

CHALMERS



CFD investigation of a 3.5 stage transonic axial compressor including real geometry effects

Master's Thesis in Solid and Fluid Mechanics

ANTON PERSSON
JOSEF RUNSTEN

Department of Applied Mechanics
Division of Fluid Mechanics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2011
Master's Thesis 2011:35

CFD investigation of a 3.5 stage transonic axial compressor
including real geometry effects

Master's Thesis in Solid and Fluid Mechanics

ANTON PERSSON

JOSEF RUNSTEN

Department of Applied Mechanics

Division of Fluid Mechanics

CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2011

CFD investigation of a 3.5 stage transonic axial compressor including real geometry effects
ANTON PERSSON
JOSEF RUNSTEN

©ANTON PERSSON, JOSEF RUNSTEN, 2011

Master's Thesis 2011:35
ISSN 1652-8557
Department of Applied Mechanics
Division of Fluid Mechanics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Cover:
Rotor blades of the 3.5 stage axial compressor Blenda.

Chalmers Reproservice
Göteborg, Sweden 2011

Abstract

When designing a transonic multi-stage compressor it is desirable to keep the development cost low and development time short while finding the optimal geometry for a given function. In order to keep the cost down CFD analysis can be used together with an optimization tool instead of expensive experiments. In the CFD optimization process, a relatively coarse mesh based on a simplified geometric definition is normally used in order to screen the design space in a reasonably short time. Once a preliminary design is set, CFD analysis using a higher fidelity model is desired to verify that the design fulfill the performance requirements. A higher fidelity model normally includes features such as fillets at the blade root section and the clearance height between the rotor tip and the casing. The computational domain should be highly resolved especially in the near wall regions.

In this project the geometry of a cold rotor blade is transformed into the shape of a blade in operating condition, by applying displacements due to centrifugal and thermal loadings and surface deviations from the manufacturing process. This is done by applying displacements of the geometry from FE-analyses which account for rotational and thermal effects together with measured surface nonconformance data originated from a manufactured geometry. An interpolation method utilizing radial basis functions is developed. With the available data in this project, the developed routine performs well in the cold to hot transformation of the rotor blade. However, the results are less reliable for the nonconformance case.

During the European 6th framework VITAL research programme a 3.5 stage transonic axial compressor was developed and tested. CFD simulations that have been carried out has shown great disagreement in performance over the second rotor, compared to the experimental data. The second rotor blade of this compressor is in this thesis analysed with real geometry effects using Ansys CFX. The lack of fillets and tip clearance in the design geometry clearly affects the passage area and thus the mass flow capability. The pressure rise is lower for the hot blade, giving a lower polytropic efficiency. A further reduction of passage area due to nonconformances yield a further decrease in mass flow.

Keywords: radial basis function interpolation, geometry transformation, Volvo Aero Corp, transonic axial compressor, Blenda

Aknowledgements

This work was conducted at the department of Aerothermodynamic at Volvo Aero Corporation. We would like to thank our tutor Lars Ellbrant for his commitment and continuous, almost constant, support during this work. The outcome of this thesis would not have been the same without his help. At Volvo Aero Corporation we would also like to thank for the great and important knowledge that Hans Mårtensson and Lars Ljungkrona has contributed with. A thank you to our examiner associate professor Tomas Grönstedt for leading us into the world of turbomachinery. We would also like to thank Karsten Østergaard Noe at the Alexandra Institute in Åhus, Denmark for his guidance in the jungle of RBF interpolation.

