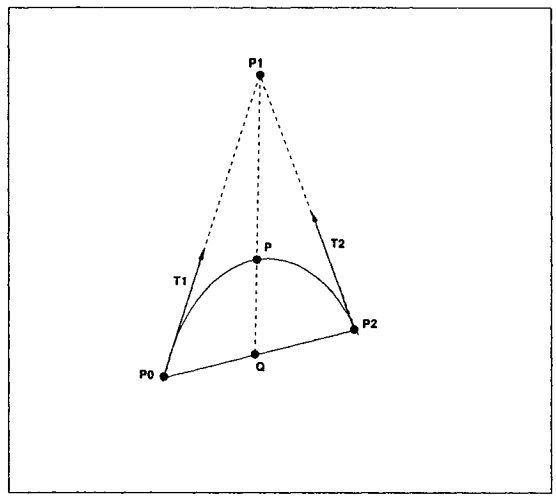

Figure 3.2: A conic curve defined by its five boundary conditions.

sign of the weight should be noted. If the value of w_1 is greater than 1.0, then the curve represents a parabola or a hyperbola. If w_1 is between 0.0 and 1.0, then the conic is an ellipse. Finally, if w_1 is less than 0.0, then the boundary conditions define a conic with an outside convex hull [6]. This further implies that the given boundary conditions define an infeasible blade geometry and should be corrected. This property allows us to incorporate a termination criteria for the blade design tool that is based on the sign of w_1.

Now consider if $\mathbf{T}_0$ and $\mathbf{T}_2$ are parallel lines. In such a case, $w_1 = 0$ and $\mathbf{P}_1$ will be parallel to $\mathbf{T}_0$. The following relation is used to determine the magnitude of $\mathbf{P}_1$ [6]:

$$P_1 = \frac{1}{f(u_0)}(P - Q) \qquad f(u_0) = \frac{2u_0(1-u_0)}{(1-u_0)^2 + u_0^2} \tag{3.38}$$

With this, we have completed the essential concepts used for developing the NURBS parametrization of the MRATD model.

3.7. NURBS Parametrization Strategies

When used in conjunction with optimization, NURBS offer some clear advantages namely, flexibility, accuracy and efficiency all at once. This being said, the next task would be to determine the best way NURBS can be implemented in order to represent the initial geometric model described in the previous section. Two possible approaches can be taken: either to fit one NURBS function across the entire blade profile, or to fit five distinct NURBS functions that would replace the five section curves of the MRATD model. Both approaches are discussed next.

3.7.1 One-NURBS curve blade shape representation

The first technique to parametrize the MRATD model using NURBS would be to fit one NURBS function through the entire geometry. The basic methodology behind this technique will be explained first.

Given a curve, we begin by specifying a number of data points lying on the curve. Using the NURBS interpolation technique described in Sec. 3.4, we can fit a B-spline curve through the given set of data points. Afterwards, a specified number of control points are eliminated, based on the curvature distribution, so that in higher curvature regions, more control points will be placed in order to have a good curve fit, while in the lower curvature regions less control points are required. Finally, an optimization routine is implemented in order to optimize the remaining control points coordinates and their weights for maximum accuracy. This method was developed by Mengistu and Ghaly and is described in more detail in [5].

When the blade profile is fitted with one NURBS function, the resulting profile is smooth and continuity of the blade shape and its derivatives is implicitly implied

across the entire profile except at the end points of the given closed curve, making this approach easier to implement than the second. However, fitting a NURBS function across the blade profile requires an optimization in itself. Furthermore the entire profile is an approximation of the original, and it is relatively difficult to preserve the original designer parameters (the 15 MRATD model parameters) embedded within the NURBS parametrization.

3.7.2 Five-NURBS curve blade shape representation

To allow for the exact representation of the conic sections, and allow for the implicit representation of the designer's parameters, the 5-NURBS representation of the blade profile is described in this subsection. Special attention will be given at the points joining these curves so as to insure continuity conditions across critical regions such as the suction side. The steps used for implementing this technique are explained first.

The first step is to use the technique described in Sec. 3.6 in order to define the NURBS parameters that will be fitted through every conic section of the blade geometry except for the suction side. The reason being that more control is usually required on the suction side, where the flow is much more sensitive than on the pressure side, therefore we would like to impose C^2 continuity across the junction points 2 and 3 (refer to Fig. 2.5).

The second step is to compute the first and second derivatives using Eq. 3.6 at key points 2 and 3. Using these derivatives as well as the methodology described in Sec. 3.4, we determine the NURBS parameters that will fit a curve through the suction side with C^2 continuity across points 2 and 3. The selection of the data points is again based on the curvature distribution of the suction side curve.

The third step consists of, if needed, the addition of control points on the uncovered surface of the blade as well as the pressure side region in order to improve the geometric model flexibility. We use the knot addition methodology described in Sec. 3.5.

Step four joins the five NURBS curves together into one single curve by joining the knot vectors. This requires that the degree p of all the sections be equal to p_{max} belonging to the suction side curve; the implementation of this degree elevation method is described in Sec. 3.5.

When the blade profile is fitted with five NURBS functions, we have a more accurate representation of the leading and trailing edges, the uncovered part of the suction surface and the pressure surface since a NURBS function can represent conic curves exactly. Furthermore, the original design parameters from the MRATD model can be easily preserved during and after optimization. In fact, the five key points of the MRATD model are always located at the same knot positions within the knot vector, and there slopes are also preserved, thus preserving the geometric constraints imposed by the MRATD designer parameters. This is a clear advantage to turbine designers who are much more comfortable dealing with those geometric parameters than with NURBS parameters, i.e., control points and weights, however such an approach involves an explicit implementation of continuity of the shape and even higher derivatives across the points joining any two NURBS functions. Further, continuity and curve smoothness issues arise when an attempt to remove the multiplicity points is made. These problems are discussed in greater detail in Ch. 4.

Nonetheless, since the advantages offered by the second approach are more practical for turbine blade designers, the second approach was chosen over the first.

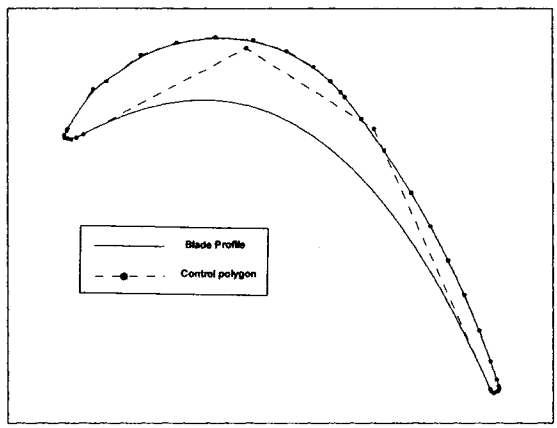

Figure 3.3: A NURBS approximation of the DFVLR geometry [4].

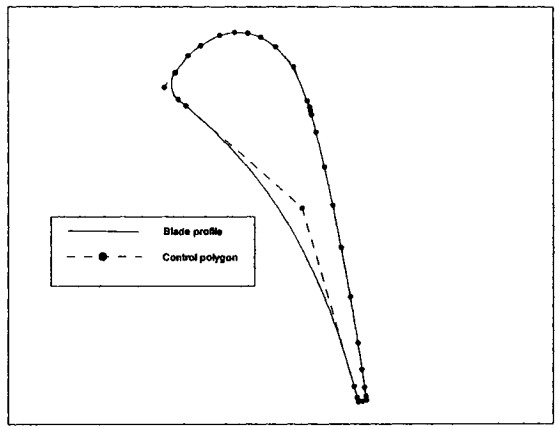

Figure 3.4: A NURBS approximation of the VKI geometry [4].

Several profiles are obtained with the NURBS parametrization and are shown together with their control polygons in Figs. 3.3 through 3.10.

3.8. NURBS Model Smoothness Assessment

In order to evaluate the suitability of the above-mentioned approximations, i.e., the one-NURBS versus the five-NURBS curve approximation, two criteria can be used namely, the curve smoothness as measured by the curvature (which can strongly

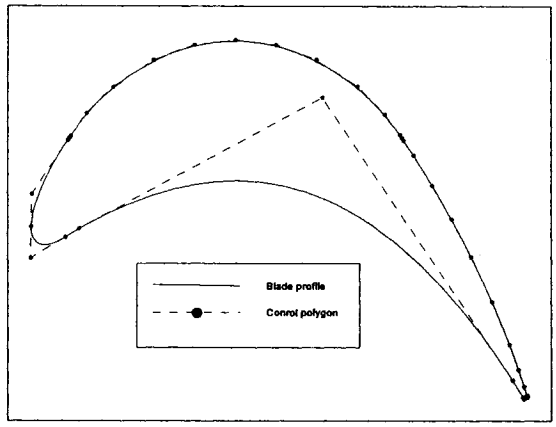

Figure 3.5: A NURBS approximation of the ETU-4 section 1 geometry [4].

