Università degli studi di Padova

*Dipartimento di Ingegneria Industriale*

Laurea Magistrale in Ingegneria Aerospaziale

# Aerodynamic multi-objective optimization of a transonic compressor rotor using meanline methods and CFD

Laureando:
**Nicola Zanoni**

Relatore:
**Prof. Ernesto Benini**

Correlatore:
**Ing. Luca Menegozzo**

Anno Accademico 2019/2020

*'The sea, once it casts its spell, holds one in its net of wonder forever.'*
JACQUES-YVES COUSTEAU
Pioneer of diving

**Abstract**

Aerodynamic optimizations of axial-flow turbomachinery have been widely explored by many researchers. Typical procedures involve the use of Computational Fluid Dynamics (CFD) simulations, which give accurate results but require high computational time and resources. In order to tackle these challenges, a procedure that couples a meanline solver and a genetic algorithm is here proposed.

The meanline solver Systematic Analysis of TURbofaN (SATURN), developed by the COMETES research group at University of Padua, is presented in Chapter 2, together with its validation on experimental data of the transonic compressor rotor NASA Rotor 37, which has been chosen as test case. The optimization is reported in Chapter 3: the results show a significant increase in pressure ratio at design condition, against an almost unchanged isentropic efficiency and operating range. Then, the methodology adopted to build a three-dimensional model starting form meanline parameters is described. A series of CFD simulations have been used to verify solver prediction on the optimized geometry. Mesh, settings and results of the simulations are shown in Chapter 4: the increase in pressure ratio has been confirmed, as for the prediction on efficiency. However, a different operating range has been calculated. Finally, an Artificial Neural Network (ANN) has been trained on meanline outputs, as explained in Chapter 5, and it has been used to predict the performance of the compressor in an optimization process analogous to the one described in Chapter 3. The optimized individual has been simulated using the meanline solver, and a good matching between ANN prediction and SATURN calculation is shown. The use of a surrogate model helped to further reduce the time required to complete the optimization procedure.

# Contents

# List of Figures

# List of Tables

# 1 | Introduction

*After a brief overview on Computational Fluid Dynamics (CFD) methods, aerody-
namic optimization problems using CFD are described. A solution that replaces CFD
with a one–dimensional meanline solver is then proposed, in order to reduce computa-
tional time needed. The test case is finally introduced.*

## 1.1 State-of-the-art CFD Methods

The behaviour of a fluid is one of the most fascinating topics in science. Using
Netwon's second law of dynamics together with fundamental principles of mass,
momentum and energy conservation, it is possible to describe the motion of a
general viscous fluid by means of Navier-Stokes (NS) equations. These equations
are one of the pillars of fluid mechanics, but proving that a smooth and globally
defined solution to these equations exists has never been done. Many scientists
tried in vain to solve NS existence and smoothness problem (Otelbaev [1] is just
an example), but today it still remains an unsolved *Millenium Prize Problem*.
NS equations consist of a system of four second-order non-linear partial differen-
tial equations: finding an analitycal solution is possible just in few particular cases,
say for inviscid and incompressible flow, and so a numerical solution in required
in almost every application. NS equations have the following expression:

$$\begin{cases} \frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot \left( \rho \vec{V} \right) = 0 \\ \rho \left[ \frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \vec{\nabla}) \vec{V} \right] = \rho \vec{g} - \vec{\nabla} p + \mu \nabla^2 \vec{V} + (\lambda + \mu) \vec{\nabla} \left( \vec{\nabla} \cdot \vec{V} \right) \end{cases} \qquad (1.1)$$

All the features of the flow motion are described, from larger eddies to tur-
bulence. It could seems obvious, at this point, that the most simple way to get
a numerical solution of a turbulent flow is to discretize NS equations in time
and space. This particular technique of CFD is called Direct Numerical Simula-
tion (DNS).
As DNS simply discretizes Equations 1.1, discretization must be so fine to capture
every aspect of turbulence in order to get an accurate result: because of this aspect,

1

DNS is extremely heavy in terms of computational costs, but on the other side its results are comparable to those given by an experiment. To have a quantitative idea, consider the DNS of the water flow through a common house tap with a diameter equal to 15 mm. Using non-dimensional analysis, it is possible to demonstrate that if the characteristic Reynolds number of the flow is $Re_1 = 6 \times 10^3$, the time required for the solution to converge would be $t_{tot,1} \propto Re_1^3 \simeq 1$ hour on a modern multicore Intel Xeon 64 GB RAM 2.00 GHz workstation (Picano [2]). Consider now the water flow through a commercial tube with a 2 m diameter characterized by a Reynolds number $Re_2 = 6 \times 10^6$. The computational time required can be calculated from the following equation:

$$\frac{t_{tot,2}}{t_{tot,1}} = \left(\frac{Re_2}{Re_1}\right)^3, \tag{1.2}$$

from which it follows that

$$t_{tot,2} = 1000^3 \cdot t_{tot,1} = 1 \times 10^9 \text{ hours,}$$

that is approximately equal to 100 000 years. This type of calculation is never used in industrial applications for obvious reasons and it is mainly used within academic research to investigate some particular aspects of turbulence. Nowadays, even the most powerful supercomputers can't afford to compute a DNS rapidly. Probably this scenario will change once the enhanced computational performance of quantum supercomputers will be state of the art (Arute et al. [3]).

It is clear that a faster CFD methodology is necessary. Under an engineering point of view, the relevant properties of a flow can often be extracted from the stationary solution: for example, an aircraft engineer can be interested in the overall effect that the fuselage has on the performance of the aircraft itself, rather than solving the small scale flow features. If so, it is sufficient to consider average velocity fields, rather than the instantaneous ones. Reynolds-Averaged Navier-Stokes equations (RANS) methodology fulfills this purpose by modifying NS equations using the so-called *Reynolds average*. A series of benefits are consequence of this assumption: is it possible to use symmetry properties as time-dependent properties are lost, less discretized grids can be employed because the focus of the analysis is on the large-scale aspects of the flow and the computational time is drastically reduced if compared to the one required by a DNS. With RANS, however, come also some negative aspects. First of all, it is not possible to capture turbulence, as it is an unsteady phenomenon. Second of all, the formulation of the problem requires further equations in additions to the Reynolds-averaged Equations 1.1 to achieve the closure of the mathematical system. A number of additional equations is then used to model turbulence behaviour depending on

(a) *RANS*          (b) *LES*          (c) *DNS*
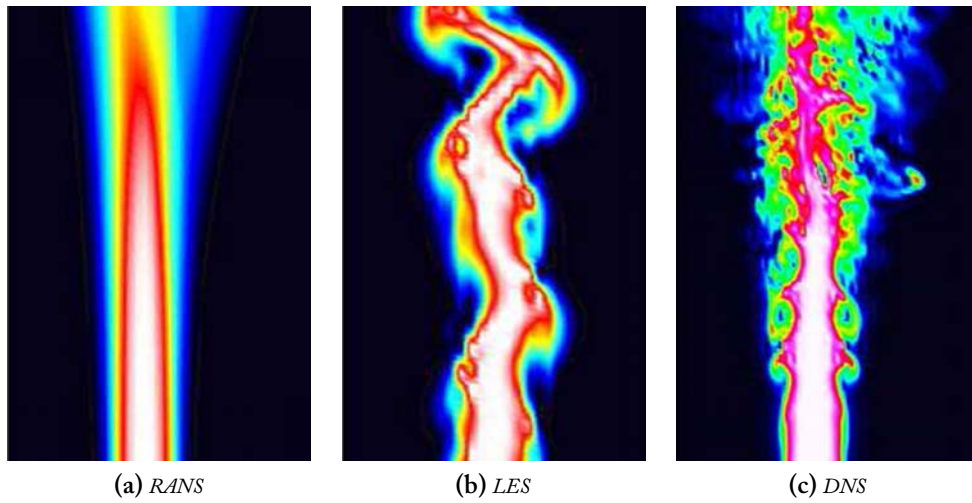
**Figure 1.1.** *Different CFD simulations of the same turbulent flow (Cuenot [4]). Notice how DNS captures the smallest eddies, while averaged flow features are captured by RANS. LES stands in–between RANS and LES*

the turbulence model adopted. As a result, the output of the simulation is dependent from these parameters, and so it is highly recommended to use RANS technique only with the support of some experimental data on which the numerical model can be validated. The turbulence model used in this thesis is a two-equations model called $k - \omega$ Shear Stress Transport (SST).

Nevertheless sometimes it could be necessary to capture the large-scale aspects of turbulence. With this goal, a technique that is in between the extreme accuracy of DNS and the averaged results of RANS simulation stands in the Large Eddy Simulation (LES). LES methodology is more advanced than RANS methodology, as it is based on the concepts of K41 theory of Kolmogorov, according to which smallest aspects of turbulence are universal while the biggest ones depend on the specific case and geometry. Following this idea, the discretization required for a LES is coarser than the one of a DNS, but more refined than the one used in a RANS: this is not to capture smallest eddies, but only the ones dependent on the particular flow in exam.

To conclude, RANS methodology is the faster way to simulate a fluid flow and is currently the most widely used CFD technique in engineering applications, while DNS remains confined in academics contexts and LES finds application in specific fields, as in meteorology.

## 1.2    Aerodynamic Optimizations using CFD

Aerodynamic optimizations of axial-flow turbomachinery based on CFD have been extensively explored by many researchers.

A common optimization procedure is to couple a bio-inspired algorithm and CFD simulations. Typically, the baseline geometry is modified by means of an arbitrary parametrization and it is then simulated using a RANS solutor. The process is usually driven by a genetic algorithm, for example, and it is repeated until convergence criteria are reached. Consider the time required for a single solution to converge being 8 hours for each individual on an Intel Pentium IV CPU of 3.00 GHz speed, like for the transonic compressor blade analyzed by Samad and Kim [5]. The analysis must be then repeated for each individual of each generation at each different operating condition. The total CPU time required is then

$$t_{tot} \propto 8 \times N_{ind} \times N_{gen} \times N_{oc} \text{ hours,} \qquad (1.3)$$

being $N_{ind}$ the number of individuals, $N_{gen}$ the number of generations and $N_{oc}$ the number of different operating conditions. Using this approach, however, $N_{ind}$ and $N_{gen}$ must be low enough to keep calculation times within reasonable limits. This is a drawback aspect of such methodology, because the limitation of the number of generations and individuals prevents to perform a thorough exploration of the design space, and the identification of the absolute optimum is not guaranteed. For example, Abate [6] proposed the optimization of a transonic fan by coupling an evolutionary optimization algorithm and CFD in order to maximize isentropic efficiency and pressure ratio. The process required 7 days on an Intel Xeon 64 GB RAM 2.00 GHz speed quadcore workstation. Generations number and individuals number have been respectively set to 10 and 6. Asghari et al. [7] presented an evolutionary algorithm optimization process of the transonic rotor Rotor 37 based on the modification of the stacking line of the blade, which led to a slight improvement of the adiabatic efficiency. The number of generations and individuals was respectively set to 8 and 20. A number of 73 CFD simulations have been carried out, since some geometries have not been simulated as they were preserved through generations because of elitism. The computational time required has not been specified by the authors, but the high number of CFD simulations led to high computational time.

Recently, some researchers tried to reduce computational time by using surrogate models to find optimal solutions, instead of CFD. Samad and Kim [5] proposed the optimization of a transonic axial compressor rotor blade using Response Surface Method. Ghalandari et al. [8] presented the optimization process of an axial compressor blade using an ANN trained on experimental data. In both cases, CFD has been used just as a verification tool and

computational time required has been of few hours.

Joly et al. [9] proposed a threefold optimization approach based on machine learning and genetic algorithm, which involved the dynamic selection and self-tuning among several surrogate models, the classification to anticipate failure of the performance evaluation and the adaptive selection of new candidates to perform CFD evaluation for updating the surrogate models, in order to facilitate design space exploration and to reduce surrogate models uncertainty. The calculations have been carried out on a 100 CPUs cluster and they required 48 hours, including 360 CFD simulations of the transonic rotor Rotor 37. Notice that CPU time is within reasonable limits, but very high computational power has been employed. The same process would have required about 120 days on an Intel Pentium IV CPU of 3.00 GHz speed, like the one used by Samad and Kim [5].

Duan et al. [10] used a data mining technique based on Proper Orthogonal Decomposition combined with a particle swarm algorithm to optimize the performance of the Rotor 37. In the first phase of the process, the creation of a database using the particle swarm algorithm has been required, and performance of each particle have been then evaluated using CFD. Since 80 particles have been created, 80 CFD simulations have been carried out. Consequently, high computational time is required by this approach too.

Mondal et al. [11] analyzed a surrogate-assisted optimization of the Rotor 37. They trained multi-fidelity surrogate models with fewer high-fidelity CFD predictions and more rapid and inexpensive lower-fidelity CFD evaluations. The approach yielded significant increase in performance within a dozen of CFD simulations, so that computational time has been kept within reasonable limits.

In this work, an optimization process based on a meanline solver is proposed, as explained in the next chapters. The aim is to reduce the use of CFD to verification purposes. The meanline solver has been developed by the COMETES research group at University of Padua and it has been extensively validated on several turbomachinery used in aeronautical propulsion systems, including the Rotor 37, developed by National Aeronautics and Space Administration (NASA), which has been chosen as test case.

## 1.3   NASA Rotor 37

The Rotor 37 is a low aspect ratio inlet rotor for an axial transonic core compressor developed by NASA Lewis Research Center.  Figure 1.2 shows its meridional section.

The Rotor 37 was designed and initially tested in the 1970s as part of a research programme involving four axial-flow compressor stages; one of them, the Stage 37, involved actually the Rotor 37. These stages were intended to cover a range of design parameters typical of aircraft turbine engine high-pressure (core) compressor inlet stages.  Some design information and overall performance of Stage 37 came from Reid and Moore [12], but more detailed measurement data were provided by the same two years later (Moore and Reid [13]).

After those tests, the Rotor 37 was tested again as an isolated component. A brief description of the related test facility was given by Suder [14], while Figure 1.2 shows the axial location of some measurement stations used to compute compressor performance.  Measurement data have been published by many authors and are used to validate numerical models for transonic compressor rotors (Suder [14]; Ameri [15]).  Design specifications for the Rotor 37 are reported in Table 1.1.
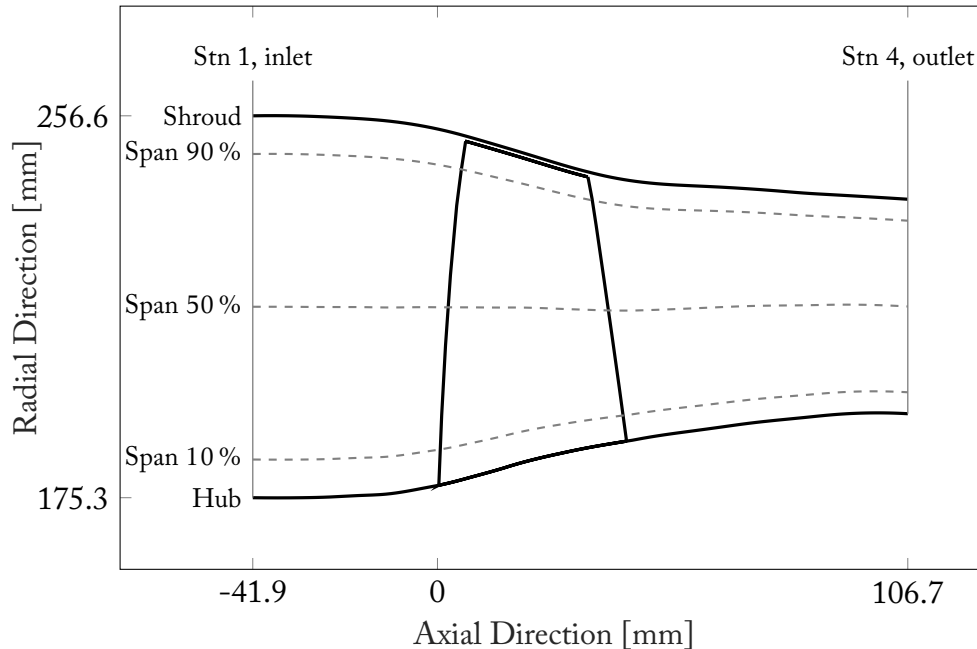


**Figure 1.2.** *Meridional section of the Rotor 37 and survey stations*

**Table 1.1.** *Design specifications of the Rotor 37 (Suder [14])*

| Mass flow rate $[\mathrm{kg\,s^{-1}}]$ | 20.188 |
|---|---|
| Rotational speed $[\mathrm{rad\,s^{-1}}]$ | 1800 |
| Total pressure ratio | 2.106 |
| Total temperature ratio | 1.270 |
| Isentropic efficiency | 0.877 |
| Inlet tip relative Mach number | 1.400 |
| Inlet hub relative Mach number | 1.130 |
| Tip solidity | 1.288 |
| Aspect ratio | 1.190 |
| Tip clearance gap [mm] | 0.356 |
| Number of blades | 36 |

More in detail, pressure ratio and isentropic efficiency are calculated as

$$\mathrm{PR} = \frac{p_4^0}{p_1^0} \quad \text{and} \quad \eta_{is} = \frac{\mathrm{PR}^{\frac{k-1}{k}} - 1}{\frac{T_4^0}{T_1^0} - 1}. \tag{1.4}$$

The subscript indicates the measurement station, while the superscript indicates stagnation quantities. In following chapters, two more quantities will be considered, these are surge margin SM and choking margin CM. They're defined as follows:

$$\mathrm{SM} = \frac{\dot{m}_{design} - \dot{m}_{surge}}{\dot{m}_{design}} \quad \text{and} \quad \mathrm{CM} = \frac{\dot{m}_{choking} - \dot{m}_{design}}{\dot{m}_{design}}. \tag{1.5}$$

In the optimization process explained in Chapter 3, PR and $\eta_{is}$ calculated at design mass rate flow will be treated as optimization objectives, while SM and CM will be treated as optimization constraints. SM and CM values of Rotor 37 are respectively 0.038 and 0.074.