Göteborg May 2011
Anton Persson, Josef Runsten

Notations

General symbols

a	Interpolation coefficients in the points of $\mathbf{S2}(R)$
A	Area
\mathbf{A}	Vector containing a 's and c 's
b	Shape factor
c	Coefficients of the linear polynom in the space of $\mathbf{S2}(R)$
C_p	Specific heat at constant pressure
C_w	Whirl velocity
\mathbf{F}	Displacement vector
\mathbf{G}	System matrix that is inverted to solve the displacements of $\mathbf{s1}$
L	Curve length
\dot{m}	Mass flow
P_0	Total pressure
r	Radius
R	Radius
s	Pitch
$\mathbf{s1}$	Arbitrary point of $\mathbf{S1}$
$\mathbf{S1}$	Points of original geometry that is to be modified
$\mathbf{s2}$	Arbitrary point of $\mathbf{S2}(R)$
$\mathbf{S2}$	Points of original geometry that has known displacements
t	Blade thickness
T_0	Total temperature
u	Displacement
w	Weight
\mathbf{U}	Displacement of $\mathbf{S2}$
U	Blade speed
x	Distance
β	Blade angle
γ	Specific heat ratio
Δ	Spatial distance
η	Efficiency
θ	Angle of rotation
Π	Pressure ratio
φ	Radial basis function
ξ	Stagger angle

Acronyms and abbreviations

CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
FEA	Finite Element Analysis
FEM	Finite Element Method
GA	Gaussian
IMQ	Inverse Multiquadric
IQ	Inverse Quadratic
Low Re	Low Reynolds
RBF	Radial Basis Function
SST	Shear Stress Transport
VITAL	EnVironmenTALly Friendly Aero Engine
WF	Wall Function

Sub- and superscripts

corr	Corrected
exp	Experiment
norm	Normalized
p	Polytropic
ps	Pressure side
ref	Reference
ss	Suction side
TT	Total-Total

Contents

Abstract	I
Preface	III
Notations	IV
Contents	VII
1 Background	1
2 Introduction	2
2.1 Purpose of thesis	3
2.2 Objectives	3
3 Theory	4
3.1 Calculating element unit normals	4
3.2 Area change due to real geometry effects	4
3.3 Rotating the FE point cloud	5
3.4 Coordinate transformation of nonconformance data points	5
3.5 Radial basis functions	6
3.6 RBF interpolation	7
3.7 Stability vs accuracy	10
3.8 Simple RBF interpolation	11
3.9 Stall margin	11
4 Method	13
4.1 Preparing the cold CAD geometry	13
4.2 FE data	13
4.3 Manufacturing nonconformances	13
4.4 Simple RBF interpolation	16
4.5 RBF interpolation cold to hot	17
4.6 RBF interpolation nonconformances	18
4.7 RBF interpolation strategy	19
4.8 Computational domain	20
4.9 Numerical simulation	21
4.9.1 Mesh configurations	22
4.9.2 Simulation settings	23
5 Results and discussion	24
5.1 Interpolation method	24
5.1.1 Hot to cold interpolation	24
5.1.2 Nonconformance interpolation	26
5.2 Difference between Design and Hot geometry	29
5.3 Passage area comparison	32
5.4 Results from CFD simulations	32
5.4.1 Overall compressor characteristics.	32
5.4.2 Stall margins	34
5.4.3 Profiles from rake positions	34
6 Conclusions and recommendations	36

1 Background

When designing a transonic multi-stage compressor it is desirable to keep the development cost low and development time short while finding the optimal geometry for a given function. In order to keep the cost down CFD analysis can be used together with an optimization tool instead of expensive experiments. Typically, the optimization process starts with basic mean line tools and loss correlation estimates to define number of stages and an initial estimate on the annulus shape. A number of constraints may be present at the start of the work, such as geometry interfaces, rotational speed etc. The work then carries over to through flow methods which help to estimate radial variation of blade angles and refine annulus shapes. It also assists in screening optimistic designs for which the off-design operation has not been properly considered. These considerations may also take place already in the mean line design phase depending on how the design tools have been programmed. A considerable effort then relates to defining a complete 3D geometry from the results coming from through flow methods. Further optimization directly applied to the CFD modeling is then used. In the CFD optimization process, a relatively coarse mesh based on a simplified geometric definition is normally used in order to screen the design space in a reasonably short time. Once a preliminary design is set, CFD analysis using a higher fidelity model is desired to verify that the design fulfill the performance requirements. A higher fidelity model normally includes features such as fillets at the blade root section and the clearance height between the rotor tip and the casing. The computational domain should be highly resolved especially in the near wall regions.

