Parameterization and design of transonic compressor blades

Master's Thesis in Applied Mechanics

HENRIK SKÄRNELL

Department of Applied Mechanics
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2013

Master's Thesis 2013:04

MASTER'S THESIS IN APPLIED MECHANICS

Parameterization and design of transonic compressor blades

HENRIK SKÄRNELL

Department of Applied Mechanics
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2013

Parameterization and design of transonic compressor blades
HENRIK SKÄRNELL

© HENRIK SKÄRNELL, 2013

Master's Thesis 2013:04
ISSN 1652-8557
Department of Applied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: +46 (0)31-722 10 00

Chalmers reproservice / Department of Applied Mechanics
Göteborg, Sweden 2013

Parameterization and design of transonic compressor blades
Master's Thesis in Applied Mechanics
HENRIK SKÄRNELL
Department of Apllied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology

Abstract

Economical and environmental requirements are driving the development of more efficient aircraft engines. For the compressor this has led to fewer stages and higher pressure ratios. To be able to fulfill these requirements while maintaining high efficiency and high stability, off design performance must be considered early in the blade design process. Recently, a new set-based design method accounting for stability at part speed as well as efficiency at the design point has been developed. This method makes use of a global optimizer where the end result is an optimal set of compressor stages, i.e. a pareto-front, which shows the trade-off between efficiency and stability. However, the optimal solutions are highly dependent on the blade parameterization tool as it sets the limitations to the allowed design space. In order push the pareto-front towards higher efficiency and higher stability, a new blade parameterization tool is developed where Bézier curves are used to represent the blade geometries. This tool, called Polly, is written in Matlab and is introduced into the existing workflow. To investigate its capabilities it is used to re-design the rotor and stator profiles positioned at 95% span of the first stage rotor of the low pressure compressor called Blenda. The investigation shows that the new parameterization is able to generate more efficient and stable blades compared to the current blade generator, called Volblade. One reason is that the suction side and pressure sides are created independently of each other, but at a cost of more parameters and longer optimization time.

Keywords: CFD, compressor, design, optimization, Bézier curve

Contents

Preface

The Master thesis presented in this report was performed during the fall of 2012 as a final part of the Masters degree at Chalmers University of Technology in Göteborg, Sweden. The work has been carried out for the department of Aerothermodynamics at GKN Aerospace Sweden AB in Trollhättan, Sweden. The thesis has been supervised by Lic. Lars Ellbrant currently a Ph. D. student at Chalmers University of Technology and employee at GKN Aerospace Sweden AB. Prof. Lars-Erik Eriksson at Chalmers University of Technology has been the thesis examiner.

I like to thank my supervisor Lars Ellbrant for the opportunity, support and sharing his knowledge in the subject with me. I like to thank Hans Mårtensson at GKN Aerospace Sweden AB for all the interesting discussions and great ideas during the project. I would like to thank my master thesis colleague Johnny Zackrisson for serving as a sounding board throughout the whole project. I would also like to thank all my colleagues at GKN Aerospace Sweden AB for making my stay there a great time.

Göteborg, April 2013
Henrik Skärnell

Nomenclature

C_p	Pressure recovery factor
f	Function
$\hat{f}$	Approximation of function f using radial basis functions.
k	Turbulence kinetic energy
lp#	Vector magnitude
P	Pressure
ps#	Vector magnitude
ss#	Vector magnitude
t	thickness
T	Temperature

Greek

α	Blade metal angle
γ	Stagger angle
η_p	Polytropic efficiency
ϕ	Radial basis function
μ	Wedge angle
σ	Scaling parameter

Subscripts

0	Total condition
1	Inlet plane
2	Outlet plane
le	Leading edge
ps	Pressure side
rel	Relative frame of reference
ss	Suction side
te	Trailing edge

1

Introduction

With rising fuel prices and larger awareness of environmental effects, such as noise and air pollution, the demand of quieter and more efficient aircraft engines are growing. This can be achieved by increasing engine component efficiency, bypass ratio and pressure ratio of the engine while keeping the total engine weight low.

The outline of a typical turbofan engine can be seen in Figure 1.1. In this configuration the air is accelerated backwards through the fan. Some of the mass flow bypasses the engine core to produce thrust while some is passed through the core to be able to power the fan. The ratio between these mass flows are known as the bypass ratio. The pressure of the air passed through the core is increased, first by a low pressure compressor and then by a high pressure compressor. Then it is lead through the combustion chamber where the temperature increase from the combustion of fuel with constant pressure. Work is then extracted from the hot high pressure gas through two turbines, first by a high pressure turbine and then later a low pressure turbine. The work extracted from the high pressure turbine powers the high pressure compressor through the high pressure-shaft, while the work extracted from the low pressure turbine power both the low pressure compressor and the fan through the low-pressure shaft. The reason for two separate turbines and compressors is the possibility to operate at two different speeds, thereby increase efficiency of the individual components. Each compressor and turbine component consists of one or more stages. Each stage in a compressor consists of a row of rotor blades followed by a row of stator blades. The working fluid is initially accelerated by the rotor blades, and then decelerated by the stator blades[1]. In the turbine the order of the stator and rotor blades are reversed in each stage as the turbine takes out work from the fluid instead of adding work as in the case of the compressor.

Current improvements to the compressor includes increased pressure ratio and/or using less stages in order to cut weight. Both leads to higher aerodynamic loading of each stage. As the aerodynamic loading increases it becomes difficult to maintain high efficiency while keeping sufficiently high stability margin along the compressors working line.

Traditional methods for blade design have only tried to maximize efficiency at the design point. With higher efficiency requirements and fewer stages this has resulted in lower stability margins. Now off-design performance must be considered early in the design process in order to increase stability margins. To keep costs down, CFD analysis together with an optimization tool can be used. However, the ability to find an optimal solution is highly dependent on the parameterization tool as this sets the limitations to

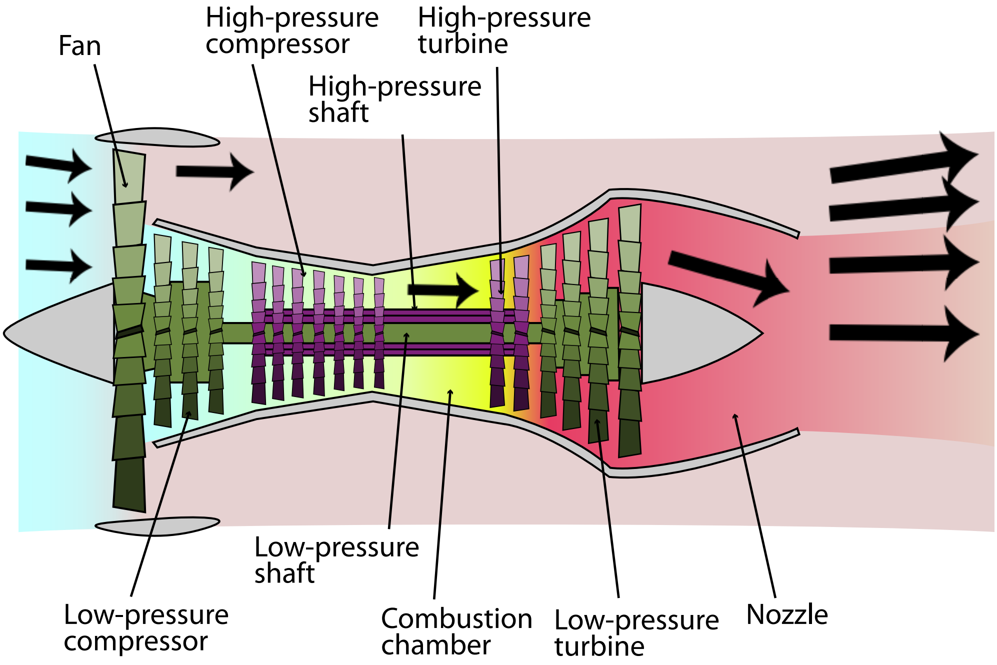


Figure 1.1: Typical turbofan engine.

the allowed design space. In order to be able to push the pareto-front towards higher efficiency and higher stability a new parameterization is suggested.

This new tool, called Polly, written in Matlab has to be further developed and introduced into the the existing workflow. In order to consider both the efficiency at design point and stability margins at off-design, a multi-objective optimization is performed to evaluate this new parameterization.

1.1 Previous research

The work in this thesis is based on a design method currently developed in a co-operation between GKN Aerospace and Chalmers University of Technology.[2]

The design process starts with preliminary design using basic mean line tools and loss correlations estimates. This is then used to create 3D blade using 2D blade profiles. If all parameters describing the 3D blade were used, the computational time required to optimize would be to long for preliminary design. The optimization process is instead divided into several steps. The first step involves a low complexity optimization of 2D blade profiles where only three different spans (10%, 50% and 95%) are considered. Then the objective functions are evaluated on stream-tubes created at these spans. By simplifying the problem to only consider stream-tubes, computation time decreases, which allows a higher number of design variables on each span to be explored. To obtain a 3D blade other spans along the stacking line are interpolated using the variables obtained from the optimal designs at each of the spans. This blade is then used as a starting point for the 3D optimization.

1.2 Scope of work

The focus of the work is to implement a new blade parameterization, called Polly, into the current design process and evaluate the aerodynamic performance, in terms of efficiency and stability. The results from the blade design are then compared to the results obtained from previous research using the Volblade parameterization.

Aspects such as aeromechanical integrity and limitation set by the manufacturing process are not evaluated, only accounted for by setting appropriate limits of the design variables.

2

Theory

THIS THESIS EXPECTS the reader to have knowledge in engineering and especially in the field of fluid mechanics. This chapter explains some theory needed for the understanding of the thesis which is not usually covered during such an education.