Experimental performance maps and spanwise plots are reported in Figure 1.3 and in Figure 1.4. It is possible to observe a severe reduction of isentropic efficiency over the span 80 %, associated with an increment of total temperature ratio. This is a typical negative feature of transonic axial flow compressor rotors, as a result of the irreversibilities that occur at the tip endwall region; these are mainly due to intense tip clearance flows, the presence of strong shock waves and casing boundary layer, as well as to their detrimental interaction (Biollo [16]).

(a) *Pressure Ratio*

(b) *Isentropic Efficiency*
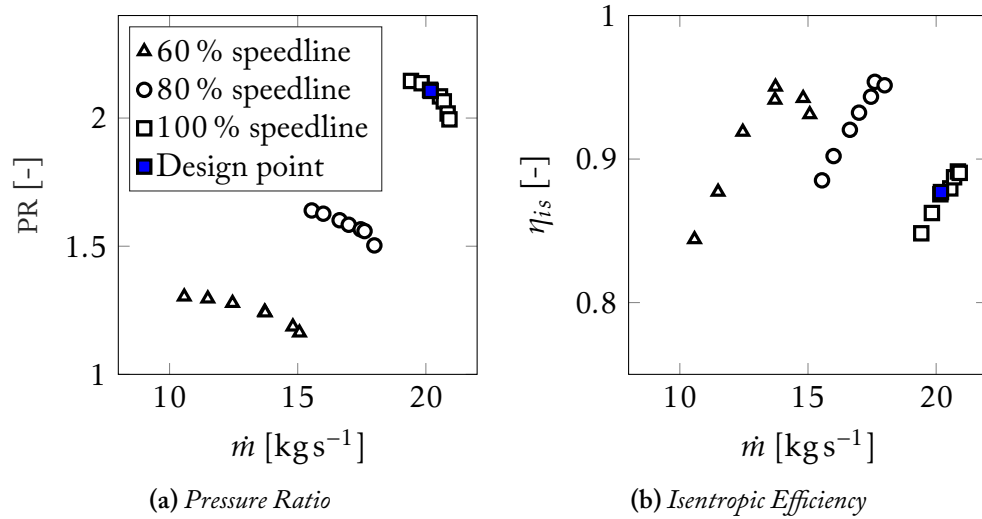
**Figure 1.3.** *Performance maps of the Rotor 37. Experimental data from Suder [14]*



(a) *Pressure Ratio*

(b) *Temperature Ratio*

(c) *Isentropic Efficiency*
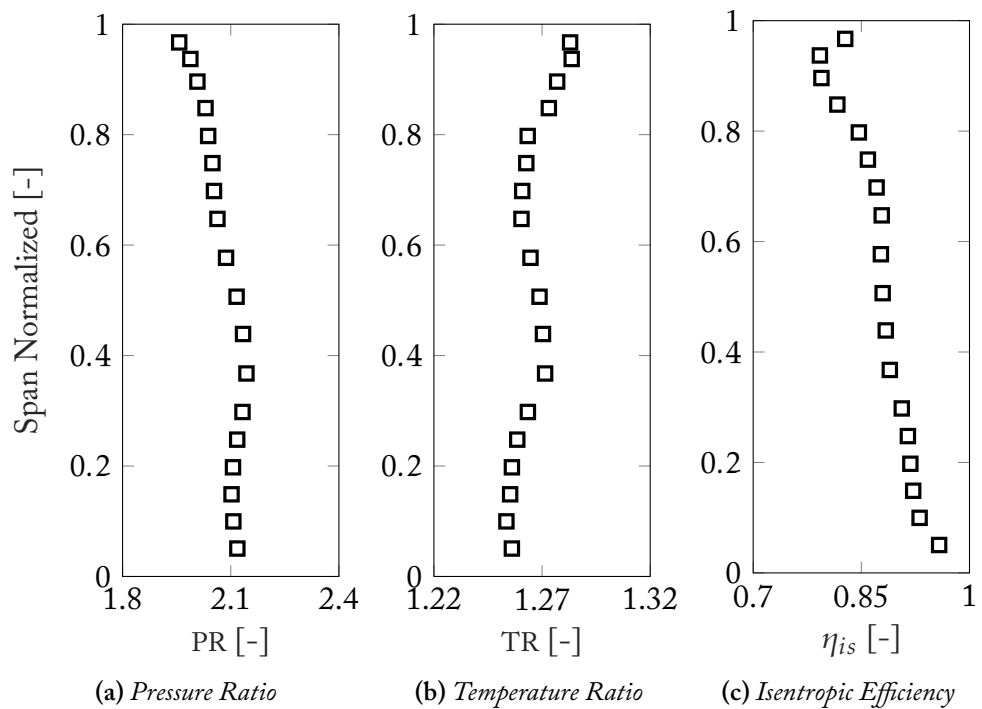
**Figure 1.4.** *Spanwise plots of PR, TR and $\eta_{is}$ at maximum efficiency mass flow rate $\dot{m} = 20.6\,\mathrm{kg\,s^{-1}}$. Experimental data from Suder [14]*

# 2 | Meanline Solver

*The meanline solver adopted in this work is here presented, together with its valida-
tion on the experimental data of the Rotor 37. Motivation, settings and results of a
Sensitivity Analysis* (SA) *based on solver's outputs are then explained.*

## 2.1 SATURN

Systematic Analysis of TURbofaN (SATURN) is a one-dimensional meanline
code developed in a C++ environment by COMETES research group at Univer-
sity of Padua. It predicts design and off-design performance of axial turboma-
chinery using fundamental equations of conservation of mass, momentum, and
energy, as well as isentropic flow equations, pressure loss and deviation models
for design and off-design conditions: solving these equations together on differ-
ent spanwise locations (specifically, 10 %, 50 % and 90 %), it determines the air
flow unknowns. The blade-to-blade geometrical properties of the airfoil at each
span given as input to SATURN and used in this work are defined by means of 5
variables, which are reported in Table 2.1.

 SATURN is able to predict the performance of several turbomachineries used in
aeronautical applications, including the transonic axial compressor Rotor 37.

**Table 2.1.** *Geometrical properties used in SATURN to define the blade–to–blade section
of the airfoil at each span*

| Definition | Variable Name |
|---|---|
| Inlet blade angle | `angBladeIn` |
| Outlet blade angle | `angBladeOut` |
| Chord length | `chord` |
| Curvature radius of leading edge | `radCurvIn` |
| Maximum thickness-to-chord ratio | `thickMaxChordRatio` |

The operating conditions, in terms of mass flow rate $\dot{m}$ and rotational speed $\omega$, are specified by the user, together with the boundary conditions at the inlet of the machine, in terms of inlet total pressure $p_{01}$, inlet total temperature $T_{01}$ and absolute inlet flow angle $\alpha_1$. If the bladed row is rotating, relative flow quantities are used. Inlet duct and outlet ducts are also considered.

A fundamental role in the solver is played by loss models. These include:

- Shock losses: they are consequence of the transonic flow regime established in the streamtube, especially at the tip, where relative velocity of the flow is higher and where shock waves are more intense. As known, entropy increases after the generation of a shock wave and a total pressure loss happens. Shock loss models account for this phenomenon;

- Blade profile losses: include losses caused both by the streamtube contraction, typical of compressors, and by the effect of the boundary layer on the suction side and pressure side of the blade, which creates drag. These losses are typical of viscid fluids;

- Off-design losses: occur when the machine works in a condition different than the design one, and so different incidence angles cause lower efficiency;

- Endwall losses: they are consequence of the secondary flows forcing the wall boundary layer toward the suction side of the rotor, where mixing of blade and wall boundary layers takes place;

- Secondary losses: they consider secondary flow losses and tip-clearance losses;

- Inlet and outlet ducts losses: an empirical loss model has been formulated using high-fidelity CFD results.

Another aspect that has been considered is the blockage: a model has been used to account for the reduction of the effective passage area caused by the presence of boundary layer both over the hub and the shroud. An empirical model has been employed, as the exact calculation of blockage factor would require the determination of boundary layers height. Furthermore, surge is detected by means of another empirical model. Finally, solver divergence for $\dot{m} > \dot{m}_{design}$ has been considered for the detection of the choking condition.

A final calibration on experimental data is required to account for models error propagation.

A complete list of the the models implemented in SATURN, together with references, is reported in Table 2.2.

**Table 2.2.** *List of the empirical models implemented in SATURN*

| Model | Reference |
| --- | --- |
| Blockage | Smith [17] |
| Reference incidence angle | Çetin et al. [18] |
| Reference deviation angle | Çetin et al. [18]; Carter and Huges [19] |
| Off-design deviation angle | Carmody and Creveling [20] |
| Blade profile losses | Lieblein [21]; Koch and Smith [22] |
| Shock losses | Koch and Smith [22]; Miller et al. [23] |
| Endwall losses | Koch and Smith [22] |
| Secondary losses | Tournier and El-Genk [24]; Yaras and Sjolander [25] |
| Off-design losses | Carmody and Creveling [20] |
| Surge detection | White et al. [26] |
| Choking detection | Solver divergence for $\dot{m} > \dot{m}_{design}$ |

The outputs of SATURN are perfomance maps (Figure 2.1) and spanwise plots of pressure ratio, isentropic efficiency and temperature ratio (Figure 2.2). The time required for the calculation of three different speedlines is about 50 s on a laptop with 8 GB RAM and an Intel i7-8550u 1.80 GHz speed processor. Calculation time reduce to 0.10 s for prediction of the performance at design condition.

Despite the simplicity of the low-order meanline code, compressor maps predictions show satisfactory accuracy, especially for pressure ratio (Figure 2.1(a)). A fine matching has been obtained at design point. Furthermore, spanwise plots of Figure 2.2 show some important aspects, as the loss in efficiency that occurs for high spans as consequence of strong shock waves and tip clearance flows.

To conlude, notice that the spanwise-average of a general quantity, $x$, is computed as

$$\bar{x} = 0.2 \cdot x_{\text{span } 10\%} + 0.6 \cdot x_{\text{span } 50\%} + 0.2 \cdot x_{\text{span } 90\%}. \tag{2.1}$$

This will help in understanding some results of the SA explained in the next Section.

(a) *Pressure Ratio*  (b) *Isentropic Efficiency*
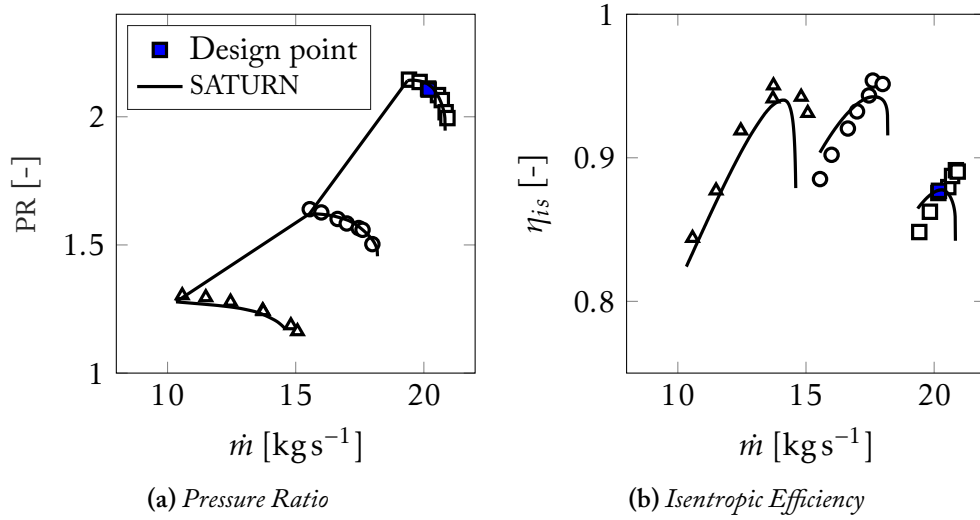
**Figure 2.1.** *Performance maps of the Rotor 37 predicted by SATURN and experimental data*



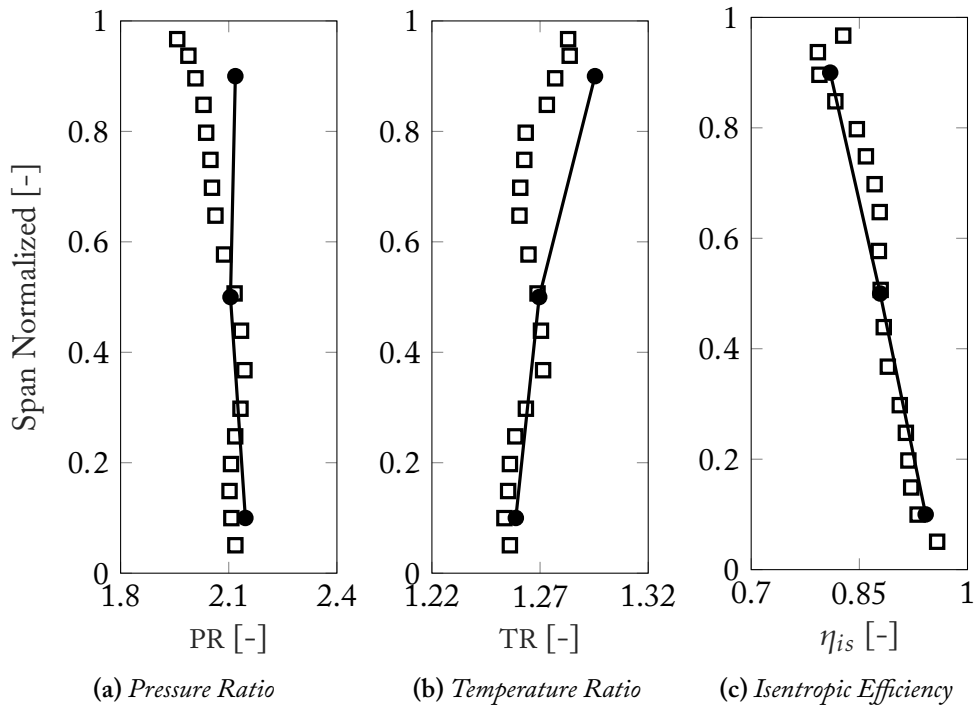(a) *Pressure Ratio*  (b) *Temperature Ratio*  (c) *Isentropic Efficiency*

**Figure 2.2.** *Spanwise plots of PR, TR and $\eta_{is}$ at maximum efficiency mass flow rate $\dot{m} = 20.6 \text{ kg s}^{-1}$. Experimental data are white squares and SATURN data are black circles*

## 2.2    Sensitivity Analysis

To understand the importance of design variables listed in Table 2.1 with respect to optimization objectives and constraints (see Equation 1.4 and Equation 1.5), a SA based on Student's t-test has been conducted. This step has been fundamental for two reasons. Firstly, it helped with the debugging of the meanline code itself, as in a first instance some incongruities with theoretical expectations arose. Secondly, it allowed to decide whether it was possible to exclude some variables from the optimization process because of their small statistical significance.

### 2.2.1    Sampling method

The first step of the analysis has been the determination of the exploration space. It has been considered reasonable to define a range of variation for each decision variable of ±7.5% centered on the Rotor 37 baseline values. This was to understand the behaviour of compressor rotors that are similar to the baseline one. Big improvements with respect to the Rotor 37 performance were not expected, since this compressor rotor was already optimized during the design phase.

In defining the exploration space, the sampling methodology adopted is very important to properly find a non-clustered distribution of samples. Various sampling methods exist. Random sampling is a completely stochastic method that, as obvious, choose value of variables in the specified exploration space randomly. Another method is called Latin Hypercube Sampling (LHS), which is similar to a random sampling but avoids any possibility of samples clustering around a specific point in the exploration space by means of a deterministic algorithm, allowing to get a more homogeneous distribution of samples. It is possible to appreciate the difference between the two sampling methods in Figure 2.3. Note how in Figure 2.3(a) some clusters of points have been generated in the center-right and in the bottom-right part of the plot, while the points are better distributed in Figure 2.3(b).
In this SA, an arbitrary number of 300 samples have been generated for each variable trough LHS.

### 2.2.2    Setup and results

The analysis has been conducted in MATLAB environment. The outputs of SATURN have been used to conduct both cross-correlation tests and Student's t-test.

Student's t-test is based on the definition of a parameter, $t$, that is a measure of the importance of a single variable with respect to a single objective (or constraint). The higher the value of $t$ and the higher is the influence of the variable on the objective (or constraint).
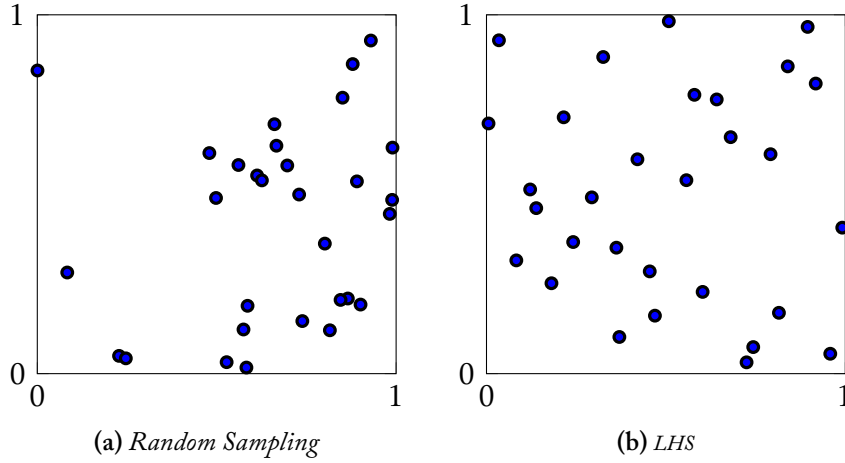
(a) *Random Sampling*  (b) *LHS*

**Figure 2.3.** *Comparison between Random sampling and Latin Hypercube sampling of 30 points in a 2D space*

The procedure adopted to determine `angBladeIn` significance on pressure ratio at a generic span is described below as an example. The same applies to other combinations of variables and objectives.