During the European 6th framework VITAL research programme the high speed transonic 3.5 stage compressor named Blenda was developed and tested. The experimental campaign was completed in end of the year 2009 and the resulting database recently became available for validation. Preliminary results have shown that the efficiency through the second stage is greatly over estimated in the CFD analysis. However, these analyses are based on the ideal geometric definition meaning that the model do not account for geometry changes that can occur when going from the aero definition to the CAD drawing, from the drawing to the manufactured geometry (nonconformances) and eventually from the geometry changes when the rotors go from a static condition (cold) to the design rotational speed (hot). In the case of Blenda, the largest change in geometric shape occurs in the manufacture process of the second rotor. The resulting rotor is thicker than intended, a feature that would most likely introduce extra losses to the system. Thus in order to make a validation of this character more accurate there is a need to have a verification process that builds a model based on the CAD drawing, measurements from the actual manufactured geometry and the information of the changes between the cold and hot state.

2 Introduction

At Volvo Aero in Trollhättan, CFD simulations have been carried out to validate experimental data on the Blenda compressor. Validated results make simulations of similar cases more reliable and can thus in a larger extent be used in the design process. The experimental and simulation results correlate well through the 1st and 3rd stage. However, significant differences, especially in total temperature and total pressure in the hub section, can be seen in the 2nd stage. The reason for this is unclear. One possible explanation is difference in blade geometry, caused by simplifications and approximations when going from design to the manufactured geometry, as well as manufacturing deviations. In this project the geometry of a cold rotor blade is transformed into the shape of a blade in operating condition, by applying displacements due to centrifugal and thermal loadings and surface deviations from the manufacturing process. Deformation due to loading are available as FEA (Finite Element Analysis) results, consisting of spatial surface coordinates with pertaining displacements. In figure 2.1 the CAD geometry, hereafter also called the Cold geometry is shown with all its subparts. It is this geometry that is deformed to the geometries that will be denoted Hot (blade under operating conditions) and Hot Nonconf (blade under operating conditions with manufacturing deviations).

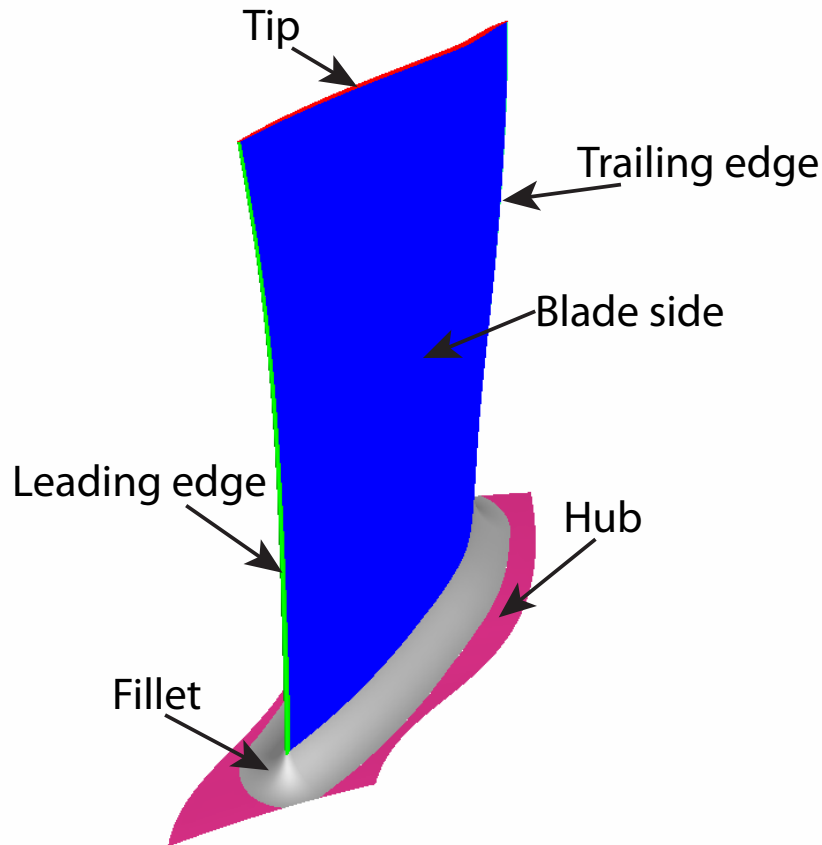


Figure 2.1: Geometry of the second rotor of the Blenda compressors.