2.1 Bézier curve

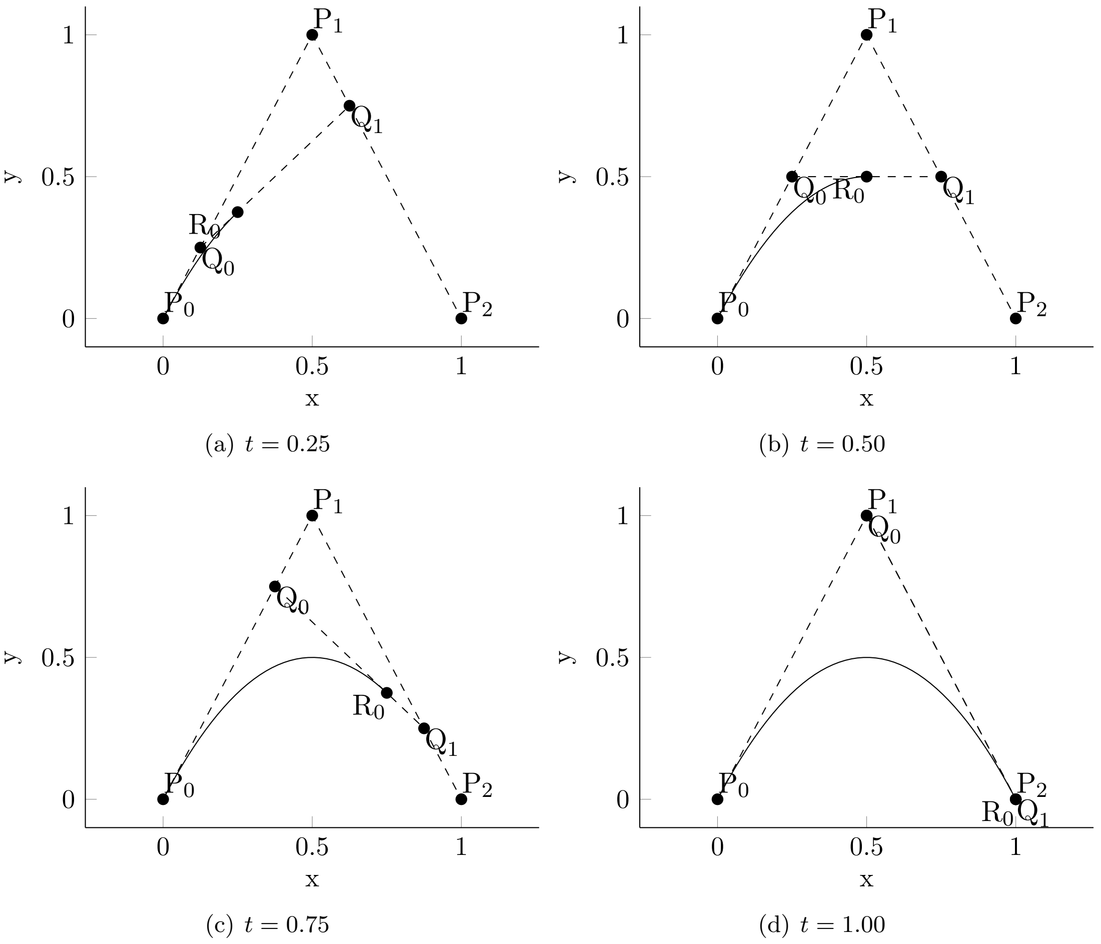
A Bézier curve is a parametric curve. It uses control points to create a polynomial curve. At first a Bézier curve may seem a little confusing since the curve does not pass through its control points, except the end-points, but when it is fully understood it is a very intuitive way to describe a smooth curve. Cubic Bézier curves are commonly used in vector graphics, for example to draw text fonts. Since they are commonly used, efficient algorithms have been developed.

Instead of node points the control points of a Bézier curve can be thought of as representing different geometric properties. The point closest to an end-point determine the direction of the curve at the end-point. The next point from the end-point controls the curvature. The curve is also guaranteed to be smooth regardless of how the control points are placed.

An example of a cubic Bézier curve can be seen in Figure 2.1. It consists of three control points. The first and third point determines the start and end of the curve while the second point gives you control over the tangent of the curve in both end-points since both points tangent pass through the second point.

A quadratic Bézier curve is shown in Figure 2.2. It makes use of one more control point compared to the cubic version. Instead of one there are two control points between the end-points. This makes it possible to control the tangents of the end-points independently and/or the curvature of the curve.

In the figures helper lines are used to help visualize how to draw the curve. For example, start to draw straight lines between all the control points. For a 3rd order Bézier curve there are four control points and three lines. If you want place the middle point of the Bézier curve you simply find the middle point on all three lines and place new control points at these locations. Draw two new lines connecting the three points. Place points in the middle of these two lines creating another set of two control points. If you draw a line between these two points the middle point on this line is the middle point of the Bézier



(a) $t = 0.25$

(b) $t = 0.50$

(c) $t = 0.75$

(d) $t = 1.00$

Figure 2.1: An example of a cubic Bézier curve. The different subfigures show different stages in plotting a Bézier curve where t is the fraction of the total curve length.

curve. This process can be repeated for all other points you want to place along the curve.

Bézier curves can be defined for any degree n, see Equation 2.1.

$$\mathbf{B}(t) = \sum_{i=0}^{n} \binom{n}{i} (1-t)^{n-i} t^i \mathbf{P}_i, \quad t \in [0,1]. \tag{2.1}$$

In this work a 5th order Bézier curve is used, see Equation 2.2.

$$\begin{aligned} \mathbf{B}(t) &= (1-t)^5 \mathbf{P}_0 + 5t(1-t)^4 \mathbf{P}_1 + 10t^2(1-t)^3 \mathbf{P}_2 \\ &+ 10t^3(1-t)^2 \mathbf{P}_3 + 5t^4(1-t)\mathbf{P}_4 + t^5 \mathbf{P}_5, \quad t \in [0,1]. \end{aligned} \tag{2.2}$$

This gives control of both the tangent and curvature of both the endpoints independently.

2.2 Radial basis function

To speed up the design process meta models are used. The choice of meta model is highly problem dependent. The best performance for this type of transonic blade design is achieved using Radial Basis Function (RBF) method[3]. The method is an interpolator which value depends on the distance from the known center points, x_i. Sums of the radial

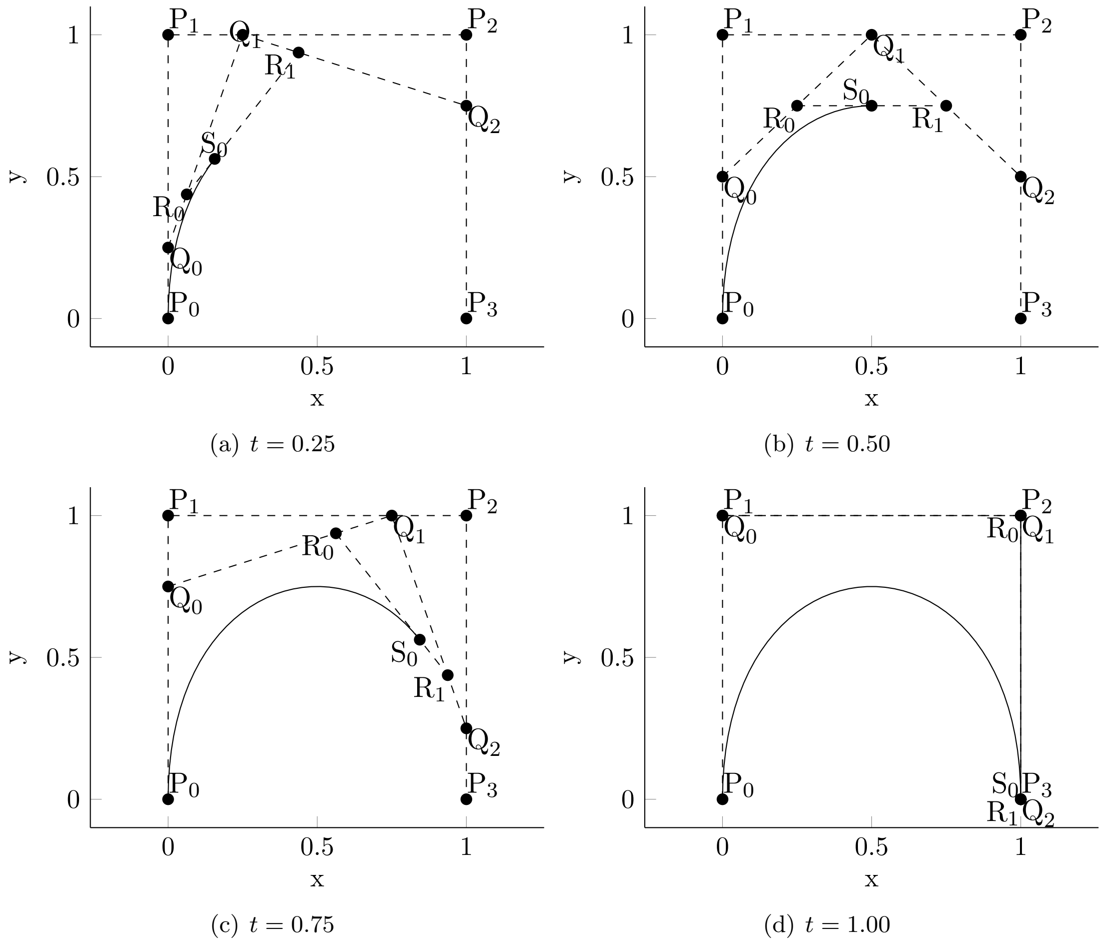


Figure 2.2: An example of a quadratic Bézier curve. The different subfigures show different stages in plotting a Bézier curve where t is the fraction of the total curve length.

basis function are then used to approximate the given function, see Equation 2.3.

$$\hat{f}(x) = \sum_{i=1}^{n} c_i \phi \left(||\mathbf{x} - \mathbf{x}_i|| / \sigma \right) \tag{2.3}$$

c_i is chosen such that the equation satisfies Equation 2.4, where x_i is a known data point.

$$\hat{f}(x_i) = f(x_i) \tag{2.4}$$

σ is a fixed scaling parameter that determines the shape of the radial function. The radial function used for this study is given by Equation 2.5.

$$\phi(r) = \sqrt{1 + r^2} \tag{2.5}$$

An example using these kind of radial basis functions can be seen in Figure 2.3. In this example $\sigma = 0.2$ and $c = 1$ for both functions.

2.3 Optimization

The optimization algorithm used in this work is the Non-dominated Sorting Genetic Algorithm-II (NSGA-II). It is a multi-objective, genetic algorithm. A genetic algorithm

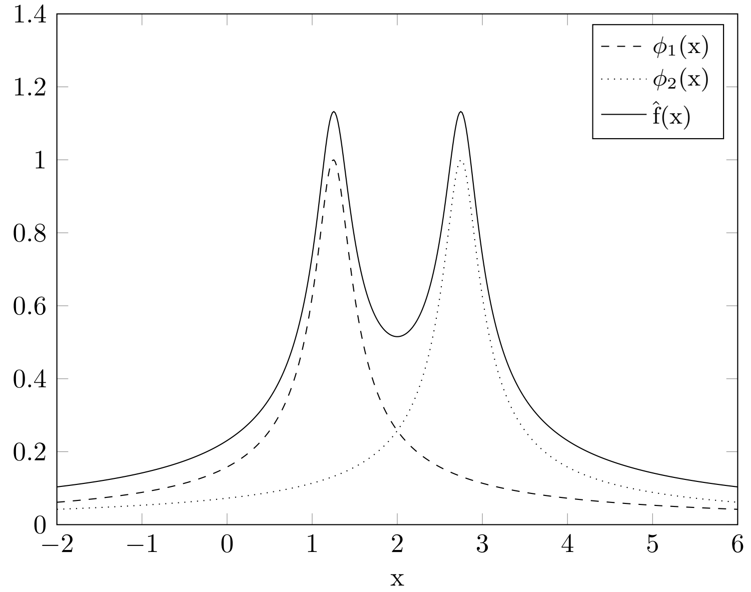


Figure 2.3: Two radial basis functions in one dimension, where $x_1 = 1.25$ and $x_2 = 2.75$. Together they form a function approximation $\hat{f}(x)$.

mimics the evolution by mixing valid individuals from each generation to form the next generation. Each individual can also get random mutations in order to introduce new properties to the population. A Latin hypercube is used to initialize the first generation of the optimization to make sure there is a good variation of all design variables.

The multi-objective optimization allows more than one objective function. The result is a set of several optimums, a so called pareto-front, compared to a single optimum point when using a single-objective optimization. An optimum is found by either maximizing or minimizing the given objective functions. A multi-objective optimization can be converted into a single-objective one by considering a weighted sum of each individual objective. In addition constraint functions are evaluated to decide if a solution is feasible. An example of a pareto-front is shown in Figure 2.4. The pareto-front visualize the trade-off between two or more conflicting objectives.