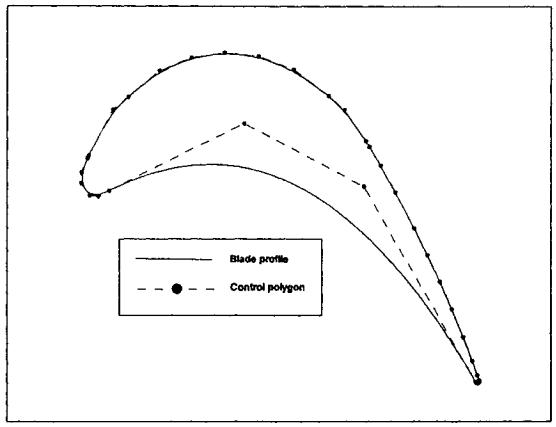

Figure 3.6: A NURBS approximation of the ETU-4 section 2 geometry [4].

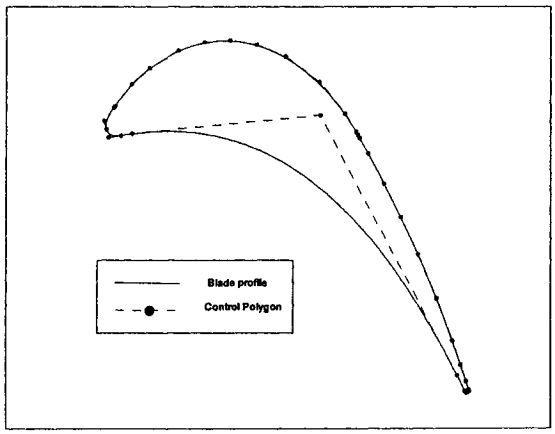

Figure 3.7: A NURBS approximation of the ETU-4 section 3 geometry [4].

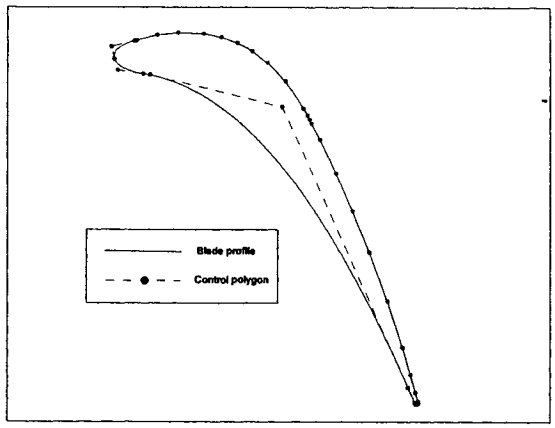

Figure 3.8: A NURBS approximation of the ETU-4 section 4 geometry [4].

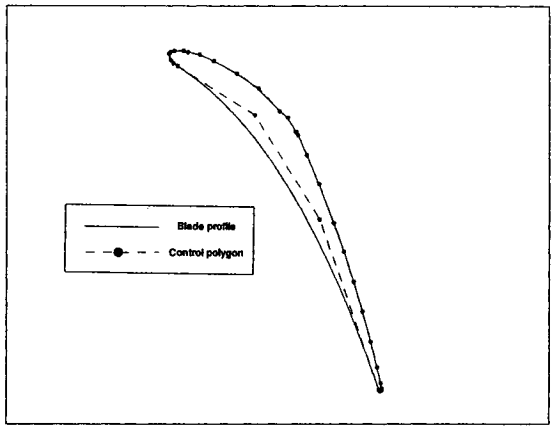

Figure 3.9: A NURBS approximation of the ETU-4 section 5 geometry [4].

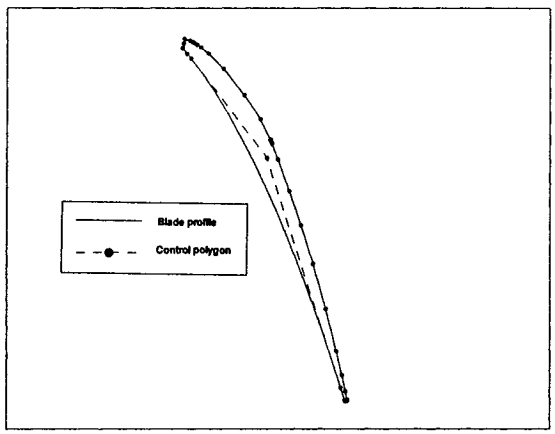

Figure 3.10: A NURBS approximation of the ETU-4 section 6 geometry [4].

impact the flow field) and the pressure field along the blades as obtained from a CFD analysis. These two criteria will provide valuable information about the accuracy and the quality of each approximation.

For this work, the DFVLR turbine blade, which is a typical subsonic turbine cascade, is chosen as the study case. Its geometry is given in [4] as a set of x and y-coordinates. Several geometric features of the blade are also provided, and that helped to determine the remaining set of designer parameters, all of which are used as input to the MRATD model. It is important to note that obtaining the original DFVLR profile exactly is not critical, since the objective is not to compare the original DFVLR geometry with the MRATD approximation, but to compare the MRATD profile with the one-NURBS and five-NURBS curve approximations. Nonetheless, it is quite possible to produce a geometry that is almost exactly that of the DFVLR by optimizing the x and y-coordinates of the control points and the weights in the NURBS parameterization, as was shown in Mansour and Ghaly [7]. Once the design parameters are determined for the MRATD model, it is necessary to define the five-NURBS curve parameterization in terms of degree, number of control points and continuity of the function. To determine those NURBS properties, one must keep in mind that we are comparing two different NURBS approximations. To have a fair comparison, this involves the development of two fairly similar NURBS functions in terms of degree of curve, and number of control points, while maintaining the smoothness of the curve at all locations. For the one-NURBS approximation, a 3rd order curve is used, with 23 control points around the entire profile, of which 6 are used for the suction side. On the other hand, the five-NURBS curve uses a combination of NURBS functions and B-spline functions. Note also that, for every degree of continuity imposed across junction points, an additional control point is

necessary. This means that inherently, the five-NURBS curve approximation will incorporate more control points on the suction side for the same level of accuracy of the one-NURBS curve approximation. Also, since the one-NURBS curve is optimized in order to minimize the number of control points, it is not surprising that a smaller number of control points are needed as compared to the five-NURBS curve. Therefore, the final result is a 3^{rd} order curve, with C^2 continuity and a total of 32 control points, of which 9 are used for the suction side.

To assess the smoothness of the NURBS approximations with respect to the MRATD model, it is important to study the curvature. The curvature is an extremely revealing feature when it comes to curve smoothness, and in CFD, the curvature of a profile will have a major influence on the flow-field properties such as the static pressure or the isentropic Mach number. Aerodynamicists also use the blade curvature profile to help them design suitable blade geometries. They usually have a general idea of the curvature distribution they wish to have and try to adjust the blade profile accordingly. Figures 3.11 and 3.12 show the curvature plots for the five-NURBS, one-NURBS and MRATD representations. The curvature is computed based on the x and y-coordinates of the blade using a three-point approximation. One can easily recognize the leading and trailing edge regions as well as the uncovered surface by the abrupt changes in curvature. Conversely, the rest of the curvature distribution on the suction side is constrained between zero and three, which are relatively small values considering the trailing edge peak of 220. By taking a closer look at the MRATD curvature plot, one can observe a very smooth and well-behaved curve, except at the junction points where the curvature is discontinuous. This is expected since the curve that is used for the suction side is a conic section. Nonetheless, the two NURBS approximations still follow with relative

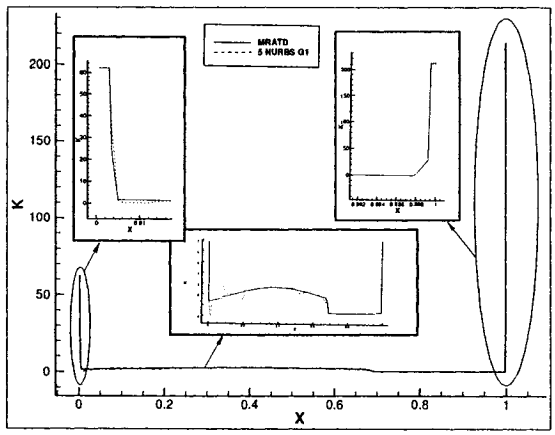

Figure 3.11: The suction side curvature distribution of the 5-NURBS approximation versus the MRATD model.

accuracy the same curvature as that of the MRATD model and succeed in capturing the curvature jumps to different degrees, even though there is some relatively small noise in the neighborhood of the LE, Throat and TE junctions. As will be shown later, this noise in the curvature is on such a small scale that its effect on the flow field is negligible.