The work done to create the geometries is conducted according to the flow chart shown in figure 2.2. Step 1 corresponds to the Cold geometry and is supplied as a CAD geometry (parasolid). In 5a FEA is performed on this geometry and in 5b nonconformance data is mapped onto the geometry. Step 6 consist of two so called radial basis function interpolations that takes the data from 5a and 5b, applies it to 1 and outputs the geometries of

7a and 7b. The major part of this work is focused on the development of the interpolation method. The remaining part of this project is the CFD analyses that is done on the second rotor of the compressor using the geometries obtained in 7a and 7b.

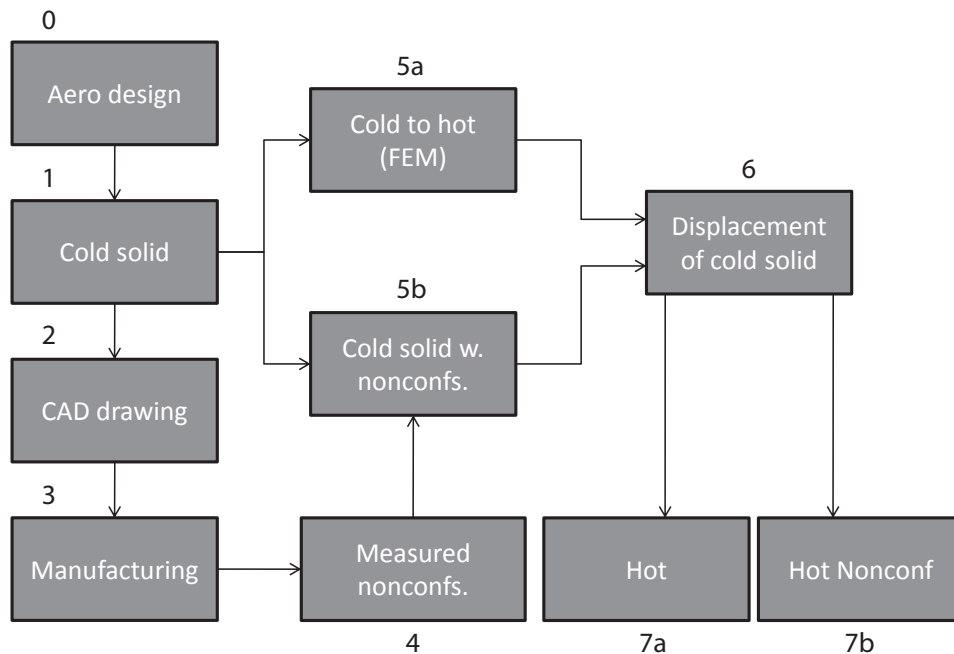


Figure 2.2: Flow chart of the geometry generation process.

2.1 Purpose of thesis

In the design optimization process CFD simulations are carried out using simplified blade and vane geometries, without hub fillets, blade-tip clearances or manufacturing nonconformances. The main focus lies on creating a method for describing a real geometry, but the work also involves analyzing how different changes of the 2nd rotor blade geometry affects the outcome of the CFD calculations. If the effect is significant, this has to be investigated further and the differences quantified.

2.2 Objectives

The objectives of this project are to fill in the gaps in the verification process:

- Develop a method that deforms the cold state geometric definition to the hot state
- Develop a method that deforms the geometry according to the measured nonconformances
- Furthermore, the developed methods are tested on the Blenda compressor where the objectives are:
 - Create the second rotor of Blenda without nonconformances (Hot)
 - Create the second rotor of Blenda with nonconformances (Hot Nonconf)
 - Investigate the performance difference between the two models and the design model (Design) using CFD.