The optimization in this thesis is done on a meta model based on results from CFD simulations. The pareto-front obtained from the optimization is re-evaluated with CFD and then used to refine the meta model and the optimization is repeated until the meta model is able to predict the CFD model.

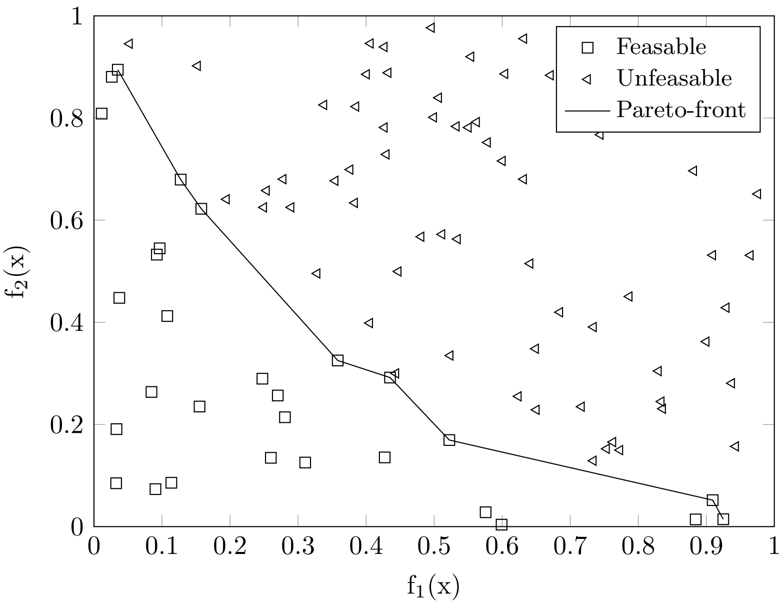


Figure 2.4: An example of a pareto-front.

3

Method

To evaluate the new blade parameterization, the same method described in Section 1.1 was used. More specifically the 2D profile of the stage at 95% of the span was optimized and compared to previous results obtained using the Volblade parameterization. In order to implement the new parameterization, it had to be altered to fit in the existing work flow. This included writing a new program, called Polly, that replaces Volblade as a geometry generator in the design process. Polly is responsible for making the whole 3D blade as well as the output of stream tubes for mesh generation and to give the optimizer information about certain blade specific data to be able to control and evaluate the design.

3.1 Polly

Polly is a program used for generating compressor blade geometries. It's created as an alternative to Volblade for use in a new design method currently developed in an co-operation between GKN Aerospace and Chalmers University of Technology. Polly expands the allowed design space by creating the suction and pressure side of the blade independently of each other instead of using a camber line and a thickness distribution. This limitation in Volblade also prevented the creation of thicker blades with high curvature, since the points that made up the curves of the suction and pressure side could be located beyond the camber lines focal point, resulting in unwanted effects. Since Polly creates each side independently from each other, some problems could arise, i.e intersection between suction and pressure sides. This is however easily avoided by choosing the parameters properly.

Polly starts with reading the input parameters of all defined spans. The blade metal angles are calculated from the incidence and deviation with the help of the flow angles from the boundary conditions. The parameters of the spans not defined is interpolated from the data available. The interpolation method can be chosen in the input file.

First the leading edge is placed at the origin and with help of the stagger angle and the axial chord length the location of the trailing edge is calculated, see Figure 3.1.

Then the axial chord length together with the leading edge and trailing edge thickness is used to calculate the normalized edge thickness. The normalized thickness is then used to place two points on equal distance from the leading and trailing edge along a line orthogonal with the metal angles of each edge, see Figure 3.2. These are the end-points for the two Bézier curves that describes the suction and pressure side respectively. In

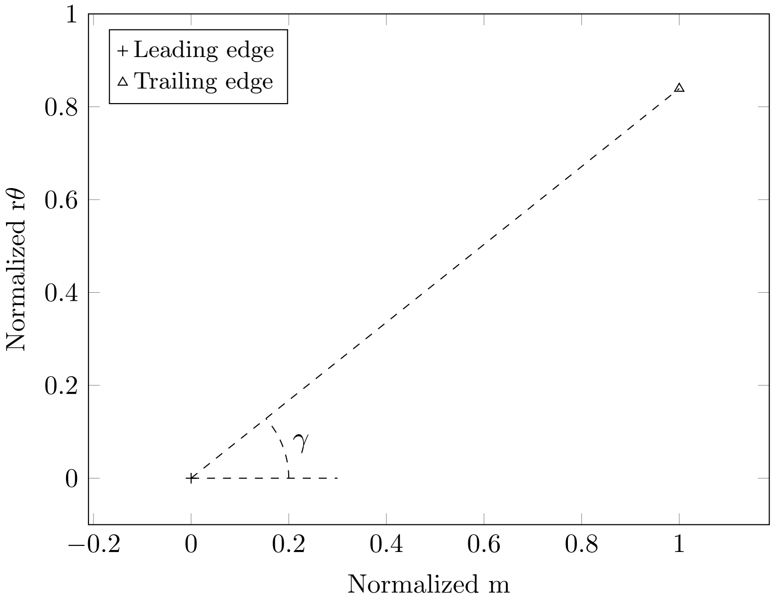


Figure 3.1: With the help of stagger angle both the leading and trailing edge are placed.

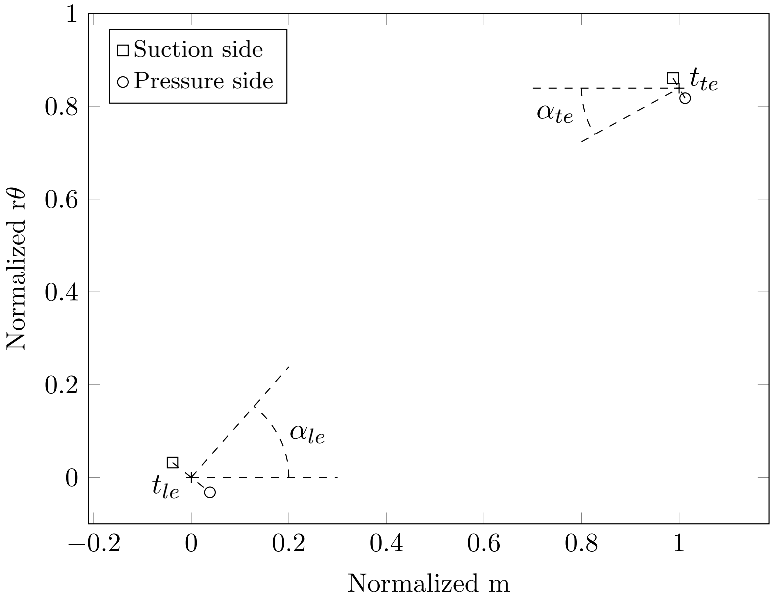


Figure 3.2: Start and end points for the Bézier curves.

order to keep track of the points on each curve they will be referred to as a number where the first point is located close to the leading edge and the sixth point is located close to the trailing edge.

Once the first and sixth control point is in place a second control point is placed with the help of two parameters. The first parameter is the wedge angle deciding the direction from the first point. The second parameter describes the distance to the second point as a fraction of the distance between the first and sixth point. See Figure 3.3 for further reference. This is also done for the fifth point but the sixth point is used for reference instead of the first point. The same procedure is then repeated for the pressure side.

The third and fourth control point is placed in between the second and fifth control

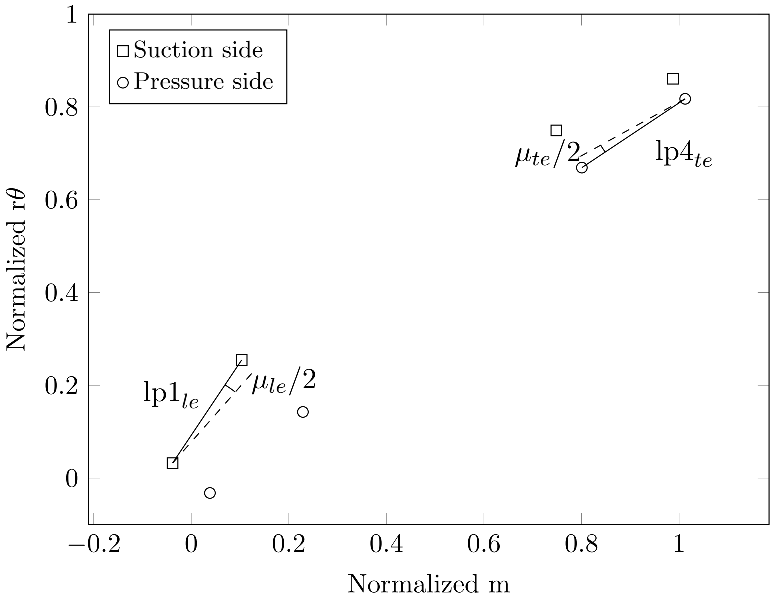


Figure 3.3: The location of the 2nd and 3rd control points are placed by a function depending on metal angle, wedge angle and a parameter defining the magnitude as a fraction of the length between the end-points.

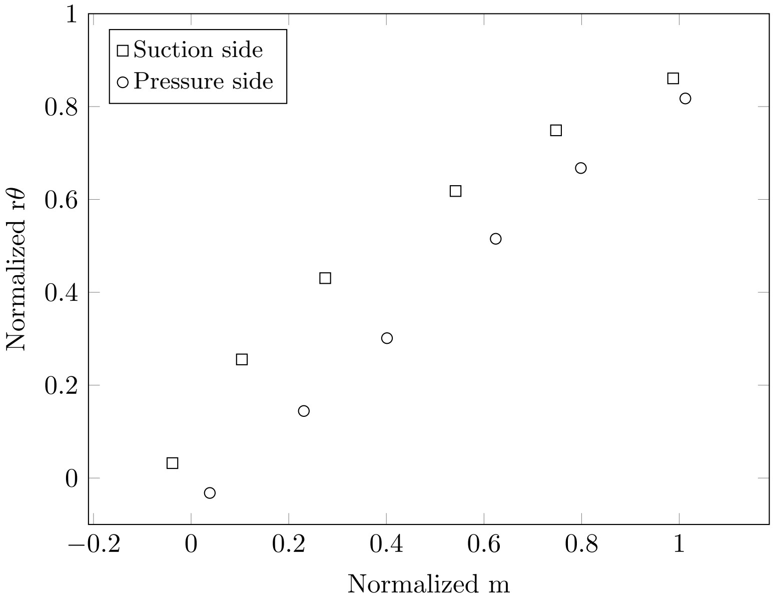


Figure 3.4: The full set of control points defining the Bézier curves.

point in a similar manner. A complete set of control points can be seen in Figure 3.4.

The control points are then used to calculate the suction and pressure side using Bézier curves, see Figure 3.5.

Finally noses, described by a 6th order polynomial curve, are fitted to connect the curves at the leading and trailing edge. These are created according to a specified aspect ratio while keeping the curvature continuous at the transition between the Bézier curves and the noses, see Figure 3.6.