1. Definition of the input matrix $\mathbf{M_{in}}$: it is a $n \times m$ matrix, where $m$ is the number of variables listed in Table 2.1 and $n$ is number of samples (300). Columns of $\mathbf{M_{in}}$ contain sampled data of each variable. Rows of $\mathbf{M_{in}}$ contain variables in the same order as they appear in Table 2.1. The first column of $\mathbf{M_{in}}$ contains 300 samples of `angBladeIn`;

2. Definition of the output matrix $\mathbf{M_{out}}$: it is a $n \times 4$ matrix. Its rows contain perfomance of compressor rotors defined by geometric variables in the rows of $\mathbf{M_{in}}$. Perfomance stored in the output matrix are, in order, pressure ratio, isentropic efficiency, surge margin and choking margin. The first column of $\mathbf{M_{out}}$ contains pressure ratio values of 300 compressor rotors;

3. The first column of $\mathbf{M_{out}}$ has been taken into account. Then, rows containing a value of PR greater than the baseline one have been marked with a +, while rows with smaller values have been marked with a −;

4. The first column of $\mathbf{M_{in}}$ has been considered. Rows have been separated according to + and − marks of the first column of $\mathbf{M_{out}}$. Two vectors have been defined, respectively $\vec{x}_{1+}$ and $\vec{x}_{1-}$. For clarity's sake, items in $\vec{x}_{1+}$ are those values of `angBladeIn` to which it corresponds a pressure ratio higher than the baseline one;

5. Student's t-parameter is defined as

$$t_1 = \frac{\overline{\vec{x}_{1+}} - \overline{\vec{x}_{1-}}}{\sqrt{\frac{s_{x_{1+}}^2}{m_{1+}} + \frac{s_{x_{1-}}^2}{m_{1-}}}} \tag{2.2}$$

where $\overline{\vec{x}_{1+}}$ and $\overline{\vec{x}_{1-}}$ are mean values of $\vec{x}_{1+}$ and $\vec{x}_{1-}$, $s_{x_{1+}}^2$ and $s_{x_{1-}}^2$ are $\vec{x}_{1+}$ and $\vec{x}_{1-}$ standard deviations and $m_{1+}$ and $m_{1-}$ are the number of elements in $\vec{x}_{1+}$ and $\vec{x}_{1-}$.

The procedure above has been repeated for each variable and each objective and constraint at each span. Significance level has been then normalized for each objective and constraint, so that

$$\sum_{j=1}^{m} t_j = 1, \tag{2.3}$$

being $m$ the number of variables. Significance level has been classified as *low* if $t \simeq 0$ and as *high* if $t$ approaches unity. Figure 2.4 shows significance of variables with respect to compressor performance. Note that there are no variables with much greater significance than others: they are all of the same order of magnitude. Moreover, some theoretical expectations have been confirmed.

First of all, is it possible to appreciate the important influence of the geometry of the blade by means of angBladeIn and angBladeOut on pressure ratio and on isentropic efficiency. The geometry of the blade contributes both to shock wave pattern and to boundary layer separation. However, blade angles are the most significant variables even for surge and choking margin. Second of all, notice that angBladeIn at span 50 % is the most influential variable on isentropic efficiency. This is due a variation in incidence and isentropic efficiency changes as a consequence. radCurvIn at span 90 % is also important for efficiency, as it is connected to the shock wave geometry at the tip. Notice, in addition, that variables at span 90 % are the most significant for surge margin. It is known that the tip of the blade is critical when it comes to stall of compressors. SATURN predictions confirm this aspect.

Moreover, angBladeIn at span 50 % has big influence on choking margin, as the solver finds difficult to reach convergence when incidence increases. Physically, on the other hand, this is related to a change in the throat area and, consequently, in the choking mass flow rate.

Generally, because of Equation 2.1, variables at span 50 % have greater significance with respect to the same variables at the other spans (except for significance on surge margin).
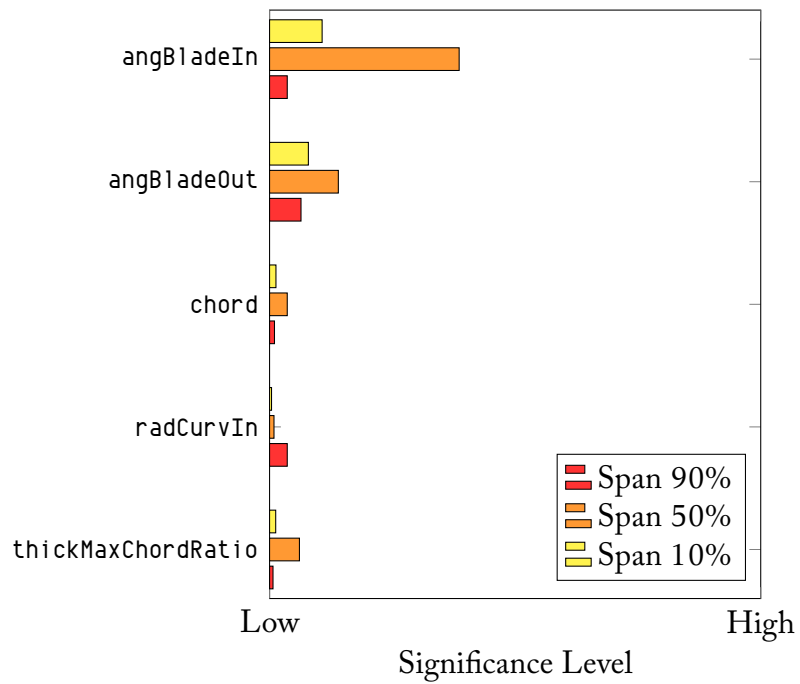
**(a)** *Pressure Ratio*
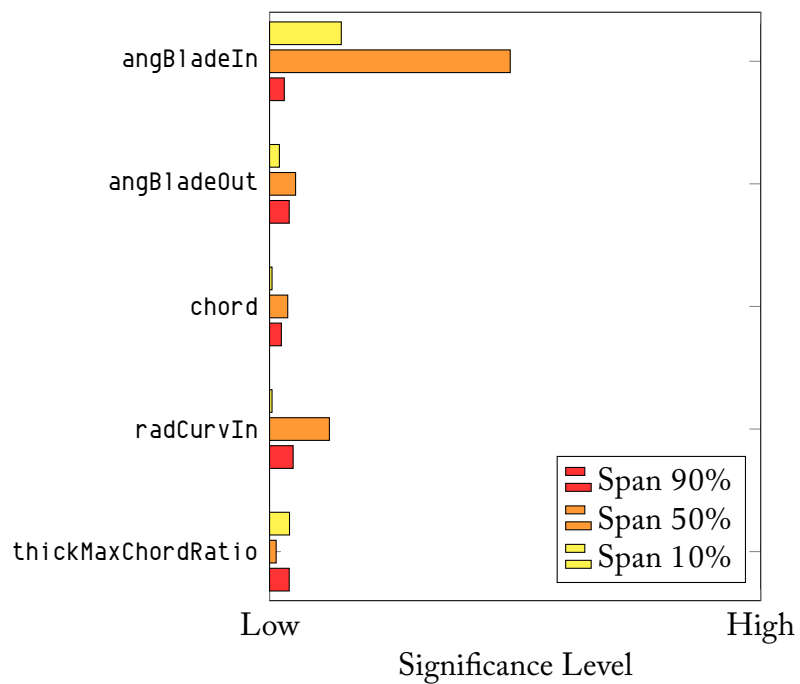


**(b)** *Isentropic Efficiency*

**Figure 2.4.** *Significance of variables with refer to compressor perfomance*
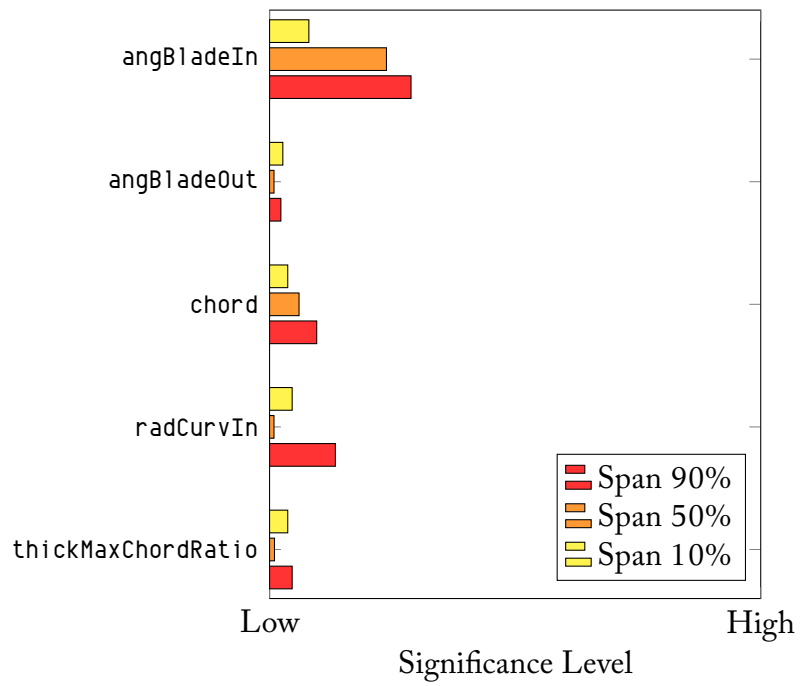
(c) *Surge Margin*
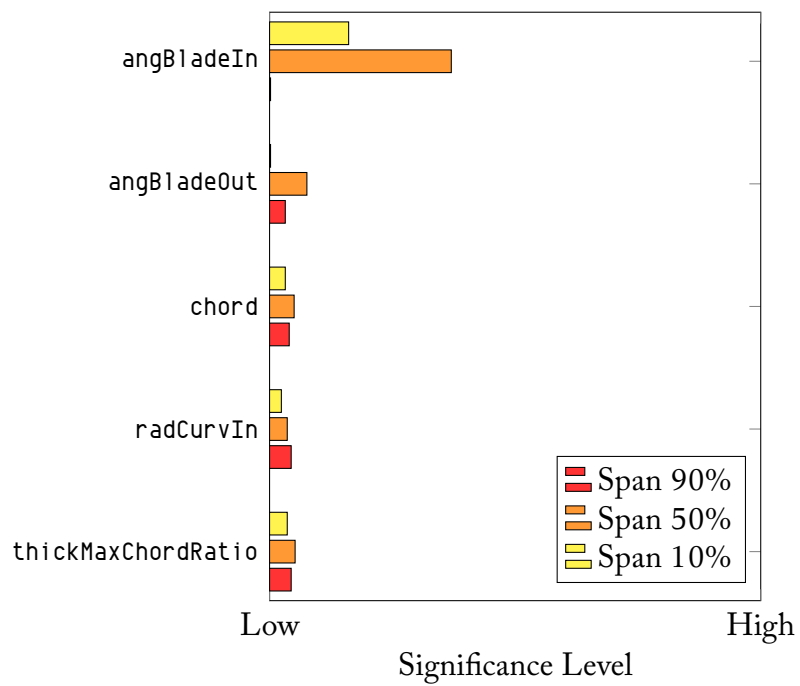


(d) *Choking Margin*

**Figure 2.4.** *Significance of variables with refer to compressor perfomance*

Cross correlation analysis has been performed too, but only one strong correlation arose, that is the one between pressure ratio and isentropic efficiency. As known, in fact, PR and $\eta_{is}$ are linked by Equation 2.1.

To conclude, no variables will be excluded from the optimization process because, for one or the other objective or constraint, their significance is of the same order of magnitude. It is not possible to reduce the number of variables and, therefore, the number of individuals will be set high enough to guarantee good genetic diversity, as explained in the next Chapter.

# 3 | Optimization

*Firstly, an introduction on genetic algorithms is presented in this chapter. Later, optimization process is described. Finally, the procedure used to build a three-dimensional geometry starting from optimized geometrical variables given by SATURN is explained.*

## 3.1 Genetic Algorithms

Genetic algorithms were been successfully used by many authors to find optimal solutions to a non-linear optimization problem (Benini and Toffolo [27]; Shahpar et al. [28]). In particular, in aerodynamic and structural problems, where an high degree of non-linearity is usually present, other optimization algorithms don't perform as good as an evolutionary algorithm (Skinner and Zare-Behtash [29]).

A genetic algorithm is a bio-inspired optimization procedure that can be defined as an heuristic search inspired by Charles Darwin's theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction in order to create offspring of the next generation.

The process of natural selection starts with the selection of fittest individuals from a population. They produce offspring which inherit the characteristics of the parents and will be added to the next generation. It is expected that offspring have better characteristics than parents, so generations keep improving along an iterative process and, at the end, a generation with the fittest individuals will be found. This notion can be applied for optimization problems of highly non-linear functions, where analytic solutions are impossible to find. Moreover, genetic algorithms are very robust in finding optimal solutions to complex problems and for this reason they're widely used in engineering optimization processes.

Six phases are considered in a genetic algorithm: initialization of initial population, fitness function definition, selection of the fittest, individual crossover, individual mutation and elitism.

### 3.1.1    Initialization of initial population

Initial population, or generation, is composed by an arbitrary number of individuals. A good-practice rule is to use a number of individuals equal to $2m \div 4m$, with $m$ the number of design variables. An individual is a binary string composed by $q$ bits, that are the *genes* of the individual. The number of bits $q$ determines the accuracy of the conversion from natural numbers to binary string: the higher the value of $q$, the higher the accuracy. It is important that individuals are uniformly distributed in the design variables exploration space: this is to help the algorithm in the research of different local minima.

### 3.1.2    Fitness function definition

Fitness function is defined as the function to be minimized in during the optimization process. The value of this function, that is the *fitness* of an individual, is used to determine which will reproduce and survive into the next generation, that is relevant to the objective functions to be optimized. The fitness of an individual represents the probability of its survival. The greater the fitness, the greater the probability it will survive.

### 3.1.3    Selection

Selection is the stage of a genetic algorithm in which individuals are chosen from a population for later breeding. One of the most common selection methods used is known as *Tournament Selection* (Figure 3.1). A random number of individuals are chosen from current generation and those with the highest fitness are selected for reproduction, so that only the best genetic makeups will survive to the next generation.
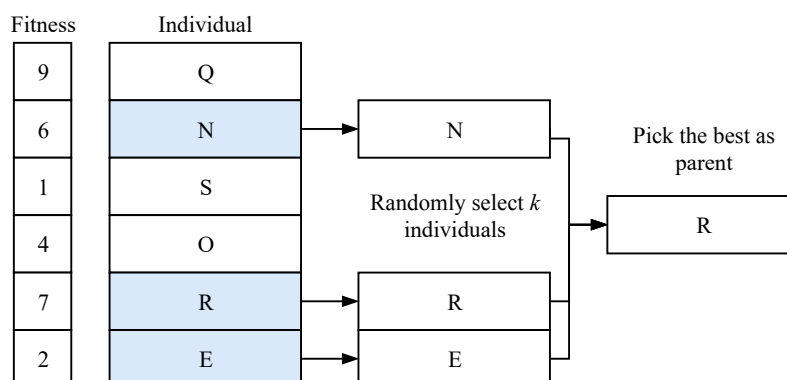


**Figure 3.1.** *Schematisation of tournament selection procedure*

### 3.1.4   Crossover

Crossover is a genetic operator used to mix the genetic information of two parents to generate new offspring. It is used to stochastically generate new individuals from an existing population, and analogous to the crossover that happens during sexual reproduction in biology. Different crossover methodologies exist. Single point crossover is shown in Figure 3.2.
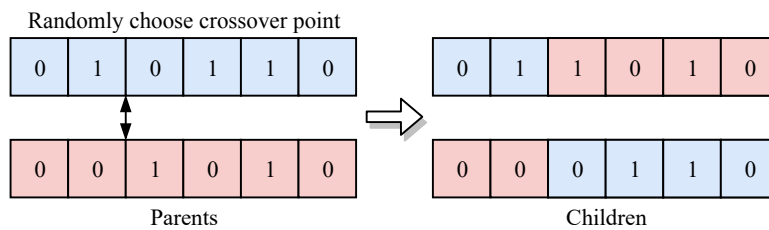


**Figure 3.2.** *Crossover between two individuals*

### 3.1.5   Mutation

Mutation is a genetic operator used to preserve genetic diversity from one generation to the next. It is analogous to biological mutation. Mutation alters one or more gene values in a chromosome from its initial state. In mutation, the solution may change entirely from the previous solution. Hence the algorithm can come to a better solution by using mutation. Mutation occurs during evolution according to a user-definable mutation fraction (or probability), that should be set low to avoid falling back into a primitive random search.
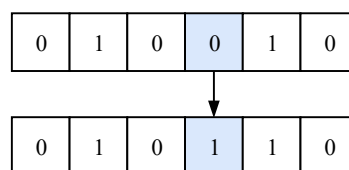


**Figure 3.3.** *Mutation of a gene*

### 3.1.6   Elitism

Elitism involves copying a small proportion of the fittest candidates, unchanged, into the next generation. Candidate solutions that are preserved unchanged through elitism remain eligible for selection as parents.