3 Theory

In this section the underlying theory of deforming a blade geometry will be explained in detail.

3.1 Calculating element unit normals

The nonconformance displacements are measured normal to the blade surface. When applying these displacements to a CAD surface it is therefore necessary to calculate the surface normals in each point. This is done by searching for neighbouring coordinates in the geometry connectivity list and taking the cross product of the two vectors forming the triangular face. For nodes being part of several faces, the normal vector of the element with center closest to the node in question is used, see figure 3.1. Since it is closest to the node it will be the best representation of the normal vector. Since the nonconformance data only includes displacements in the x- and z-directions, the normal y-component is set to zero. The normal is then normalized to achieve unit length.

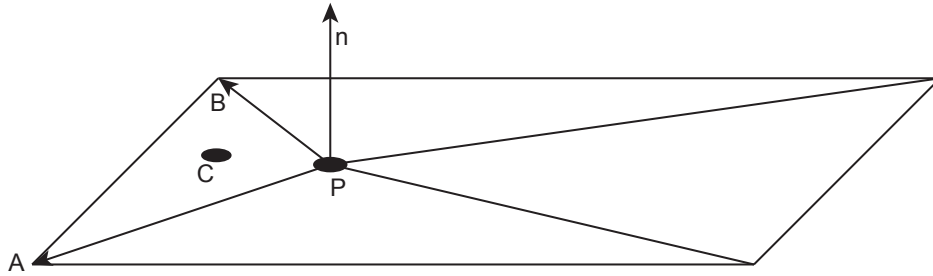


Figure 3.1: The normal in point P is taken as $A \times B$ of the face whose center, C, is closest to P.

3.2 Area change due to real geometry effects

The rotor geometry used in the design optimization process does not include any fillet at the hub or any clearance between the blade tip and shroud. In reality, these features effectively reduces the area between the blades. The fillet introduces an actual material blockage, while the small space of the tip clearance reduces the fluid velocity in the region to negligible values. The reduction in passage area is estimated, according to figure 3.2. The area of the passage with no blockage is calculated as

$$A = (r_{shroud} - r_{hub}) \frac{1}{2} [(s \cos \xi - t)_{shroud} + (s \cos \xi - t)_{hub}] \quad (3.1)$$

where r is the radial position, s is the blade pitch, t is the blade thickness and ξ is the stagger angle. The area of the tip clearance blockage is calculated in the same manner, assuming that s , t and ξ are the same at the shroud and tip radial positions.

$$A_{tip\,clearance} = (r_{shroud} - r_{tip}) (s \cos \xi - t)_{shroud} \quad (3.2)$$

The fillet, with radius R , is estimated as the difference of a quarter square with side length $2R$ and a circle with radius R , see figure 3.3.

$$A_{fillet} = \frac{(2R)^2 - \pi R^2}{2} \quad (3.3)$$

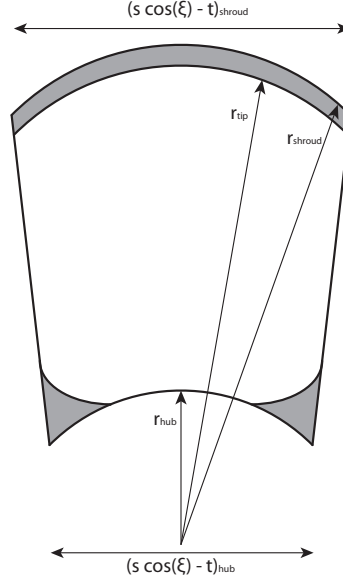


Figure 3.2: Estimate of the effective area blockage.