When the profiles have been created they are stacked on stream lines or geometric

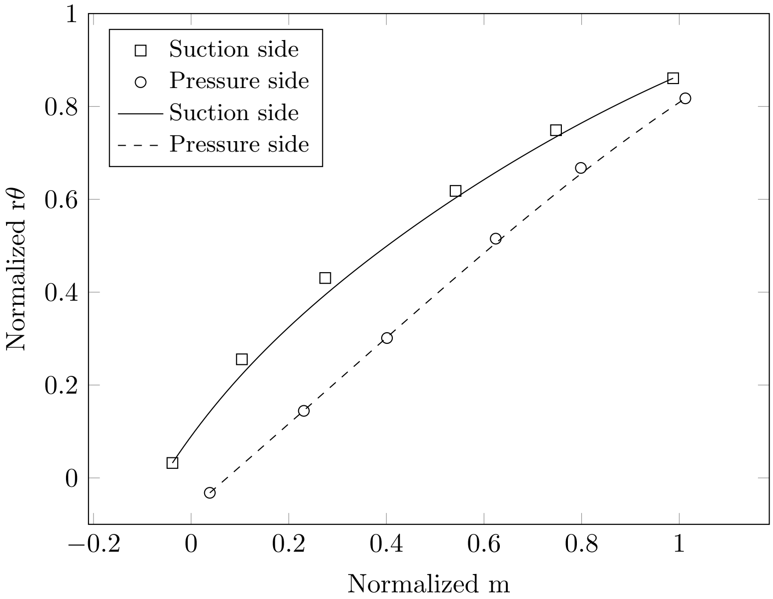


Figure 3.5: The Bézier curves before noses are fitted at the leading and trailing edges.

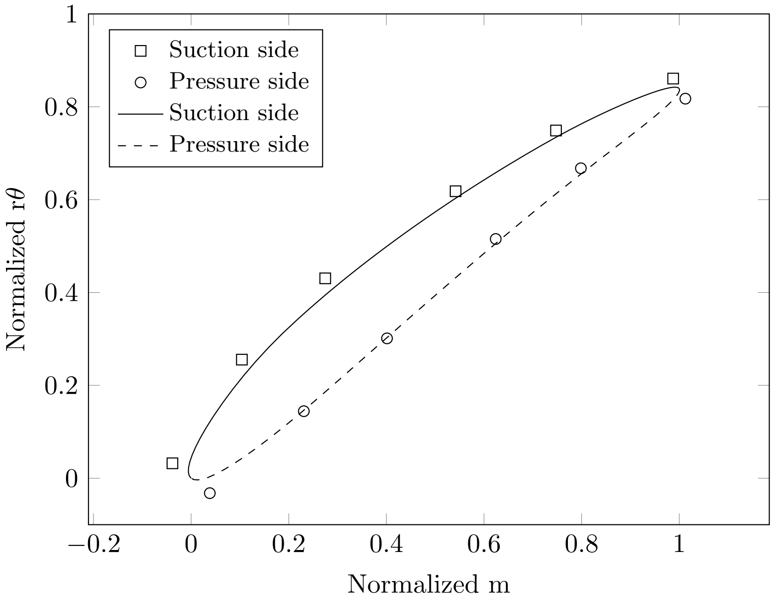


Figure 3.6: Completed blade profile. Note that the blade profile does not have to pass through any of the control points.

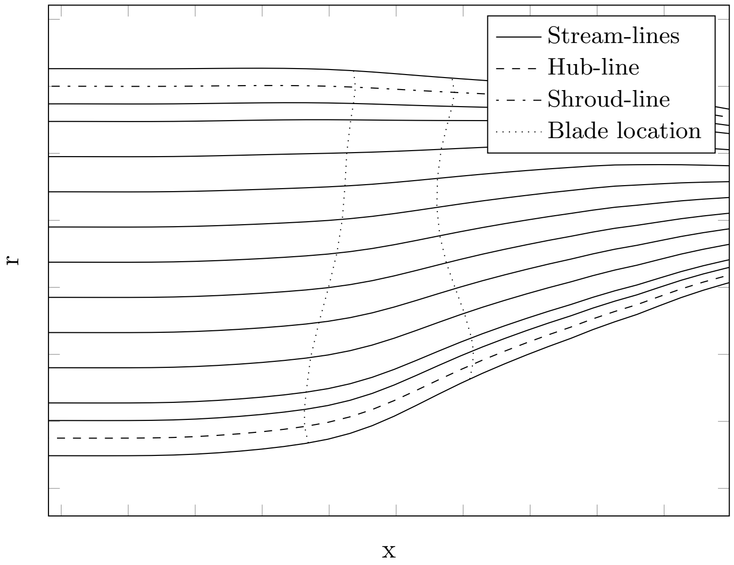


Figure 3.7: Stream-lines and compressor blade in the meridional view.

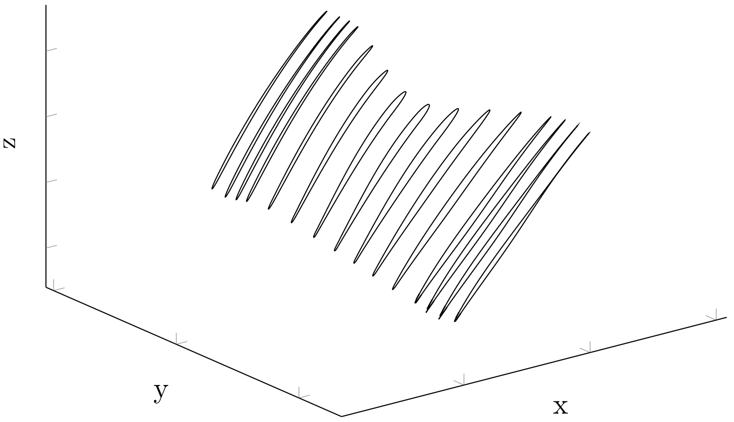


Figure 3.8: Complete 3D blade.

spans along a stack line, specified by a Bézier curve, to create the final blade geometry. The geometric spans are calculated from the hub and shroud lines, see Figure 3.7. The stream lines are created with the help of Bézier curves with the end-points defined in the geometric spans at the inlet and outlet of the blade row. With the use of flow angles a stream line can be calculated and flow angles can be calculated at the beginning and end of the blade to calculate metal angles of the blade from given incidence and deviation angles. The stacked profiles can be seen in Figure 3.8.

Polly also outputs an extended parameter file where additional blade specific data is stored. This information, for example maximum blade thickness, can be used when the blades are evaluated in an optimization process as a way to set limits on properties not directly controllable by the input parameters. It also outputs the boundaries for the stream tubes used to create meshes for the quasi 3D CFD analysis used in this thesis. Some of this output can be used when comparing blade different blade properties, for example the cascade areas of the blade.

The cascade area is calculated by finding the distance between two blades in a blade row, see Figure 3.9. The variation of the cascade area through the gas channel can be visualized, see Figure 3.10. This can be used to compare different blade designs. The tool

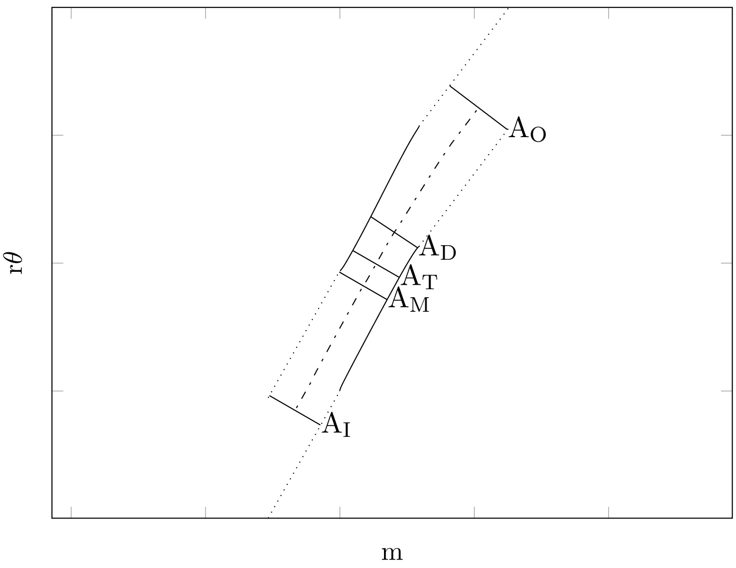


Figure 3.9: A 2D representation of the cascade area. The areas are located at the inlet, mouth, throat, discharge and outlet.

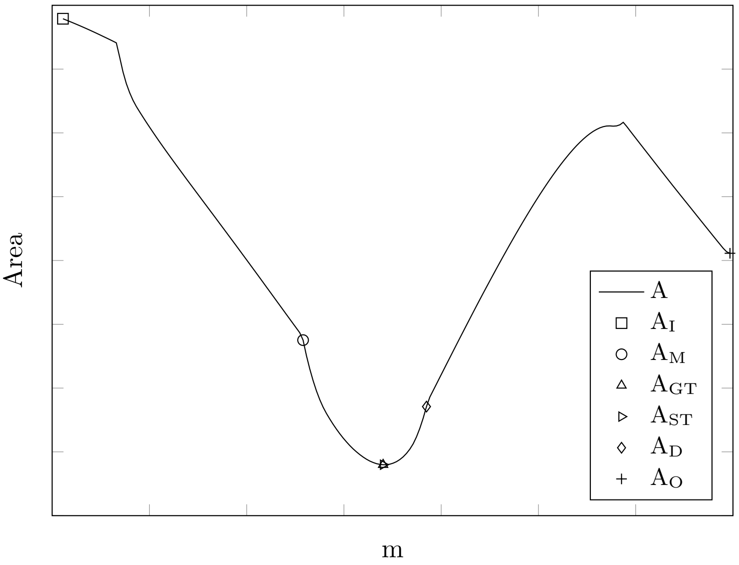


Figure 3.10: The 2D length together with the height of each stream tube is used to produce the following cascade area plot. Note that the location of the throat does not have to be the same as in the 2-dimensional case since the stream tube height is varying.

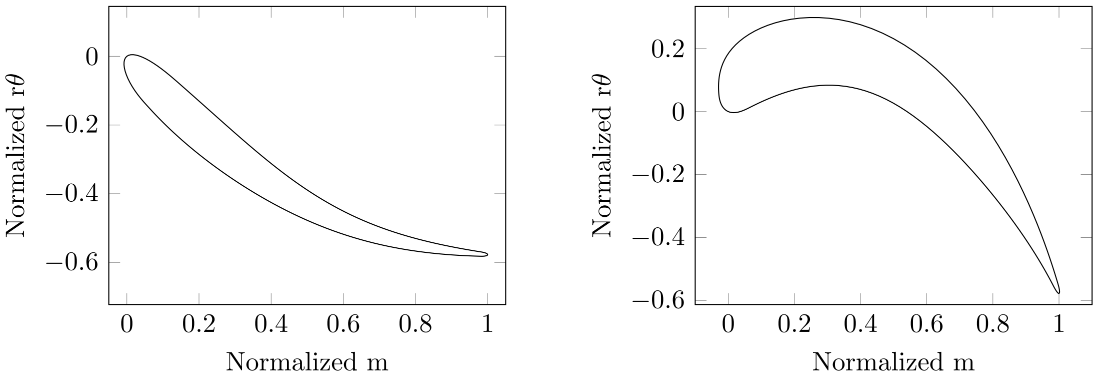
is not limited to Polly, as it can read any blade geometry on the VAC file format[4].

In this thesis, Polly was used to design a compressor stage, but can be used to create any type of blade profile. Examples of a compressor blade and a turbine blade can be seen in Figure 3.11.