Further, this noise can be significantly reduced by raising the degree of the curve and by adding a greater number of control points. For instance, if we take the five-NURBS parametrization as an example and the DFVLR profile used previously, and fit a 10^{th} degree NURBS function through that geometry with 30 control points on the suction side surface, we get the curvature profile shown in Fig. 3.13. This figure clearly shows that the NURBS curve is very smooth, continuous and follows the MRATD curvature distribution very precisely, especially on regions that are away from the junction points. However, near the leading edge junction point on the suction side, the NURBS function experiences a sharp overshoot in the curvature. This problem is mainly due to the C^2 continuity condition imposed on the geometry at key point 3. Nonetheless, this overshoot can be removed as shown in Ch. 4 by

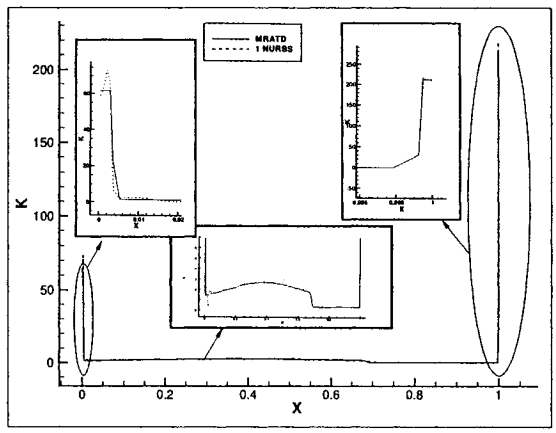

Figure 3.12: The suction side curvature distribution of the 1-NURBS approximation versus the MRATD model.

modifying one control point.

3.9. NURBS Model Accuracy Assessment

To assess the two NURBS approximations, their effect on the pressure distribution is presented in this section. A CFD analysis was carried out and the results thus obtained are compared with those of the MRATD model.

In the CFD analysis, the steady two-dimensional turbulent compressible flow in a linear cascade is simulated using a cell-vertex finite volume space discretization method on an unstructured triangular mesh. The steady state solution is reached by pseudo-time marching the Reynolds-averaged Navier-Stokes (RANS) equations using an explicit five-stage Runge-Kutta scheme. Baldwin-Lomax turbulence model was used to represent turbulence effects. Local time stepping and implicit residual smoothing were used for convergence acceleration. The non-linear blend of second and fourth order artificial viscosity was found to be successful in capturing shocks and eliminating pressure-velocity decoupling with minimal numerical diffusion. The

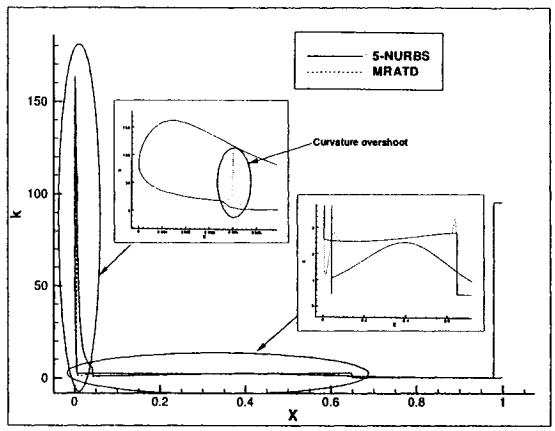

Figure 3.13: The curvature distribution of the DFVLR blade using the MRATD and a NURBS approximation with degree 10 and 30 control points on the suction side surface.

method of characteristics was used to impose inflow and outflow boundary conditions. The boundary conditions imposed at inlet are total pressure, total temperature and inlet flow angle. At the exit plane, the exit static to inlet total pressure is specified. More details on the discretization method for inviscid flows can be found in Ahmadi and Ghaly [8]. The implementation of RANS equations was provided by Daneshkheh [9] in the form of a computer code that was used to analyze the cases presented in this work.

For the DFVLR cascade, given in Fottner [4], the spacing-to-chord ratio is 0.687, the inlet and exit flow angles are 37.7^o and -49^o, respectively, and the exit-static-to-inlet-total pressure is 0.79. This case was simulated as a laminar flow case at a Reynolds number of 500. The CFD simulation was carried out on a relatively fine mesh so as to capture accurately the leading/trailing edge regions and to resolve the boundary layer region as well. The hybrid mesh, shown in Fig. 3.14, was used in the three models presented in Sec. 3.8. It allows for an accurate resolution of the boundary layer as well as the LE and TE regions, with a reduced number of wall

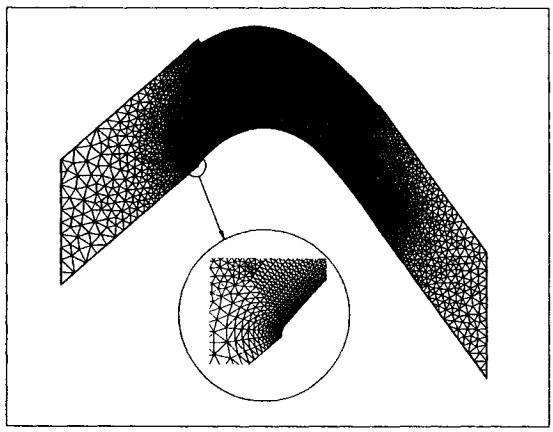

Figure 3.14: The hybrid mesh used for the CFD analysis.

nodes, and without any significant mesh distortion. The residual error for every simulation was reduced by six orders of magnitude.

The pressure distribution on the blade, which is shown for the three cases in Figs. 3.15 and 3.16, is practically identical, except near the leading and trailing edge regions, where one can observe a slight but negligible difference. These results confirm the quality of the approximations, and support the assumption that the curvature noise effects on the pressure are negligible. Furthermore, if one wishes to obtain identical results as those of the MRATD model, one can impose C^1 continuity across the junction points, thus exactly replicating the MRATD geometry at the junctions by using NURBS.

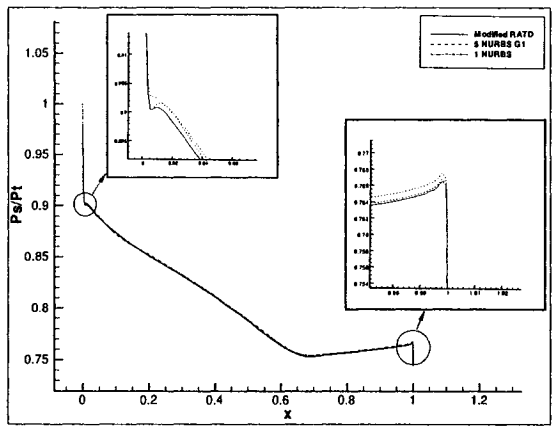

Figure 3.15: Suction side pressure distribution.

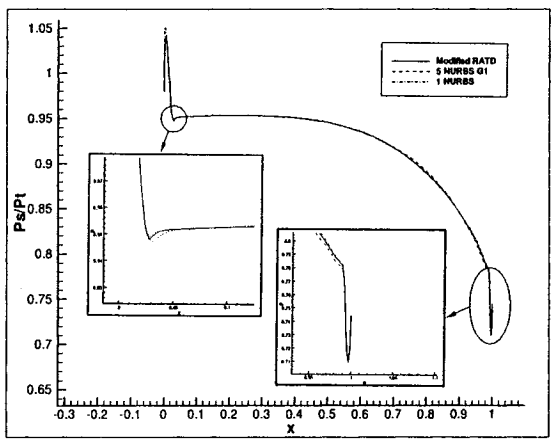

Figure 3.16: Pressure side pressure distribution.

Chapter 4

Shape Optimization Using the MRATD Model

The MRATD model, which is a low-order global representation of turbine baldes, and the five-NURBS parametrization, which gives a high order flexible and accurate representation, were successfully developed and assessed for their flexibility, smoothness and accuracy. However, in order to fully evaluate their performance, they need to be tested in an aerodynamic optimization process.