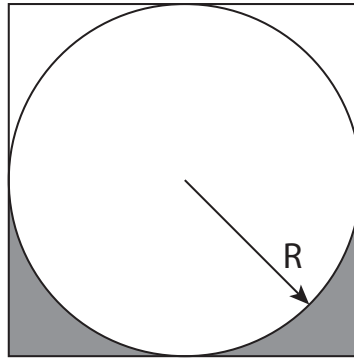


Figure 3.3: The simple fillet representation.

3.3 Rotating the FE point cloud

In order to ensure periodic boundaries at the hub, coordinates in the FE point cloud are copied and rotated to create a continuous hub section, see figure 4.1. This is done using a standard rotation matrix:

$$\mathbf{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} \quad (3.4)$$

where θ is the angle of rotation.

3.4 Coordinate transformation of nonconformance data points

Data of nonconformances exist, but their coordinates are given in the frame of reference of the measuring machine. The corresponding positions on the CAD-surface need to be estimated. This is done by distributing the existing data along the blade surface according to percentage of profile length at given y-coordinate. The distances, Δ_j , between consecutive

points are added cumulatively and used to determine a normalized position of each point i . For n points in a one dimensional case, finding the position of the i^{th} point is done by:

$$\mathbf{x}_{i,norm} = \frac{\sum_{j=1}^i \Delta_j}{\sum_{j=1}^n \Delta_j} \quad (3.5)$$

where $\sum_{j=1}^n \Delta_j$ is the entire length of the curve. The same is done on the curves located in the coordinate system of which we want to place the points. The two normalized vectors are then used to place the points and corresponding displacements in the correct coordinate system, see figure 3.4

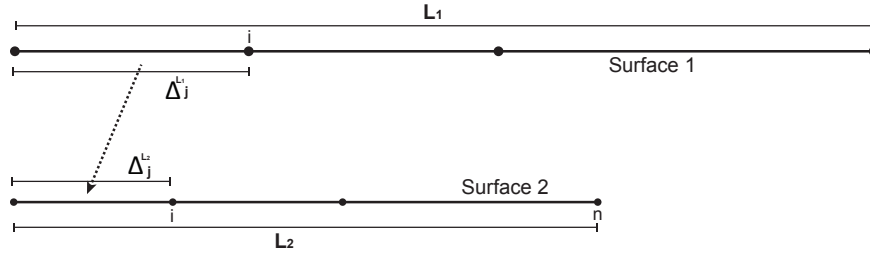


Figure 3.4: Normalized positions are calculated in both coordinate systems, allowing us to transform the given data onto the blade geometry. In the figure above, the normalized distances $\Delta_j^{L1}/L1 = \Delta_j^{L2}/L2$.