To have a consistent definition of the swirl angle, the blade metal angle is defined according to Equation 3.1[5].

$$\tan(\alpha) = \frac{r\mathrm{d}\theta}{\mathrm{d}m} \tag{3.1}$$

Where $\mathrm{d}m$ is a function of x and r defined in Equation 3.2.



(a) Compressor blade

(b) Turbine blade

Figure 3.11: Polly is capable to create any type of blade profile.

$$\mathrm{d}m = \sqrt{\mathrm{d}x^2 + \mathrm{d}r^2} \tag{3.2}$$

3.2 CFD analysis

As a CFD solver an in-house GKN code called Volsol is used. It is a finite-volume, density based solver. It uses Runge-Kutta time marching with a third order accurate upwind-biased scheme for convective terms and a second accurate compact centered scheme for all diffusive terms. A k-ϵ turbulence model with a Kato-Launder limiter and wall-functions is used. Local adaptive dampening is applied around shocks in order to reduce numerical oscillations.

3.2.1 Computational grid

The computational grid is generated with G3dmesh, an in-house GKN meshing tool. Each mesh consists of an o-grid around the blade connected to h-grids, see Figure 3.12.

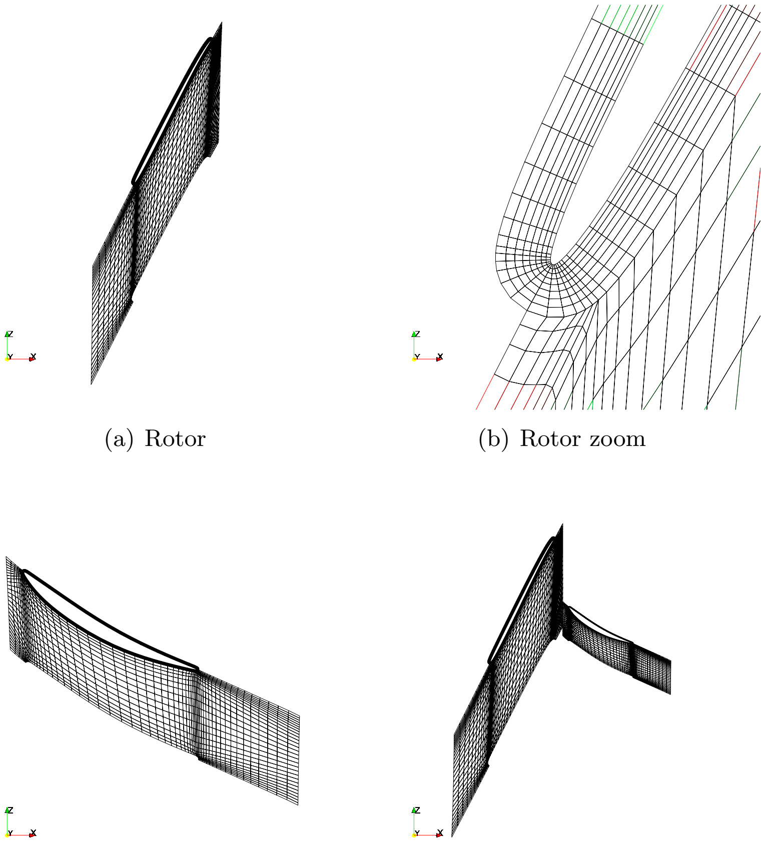
A mesh study was done prior to this work and since the new blade parameterization will use the same running conditions and will produce similar blade geometries, a new mesh study was not performed, instead the recommended settings are used.[2]

3.3 Optimization

Before the optimization can start reasonable limits have to be set to the variables in the design space. This was done by adjusting the parameters such that it resembled an existing blade.The parameter limits was set to vary around these values. As a starting point for the 95% span, all the angles was allowed to vary ±3° and the rest of the parameters ±10% or by recommendations for compressor blade design[6].

An overview of the optimization process for the blade designs can be seen in Figure 3.13. This is the same method used in earlier studies[2] in order to be able to compare the different parameterizations.

An uniform Latin hypercube was used to populate the initial training-set. The initial training-set consisted of 430 individuals, which was evaluated using the CFD solver.



(a) Rotor

(b) Rotor zoom

(c) Stator

(d) Assembly

Figure 3.12: Mesh

To rank the blades, two objective functions were used. The first was the polytropic efficiency at design point, η_p, defined in Equation 3.3.

$$\eta_p = \frac{\gamma - 1}{\gamma} \frac{\ln\left(\frac{P_{02}}{P_{01}}\right)}{\ln\left(\frac{T_{02}}{T_{01}}\right)} \tag{3.3}$$

The other objective function is stability and was quantified as static pressure recovery at off design, defined in Equation 3.4

$$C_p = \frac{P_2 - P_1}{P_{01,rel} - P_{01}} \tag{3.4}$$

The outlet flow angle and the normalized mass-flow are constrained such that the stage is matched properly within the multi-stage environment. Further more with this parameterization the maximum blade thickness had to be constrained to get blades that fulfill the aeromechanical limitations. Since this parameterization does not define the thickness distribution explicitly, the maximum thickness was controlled by a constraint.

The design process itself uses an outer and an inner optimization loop. The inner optimization loop is used to speed up the process and to save computational time by evaluate the objective functions and the constraints using radial basis functions. The constants used by the radial basis function is recalculated every outer optimization loop adding the individuals that was evaluated with CFD to the training-set. In this way, the optimization process is accelerated by only refining the model at the location where the optimizer currently finds the best candidates.

The optimization process is implemented in an optimization software called modeFRONTIER, which governs the optimization process. Python scripts are used to shuffle information between different modules of the design process. The outer loop of the optimization process is repeated until the pareto-front is converged.

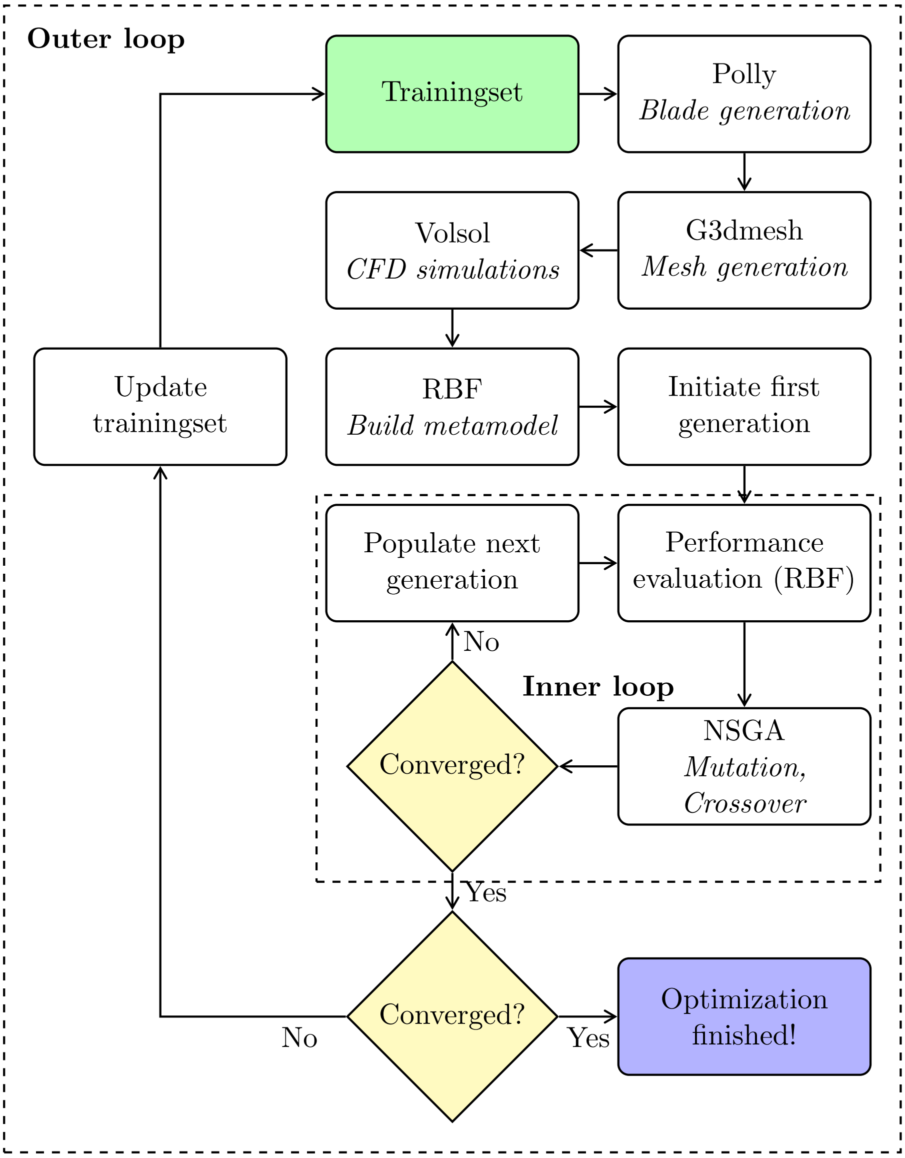


Figure 3.13: Blade design process.

4

Results

The pareto-front from the optimizations can be seen in Figure 4.1. The baseline is the pareto-front achieved by using Volblade. The pareto-front with constrained t_{max} sets a minimum of the maximum blade thickness, to prevent blades to get to thin. The unconstrained pareto-front has no constraint on t_{max}. The optimization has favored thin blades and created profiles with a t_{max} of 10-15% below current manufacturing limits. The new parameterization was able to push the pareto-front towards higher η and C_p relative the current version. There are two pareto-fronts shown that uses the new parameterization. The reason for a second optimization was because of blades with a maximum thickness less than the lower limit was favored. Since there is no parameter in the parameterization for blade thickness, a constraint was set to force the optimization process to find blades that fulfill the aeromechanical and manufacturing constraints. Both pareto-fronts are shown in order to point out that if the manufacturing process and aeromechanical limitations would allow for thinner blades there are possible gains in efficiency and stability.