In this chapter we propose a strategy that will allow for an efficient, accurate and robust optimization scheme. We begin with an aerodynamic shape optimization using the low-order global blade representation, the MRATD model. This step is necessary in order to effectively perform a global sweep of the design space, hence improving the chances of locating regions where a potential optimal solution might be found. Once a set of MRATD design parameters is found, we propose an approach to continue with a local performance improvement using the NURBS parametrization. (This also demonstrates the power of the NURBS representation.) We propose to take the global optimal case obtained from the MRATD model, locate the regions where the blade profile behaves poorly and focus on improving the blade smoothness near the LE.

The optimization scheme used in this chapter is the object of the Ph.D. thesis work of Mr. Mengistu, graduate student in the Mechanical and Industrial department, who should be defending his thesis soon, [10].

4.1. Optimization Problem Definition

For this evaluation, the 3^{rd} section of the ETU-4 turbine blade, a typical 3D blade with a free vortex design, is used as a case study. Its geometry was given in [4] as a set of x and y-coordinates. Several geometric features of the blade were also provided, and that helped to determine certain designer parameters that are used as input for the MRATD model.

The CFD analysis code is the same as the one described in Sec. 3.9. The mesh used for this work is a semi-structured H-mesh with 12000 points. The CFD simulations are stopped when the norm of the residual on the momentum equations reaches 10^{-8}, which requires approximately 50 minutes of CPU time on a pentium 4 PC with 3 GHz and 2GB of RAM. The original and the approximated profiles are shown in Fig. 2.10. The MRATD design parameters for the MRATD approximation are listed in Table 4.1. The turbine blade rotates at 7500 rpm, and passes an air mass flow rate of 7.8 kg/sec. The inlet pressure and temperature are 2.6 bar and 413 K respectively, the outlet pressure is 1.022 bar. The blade has an efficiency of 0.913 and a stage reaction of 0.5. From these data, the inlet conditions used in the numerical simulation are: the total pressure and temperature of 135404.7 Pa and 341.7 K respectively, the absolute inlet velocity of 79.1 m/s at an angle of $\alpha_1 = 6.00^o$ (see Fig. 4.1).

For this optimization, the MRATD model is used to generate the blade profiles, however the NURBS parametrization is used in generating the geometry that is

Design Parameter	Value
Number of Blades	30
Radius	0.162 m
Axial chord	0.0396472 m
Tangential chord	0.031 m
Throat	0.01330
Unguided turning	12.00^o
Trailing edge radius	0.00022
Inlet metal angle	39.4^o
Exit metal angle	-66.0^o
Inlet wedge angle for suction side	15^o
Inlet wedge angle for pressure side	30^o
Outlet wedge angle for pressure side	2.5^o
Maximum thickness	0.01065
Axial location of maximum thickness	0.35
Minor diameter of the ellipse leading edge	0.002
Major diameter of the ellipse leading edge	0.005

Table 4.1: Design parameters for the ETU-4 section 3 turbine blade.

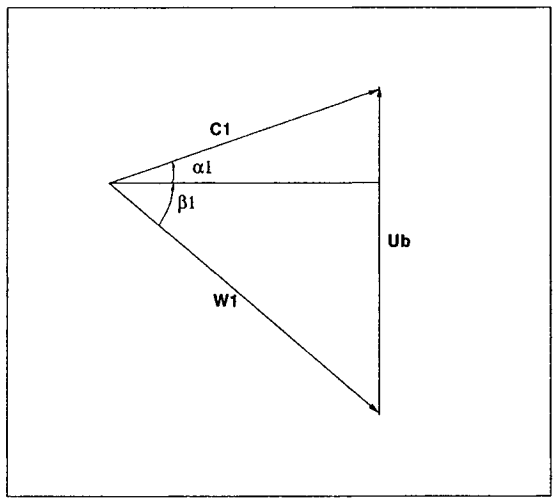

Figure 4.1: Velocity triangles for the ETU-4 section 3 turbine blade.

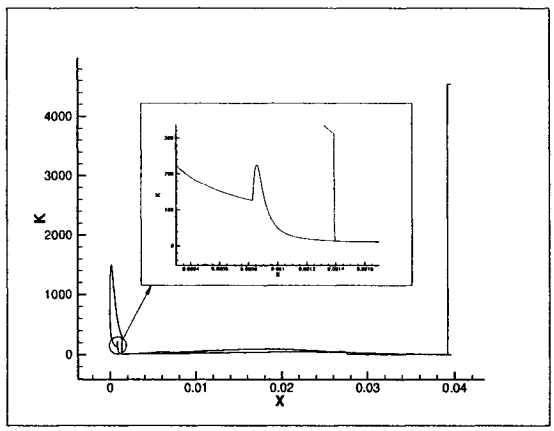

Figure 4.2: Curvature of the NURBS parametrization of the section 3 ETU turbine blade (Original).

meshed and analyzed in the optimization scheme to ensure 2^{nd} order continuity at the junction points. The NURBS parametrization for this case uses a 5^{th} degree curve with 13 control points on the suction side region. The curvature profile of the NURBS parametrization is smooth everywhere across the suction side except at the junction points where the curvature experiences a sudden jump (see Fig. 4.2).

The CFD results confirm the accuracy of the NURBS parametrization with respect to the MRATD model in terms of pressure distribution (see Fig. 4.3). From the CFD analysis, and the resulting pressure distribution, we are made aware of some important flow properties. First the pressure side region behaves very well; the flow is accelerating across that region of the blade, and the pressure decreases gradually, eliminating any chances of flow separation. In contrast, the suction side region, which is a much more sensitive region, behaves rather poorly, especially near the junction points and is characterized by a diffusion region between 65% chord and the TE where static pressure is rising. The sharp pressure fluctuations along the blade surface can result in a poor flow behavior. This further indicates that the suction side region needs to be redesigned in order to improve the blade performance.

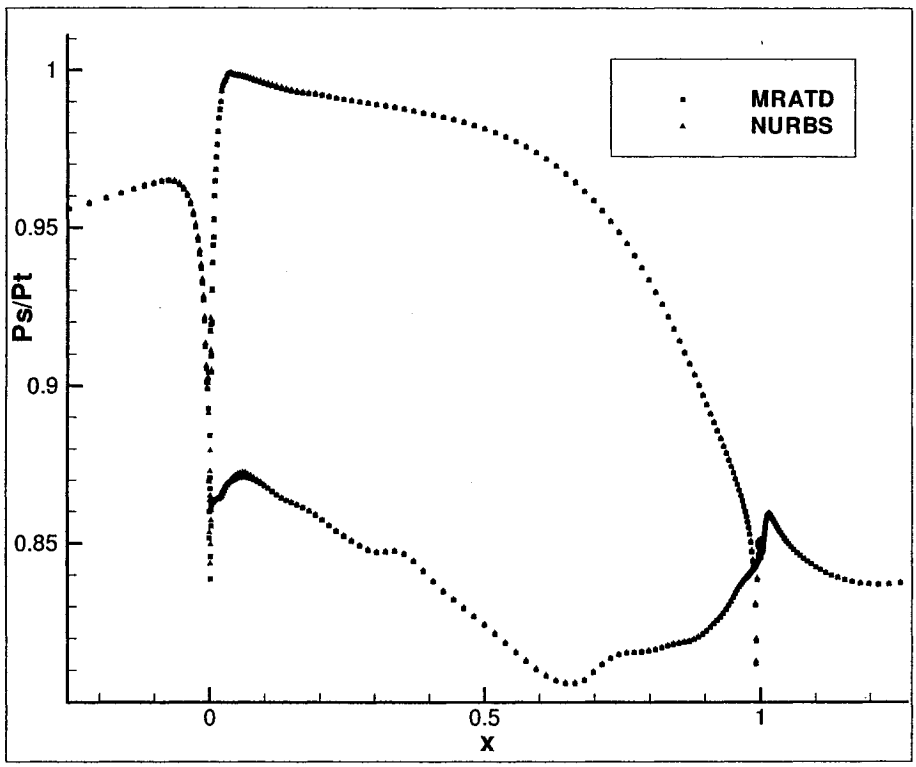

Figure 4.3: The pressure distribution of the section 3 ETU-4 blade geometry using the the MRATD model and its NURBS parametrization.

The objective of this exercise is to optimize the blade performance at the design point. The optimization uses a genetic algorithm for a global sweep of the design space. The objective function is a weighted sum of all individual objectives, namely to maximize efficiency, minimize the total pressure loss, and is penalized with all the constraints in order to maintain or increase the pressure ratio and maintain the same operating conditions (i.e., fixed rotor speed, fixed inlet and outlet boundary conditions). All the objective parameters are included in the objective function, although the weights associated with each objective might vary according to importance. As will be shown later, varying the weights has a significant impact on the outcome of the resulting optimal blade shape.