## 3.2  Optimization Setup

Two optimization objectives have been identified, as introduced in Section 1.3: total pressure ratio PR and isentropic efficiency $\eta_{is}$, both calculated at design condition. The first one is important because it allows to produce more specific work by keeping unchanged overall dimensions of the machine. The second one, instead, is important for obvious reasons: the greater the isentropic efficiency and the lower the amount of work that need to be given by a turbine to the compressor. Moreover, two constraints have been introduced in order to preserve the original operating range of the compressor: surge margin SM and choking margin CM.

The optimization has been carried out using the multi-objective genetic algorithm `gamultiobj`, which belongs to the *Global Optimization Toolbox* of MATLAB and extensively described by Konak et al. [30]. The algorithm requires the formulation of a fitness function to be minimized. For this reason, objectives have been defined as:

$$y_1 = \frac{1}{\text{PR}} \quad \text{and} \quad y_2 = \frac{1}{\eta_{is}}. \tag{3.1}$$

Two external penalty function have been used to account for constraints. Equations of constraints are defined as follow:

$$g_1(\vec{x}) = 0.9 - \frac{\text{SM}}{\text{SM}_{\text{Rotor 37}}} \leq 0 \tag{3.2}$$

$$g_2(\vec{x}) = 0.9 - \frac{\text{CM}}{\text{CM}_{\text{Rotor 37}}} \leq 0. \tag{3.3}$$

where $\vec{x}$ is a vector containing optimization variables. Penalty functions, then, are

$$G_j(\vec{x}) = \max\left[0, \, g_j(\vec{x})\right], \qquad j = 1, 2. \tag{3.4}$$

Penalty functions assume that particular formulation because it has been considered acceptable to have a decrease by 10 % of SM and CM baseline values. Margins lower than these values are not accepted and the fitness of the individual gets penalized: the lower the margins and the higher the penalization. Penalization is zero when margins are greater than the baseline ones.

Finally, the fitness function has been formulated as

$$\varphi(\vec{x}) = \vec{y}(\vec{x}) + C \cdot (G_1(\vec{x}) + G_2(\vec{x})), \tag{3.5}$$

where $\vec{y}(\vec{x}) = (y_1(\vec{x}), \, y_2(\vec{x}))$ and $C$ is a coefficient used to weight the constraints and that varies during generations following a cubic function. By doing this, the constraints will be more flexible for initial generations and more rigid with the advancement of generations number.

**Table 3.1.** *Setup of the optimization trials. Calculations have been carried out on a laptop with* 8 GB *RAM and an Intel i7-8550u* 1.80 GHz *speed processor*

| Optimization ID | opt1 | opt2 | opt3 | opt4 |
|---|---|---|---|---|
| Generations | 30 | 35 | 50 | 90 |
| Individuals | 15 | 45 | 50 | 50 |
| Crossover fraction | 0.75 | 0.75 | 0.75 | 0.8 |
| Mutation fraction | 0.24 | 0.24 | 0.24 | 0.2 |
| Elitism fraction | 0.01 | 0.01 | 0.01 | 0 |

Once defined the fitness function, 4 different optimization trials have been set. The initial population has been sampled through LHS and Rotor 37 has been included in it to help the algorithm in its optimum research. On the other side, the trials differ in some aspects. To begin with, opt1 is an optimization trial where optimization variables are only those at 50 % span, as the SA proved that they have a slightly higher significance than other variables with respect to optimization objectives. For opt1, the number of generations has been kept low because a low CPU time has been preferred instead of high performance improvements. For opt2, opt3 and opt4, on the other side, all spanwise positions have been considered. Lower and upper limits of variables are respectively −7.5% and +7.5% with respect to Rotor 37 values. Secondly, trials differ in generations and individuals number. The number of individuals is linked to the number of optimization variables. Generally, it is approximately equal to three times the number of variables. Generations number is different too, and this is to find the right trade-off between CPU time and performance improvement. Mutation fraction has been set relatively high to widely explore possible local minima for each optimization trial. Detailed data of optimization trials are reported in Table 3.1. The whole procedure has been implemented in MATLAB environment and it is reported in Figure 3.4.
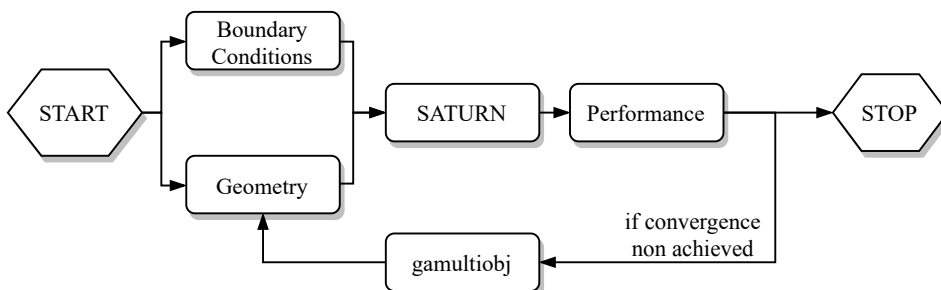


**Figure 3.4.** *Workflow of the optimization algorithm*

## 3.3    Optimization Results

The results of a multi-objective optimization are expressed in term of *Pareto front*. A Pareto front is defined as a set of non-dominated solutions of an optimization problem. Consider two individuals on the Pareto front of a two-objectives ($y_1$ and $y_2$) optimization problem. The first individual is better than the second one for what it concerns objective $y_1$, but it is worst for $y_2$. Then, it is impossible to state which of the two individuals is better than the other one: the two individuals are two non-dominated solutions of the optimization problem. In this specific work, however, the objectives are linked by Equation 1.4 and a strong correlation between them exists. As a result, Pareto front degenerates in a single point for each optimization trial. Results are reported in Table 3.2.

Firstly, notice that every trial produced a considerable increment on pressure ratio and on isentropic efficiency. Secondly, constraints have been respected, even if opt1, opt3 and opt4 have lower surge margin with respect to Rotor 37 value. Furthermore, opt1 has also a lower choking margin. A good trade-off between CPU time and performance improvement has been found, that is opt3. Note, in fact, that despite a higher calculation time, the performance of opt4 is not so much better than the one of opt3, so opt3 has been preferred. Optimization trials have recommended to significantly reduce by 7 % the variable `angBladeOut` at span 50 %, so that its value is very close to the lower bound of `angBladeOut`. Other optimized values, instead, place more distant from their upper and lower bounds, giving more value to the optimization process. As for the variable `radCurvIn`, it is almost unchanged in every trial. Moreover, camber angle of airfoils is increased at every spanwise location: probably, the increase in pressure ratio is caused by these variations, but confirmation will be given by CFD simulations of the next Chapter. Finally, shifting the attention to opt3, notice how the result of the optimization is a blade with lower thickness with respect to Rotor 37 (Table 3.3). Perfomance maps of opt3 are reported in Figure 3.5.

**Table 3.2.** *Results of the optimization trials. Improvements are referred to performance of the Rotor 37*

| Optimization ID | opt1 | opt2 | opt3 | opt4 |
|---|---|---|---|---|
| $\Delta$PR [%] | 3.18 | 3.23 | 4.75 | 4.96 |
| $\Delta\eta_{is}$ [%] | 0.62 | 0.71 | 1.31 | 1.36 |
| $\Delta$SM [%] | −1.33 | 2.62 | −7.96 | −5.32 |
| $\Delta$CM [%] | −1.41 | 2.71 | 5.69 | 1.25 |
| CPU Time [h] | 3.35 | 9.45 | 13.52 | 23.82 |

**Table 3.3.** *Geometrical properties of opt3*

| Variable Name | Span [%] | Rotor 37 | opt3 | Δ [%] |
|---|---|---|---|---|
| angBladeIn | 10 | 52.37° | 53.83° | 2.79 |
| | 50 | 56.53° | 58.14° | 2.85 |
| | 90 | 60.76° | 61.08° | 0.53 |
| angBladeOut | 10 | 22.68° | 21.83° | −3.75 |
| | 50 | 38.87° | 36.10° | −7.13 |
| | 90 | 48.18° | 47.09° | −2.26 |
| chord | 10 | 56.04 mm | 58.03 mm | 3.55 |
| | 50 | 55.70 mm | 53.97 mm | −3.11 |
| | 90 | 56.03 mm | 57.13 mm | 1.96 |
| radCurvIn | 10 | 0.24 mm | 0.25 mm | 0.48 |
| | 50 | 0.19 mm | 0.19 mm | 0.50 |
| | 90 | 0.14 mm | 0.14 mm | 0.29 |
| thickMaxChordRatio | 10 | 0.0758 | 0.0733 | −3.32 |
| | 50 | 0.0544 | 0.0517 | −4.90 |
| | 90 | 0.0355 | 0.0347 | −2.27 |



(a) *Pressure Ratio*

(b) *Isentropic Efficiency*
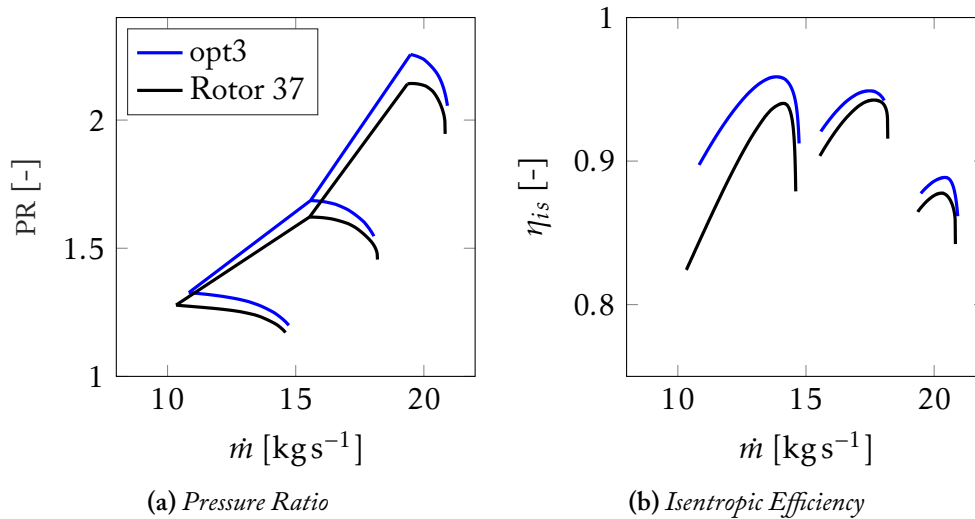
**Figure 3.5.** *Performance maps of the Rotor 37 and of opt3 predicted by SATURN*

# 3.4 Optimized Blade Geometry

The outputs of the optimization are expressed in terms of geometrical properties of the blade at different spanwise locations. It is necessary, then, to convert those properties into a three-dimensional model of the blade which is going to be simulated using CFD for the verification of its performance. The first step is the parametrization of the baseline blade airfoil shape at span 10 %, 50 % and 90 %. These airfoils will then be modified according to values listed in Table 3.3 to obtain the optimized blade.

## 3.4.1 Baseline blade parametrization

The geometry of the Rotor 37 is found in ANSYS Turbogrid example files and it is expressed in terms of `.curve` files. Such files contains the cartesian coordinates $x, y, z$ of the airfoil at 6 different spanwise locations: 0 %, 20 %, 40 %, 60 %, 80 % and 100 % of the blade height. Airfoils at intermediate locations have been obtained through linear interpolation. Then, cartesian coordinates have been converted to cylindrical coordinates $r, \vartheta, z$ to facilitate the parametrization process.

Different methods exist to uniquely define an airfoil shape: in this work, parametrization will be done by means of the definition of its camber line and its thicknesses distribution. Consider for example the Rotor 37 airfoil shape at span 10 % (Figure 3.6(a)).
Camber line is defined as the line that joins the leading and trailing edges and that is equidistant from the suction side and the pressure side of the airfoil. Once calculated, it has been parametrized through a Bézier curve. Bézier curves are often used when it comes to parametrize a curve, since the shape of the curve can be easily modified by editing control points. Bézier curve $B(x)$ is defined by the following equation:

$$B(x) = \sum_{i=0}^{n} \binom{n}{i} (1 - x)^{n-i} x^i P_i, \qquad (3.6)$$

where $P_i$ are the coordinates of the control points. In this work, camber line has been parametrized using 5 control points (Figure 3.6(b)). Control points coordinates have been found by means of a genetic algorithm that has minimized the discrepancy between real camber line and interpolated camber line.
Thickness, instead, is defined as the perpendicular distance between a point on the camber line and a point on the suction (or pressure) side. Thickness distribution of the airfoil at span 10 % is reported in Figure 3.6(c) [1].
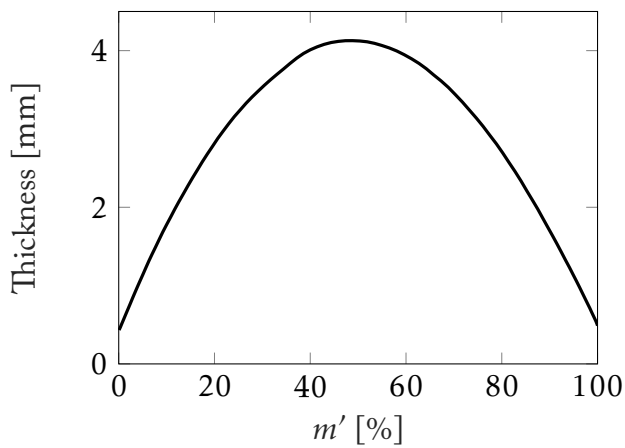
---

[1] $m' = \frac{\sqrt{\partial r \partial r + \partial z \partial z}}{R}$ is the radius normalized distance along meridional curve
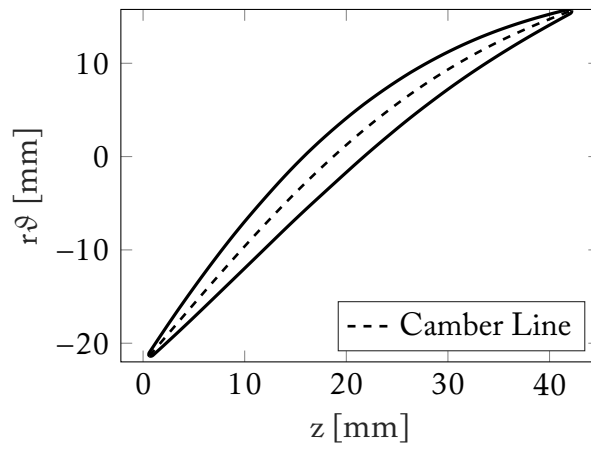
(a) *Airfoil shape and camber line*
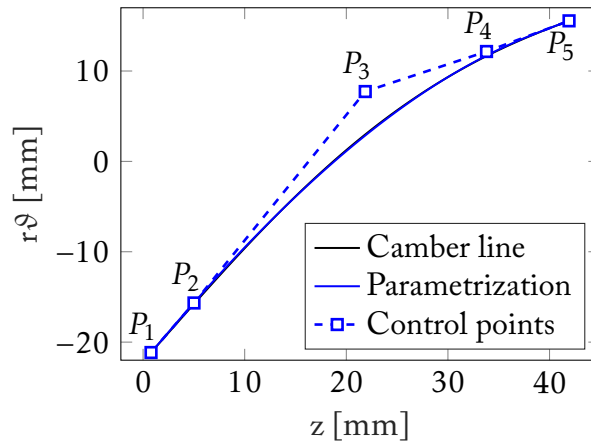


(b) *Parametrization of the camber line*



(c) *Thickness distribution*

**Figure 3.6.** *Parametrization of the baseline airfoil at span* 10 %

### 3.4.2 Optimized blade reconstruction

Starting from the curves of Figure 3.6, the optimized blade has been reconstructed by following the opposite process. First of all, geometrical parameters of the blade have been adjusted according to Table 3.3. Then, camber line and thickness distribution curves have been imported into ANSYS BladeGen to build a three-dimensional geometry.

The parametrization of the camber line by means of a Bézier curve allowed to easy modify the values of `chord`, `angBladeIn` and `angBladeOut`. With respect to Figure 3.6(b), it should be noted that the Bézier curve is tangent to the segments that join $P_1$ to $P_2$ and $P_4$ to $P_5$.

First of all, chord length has been modified. Call $\Delta c = \text{chord}_{opt3} - \text{chord}_{\text{Rotor 37}}$, $Q_i$ the control points of the optimized camber line and $\psi$ the angle defined by

$$\tan \psi = \frac{y_{P_5} - y_{P_1}}{x_{P_5} - x_{P_1}}, \tag{3.7}$$

then points $Q_1$ and $Q_5$ have been calculated as:

$$\begin{cases} x_{Q_1} = x_{P_1} - 0.5 \cdot \Delta c \cdot \cos \psi \\ y_{Q_1} = x_{P_1} - 0.5 \cdot \Delta c \cdot \sin \psi \\ x_{Q_5} = x_{P_5} + 0.5 \cdot \Delta c \cdot \cos \psi \\ y_{Q_5} = x_{P_5} + 0.5 \cdot \Delta c \cdot \sin \psi. \end{cases} \tag{3.8}$$

Second of all, $Q_2$ and $Q_4$ have been set according to `angBladeIn` and `angBladeOut` optimized values. More in detail, since `angBladeIn` is defined by the tangent to the camber line from the leading edge, $Q_2$ coordinates are such as to respect the following condition:

$$\tan \left( \text{angBladeIn}_{opt3} \right) = \frac{y_{Q_2} - y_{Q_1}}{x_{Q_2} - x_{Q_1}}. \tag{3.9}$$

By doing so, once $x_{Q_2}$ has been arbitarly fixed, the value of $y_{Q_2}$ is calculated. Similarly, $Q_4$ coordinates have been calulated according to

$$\tan \left( \text{angBladeOut}_{opt3} \right) = \frac{y_{Q_5} - y_{Q_4}}{x_{Q_5} - x_{Q_4}}. \tag{3.10}$$

Finally, the position of $Q_3$ has been set such that the camber line had a reasonable curvature:

$$\begin{cases} x_{Q_3} = x_{Q_2} + 0.5 \cdot \left( x_{Q_4} - x_{Q_2} \right) \\ y_{Q_3} = y_{Q_2} + 3 \cdot \left[ \left( y_{Q_2} - y_{P_2} \right) + \left( y_{Q_4} - y_{P_4} \right) \right] \end{cases} \tag{3.11}$$

Optimized thickness distribution $t_{opt}(m')$ has been calculated using the following equation:

$$t_{opt}(m') = \left( \frac{\texttt{chord}_{opt3}}{\texttt{chord}_{\text{Rotor 37}}} \right) \left( \frac{\texttt{thickMaxChordRatio}_{opt3}}{\texttt{thickMaxChordRatio}_{\text{Rotor 37}}} \right) \cdot t_{\text{Rotor 37}}(m'). \quad (3.12)$$

As for the variables `radCurvIn`, they've not been considered as they've remained almost unchanged during the optimization process. Procedure described above has been applied for each spanwise location. Optimized airfoil shapes are reported in Figure 3.7.