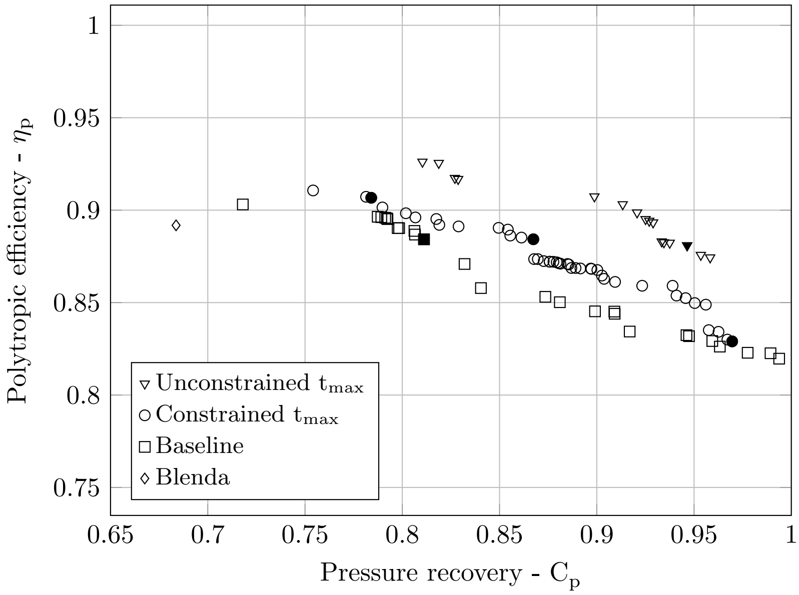
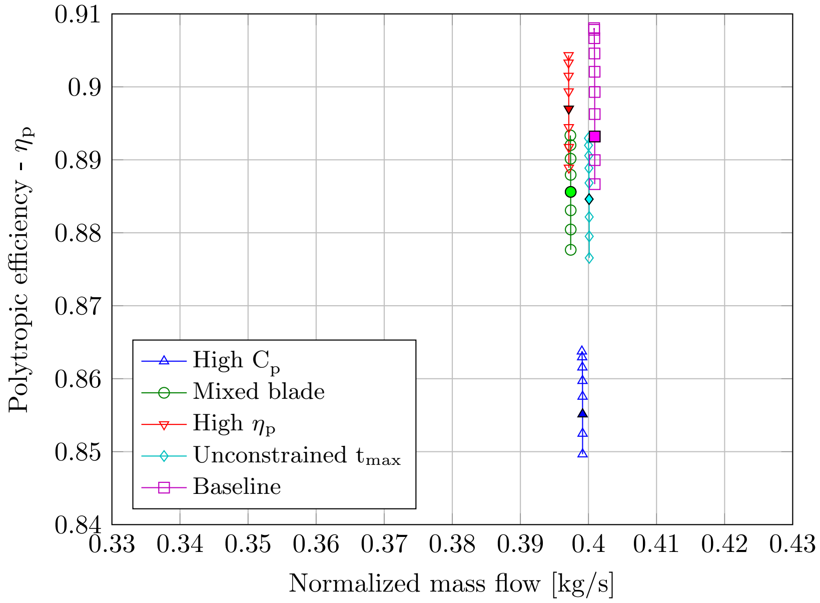
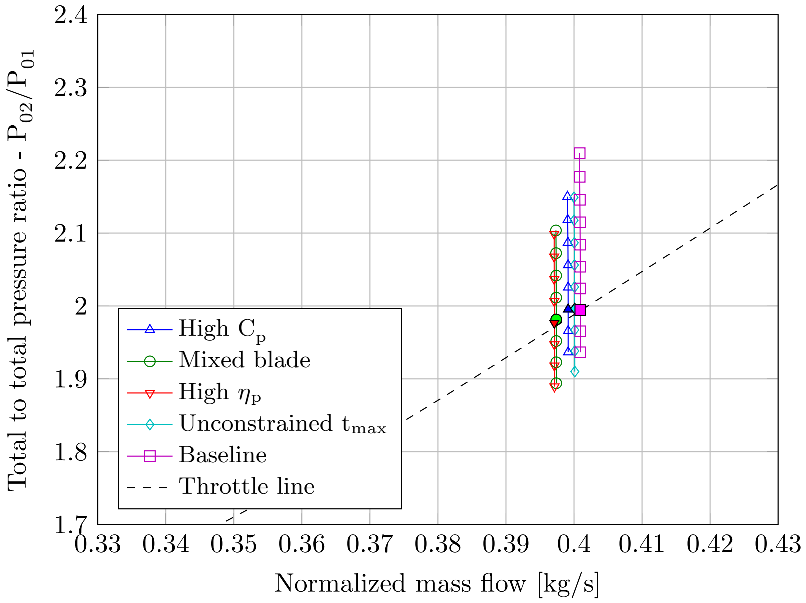


Figure 4.1: The pareto-front of the three optimizations.



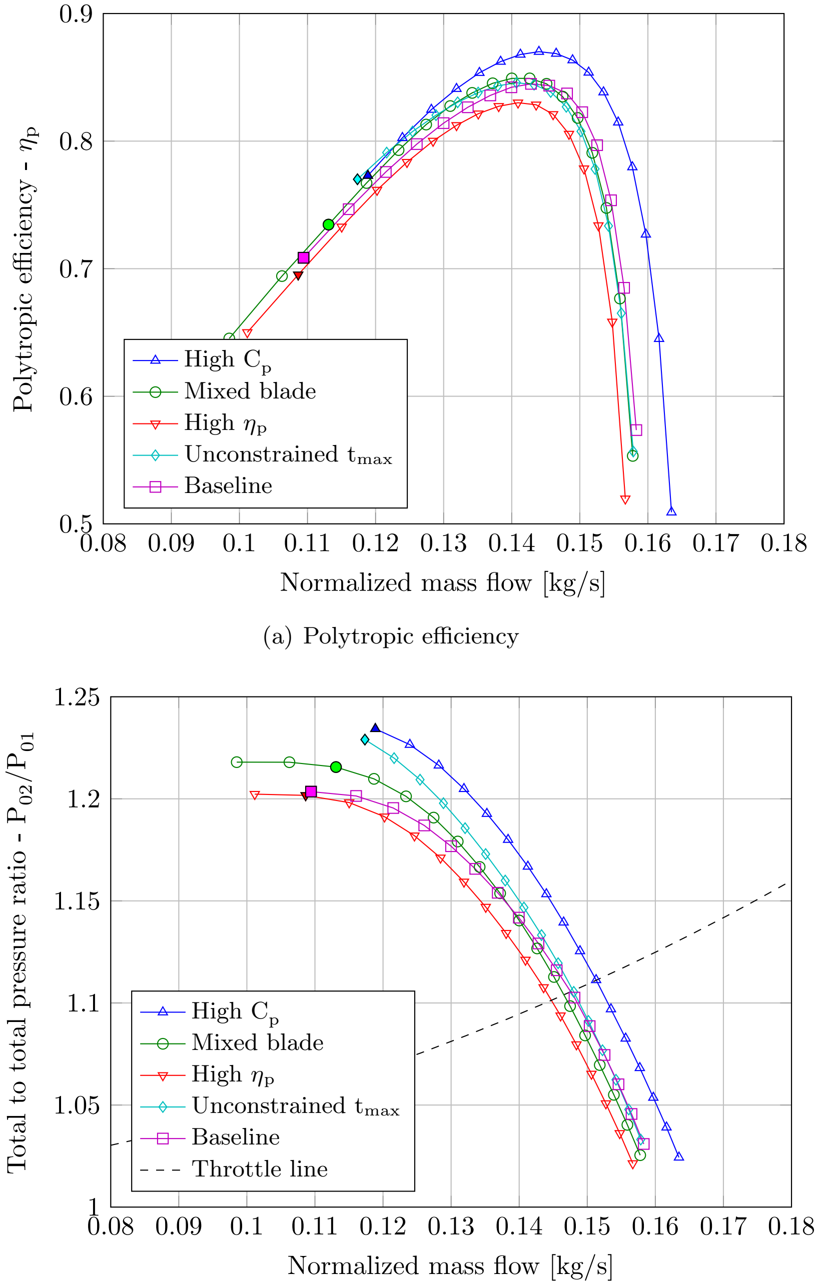
(a) Polytropic efficiency



(b) Total to total pressure ratio

Figure 4.2: Polytropic efficiency and total pressure ratio at design speed.

To verify the data on the pareto-front with constrained t_{max}, three designs was chosen for a more detailed study. Likewise one point on each pareto-front with similar η was chosen to be able to compare between the pareto-fronts. These points are marked in Figure 4.1. Each blade was evaluated at several flow condition, both at design speed, see Figure 4.2, and at part speed, see Figure 4.3. In the case of partial speed performance it looks like the mixed blade has the best stability margin since it has more points towards the surge line. This is not expected. To make the part speed easier to compare, the pressure recovery, C_p, is visualized in Figure 4.4. Here we clearly see that the pressure recovery for the different blades is as expected from the optimization result.



(a) Polytropic efficiency

(b) Total to total pressure ratio

Figure 4.3: Polytropic efficiency and total pressure ratio at part speed (55%).

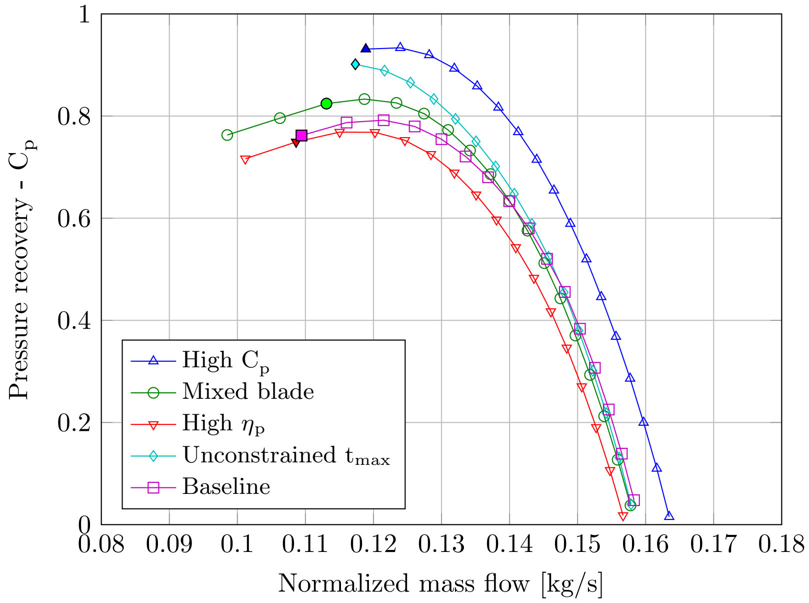


Figure 4.4: Pressure recovery at part speed (55%).