The objective function in this case is computed using an artificial neural network (ANN) which is a low-fidelity approximation model. This avoids using the expensive and time consuming CFD code for computing the objective function in the optimization cycle. To train and test this ANN model, fifty candidate profiles are generated with the MRATD model by randomly varying the design parameters by $\pm 25\%$. Of these fifty cases 35 are selected for ANN training and 15 for ANN testing, with a total training time of 6 hours and a 3% maximum error of objective function evaluation during testing.

4.2. Optimization Using the MRATD Model

Since the flow is most sensitive to the shape of the suction side region and the uncovered surface, the MRATD design parameters controlling both of these regions, namely the tangential chord, the throat, the unguided turning, the suction side inlet wedge angle, the maximum thickness and the major diameter for the leading edge ellipse, were selected as the design parameters.

The results of the optimization are presented in Tables 4.2 and 4.3, as well as Figs. 4.4 and 4.5. There are several optimal design solutions, designated as $optimal_1$ to $optimal_3$, each one corresponding to a specific objective function. The objective function is varied by modifying the weights associated to the various objective parameters. For all optimal cases, the design conditions for the blade are in general equivalent to the original design specifications. From the results, we notice that the most encouraging solution is $optimal_2$. For that case, the pressure loss is reduced by 0.386%, the efficiency is increased by 0.388%, the pressure distribution is almost unchanged, the maximum mass flow rate was increased by 0.52% and the inlet and outlet flow conditions are maintained to the original specifications.

A few remarks should be made at this point. First, note the global shape change, depicted in Fig. 4.4, results from changing only six design variables. The relatively low number of design variables allows the numerical optimization method to find the global optimum. Given that the design variables are also the designer's parameters, any geometry that is generated during the optimization process is a feasible geometry, which effectively limits the design space to feasible geometries and ensures that the optimized geometry will also be a feasible one.

Let us also note that, if this geometry were to be parametrized using NURBS, about thirty control points would be needed to represent the entire blade shape, which would result in approximately sixty design variables, instead of 6 for the MRATD representation. This would result in a much more complicated design space and the optimizer might have difficulty carrying out the optimization or may not produce a feasible blade geometry. Such an observation might explain the absence of any literature on the use of NURBS in shape optimization that would result in a global change such as the one shown in Fig. 4.4.

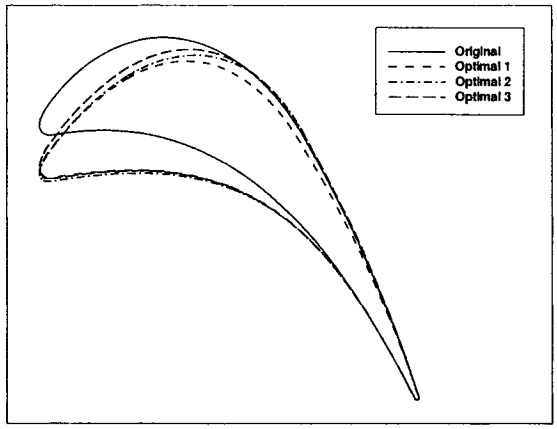

Figure 4.4: The optimized profiles vs the original for section 3 of the ETU blade.

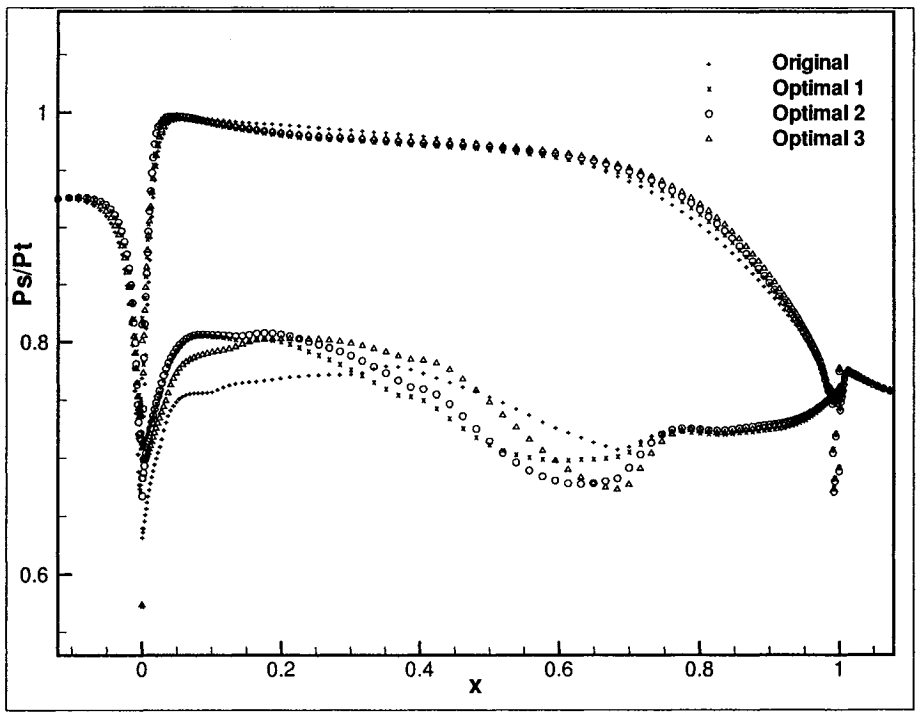

Figure 4.5: The pressure distribution for all the optimal cases provided by the optimization routine.

Design Parameters	Original	Optimal 1	Optimal 2	Optimal 3
Tangential Chord	0.031	0.02639	0.02609	0.02653
Throat	0.01331	0.01388	0.01312	0.01316
Unguided Turning	12.0	12.92	9.95	12.3
SS Inlet Wedge Angle	15.0	12.63	14.83	14.44
Maximum Thickness	0.01065	0.01146	0.0122	0.0127
Major Diameter of LE Ellipse	0.0020	0.001664	0.001828	0.002172

Table 4.2: Design parameters for the ETU-4 section 3 turbine blade.

Cases	Ploss %	Eff. %	PR in/out	Mass	Inlet flow angle, deg	Exit flow angle, deg
Original	17.263	91.674	1.51849	0.191	57.37	-65.83
Optimal 1	14.186	91.681	1.51000	0.201	56.02	-64.90
Optimal 2	16.877	92.062	1.51800	0.192	57.35	-65.93
Optimal 3	17.045	92.052	1.51851	0.191	57.47	-66.00

Table 4.3: Design parameters for the ETU-4 section 3 turbine blade.

It should also be noted that this case involves an already good design and given that the flow is subsonic and attached, the only aerodynamic loss is the profile loss due to boundary layer development. This type of loss is the hardest to reduce. Nonetheless, the optimization process was able to suggest a better performing blade shape.

4.3. Local Performance Improvement Using the NURBS Parametrization

The global optimization results obtained in the previous section are very encouraging, given that the original geometry has a high efficiency (91.67%), that the case is subsonic and that only the global design parameters are used in the optimization

scheme. Several optimal design points are determined, and the design is improved with respect to the original one. Further, since the flow is subsonic across the entire blade surface, increasing the efficiency by 0.388% is not a trivial matter since the aerodynamic losses are mainly due to 2D viscous effects. Local optimization on the other hand is necessary in order to narrow down the results to a global optimal point. Work on local 2D aerodynamic optimization of turbine blades has been implemented using B-splines or Bézier curves by [3, 11, 12]. Furthermore, it was demonstrated in [13] that similar schemes that use NURBS as the parametrization function are much more robust in finding an optimal geometry due to their increased flexibility and accuracy.

By studying the curvature distribution and the aerodynamic characteristics of the initialization geometry, one would determine the regions of the blade geometry that require some improvement. For instance, after studying the curvature and the pressure distribution of the optimized MRATD geometry, we observe certain undesirable features (see Figs. 4.2, 4.5). This adverse behavior needs to be eliminated in order to see any further significant improvement in the blade performance.