(a) *Span* 10 %
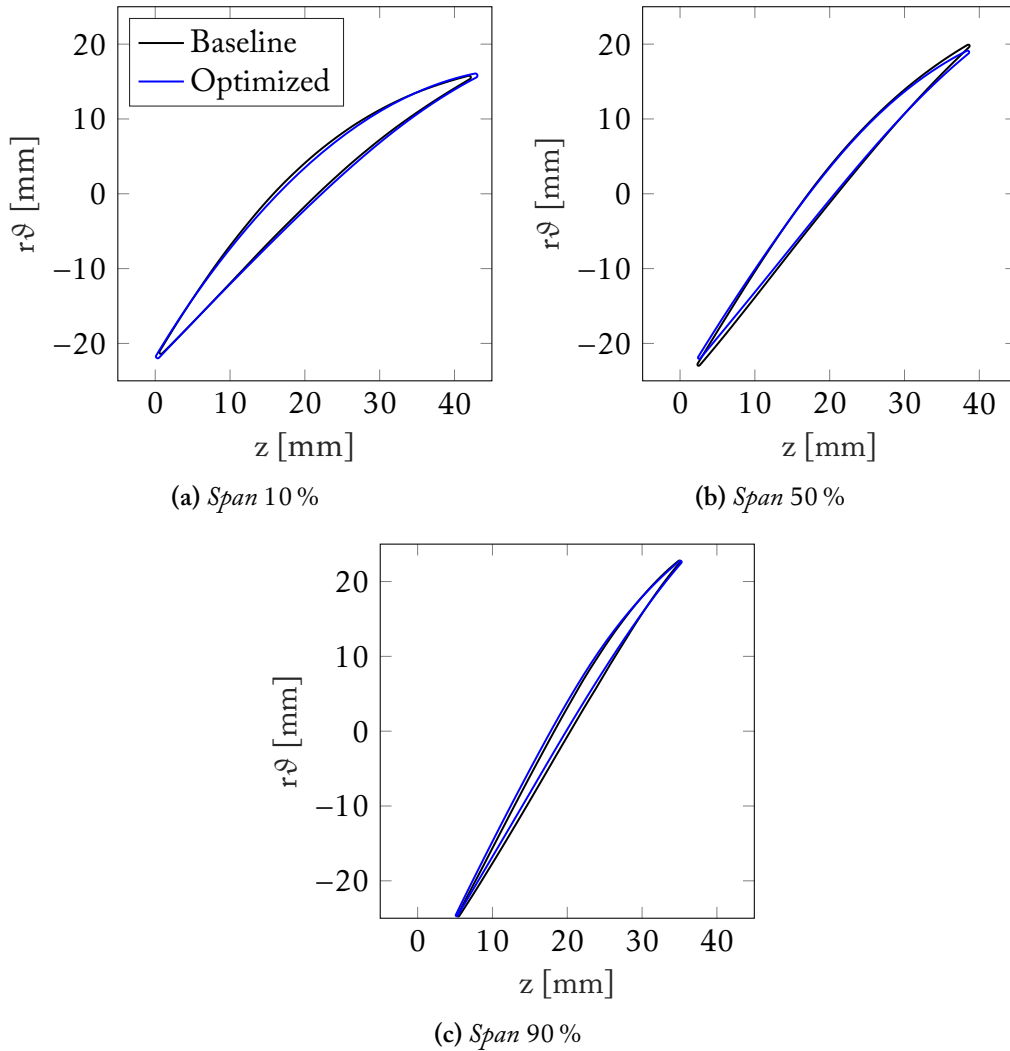
(b) *Span* 50 %

(c) *Span* 90 %

**Figure 3.7.** *Comparison between baseline and optimized airfoil shapes*

Finally, optimized thickness distributions $t_{opt}(m')$ and optimized camber lines descibed by the curve $\vartheta(m')$ have been imported into ANSYS BladeGen, so that to build a three-dimensional geometry (Figure 3.8).
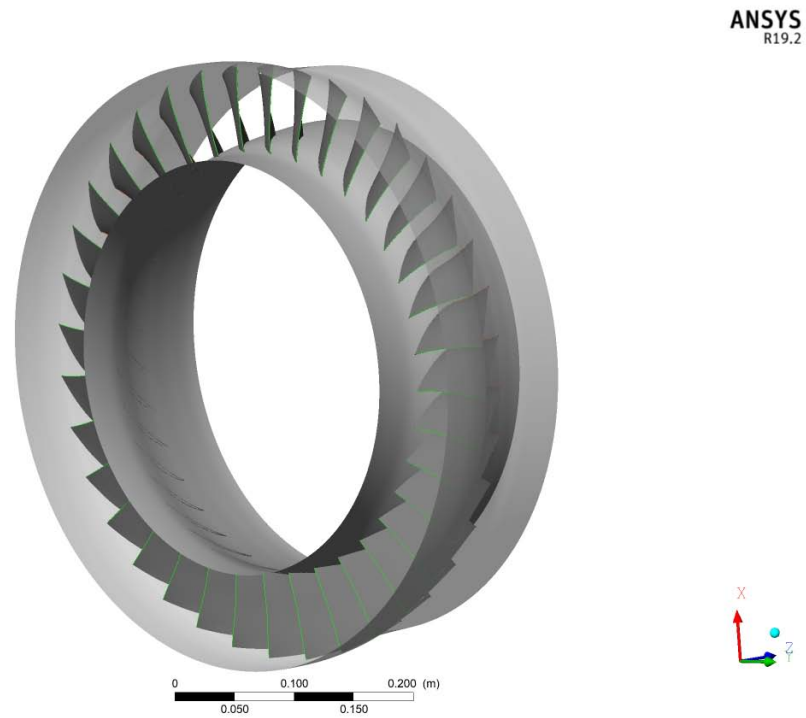


**Figure 3.8.** *3D geometry of opt3. Rotation axis is +z*

# 4 | CFD Analysis

*The optimized geometry has been simulated through CFD using a 3D compressible flow RANS solver, that is ANSYS CFX, and a comparison whith the prediction of the mean-line code is made. The creation of the mesh, the setup of the simulations and the post processing of the results are described in this chapter. An hybrid solution with higher efficiency is finally proposed.*

## 4.1 Mesh and Turbulence Model

The first important step when it comes to set-up a CFD analysis is the creation of the mesh, which consists of a number of finite control volumes where CFD equations are discretized and solved iteratively. In this work, the mesh has been created with ANSYS TurboGrid, as it easily allows to produce high quality meshes for turbomachinery. The number of mesh elements for a RANS simulation is related to the turbulence model adopted for the solution by means of the non-dimensional wall distance $y^+$, which is defined as

$$y^+ = \frac{u^* h}{\nu} \tag{4.1}$$

and depends on the placement of the first cell node above the wall ($h$) and on the flow features, that are the kinematic viscosity of the fluid ($\nu$) and the friction velocity calculated at the nearest wall ($u^*$). Different turbulence models exist. The most used in engineering applications are called $k - \epsilon$, $k - \omega$ and $k - \omega$ SST and they're employed in different situations:

- $y^+ < 5$: this condition is verified when the first cell node is in the viscous sub-layer, and $k - \omega$ model is used;

- $5 < y^+ < 30$: no accurate turbulence models exist for this situation, when the first cell node is found in the buffer layer;

– $y^+ > 30$: the first node is above the buffer layer, and it is found in the log
layer. In this situation, $k - \epsilon$ model is used and wall turbulence is modeled
by means of wall functions.

The $k - \omega$ SST turbulence model has been used in this work: it is a two-equation
turbulence model that has been developed for the accurate prediction of aero-
nautics flows with strong adverse pressure gradients and flow separation (Menter
et al. [31]). The SST formulation combines the best elements of $k - \epsilon$ and $k - \omega$
turbulence models. The model uses a $k - \omega$ formulation in boundary layer region,
making the model directly usable all the way down to the wall through the vis-
cous sub-layer, and switches to a $k - \epsilon$ behaviour in the free-stream, avoiding the
common $k - \omega$ problem that the model is too sensitive to the inlet free-stream
turbulence properties. The Reattachment Modification option has been enabled
within the $k - \omega$ SST turbulence model, as it has been proved to be beneficial in
order to reduce excessive flow separation in the suction side surface (Porro [32]).

The use of $k - \omega$ SST turbulence model required to have a very fine mesh near
to the walls, so that $y^+ < 5$. Mesh parameters of ANSYS TurboGrid have been
set according to the condition on $y^+$ value. The *Topology Set* is created with the
Automatic Topology and Meshing (ATM Optimized) with the method *Single
Round Round Refined*. This topology type enables to create high quality meshes
with minimal effort, and there is no need for control point adjustment. Then,
5 equidistant layers have been used in the spanwise direction to reduce mesh
distortion. The *Global Size Factor* has been set to 1.31 to have an average element
size of 0.05 mm. The size of the elements is progressively reduced near the walls
and in the tip gap, as it has been considered that these regions have a relevant
influence on efficiency losses. The resulting mesh of one blade passage has 0.99
million hexahedral elements and has been validated on Rotor 37 experimental
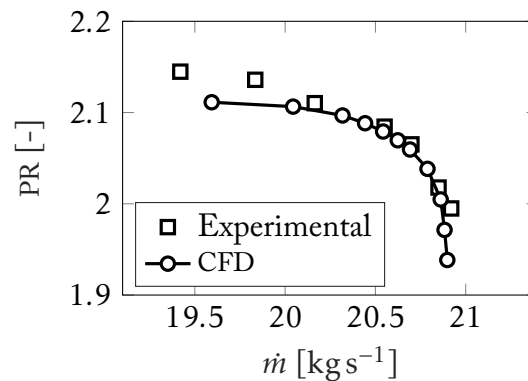data (Figure 4.1). The parameter $y^+$ is less than 1 everywhere.
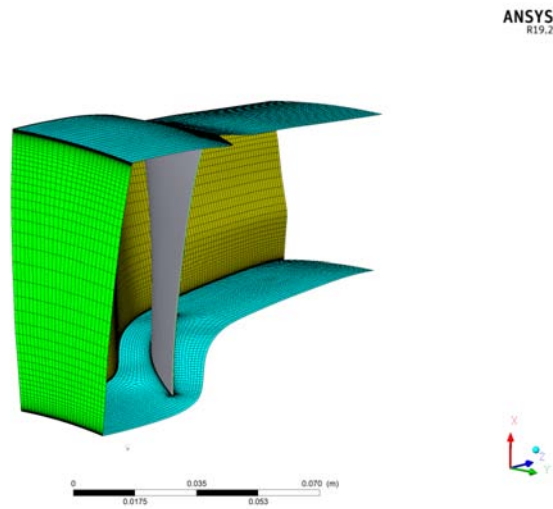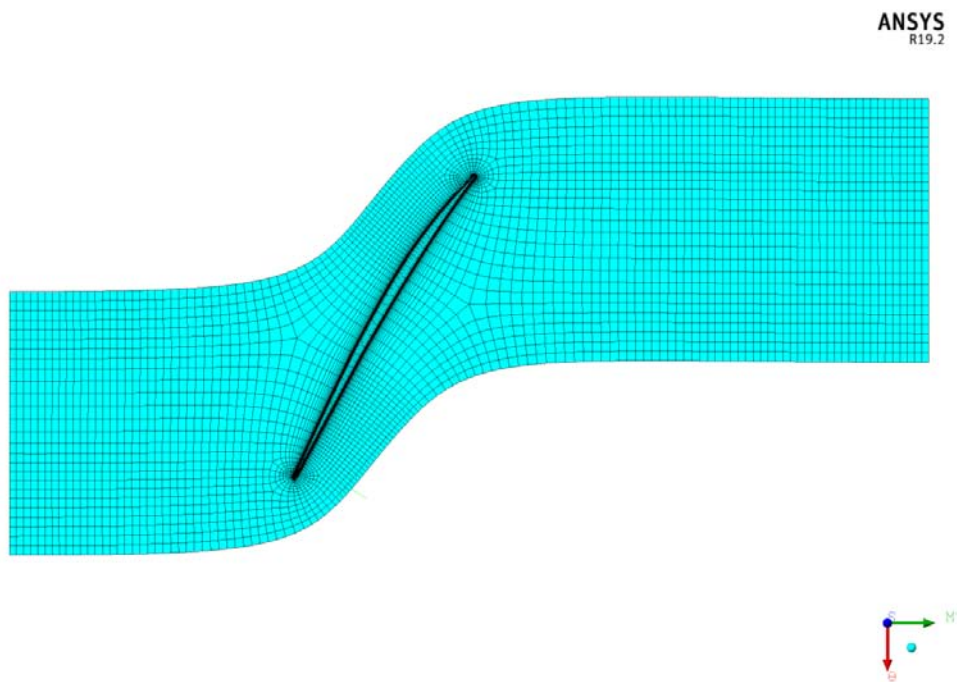


**Figure 4.1.** *Mesh validation on pressure ratio experimental data of the Rotor 37*

**(a)** *3D view of the mesh*



**(b)** *Mesh topology at the shroud. Note the high mesh density near to the blade to respect the condition $y^+ < 5$*

**Figure 4.2.** *Mesh of the optimized geometry. A single passage is shown*

## 4.2 Setup of the Simulation

It has been decided to use a commercial 3D compressible flow RANS solver, that is ANSYS CFX, to carry out the simulation of the nominal speedline. ANSYS CFX-Pre has been used to setup the simulation. Boundary conditions have been set as follows:

- rotational speed $\omega = 1800\,\text{rad}\,\text{s}^{-1}$;

- inlet total pressure $p_1^0 = 101\,325\,\text{Pa}$;

- inlet total temperature $T_1^0 = 288.15\,\text{K}$;

- to obtain the complete speedline, outlet static pressure $p_4$ varied for each simulation in order to simulate different mass flow rates ranging from the choking condition to the surge onset.

Thanks both to geometrical symmetry of the turbomachinery and symmetric properties of the boundary conditions, it has been possible to simulate one single blade passage instead of the entire turbomachinery: this allowed to drastically reduce computational time. Periodic boundary conditions have been set as a consequence of this assumption. The turbulence model has been set to $k-\omega$ SST and a limit of 1000 iterations was imposed. Other parameters are the defaults ones.

Control of the simulation has been carried out using CFX-Solver Manager, which allowed to select the numerical precision and to choose between the serial and the parallel run mode. For this study, the double precision scheme has been used, as it generally allows higher numerical accuracy for mathematical operations and reduces numerical errors propagation. The computer used for the simulation is a laptop with 8 GB RAM and an Intel i7-8550u 1.80 GHz speed quadcore processor. Simulations ran in parallel mode and four physical cores have been used in order to accelerate the convergence process. Each simulation has been initialized with the results of the previous one. The time required to reach convergence was about 7 hours for each simulation.

The results file of each simulation has been then loaded into CFX-Post for the post processing. After the initialization of the results, it is possible to define the performance of the compressor, isentropic efficiency and pressure ratio, by defining them in the *Expression Tab* according to Equations 1.4.

The same setup has been used to validate the mesh. This has been done to have comparable results. In the following analyses, the parameter $y^+$ has always been kept less than 1.

## 4.3    Performance of opt3

Compressor maps of opt3 have been obtained in a total time of approximately 50 hours (Figure 4.3).

To begin with, note the difference regarding the operating range: surge margin and choking margin are different to the ones predicted by SATURN (see Figure 3.5). The reason of this discrepancy is related to the blockage factor $\xi$, which accounts for the presence of the boundary layer on the walls and reduces the effective passage area.

As said in Section 2.1, SATURN uses some empirical models to calculate the performance of a machine, and one model is used to calculate the blockage factor (Smith [17]). The solver is then calibrated on experimental data by means of some coefficients. Blockage factor is finally calculated by SATURN as $\xi = \xi_{empirical} \cdot K_\xi$, where $\xi_{empirical}$ is calculated according to Smith [17] and $K_\xi$ is a calibration factor. Problems may arise when the geometry of the machine get changed, as it happened in the optimization process, because the calibration factor remains constant and loses its meaning. Since the blockage factor is linked to the mass flow rate by the following equation

$$\dot{m} = \frac{A \xi p^0}{\sqrt{T^0}} \sqrt{\frac{k}{R}} M \left( 1 + \frac{k-1}{2} M^2 \right)^{-\frac{k-1}{2(k-1)}}, \tag{4.2}$$

the overall effect of keeping $K_\xi$ unchanged is a translation on the $x$-axis of the performance curves, which caused the discrepancy. However, the analytic prediction of the blockage factor is challenging, and the use of a model is considered as an acceptable compromise.