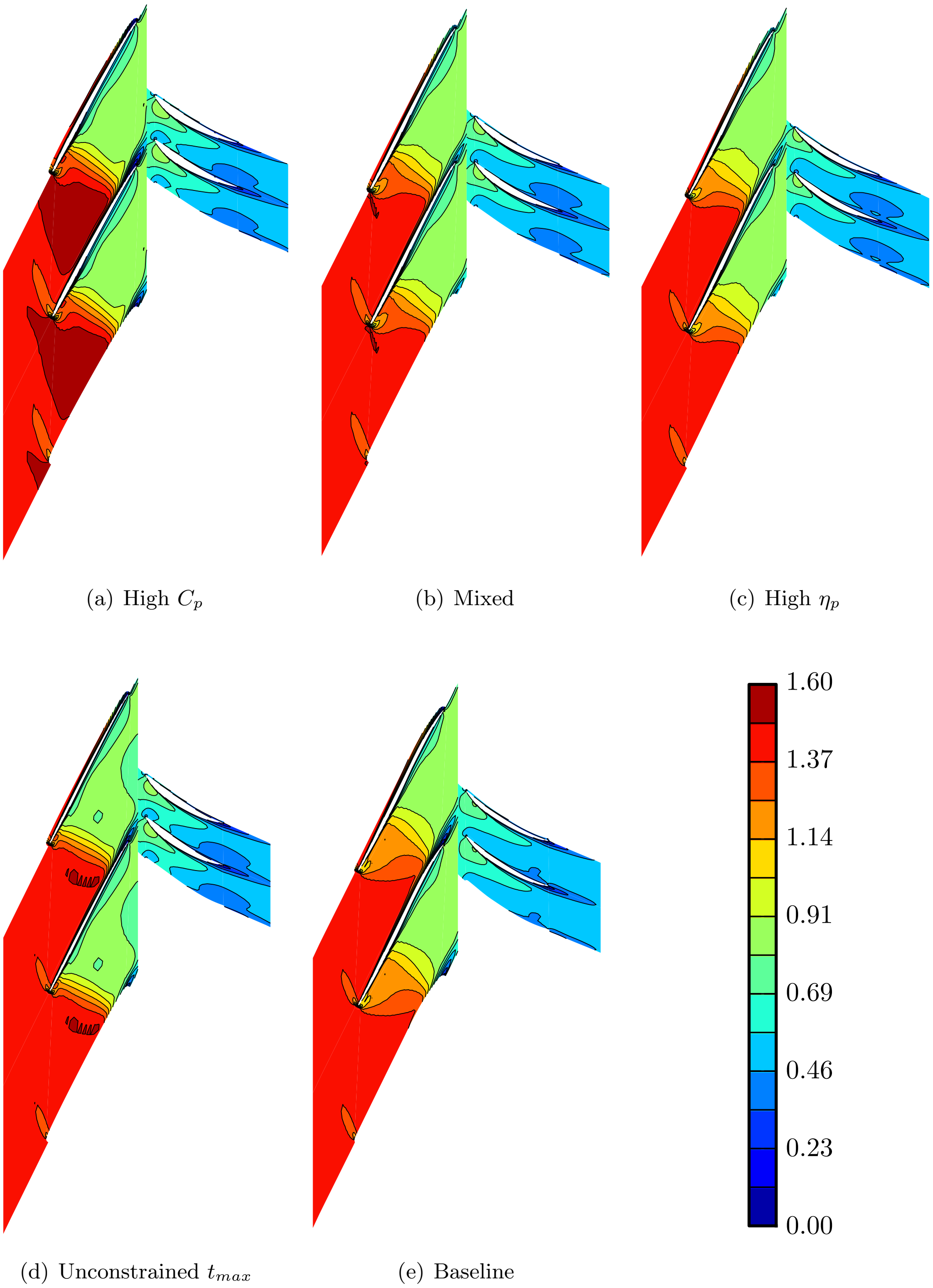
The contour plots of the Mach number at the design speed and at part speed is shown in Figure 4.5 and Figure 4.6. Here we can see the difference in shock pattern. The blade with high C_p has more acceleration on the suction side and thereby a stronger shock than the others and this would explain its low efficiency. The high η_p blade have a weaker shock which explain the higher efficiency. Note the small wake of the mixed and high efficiency blade, at the design speed. When comparing the high η_p blade to the baseline the wake is smaller on the high η_p.

There are three rotor blades that have an s-shaped form, baseline, unconstrained t_{max} and the high η_p. The rotor blade with high C_p has a convex suction side and a concave pressure side. The rotor of the mixed blade has a convex suction side and an s-shaped pressure side. The blade with unconstrained t_{max} is thinner than the rest. Note the thinner stator of the baseline.

From the Mach contours in Figure 4.6 it is difficult to draw conclusions about stability. Instead axial velocity of the baseline and the mixed blade can be compared in Figure 4.7. There is a larger separation area on the suction side of the baseline blade. For the mixed blade there is only a small area with negative axial velocity located close to the shock on the suction side.

Furthermore the curvature between the two parameterization tools are compared. In Figure 4.8 a comparison of the curvature at the leading edge of the blade is shown. Note the discontinuous curvature of the blade created by Volblade where the nose is fitted to the rest of the blade. The implementation in Polly requires the transition between both the pressure and suction side and the nose to have a continuos curvature, a requirement Volblade lacks. The effect on the flow can be seen in Figure 4.9. There is a local increase of the Mach number at both the pressure and suction side where the nose is attached to the rest of the blade. This is due to the lack of continuity of the curvature in the Volblade implementation. There is no disturbance like this at part speed, see Figure 4.10, instead the flow separates, see Figure 4.7.

The curvature of the suction and pressure side are shown in Figure 4.11. The baseline blade, generated with Volblade, has a curvature that changes signs six times along the



(a) High C_p

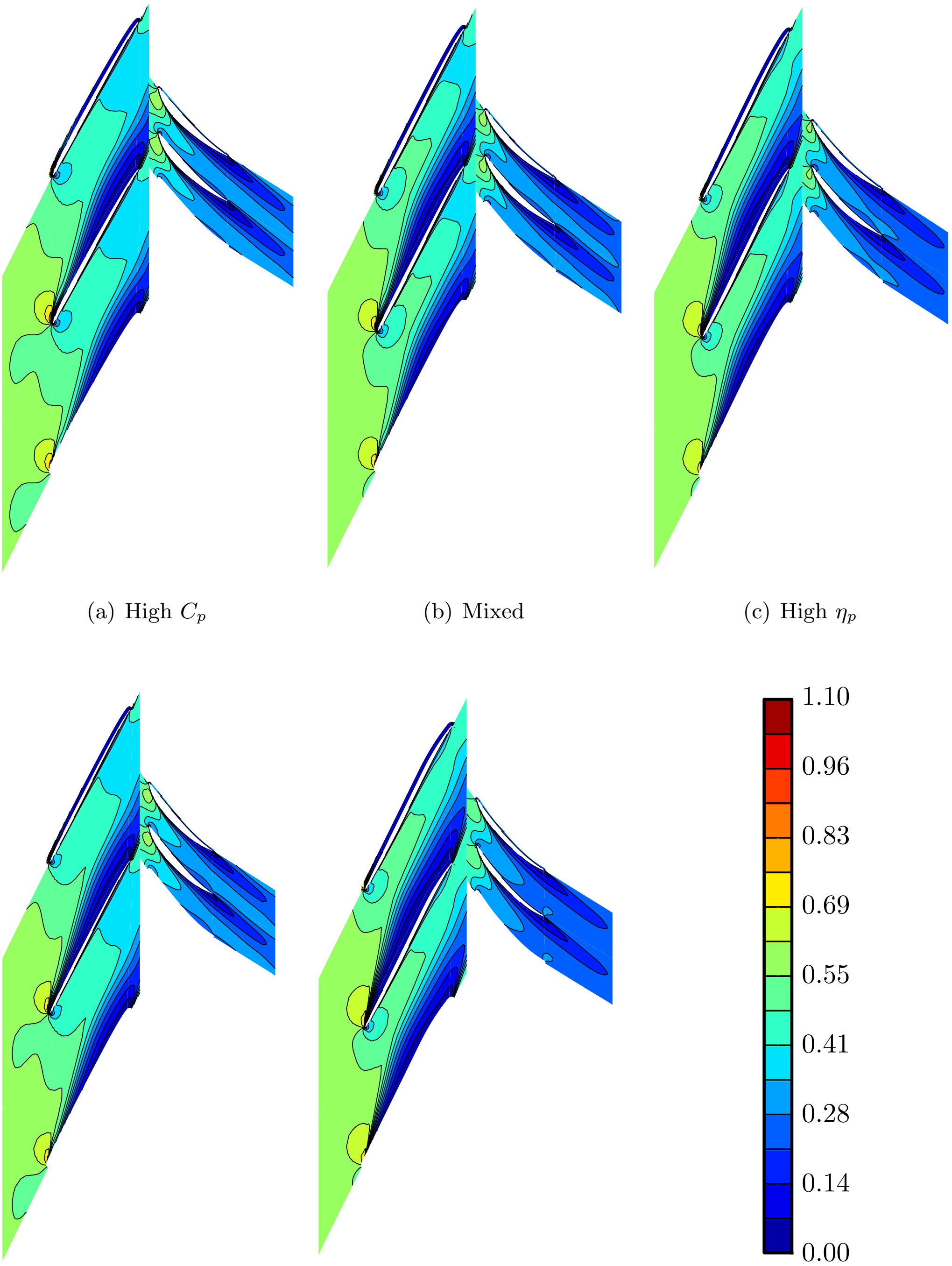
(b) Mixed

(c) High η_p

(d) Unconstrained t_{max}

(e) Baseline

Figure 4.5: Mach contour plots of 95% span at the design point.



(a) High C_p

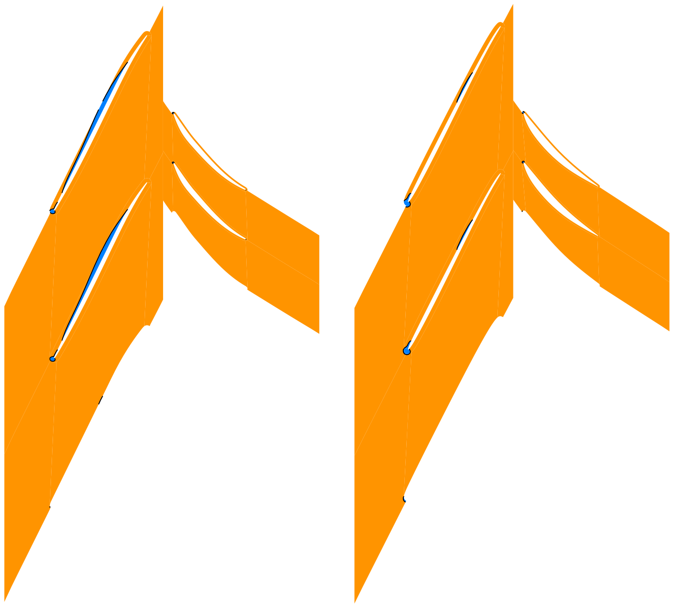
(b) Mixed

(c) High η_p

(d) Unrestricted

(e) Baseline

Figure 4.6: Mach contour plots of 95% span, near stall, at part speed (55%).

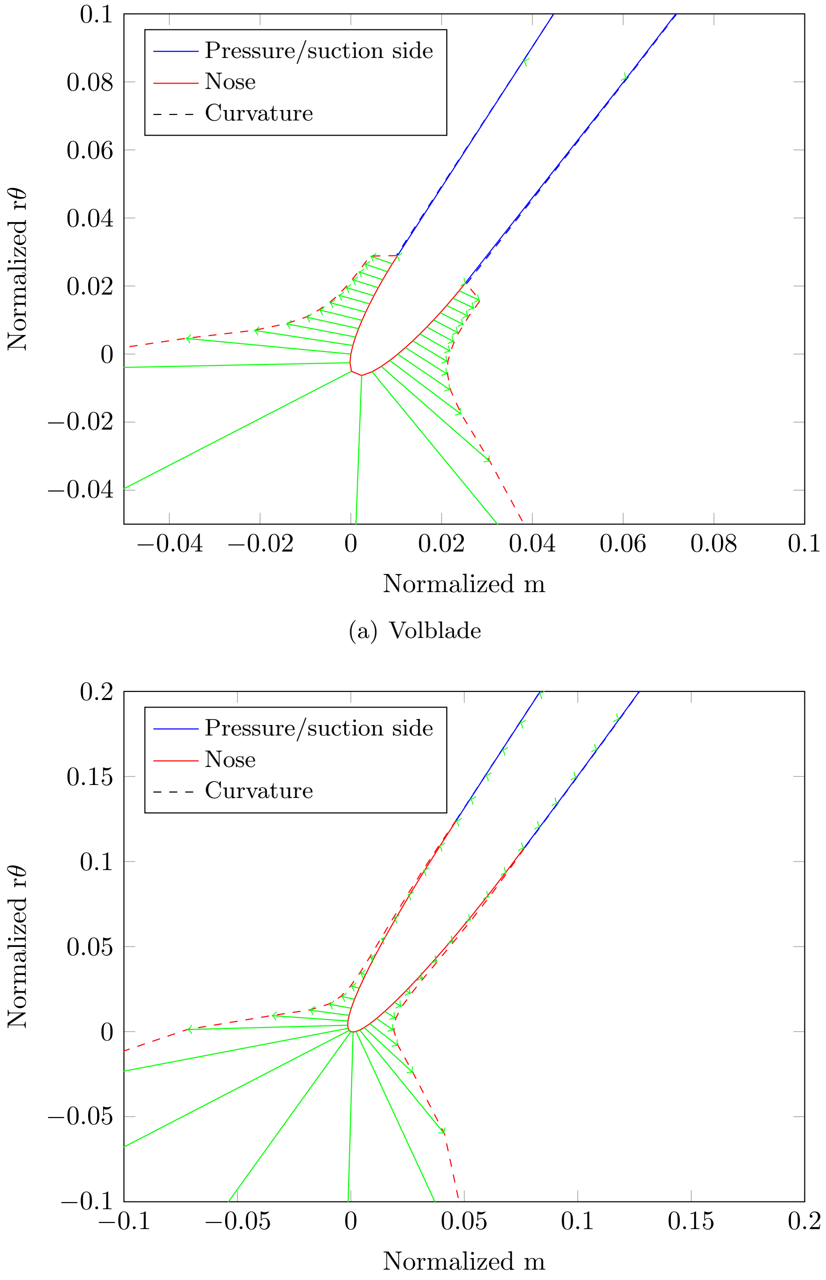


(a) Baseline

(b) Mixed

Figure 4.7: Axial velocity at part speed (55%). Orange indicates a positive axial velocity, and blue a negative axial velocity.