In order to demonstrate the added flexibility of using NURBS functions in order to modify the blade geometry, a simple exercise is carried out on the $optimal_2$ geometry of section 3 of the ETU-4 blade. The curvature of this geometry at the junction point between the LE and the suction side is characterized by a sharp rise, that would be practically impossible to dampen using a low-order geometric model (see Fig. 4.2). In order to eliminate that curvature jump, it is necessary to adjust the control points and the corresponding weights that define that region. This exercise is relatively simple to perform, since the curvature distribution will either improve or worsen depending on the direction (negative or positive) and size of the change in control point location and weight. An example of an improved curvature is shown

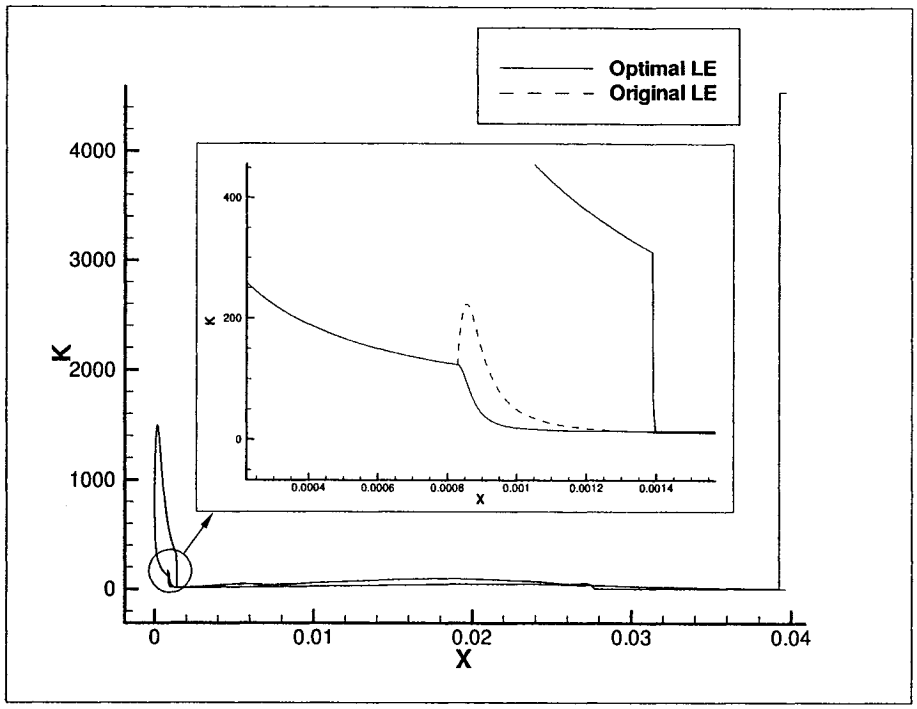

Figure 4.6: Improved curvature of the NURBS parametrization of the section 3 ETU turbine blade.

in Fig. 4.6, where one control point and weight were manually changed and resulted in removing entirely the overshoot in curvature that was prevailing near the LE. It should be noted that having the added flexibility of the weights in the NURBS function was critical in order to obtain the given results.

Finally, in order to assess the impact of this curvature adjustment on the aerodynamic performance of the blade, a CFD analysis is carried out on both geometries. The results, shown in Fig. 4.7, reflect an improved pressure distribution where the flow experiences a more controlled and less severe pressure increase at the leading edge on the suction side, which implies an increased loading of the blade. Furthermore, the aerodynamic efficiency increases by 0.055% to 92.117% and the pressure loss is reduced by 0.327% to 16.55%. Such an improvement can be automatically captured when using a local optimization scheme coupled to the present NURBS

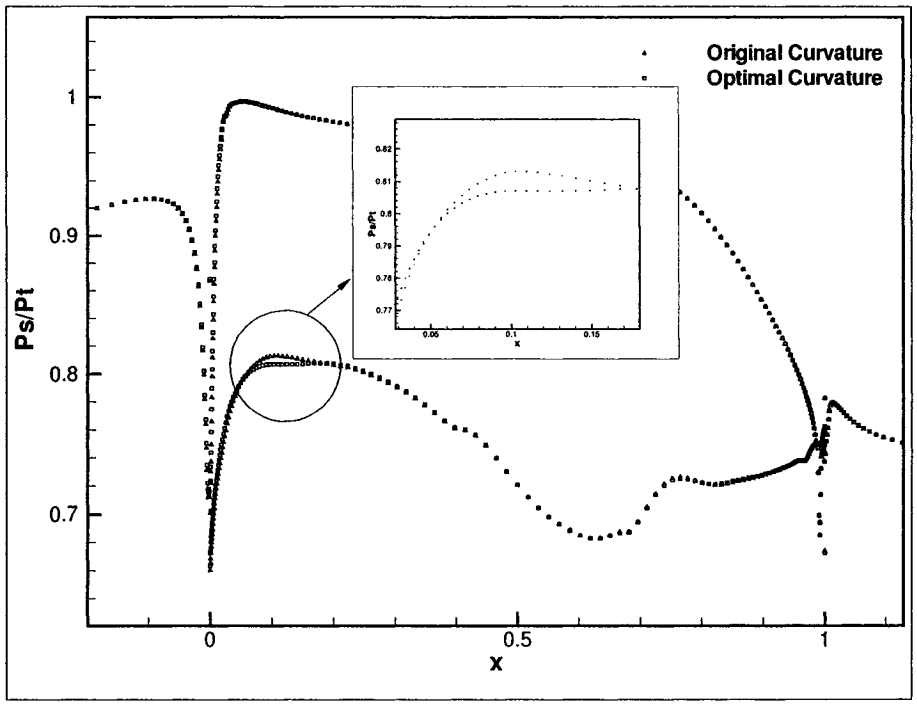

Figure 4.7: Pressure distribution of the ETU section 3 geometry original curvature v.s. optimal LE curvature.

parametrization.

Chapter 5

3D NURBS Parametrization of Turbine Blades

In the previous chapters, two geometric models were presented and were assessed for 2D aerodynamic optimization. However, in highly loaded gas turbine blades, the flow is strongly three-dimensional particularly when the chord to height ratio is large, hence 3D aerodynamic effects become significant and should be accounted for in order to capture accurately all the flow characteristics. In order to properly design a 3D turbine/compressor blade, the 2D sections at different radial locations can be first designed and analyzed using the MRATD model and its NURBS parametrization; this is followed by a CFD analysis of the 3D geometry. The 3D analysis can be taken one step further by performing a 3D aerodynamic optimization of the blade geometry. In this chapter, a 3D NURBS geometric model for turbine/compressor blades is developed and assessed. This model should have the same characteristics as those of the 2D models so as to be suitable for 3D aerodynamic optimization, i.e., the model must be flexible, accurate, efficient and robust. It was found that the NURBS skinning technique is one of the best possible solutions to fit a smooth surface through several 2D blade sections stacked in the radial direction. First, the model is presented in detail, and afterwards it is assessed in terms of smoothness

and accuracy.

5.1. 3D NURBS Model

In order to design a 3D blade surface, an aerodynamicist usually starts by designing several 2D blade profiles at different radial locations. A 3D geometry is then obtained by fitting the 2D sections with a smooth surface using a given surface fitting technique. Several techniques were studied for this work, and it was concluded that the NURBS skinning technique allows for a more accurate, smooth and straightforward means of accomplishing this task. The methodology used to develop the 3D model is thoroughly discussed next.

A NURBS surface is defined as follows:

$$\mathbf{S}(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)\mathbf{P}_{i,j}\mathbf{w}_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)\mathbf{w}_{i,j}}, \tag{5.1}$$

where $\mathbf{S}(u,v)$ are the x, y and z-coordinates of the surface being generated, n is the number of control points in the u direction and m is the number of control points in the v direction, $\mathbf{P}_{i,j}$ are x, y and z-coordinates of the control point, $w_{i,j}$ the weights and $N_{i,p}(u)$ are the basis functions in the u direction and $N_{j,q}(v)$ are the basis functions in the v direction defined on the non-periodic and nonuniform knot vectors $\mathbf{U}$ and $\mathbf{V}$ [6]:

$$\mathbf{U} = \{\underbrace{a,\ldots,a}_{p+1}, u_{p+1},\ldots,u_{r-p-1},\underbrace{b,\ldots,b}_{p+1}\} \quad \mathbf{V} = \{\underbrace{c,\ldots,c}_{q+1}, v_{q+1},\ldots,v_{s-q-1},\underbrace{b,\ldots,b}_{p+1}\}$$

where $r = n + p + 1$ and $s = m + q + 1$.