Nevertheless, note that the performance of opt3 is better than the baseline one. Consider the performance at maximum efficiency mass flow rate for both the individuals: the optimized compressor has a considerable increase by 6.83 % on pressure ratio and an almost unchanged isentropic efficiency (Table 4.1). The operating range is slightly narrower in the optimized configuration.

**Table 4.1.** *Comparison between baseline and optimized performance at maximum efficiency mass flow rate (CFD results)*

|  | PR | $\eta_{is}$ | $\dot{m}_{surge}$ [kg s$^{-1}$] | $\dot{m}_{choking}$ [kg s$^{-1}$] |
|---|---|---|---|---|
| Rotor 37 | 2.04 | 0.8613 | 19.59 | 20.98 |
| opt3 | 2.18 | 0.8614 | 20.30 | 21.33 |
| $\Delta$ [%] | 6.83 | 0.01 | – | – |

(a) *PR*

(b) $\eta_{is}$
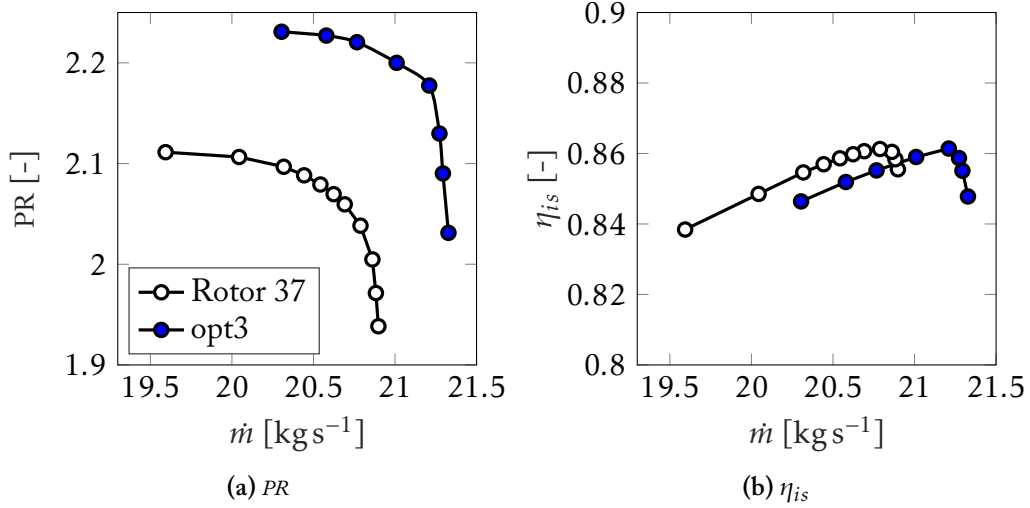
**Figure 4.3.** *Comparison between baseline and optimized nominal speedline (CFD results)*



(a) *Pressure Ratio*

(b) *Temperature Ratio*
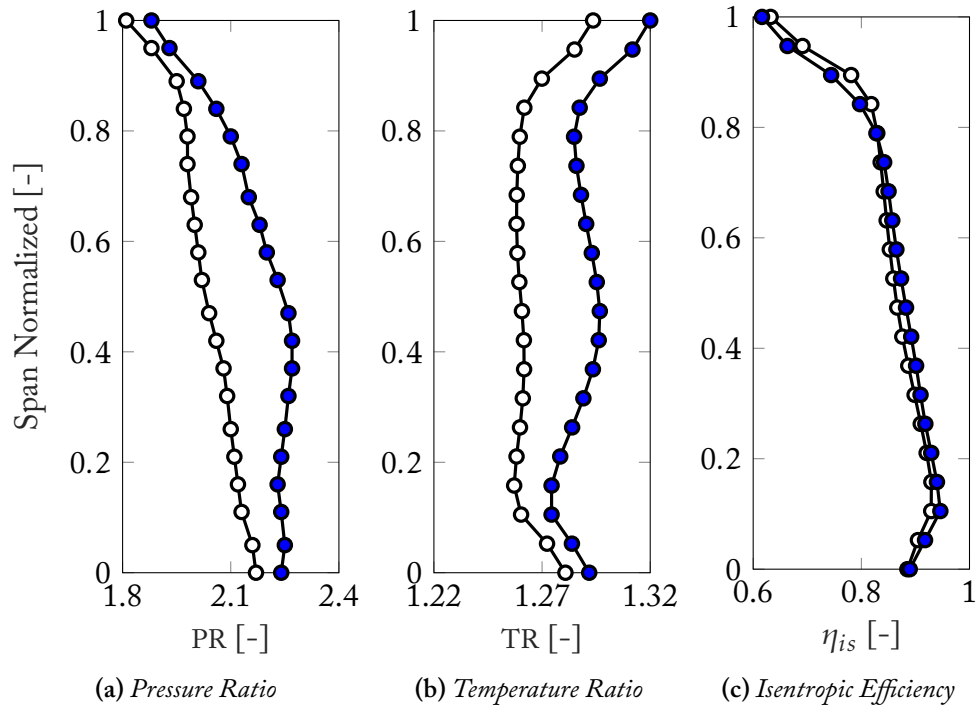
(c) *Isentropic Efficiency*

**Figure 4.4.** *Spanwise plots comparison between baseline and optimized geometry performance at maximum efficiency mass flow rate (CFD results)*

Considering spanwise plots of pressure ratio, temperature ratio and isentropic efficiency, it is possible to observe that:

- The optimized geometry produce a pressure ratio higher than the baseline in every spanwise location, and especially at mid-low spans. The optimized geometry, which is characterized by a greater camber angle, modifies the shock wave pattern and is capable of producing greater outlet total pressure, resulting in higher pressure ratio. This is clearly appreciable from total pressure contours of Figures 4.5 and 4.6;

- Temperature ratio of the optimized blade is higher than the baseline one in every spanwise location, especially for span greater than 10 % of the blade height;

- The isentropic efficiency of the optimized blade is slightly higher for spans < 80 %, whereas lower values are observed for the tip region. In both cases, the shock wave is generated at the leading edge and intersects the next blade at about 60 % of the chord. For high spanwise locations, however, the optimized geometry produces a more intense shock wave whose interaction with the boundary layer on the blade produces a more marked separation of the latter on the suction side surface. For this reason, the isentropic efficiency is lower than the baseline one for high spans. Total pressure contours of Figures 4.5(a) and 4.6(a) show this aspect. Notice, in fact, the larger red wake near to the trailing edge of opt3 at span 90 %, that indicates a more consistent separation of the boundary layer.

As a results, the optimized blade has better overall performance, but its efficiency is lower for high spans. It has been decided, so, to build an *hybrid* geometry that has the same geometrical properties of opt3 at spans 10 % and 50 %, but whose properties at span 90 % are an average between the baseline and optimized configuration (Table 4.2). In doing so, it has been tried to gain some efficiency at high spanwise locations, where the baseline geometry performs better.

**Table 4.2.** *Geometrical properties of hybrid compressor at span* 90 %

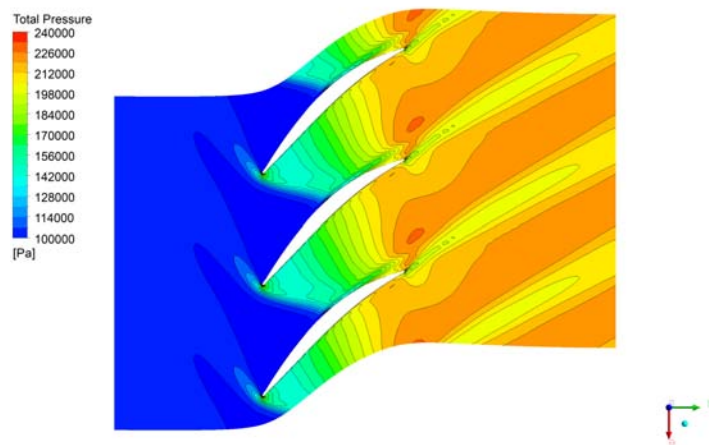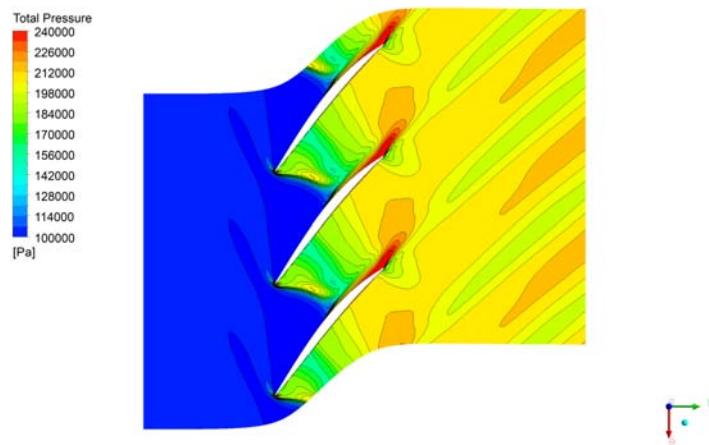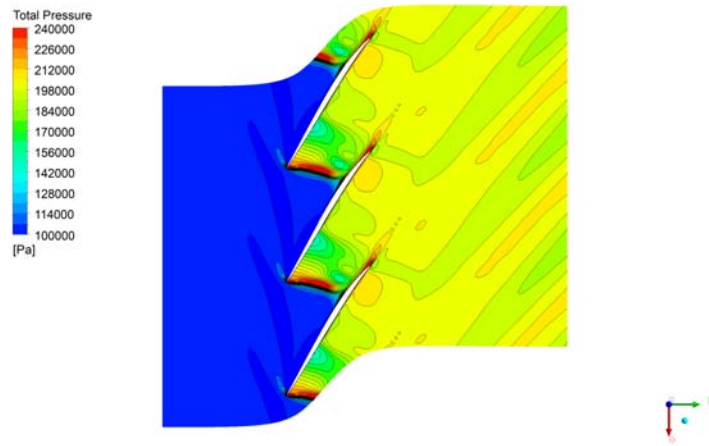| Variable Name | Rotor 37 | opt3 | hybrid |
|---|---|---|---|
| angBladeIn | 60.76° | 61.08° | 60.92° |
| angBladeOut | 48.18° | 47.09° | 47.64° |
| chord | 56.03 mm | 57.13 mm | 56.58 mm |
| radCurvIn | 0.14 mm | 0.14 mm | 0.14 mm |
| thickMaxChordRatio | 0.0355 | 0.0351 | 0.0347 |

(a) *Span* 90 %



(b) *Span* 50 %



(c) *Span* 10 %

**Figure 4.5.** *Total pressure contours of the baseline geometry (Rotor 37)*

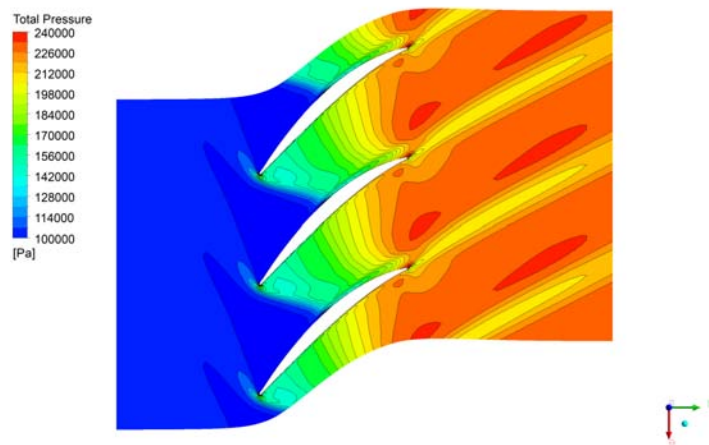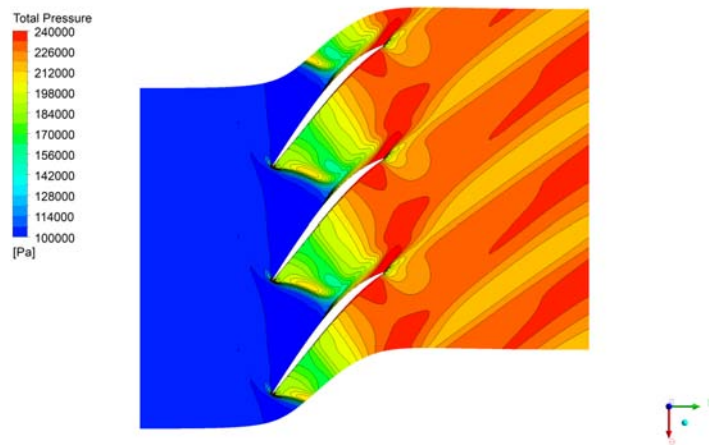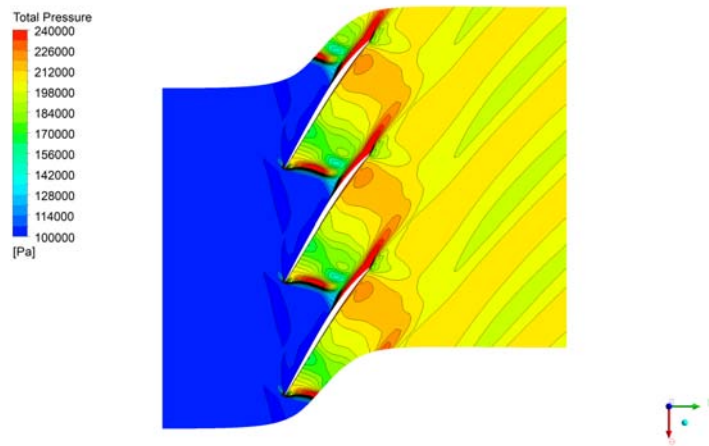(a) *Span* 90 %



(b) *Span* 50 %



(c) *Span* 10 %

**Figure 4.6.** *Total pressure contours of the optimized geometry (opt3)*

## 4.4 Performance of hybrid

The three-dimensional geometry of the hybrid individual has been reconstructed by modifying the airfoil shape of opt3 at span 90 %. The same procedure adopted for opt3 and described in Section 3.4 has been used. The hybrid blade is nearly the same of opt3 for lower spans, while it is modified in its upper part. This has been done in order to increase its isentropic efficiency.

Perfomance maps and spanwise plots of the hybrid individual are shown in Figure 4.7 and 4.8.

Despite similar pressure ratio performance curve (Figure 4.7(a)), a small improvement in the isentropic efficiency was obtained (Figure 4.7(b)), as a consequence of the modification of the airfoil shape at span 90 %. This is related to a decrease in the total temperature ratio for spanwise positions > 0.8, which takes on intermediate values between the baseline individual and the optimized one (Figure 4.8(b)). Regarding the spanwise pressure ratio, it remains almost unchanged (Figure 4.8(a)) and, as a results, the isentropic efficiency at high spanwise locations is similiar to the baseline one (Figure 4.8(c)). The overall results is an increment of 0.41 % on peak isentropic efficiency. Perfomance for spans < 0.5 is unchanged. The operating range is the same of opt3.

Switching the attention to the Relative Mach contour plots of Figure 4.9, it is possible to appreciate that a weaker shock wave is generated at span 90 %, and so its interaction with the boundary layer is less detrimental compared to opt3. The wake downstream of the hybrid airfoil is thinner than the wake of opt3, but is still wider than the baseline one. Moreover, looking at meridional plots of Figure 4.10, it is possible to appreciate that the relative Mach number downstream the blade is higher for the baseline geometry, especially at high spans. Nevertheless, the hybrid configuration shows results which are a compromise between the Rotor 37 and opt3, confirming that it is affected by a weaker shock wave with respect to opt3, so that its efficiency at high spanwise locations is near to the baseline one.