suction side. The magnitude of the curvature is not high but still it indicates that Volblade does not create blades with smooth surfaces. The curvature of the blade created in Polly is both continuos and smooth around the whole blade.



(a) Volblade

(b) Polly

Figure 4.8: Curvature of the two different parameterizations. The dashed line is a measure of curvature. This line is created by taking each point on the profile and translating it in its normal direction by a distance set by the magnitude of the curvature. The further away, the greater is the curvature.

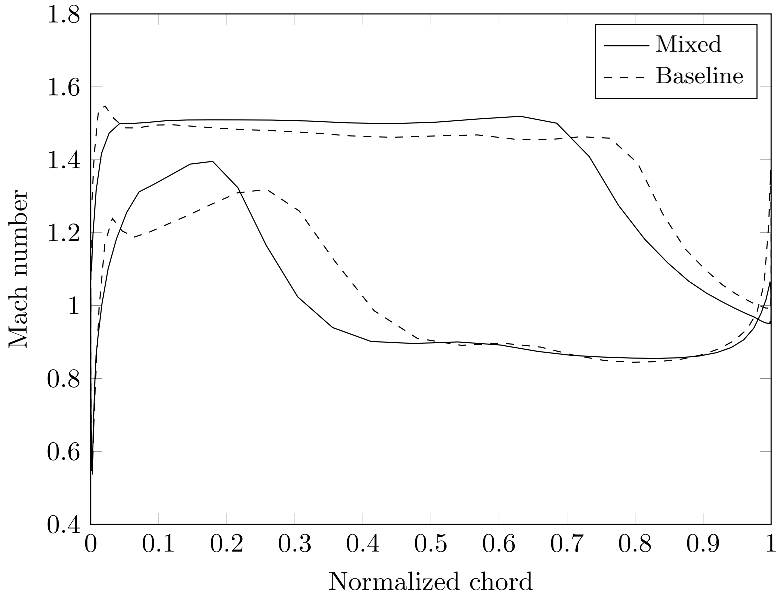


Figure 4.9: Comparison of the Mach number along the surface of the blade at design point.

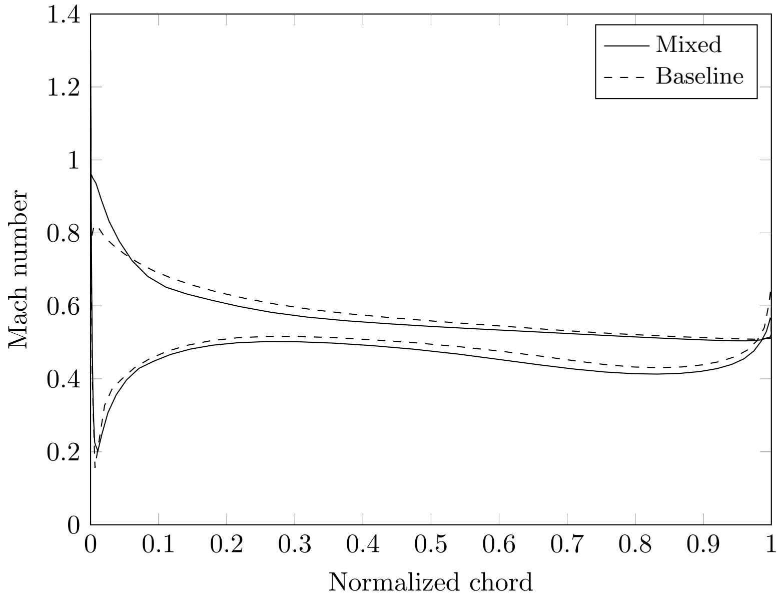


Figure 4.10: Comparison between the Mach numbers along the surface of the blade, near stall, at part speed (55%).

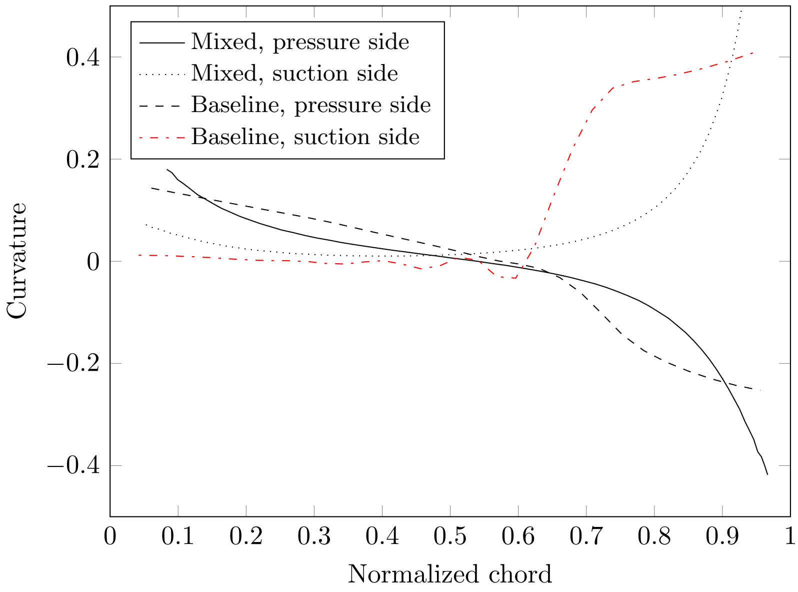


Figure 4.11: Curvature comparison between Volblade and Polly.

5

Discussion

One strength of the parameterization used in Polly is the ability to control the shape of each side of the blade independently. The mixed blade is such a blade with one side s-shaped and the other convex. As it is a part of the pareto-front it shows the importance of this independence. Since Volblade was not able to produce this type of blades, this region of the design space is previously unexplored in the design method used in this thesis. Another strength of Polly, is the continuous curvature and smooth curves produced, compared to Volblade, where the curvature is discontinuous at the transition between the separate curve segments that together describes the profile. The discontinuous change in curvature, where the nose is attached to the blade, accelerates the flow. Since the change in curvature is discontinuous, the change in the flow should in theory be instantaneous. Since this is not possible there is instead a risk of separation. This discontinuity of the curvature can be the reason for the separation of the flow at off-design for the baseline blade, seen in Figure 4.7.

As well as strengths, Polly also have some weaknesses compared to Volblade. With the use of 20 instead of 11 design variables for each span of every blade, there is a need of more CFD evaluations. The number of individuals evaluated in the design process increases exponentially. When the number of parameters doubles this means that the number of evaluations needed is approximately four times as many. One way to decrease this is to reduce the number of parameters used by the optimization. This can be done in several ways. Either by removing the ability to create the suction and pressure side independently of each other by using the same parameter values on both sides when placing the third and fourth control points in the Bézier curve. This would reduce the number of needed parameters by four. Another way is to make all the variables used to place these points constant, keeping the continuous curvature benefits of Polly, but limit the control of the shape of the suction and pressure side, reducing the number of variables to 10. This is not recommended as the performance of the blade is closely connected to several of these parameters. If a parameter reduction is required, a pre-optimization with a simple model, to determine the values of some variables, is suggested. Then a full evaluation on the remaining parameters is proposed.

Another issue with the new parameterization introduced by Polly is that the parameters are not completely independent of each other. The placement of the four control points between the leading and trailing edge are all based on the placement of previously placed points. The reason for this methodology is to only create reasonable blades, but the dependency can create convergence problems for the pareto-front.

Another reason to the increased optimization time was the fact that the number of individuals used in the first training-set was reduced by a factor two, compared to earlier work, in a try to speed up overall process. This lead to more outer loop generations in the design process and about the same total number of evaluated designs. This was confirmed by reducing the number of individuals in the first training-set when optimizing with the original parameterization.

Polly was able to find blades with both higher efficiency and higher stability. The reason for this is probably because of the higher number of parameters that can be altered. Since there is no parameter for setting the blade thickness a constraint on the minimum thickness of the blade at its thickest point had to be used. More constraints and more parameters leads to longer optimizations, which has to be considered. Polly has a lot of potential but since no optimization of the full 3D blade was done it is hard to tell how much the blade was improved compared to the increased design time.

6

Conclusions and future work

The new parameterization has successfully been implemented into the existing work flow. The result of the optimization suggest blade designs with both higher efficiency and higher stability. This gain comes at a cost of a longer optimization, since twice the amount of parameters are used.

To further improve the design process, use of a different optimization software is suggested, to be able to automize more and run more evaluations in parallel without being limited by software licenses. There are a free, open source software, called Dakota, which shows much potential. The downside in Dakota is the lack of possibility to post-process the data. This is not a problem since modeFRONTIER could still be used for post-processing purposes.

Another aspect to investigate is the interrelation between input parameters. More specifically the method of placing the third and fourth control point of the Bézier curves used to create the suction and pressure side should be investigated since under certain circumstances some parameters have no effect.

Bibliography

[1] H. Saravanamuttoo, G. Rogers, H. Cohen, S. P.V, Gas turbine theory, 6th Edition, Pearson Education, Longman, 2008.

[2] L. Ellbrant, Optimization and Model Validation of Transonic Compressors, Licentiate thesis, Chalmers University of Technology (2012).

[3] L. Ellbrant, L.-E. Eriksson, H. Mårtensson, CFD Optimization of a Transonic Compressor Using Multiobjective GA and Metamodels, in: 28th International Congress of the Aeronautical Sciences, no. ICAS 2012 Paper no. 267, 2012.

[4] VAC file format, a GKN internal document.

[5] K. Siddappaji, M. G. Turner, General Capability of Parametric 3D Blade Design Tool for Turbomachinery, in: ASME Turbo-expo 2012, no. GT2012-69756, 2012.

[6] H. Mårtensson, Basic design of compressor blades, a GKN internal document (2010).