In order to obtain the 3D surface using Eq. 5.1, each 2D NURBS function must be correctly formed. Firstly, using the 2D NURBS parametrization and the

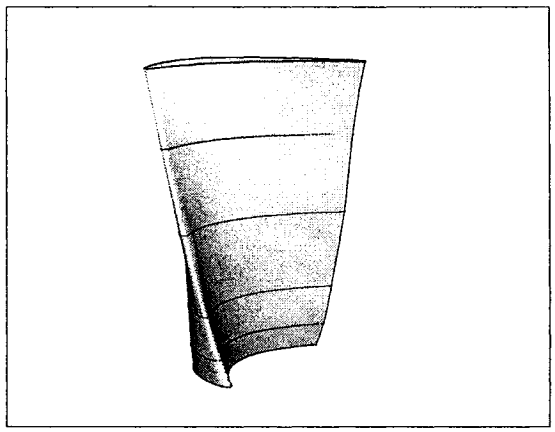

Figure 5.1: The 3D ETU-4 Blade Generated Using NURBS Skinning Technique.

knot refinement techniques discussed in Sec. 3.5, all 2D NURBS parametrization used for generating the 3D blade are made compatible in terms of number of control points and identical knot vectors. Once this is complete, the curve interpolation technique discussed in Sec. 3.4 is used in order to determine the new set of control points in the radial direction. Therefore, the 3D blade surface is defined by the 2D blade sections that are stacked radially, and the x, y and z-coordinates of the blade can be obtained and plotted. More detail about this technique can be found in [6]. Figure 5.1 shows the approximate representation of the ETU-4 blade where sections at various radial locations from hub to tip are shown in Figs. 2.8 to 2.13 respectively, as well as in Fig. 5.2.

5.2. 3D NURBS Model Assessment

In order to assess the 3D model in terms of accuracy and smoothness, the blade shape and curvature evaluated at different 2D radial sections are examined. When moving radially out from hub to tip, the blade shape and curvature should transit smoothly at radial positions located between any two original stacked sections. Figures 5.3

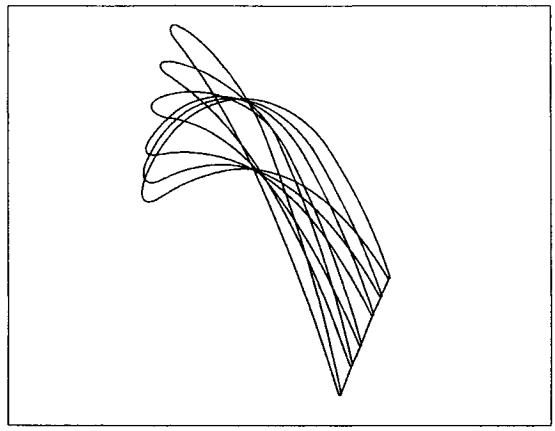

Figure 5.2: The 6 stacked sections used to generate the 3D surface of the ETU-4 blade.

to 5.6 show the blade geometry and curvature at 32.7% and 59.8 % span of the ETU-4 blade generated using the 5-NURBS parametrization technique applied to the original 2D sections. The blade shapes, given in Figs. 5.3 and 5.5, look smooth however a close inspection of the curvature distribution, given in Figs. 5.4 and 5.6, reveals some noise in the curvature at points where the original sections have a multiplicity in the knot vector. Two approaches can be used to remove this noise. One way is to eliminate the multiplicity in the knot vector and optimize the five-NURBS curves representation using the approach developed in [5]. Another approach would be to represent each of the 2D sections with one NURBS curve again using the work of Ghaly and Mengistu [5].

A possible approach to designing 3D sections is to do it in two steps. Start first with a global search of the 3D design space using the MRATD model at selected 2D sections that are then fitted with a 3D NURBS to generate a continuous 3D blade representation. The MRATD model ensures a small number of design variables and the NURBS skinning technique ensures a smooth 3D profile. An alternate approach to using the designer parameters at the selected 2D sections is to follow

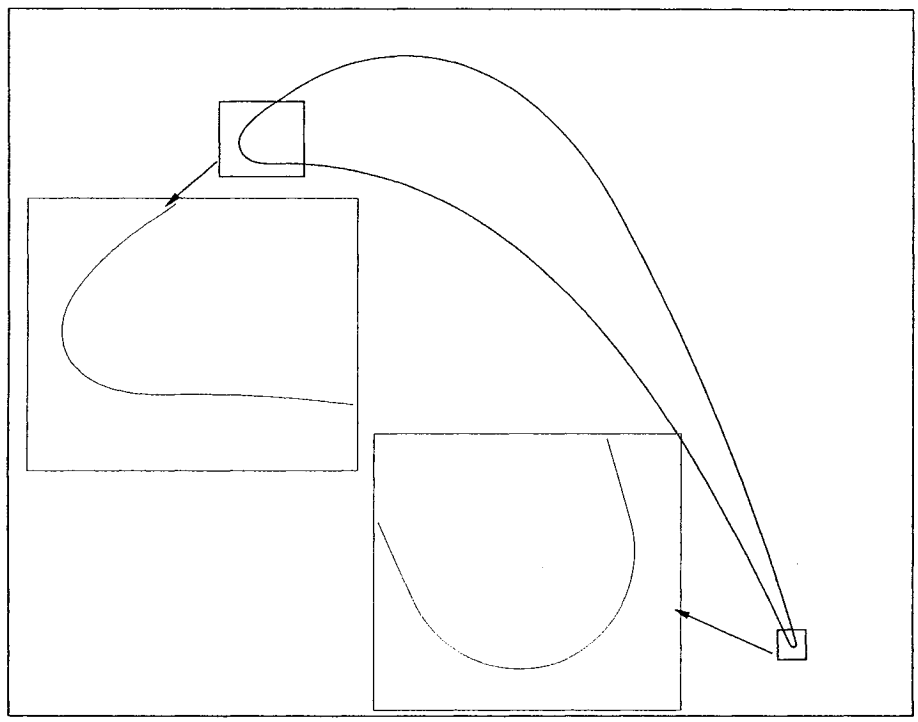

Figure 5.3: The ETU 3D blade profile at 32.7% span using a 5-NURBS parametrization technique.

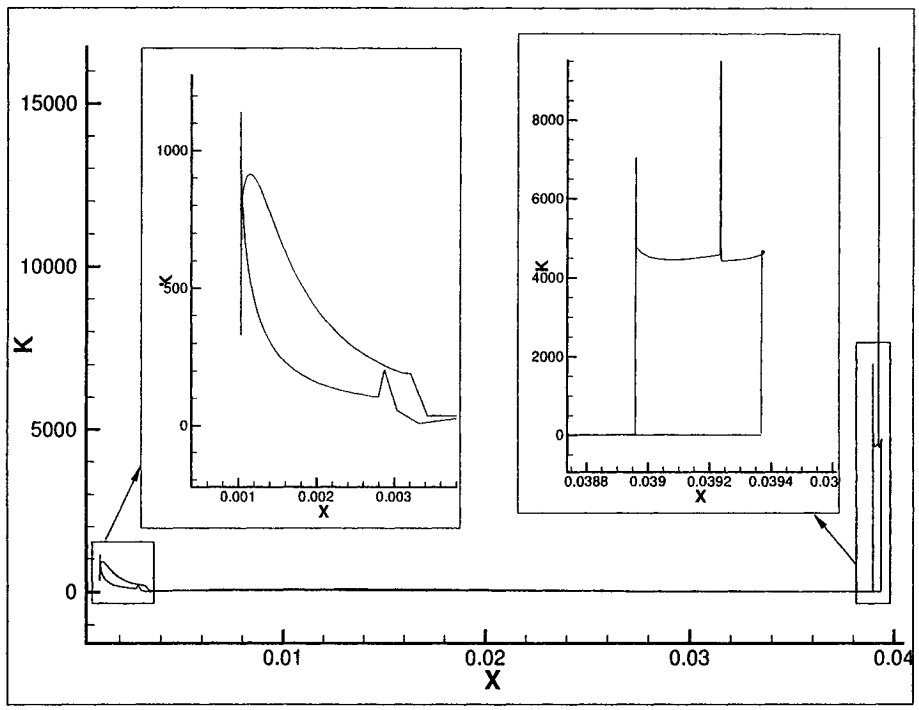

Figure 5.4: Curvature of the ETU 3D blade at 32.7% span using a 5-NURBS parametrization technique.