**Table 4.3.** *Comparison between baseline, optimized and hybrid performance at maximum efficiency mass flow rate (CFD results)*

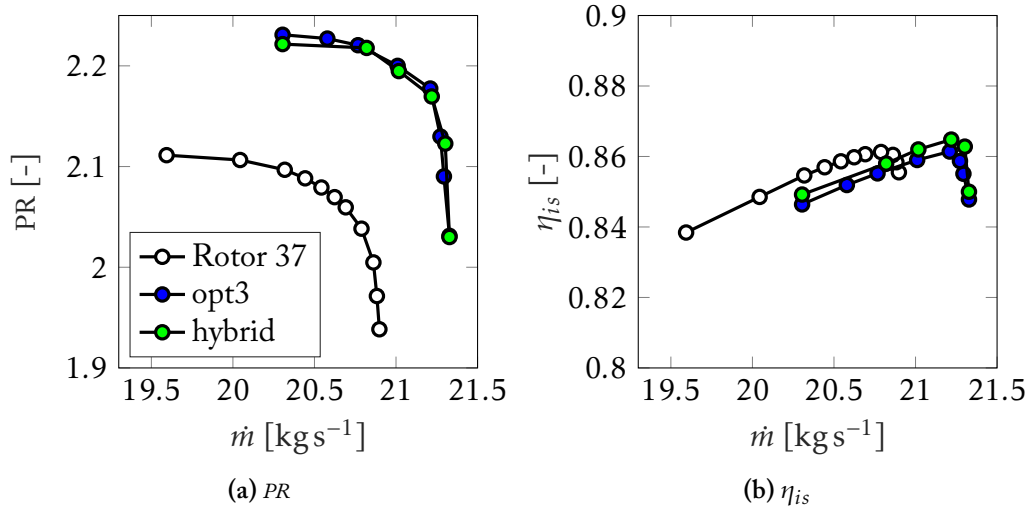|  | PR | $\eta_{is}$ | $\dot{m}_{surge}$ $[\mathrm{kg\,s^{-1}}]$ | $\dot{m}_{choking}$ $[\mathrm{kg\,s^{-1}}]$ |
|---|---|---|---|---|
| Rotor 37 | 2.04 | 0.8613 | 19.59 | 20.98 |
| opt3 | 2.18 | 0.8614 | 20.30 | 21.33 |
| $\Delta_{opt3}$ [%] | 6.83 | 0.01 | – | – |
| hybrid | 2.17 | 0.8648 | 20.30 | 21.33 |
| $\Delta_{hybrid}$ [%] | 6.44 | 0.41 | – | – |

**(a)** *PR*

**(b)** $\eta_{is}$

**Figure 4.7.** *Comparison between baseline, optimized and hybrid nominal speedline (CFD results)*



**(a)** *Pressure Ratio*

**(b)** *Temperature Ratio*

**(c)** *Isentropic Efficiency*
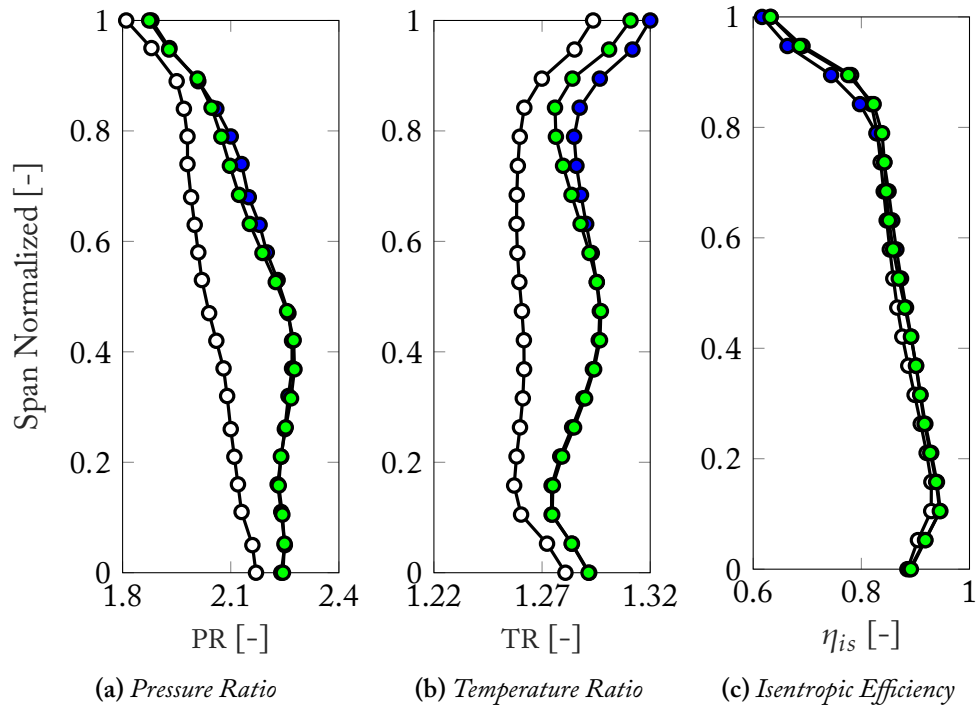
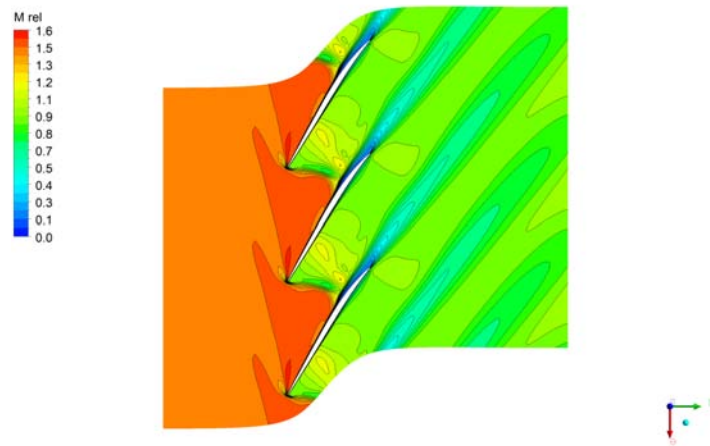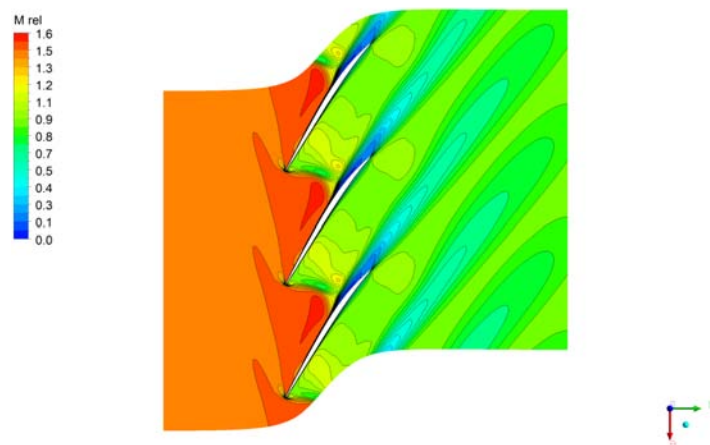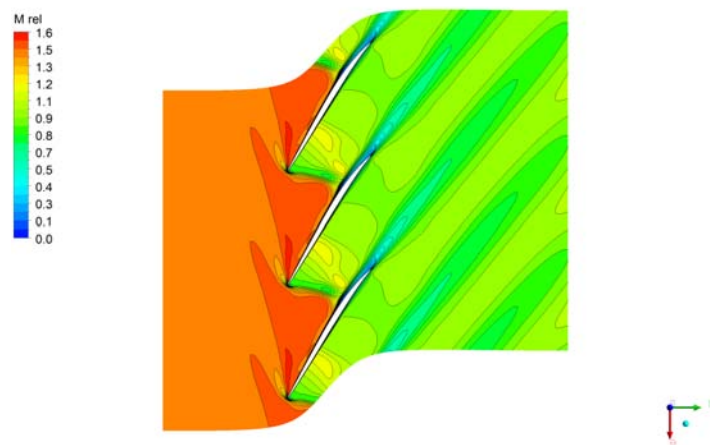**Figure 4.8.** *Spanwise plots comparison between baseline, optimized and hybrid geometry performance at maximum efficiency mass flow (CFD results)*

(a) *Hybrid*



(b) *opt3*
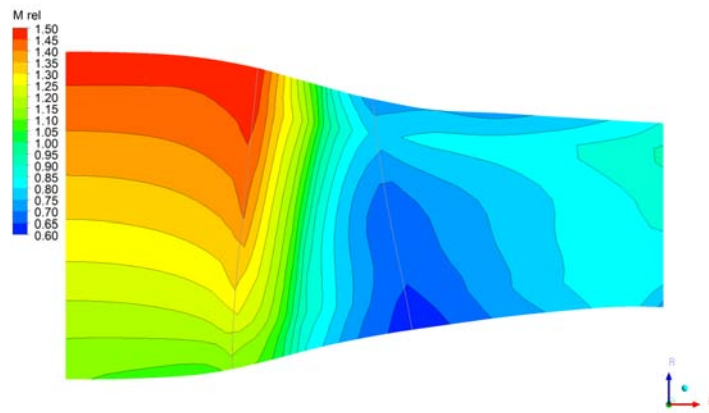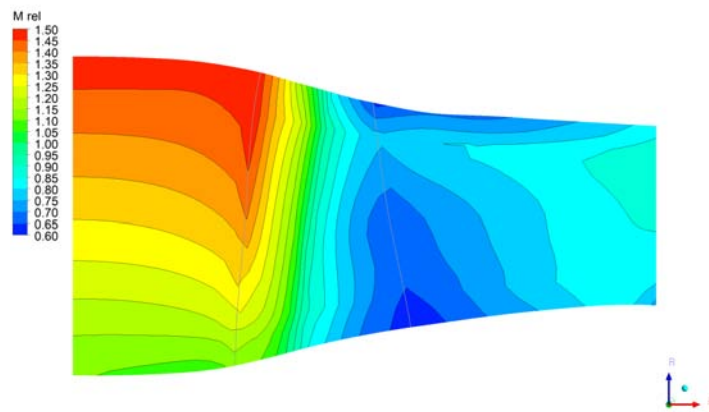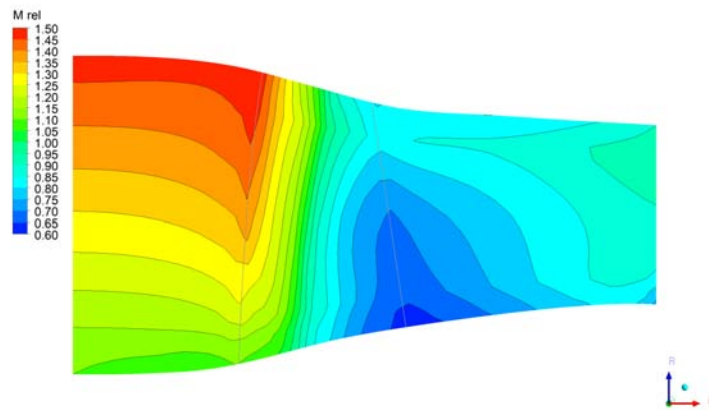


(c) *Rotor 37*

**Figure 4.9.** *Comparison of Relative Mach Number contours at span* 90 %
*(blade–to–blade plots)*

(a) *Hybrid*



(b) *opt3*



(c) *Rotor 37*

**Figure 4.10.** *Comparison of Relative Mach Number contours (meridional plots)*

### 4.4.1 Modal analysis of hybrid

The hybrid configuration is characterized by thinner blades with respect to the Rotor 37. A preliminary modal analysis has been conducted in order to compare the first four vibrational modes of the two cases.

The analysis has been carried out using ANSYS Mechanical. First of all, the single blade geometry has been imported. Then a 350 000 hexahedral elements mesh has been created with the embedded meshing tool by using default parameters, but enabling the *Face Meshing* options, in order to have a structured mesh on pressure side and suction side surfaces. Following this, a material property has been assigned to the blade volume. Since no information was found regarding the materials used for Rotor 37 construction, a titanium alloy was chosen, that is Ti-6Al-4V, as it is widely used in these applications. Finally, the blade has been constrained with a fixed joint at its basis to simulate its connection with the hub. No applied forces have been considered. The first four vibrational modes natural frequencies are reported in Table 4.4.

Notice how the natural frequencies of hybrid are lower than the Rotor 37 ones. As expected, in fact, this is because of its reduced thickness. These aspects could be critical for the problem of flutter, a phenomenon of aeroelastic instability that can lead to the structural failure of the blade. However, for a more detailed and accurate analysis, it is also necessary to consider the centrifugal forces that act on the blade istelf and the fluid-structure interaction, that modify the aeroelastic behaviour of the blade.

**Table 4.4.** *Vibrational modes of the baseline and of the hybrid configuration*

|             | Rotor 37 | hybrid  | $\Delta$ [%] |
| ----------- | -------- | ------- | ------------ |
| Mode 1 [Hz] | 880.90   | 846.23  | −3.94        |
| Mode 2 [Hz] | 2471.90  | 2348.40 | −5.00        |
| Mode 3 [Hz] | 3158.70  | 2990.60 | −5.32        |
| Mode 4 [Hz] | 4721.00  | 4328.60 | −8.31        |

# 5 | Optimization using ANNs

*A system of ANNs has been trained on SATURN outputs to further reduce the time required for the optimization process. In this chapter its performances are shown, in terms of accuracy and calculation speed.*

## 5.1 Introduction on ANNs

Neural networks and deep learning are big topics in Computer Science and engineering. They as of now give the best solutions to many problems, related for example to autonomous driving vehicles, risk managements in business and converting acoustic patterns of a human voice for vocal assistants.

A neural network, more appropriately called *artificial* neural network (ANN), is a computational model inspired by the method through which biological neural networks collect knowledge about the human brain processes. The basic computational unit of the brain is a neuron. Many billions of neurons can be found in in the nervous system of a human, and they're connected through *synapses*. The basic unit of computation in a neural network is the neuron, often called a node or unit. It receives input from some other nodes and computes an output. Each input has an associated weight ($w$), which is assigned on the basis of its relative importance to other inputs. The node applies a function to the weighted sum of its inputs. In the biological model, the dendrites carry the signal to the cell body where they all get summed. If the final sum is above a certain threshold, the neuron can send a spike along its nerve fiber. In the computational model, the precise timings of the spikes do not matter, and only the frequency of the firing communicates information, which is modeled with an activation function (Menshawy [33]), through which an output depending on input data is then processed. According to these facts, the ANN can be used as a transfer function that simulates the physical behavior of the system of interest. Starting from geometrical data, for example, a suitably trained neural network can replace a CFD simulation, and so the advantages related to the use of a network are obvious.

Consider the schematzation of the artificial neuron proposed in Figure 5.1. It is composed by:

- Input nodes: they represent input data. These nodes pass the information to the next node without doing any calculation. They may also include a *bias*. A block of nodes is also called *layer*;

- Hidden node: it performs computations and then transfer its weighted output to the following node. Generally, most of the neural networks are composed by 2 hidden layers (Heaton [34]). All the computation is done in these layers;

- Connection and weights: the network consists of connections. Each connection transfers the output of a neuron $i$ to the input of a neuron $j$. In this sense, $i$ is the predecessor of $j$ and $j$ is the successor of $i$. Each connection is assigned a weight $w_{ij}$.

- Output node: it receives the weighted inputs from the previous layer. Then, outputs are calculated according to the activation function adopted.

- Activation function: it defines if a given node should be activated or not based on the weighted sum of its inputs, which may depends on the bias value, and on its threshold. Many activation function exist, as for example the *Sigmoid Function*;

- Learning rule: it is an algorithm which modifies the parameters of the neural network, in order for a given input to the network to produce a desired output. This learning process, also called *training* of the network, typically amounts to modifying the weights and thresholds in order to minimize the difference between target outputs and calculated outputs. A widely used learning algorithm is called *Back Propagation*.
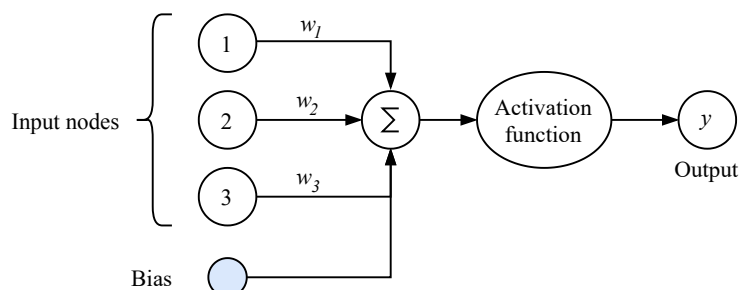


**Figure 5.1.** *Artifical neuron*

## 5.2 Training

In this work, four ANNs have been trained to predict the outputs of SATURN. This was done to further reduce the calculation time of the optimization through genetic algorithm. If 7 seconds were required for the prediction of performance of each individual using the meanline code, by using the neural network the same process would take a fraction of a second, so that the whole optimization would be completed in few seconds. In order to be used for this purpose, however, the neural networks must be properly trained. Only in this way, in fact, their predictions are accurate and can replace the use of SATURN.

A large database is necessary for training: starting from the inputs, the learning algorithm adjusts the weights in order to minimize the gap between the target outputs (provided by the database) and the outputs calculated by the network. A database of 1500 individuals has been created for this purpose. LHS has been used as sampling method for individuals, whose geometrical variables vary by ±7.50 % centered on the baseline value. The creation of the database, including simulations of individuals with SATURN for the creation of target outputs, required 3 hours.

Each ANN has been created using the *Neural Fitting Toolbox* of MATLAB, which provides a graphical interface that allows the user to train a neural network in few simple steps. Firstly, the topology of the neural network has been defined. The input layer is composed by 15 nodes, each one corresponding to a geometrical variable (see Table 2.1 on page 9). Moreover, it was chosen to use a single hidden layer consisting of 10 nodes, as recommended by the toolbox parameters. The output layer of each network, composed by a single node, corresponds to one of the objectives or constraints of the optimization process. By doing this, one ANN has been trained to predict pressure ratio, another one to predict isentropic efficiency, while the remaining two are used to predict surge margin and choking margin. The *Bayesian-regularization* algorithm has been chosen for the learning process, since it is suitable for fitting complex functions. The database has been divided into 2 parts: 80 % of it was used to train the ANN and the remaining 20 % was used for testing the prediction of the ANN.

The performance of the neural networks was evaluated in terms of correlation coefficient, $R$, between target outputs and calculated outputs. The more the coefficient approached unity, the more accurate the net was considered. Performance is reported in Table 5.1. Notice how three of the four networks provided excellent performance (ANN 1,2 and 3). On the other hand, however, the neural network trained to predict the choking margin is not so performing. The reason is attributable to the fact that the choking detection performed by SATURN is based on the solver divergence rather than on a correlation. Therefore, despite

**Table 5.1.** *Performance of the ANNs*

| ANN ID | ANN output | $R_{training}$ [%] | $R_{test}$ [%] |
|--------|------------|--------------------|----------------|
| 1 | Pressure ratio | 99.38 | 98.48 |
| 2 | Isentropic efficiency | 99.45 | 98.16 |
| 3 | Surge margin | 98.78 | 96.41 |
| 4 | Choking margin | 87.88 | 79.49 |

the correct choking prediction of the meanline code itself, the definition of a transfer function based on the solver divergence for several test cases may not be straightforward, due to the numerical instability arising close to the choking mass flow (see Table 2.2 on page 11). As a consequence, the performance of ANN 4 are slightly worse than others, but a reasonable trend is found anyway.

The plots from which the correlation index $R_{training}$ of ANN 1 and of ANN 2 has been calculated are reported in Figure 5.2. The straight line representing the linear interpolation of the data is called *fit*, while the line which corresponds to a correlation factor $R = 1$ is defined by the equation $y = t$. Note how, in Figure 5.2(b), the two lines are clearly misaligned, while in Figure 5.2(a) they're are superimposed. Not only that, but also note how the points of Figure 5.2(b) form a larger cloud, indicating a lower precision of the network in question.



(a) *ANN 1*                    (b) *ANN 4*
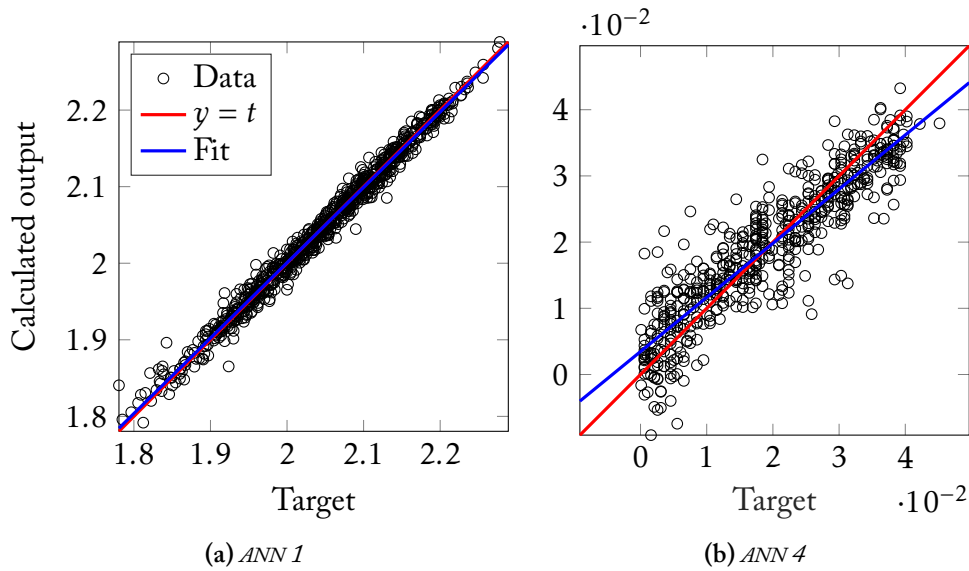
**Figure 5.2.** *Visualization of coefficient $R_{training}$ related to ANN 1 and ANN 4*

## 5.3   Optimization and Results

Given the accuracy of the neural networks, it has been decided to conduct an optimization on the Rotor 37 using genetic algorithm. The algorithm settings, the optimization variables, the objectives and the constraints are set in the same way as for opt3, in order to be able to conduct a comparison between the two optimizations. The only difference is in how fitness and constraints are calculated: if the meanline code was used in opt3 to calculate these values, in this optimization, optNN, ANNs are used. Total calculation time required by optNN is 1.20 seconds.

Optimization optNN was repeated 100 times, so as to eliminate the random component from the optimization results. The results, in the space of fitnesses, are shown in the Figure 5.3: optNN always finds an optimized individual, as expected, whose fitness values cluster in an opt3 neighborhood, but in most cases has better characteristics. Average performance of optNN individuals are shown in the Table 5.2. Notice how the relative error on objectives is low, while the one on constraints is higher, as they're value can vary by $\pm 10\%$ on baseline value.

Finally, an individual has been simulated using SATURN, to further verify the accuracy of the neural network. Input variables of this individual, indicated in Figure 5.3 as optNNx, are reported in Table 5.3 where it is possible to appreciate the comparison with opt3 variables. A fine matching is found. Performance maps are shown in Figure 5.4. It can be appreciated that performance maps of optNNx and opt3 are almost identical. This is because the ANNs are not as accurate as the meanline. To improve the accuracy of neural networks, however, the database can be integrated with experimental results and outputs of CFD simulations.

This optimization trial has proven to be successful: firstly because it has extremely reduced the computational times required by the optimization process, secondly because it has shown that it can replace the meanline code safely and accurately. Even if the definition of the acpANN can be further improved, the methodology presented here showed satisfactory results.

**Table 5.2.** *Average performance of optNN individuals at design point*

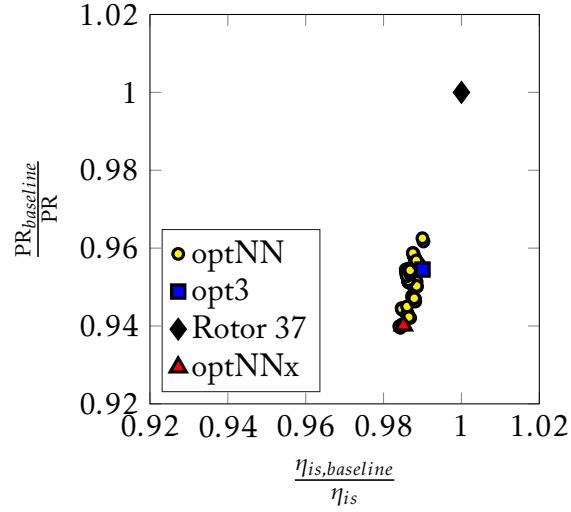|  | Rotor 37 | opt3 | optNN |
|---|---|---|---|
| PR | 2.11 | 2.21 | $2.22 \pm 0.32\%$ |
| $\eta_{is}$ | 0.8770 | 0.8885 | $0.8879 \pm 1.29\%$ |
| SM | 0.038 | 0.035 | $0.04 \pm 5.24\%$ |
| CM | 0.074 | 0.079 | $0.07 \pm 4.26\%$ |

**Figure 5.3.** *Results of optNN*

**Table 5.3.** *Comparison between geometrical properties of optNNx and opt3*

| Variable Name | Span [%] | opt3 | optNNx | Δ [%] |
|---|---|---|---|---|
| angBladeIn | 10 | 53.83° | 53.72° | −0.20 |
| | 50 | 58.14° | 57.85° | −0.50 |
| | 90 | 61.08° | 61.77° | 1.13 |
| angBladeOut | 10 | 21.83° | 21.77° | −0.27 |
| | 50 | 36.10° | 35.98° | −0.33 |
| | 90 | 47.09° | 46.29° | −1.70 |
| chord | 10 | 58.03 mm | 56.48 mm | −2.66 |
| | 50 | 53.97 mm | 55.49 mm | 2.83 |
| | 90 | 57.13 mm | 55.10 mm | −3.55 |
| radCurvIn | 10 | 0.25 mm | 0.24 mm | −0.41 |
| | 50 | 0.19 mm | 0.19 mm | −0.52 |
| | 90 | 0.14 mm | 0.14 mm | −0.70 |
| thickMaxChordRatio | 10 | 0.0733 | 0.0745 | 1.65 |
| | 50 | 0.0517 | 0.0545 | 5.39 |
| | 90 | 0.0347 | 0.0343 | −1.16 |

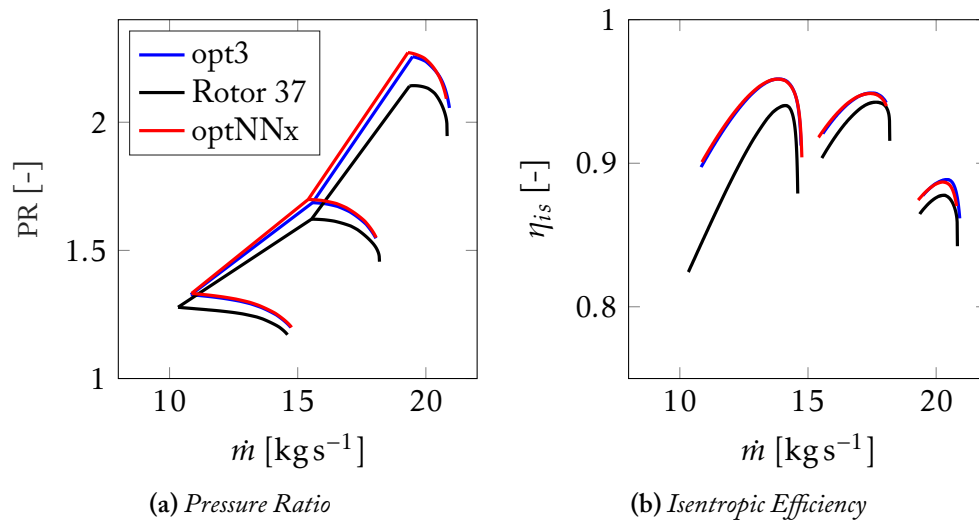(a) *Pressure Ratio*     (b) *Isentropic Efficiency*

**Figure 5.4.** *Performance maps of the Rotor 37, opt3 and optNNx predicted by SATURN*

# 6 | Conclusions

An optimization procedure that couples a one-dimensional meanline solver (SATURN) with a genetic algorithm has been proposed in this work. The aim is to lower the required computational time with respect to a more common optimization methodology, which combines CFD with evolutionary algorithms. If the latter generally takes a few days to deliver a result, the proposed procedure takes a few hours. The transonic axial-flow compressor rotor Rotor 37, developed by NASA, has been chosen as test case.

Various optimization trials have been set-up and a good trade-off between computational time and increase in performance has been individuated. A remarkable increase of +4.75% on pressure ratio at design point has been predicted, against a slightly improved isentropic efficiency (+1.31%) and an almost unchanged operating range (−2.88%). The procedure required 12 hours. Starting from optimized geometric properties, the one-dimensional geometry provided by SATURN has been parametrized to obtain a three-dimensional geometry, which has been simulated with a CFD anslysis by means of ANSYS CFX.

The comparison between the prediction of the meanline code and CFD results of the optimized geometry has shown that operating ranges are significantly different. This is attributable to the setting of a parameter used to calibrate the blockage factor, whose effect on the performance maps is a translation along the $x$-axis of the curves. On the other hand, predictions on pressure ratio and isentropic efficiency are quite accurate. The analysis showed that the optimized individual has a pressure ratio equal to 2.18 (+6.83%) and an isentropic efficiency equal to 0.8614 (+0.01%) at maximum efficiency mass flow rate. However, the optimized configuration has a lower isentropic efficiency than the baseline for high spanwise locations, which is why a hybrid geometry has been identified. Its airfoils, for span > 0.8, are a trade-off between the baseline configuration and the optimized configuration. The improvement in terms of isentropic efficiency flow with respect to the baseline is +0.41% calculated at at maximum efficiency mass flow rate.

To conclude, a system of ANNs has been trained on the outputs of SATURN, in order to replace the role of the meanline solver inside the optimization loop to

further reduce the calculation time. Indeed, the time required by this optimization process is equal to 1.2 seconds, and its outputs are in good agreement with the prediction of the meanline code.

Future works may involve the verification of the prediction of the system of ANNs with CFD results, as well as the search of a methodology to match the operating range predicted by SATURN with the one calculated with CFD simulations, which may be obtained by employing an ANN trained to predict the boundary layers height.

# Bibliography

[1] M. Otelbaev, "The existence of a strong solution to the Navier-Stokes equations," *Matematiceskij Zurnal*, vol. 2013, 01 2014. [Online]. Available: https://www.researchgate.net/publication/259705678_The_existence_of_a_strong_solution_to_the_Navier-Stokes_equations

[2] F. Picano, "Lecture notes in Aerodynamics 2," PDF file, 2019.

[3] F. Arute, K. Arya, R. Babbush, D. Bacon, J. Bardin, and collaborators, "Quantum Supremacy using a Programmable Superconducting Processor," *Nature*, vol. 574, p. 505–510, 2019. [Online]. Available: https://www.nature.com/articles/s41586-019-1666-5

[4] B. Cuenot, "Introduction à la modélisation de la combustion turbulente." Toulouse, France: CERFACS-CFD, 2005.

[5] A. Samad and K.-Y. Kim, "Shape Optimization of an Axial Compressor Blade by Multiobjective Genetic Algorithm," vol. 222. Institution of Mechanical Engineers - Part A, 09 2008, pp. 599–611.

[6] G. Abate, "Aerodynamic Optimization of a Transonic Axial Compressor Rotor," Master's thesis, University of Padua, 2012.

[7] M. Asghari, M. Mirzabozorg, and M. Adami, "Aerodynamic optimization of the tangential stacking line of a transonic axial flow compressor rotor using genetic algorithm," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 41, 01 2019.

[8] M. Ghalandari, A. Ziamolki, A. Mosavi, S. Shamshirband, K.-W. Chau, and S. Bornassi, "Aeromechanical optimization of first row compressor test stand blades using a hybrid machine learning model of genetic algorithm, artificial neural networks and design of experiments," *Engineering Applications of Computational Fluid Mechanics*, vol. 13, no. 1, pp. 892–904, 2019.

[9] M. Joly, S. Sarkar, and D. Mehta, "Machine Learning Enabled Adaptive Optimization of a Transonic Compressor Rotor With Precompression," *Journal of Turbomachinery*, vol. 141, no. 5, 01 2019, 051011.

[10] Y. Duan, W. Wu, P. Zhang, F. Tong, Z. Fan, G. Zhou, and J. Luo, "Performance improvement of optimization solutions by POD-based data mining," *Chinese Journal of Aeronautics*, vol. 32, no. 4, pp. 826 – 838, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1000936119300627

[11] S. Mondal, M. Joly, and S. Sarkar, "Multi-Fidelity Global-Local Optimization of a Transonic Compressor Rotor," 06 2019.

[12] L. Reid and R. D. Moore, "Design and Overall Performance of Four Highly-Loaded, High Speed Inlet Stages for an Advanced High-Pressure Ratio Core Compressor," 11 1978.

[13] R. D. Moore and L. Reid, "Performance of Single-Stage Axial Flow Transonic Compressor With Rotor and Stator Aspect Ratios of 1.19 and 1.26, Respectively, and With Design Pressure Ratio of 2.05," NASA, Tech. Rep., 1980.

[14] K. L. Suder, "Experimental Investigation of the Flow Field in a Transonic, Axial Flow Compressor with Respect to the Development of Blockage and Loss," Ph.D. dissertation, Case Western Reserve University, 1996.

[15] A. Ameri, "NASA ROTOR 37 CFD CODE validation: Glenn-HT code," 01 2009.

[16] R. Biollo, "Systematic Investigation on Swept and Leaned Transonic Compressor Rotor Blades," Ph.D. dissertation, University of Padua, 2008.

[17] L. H. Smith, "Casing Boundary Layers in Multistage Axial-Flow Compressors," *Flow Research on Blading*, pp. 275–304, 1970. [Online]. Available: https://ci.nii.ac.jp/naid/10027844823/en/

[18] M. Çetin, A. S. Üçer, C. Hirsch, and G. K. Serovy, "Application of Modified Loss and Deviation Correlations to Transonic Axial Compressors," AGARD, Tech. Rep., 1987. [Online]. Available: https://books.google.it/books?id=PT9UGQAACAAJ

[19] A. D. S. Carter and H. P. Huges, "A Theoretical Investigation into the Effect of Profile Shape on the Performance of Airfoils in Cascade," Aeronautical Research Council, London, Tech. Rep. 2384, 03 1950.

[20] R. H. Carmody and H. Creveling, "Computer program for calculating off-design aerodynamic performance of axial flow compressor," NASA, Tech. Rep., 08 1968. [Online]. Available: http://hdl.handle.net/2060/19690001542

[21] S. Lieblein, "Experimental Flow in Two-Dimensional Cascades," 1965.

[22] C. C. Koch and L. H. Smith, "Loss Sources and Magnitudes in Axial-Flow Compressors," *Journal of Engineering for Power*, vol. 98, no. 3, pp. 411–424, 07 1976.

[23] G. R. Miller, J. G.W. Lewis, and M. J. Hartmann, "Shock Losses in Transonic Compressor Blade Rows," *Journal of Engineering for Power*, pp. 235–241, 07 1961.

[24] J. Tournier and M. S. El-Genk, "Axial Flow, Multistage Turbine and Compressor Models," *Energy Conversion and Management*, pp. 16–29, 01 2010.

[25] M. I. Yaras and S. A. Sjolander, "Prediction of Tip-Leakage Losses in Axial Turbines," *Journal of Turbomachinery*, vol. 114, no. 1, pp. 204–210, 01 1992.

[26] N. White, A. Tourlidakis, and R. Elder, "Axial compressor performance modelling with a quasi-one-dimensional approach," vol. 216. Institution of Mechanical Engineers - Part A, 03 2002, pp. 181–193.

[27] E. Benini and A. Toffolo, "Development of High-Performance Airfoils for Axial Flow Compressors Using Evolutionary Computation," *Journal of Propulsion and Power*, vol. 18, no. 3, 06 2002.

[28] S. Shahpar, D. Sasaki, and A. Keane, "Multi-objective Evolutionary Optimization of a Compressor Stage using a Grid-Enabled Environment," 01 2006.

[29] S. N. Skinner and H. Zare-Behtash, "State-of-the-art in aerodynamic shape optimisation methods," *Applied Soft Computing*, vol. 62, pp. 933–962, 01 2018.

[30] A. Konak, D. Coit, and A. Smith, "Multi-objective Optimization using Genetic Algorithms: a Tutorial," *Reliability Engineering and System Safety*, vol. 91, pp. 992–1007, 09 2006.

[31] F. Menter, M. Kuntz, and R. Langtry, "Ten years of industrial experience with the SST turbulence model," *Heat and Mass Transfer*, vol. 4, 01 2003.

[32] M. Porro, "Axial Flow Compressor Stability Enhancement: Code Validation and Circumferential Casing Grooves Investigation," Master's thesis, University of Nottingham, Notthingam, UK, 04 2019.

[33] A. Menshawy, *Deep Learning By Example*. Packt Publishing, 02 2018.

[34] J. Heaton, *Artificial Intelligence for Humans*. Createspace Independent Publishing Platform, 10 2015, vol. 3.

# List of acronyms