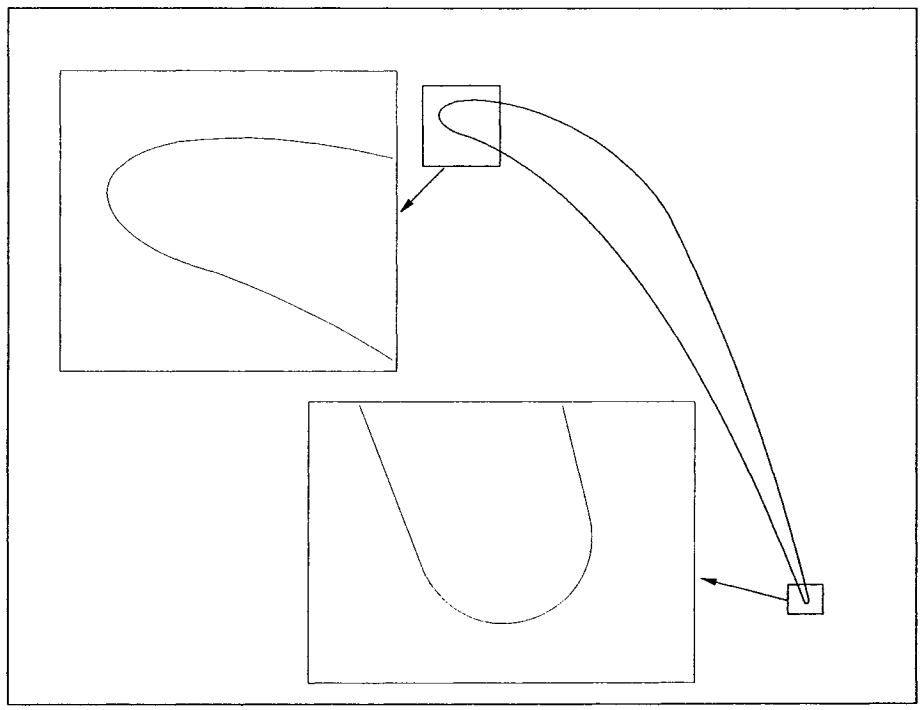

Figure 5.5: The ETU 3D blade profile at 59.8% span using a 5-NURBS parametrization technique.

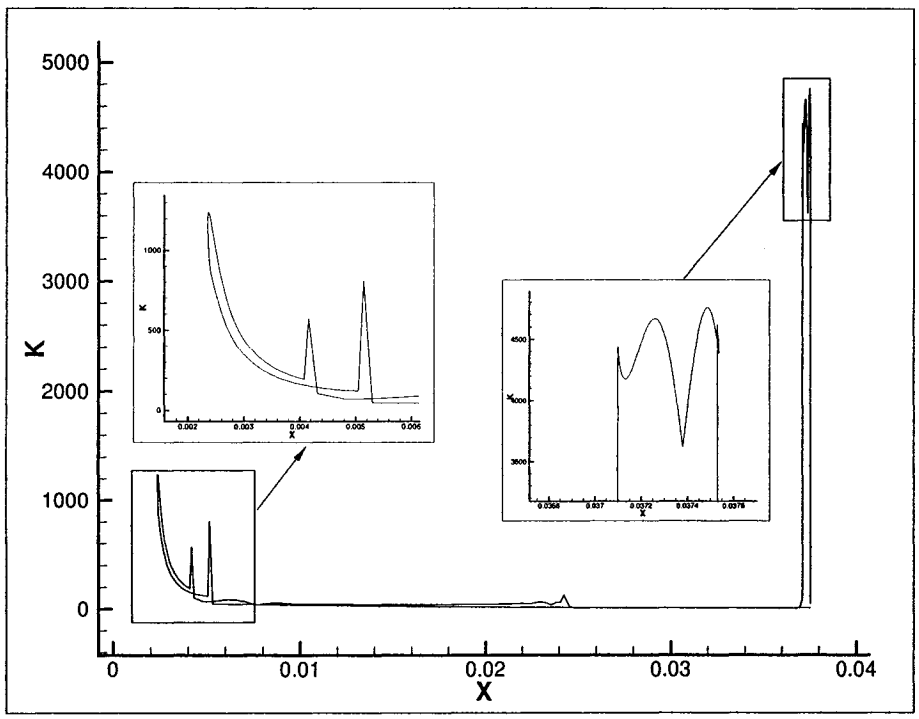

Figure 5.6: Curvature of the ETU 3D blade at 59.8% span using a 5-NURBS parametrization technique.

the approach developed by Lai and Yuan [14], where the 2D sections of the 3D blade are designated as design parameters and they are manipulated using transformation matrices in translation, rotation and scaling. Either of these two approaches allows for a significant reduction in the complexity of the design space, eliminates a large region of unfeasible designs, and allows enough flexibility for the optimizer to explore different design alternatives.

Once the global optimum shape is obtained, local refinement of the blade shape can then be done using the NURBS representation of the blades. Note that only very few control points would be used in the local refinement of the already globally optimized 3D blade geometry.

Chapter 6

CONCLUSION

6.1. Completed Work

The target for this work was to develop an implicit geometric representation of gas turbine blades that incorporate the designer parameters and yet is robust, flexible and accurate so as to be used efficiently in automatic shape optimization. This led to the development of the MRATD model, and a parametrization of that model using NURBS functions. The flexibility of both models was measured by reproducing several existing turbine blade profiles with remarkable accuracy. Also, the smoothness and accuracy of the five-NURBS curve parameterization was also measured and compared with those of the MRATD model as well as the one-NURBS representation. Differences between the five-NURBS curve and the one-NURBS representations were found to have insignificant effect on the pressure field. In addition, the MRATD model was successfully implemented in an optimization scheme. Several optimal blade geometries were obtained for different choices of the objective function. The five-NURBS parametrization was also used in order to smoothen the curvature at the leading edge near the junction point between the suction side and

the leading edge arc. This was accomplished through minor profile modifications using the NURBS control points and as a result, the aerodynamic performance of the blade was enhanced quite significantly.

Finally a 3D geometric model is developed and assessed based on the curvature distributions obtained across the radial direction. Through the curvature analysis, it is clear that the 5-NURBS parametrization is disadvantaged by the multiplicities in the knot vector, creating discontinuities in the curvature at various locations across the airfoil span.

6.2. Future Work

For future work, several issues need to be accounted for with respect to this work. Firstly, a fully automated high-fidelity optimization process must be elaborated and performed in order to fully validate the NURBS parametrization of the MRATD model. Furthermore, a more detailed assessment of the 3D geometric model must be performed. The model should be evaluated through a CFD analysis in order to measure its performance with respect to existing blade geometries. Further, a 3D aerodynamic optimization can be carried out to evaluate the feasibility of such a method, and whether the model behaves adequately when the geometry is modified. Finally, this work can be easily extended for compressor applications. Acknowledging the fact that 2D compressor blade geometries can be modelled based on a similar set of design parameters that one finds in the MRATD model, it would be interesting to study how such a model would perform in an optimization scheme. This work has set the foundation for the realization of the above mentioned future objectives.

Bibliography

[1] L. Pritchard, "An eleven parameter axial turbine airfoil geometry model," *85-GT-219*, 1985.

[2] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design.* Academic Press, 2nd ed., 1993.

[3] B. H. Dennis, G. S. Dulikravich, and Z.-X. Han, "Constrained shape optimization of airfoil cascades using a navier-stokes solver and a genetic/sqp algorthm," *ASME paper 99-GT-441*, 1999.

[4] L. Fottner, "Test cases for computation of internal flows in aero engine components," *AGARD-AR-275*, 1990. Propulsion and Energetics Panel.

[5] W. S. Ghaly and T. T. Mengistu, "Optimal geometric representations of turbomachinery cascades using nurbs," *Inverse Problems in Engineering*, vol. 11, no. 5, pp. 359–373, 2003.

[6] L. Piegl and W. Tiller, *The NURBS Book.* Springer, 1995.

[7] T. Mansour and W. Ghaly, "An implicit geometric representation of turbine blades using nurbs," *The Eleventh Annual Conference of the CFD Society of Canada*, 2003.

[8] M. Ahmadi and W. Ghaly, "A finite volume method for inviscid transonic cascade flow with solution adaptation on unstructured mesh," *CASI Journal*, vol. 44, pp. 175–181, 1998.

[9] K. Daneshkheh, "Private communications." Concordia University, Department of Mechanical and Industrial Engineering, 2004.

[10] T. Mengistu, "Aerodynamic design and optimization of turbomachinery blading." Concordia University, Department of Mechanical and Industrial Engineering, April 2005.

[11] A. Oyama, M.-S. Liou, and S. Obayashi, "Transonic axial-flow blade shape optimization using evolutionary algorithm and three dimensional navier-stokes solver," *AIAA paper 2002-5642*, 2002.

[12] S. K. T. Rogalsky and R. Derksen, "Differential evolution in aerodynamic optimization," *Canadian Aeronautics and Space Journal, Vol. 46, No. 4*, 2000.

[13] T. T. Mengistu and W. Ghaly, "Aerodynamic shape optimization of transonic cascades using an euler solver and a heuristic approach," *CASI*, April 28-30 2003.

[14] Y. yang Lai and X. Yuan, "Blade design with three-dimensional viscous analysis and hybrid optimization approach," *AIAA paper 2002-5658*, 2002.