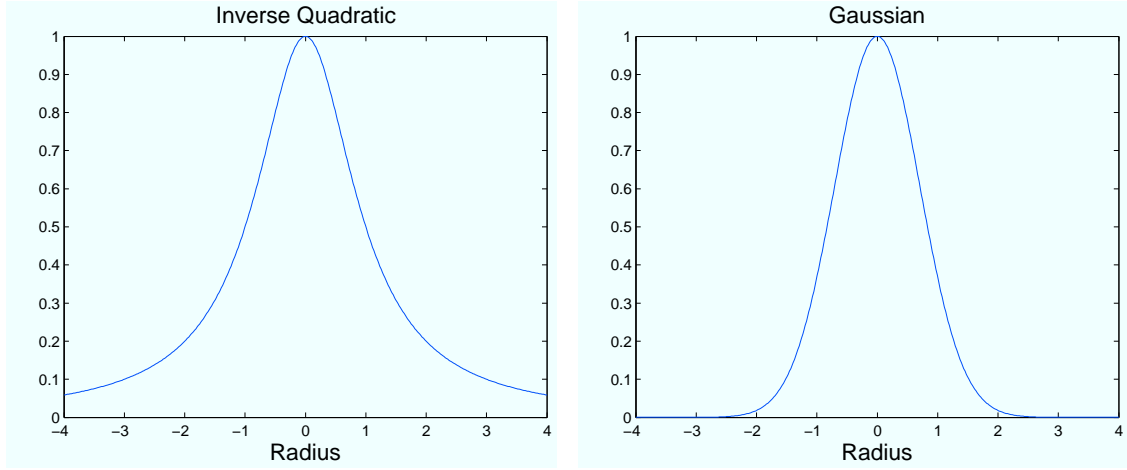
3.5 Radial basis functions

Radial basis functions are functions that gives a distribution depending on a distance or radius. This distribution can be used to rank the importance of data positioned at different locations along the distance. Depending on what problem that is to be solved, different functions can be used. In Table 3.1 some of the most common radial basis functions (RBF's) are listed. In the functions, r is the radius and b a so called shape factor. This shape factor can be used to scale the RBF, which can be useful during certain circumstances. An example of this is shown in section 4.5. To demonstrate the behaviour

Table 3.1: Radial basis functions [1]

$\varphi = e^{-br^2}$	GA, Gaussian.
$\varphi = \frac{1}{\sqrt{b^2+r^2}}$	IMQ, Inverse Multiquadric.
$\varphi = \frac{1}{b^2+r^2}$	IQ, Inverse Quadratic.

of an RBF, two of the listed functions are plotted with $b = 1$ in figure 3.5a and 3.5b.



(a) Inverse quadratic radial basis function

(b) Gaussian radial basis function

Figure 3.5: Example two radial basis functions

The figures show that the two RBF's weigh data at the same distance differently. Also seen is that the functions are symmetric about the y-axis, hence the sign of the distance is of no importance. In the next section it will also be shown that the distance used when performing RBF interpolation is always positive.

3.6 RBF interpolation

RBF is short for Radial Basis Functions and in [2] it is stated that: "Scientific Computing with Radial Basis Functions focuses on the reconstruction of *unknown functions* from *known data*". This is analogous with reconstructing a set of data points according to the displacements (the *known data*) of another set of data points that represents the same surface. How to perform an interpolation using RBF's is described in this section.

When computing the new positions of a dataset, every point needs to be evaluated separately. Assume that two pointsets represent the same geometry, where the first consists of $i = 1, 2, \dots, N$ number of points stored in $\mathbf{S1}_i$ with unknown displacements and the second consists of $j = 1, 2, \dots, M$ number of points stored in $\mathbf{S2}_j$ which have the known displacements \mathbf{U}_j . All sets are in x, y, z space. The principle of the problem is illustrated using a two dimensional case in figure 3.6.

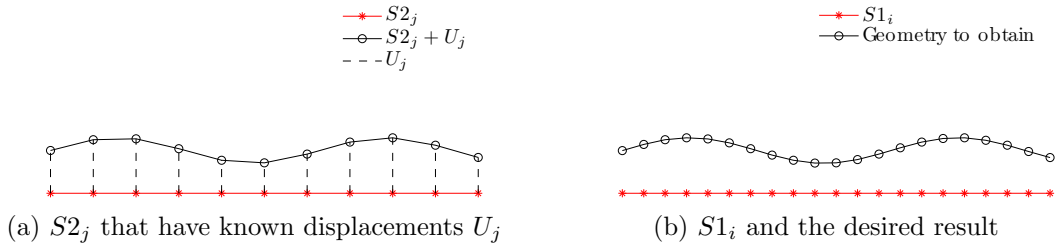


Figure 3.6: Principle of the problem that can be solved using RBF interpolation

Say that one wants to obtain the new position of the arbitrary point, $\mathbf{s1}$ of $\mathbf{S1}_i$. Since all coordinates of both pointsets are known it is possible to compute the distances from

$\mathbf{s1}$ to all points of $\mathbf{S2}_j$. It is reasonable to assume that the displacements of the points in $\mathbf{S2}_j$ closest to $\mathbf{s1}$ should influence its new position the most. Depending on the number of points and the nature of the geometry it is not mathematically possible or geometrically correct to include all points of $\mathbf{S2}_j$ when calculating the new position of $\mathbf{s1}$ [3]. This is most often the case and therefore some kind of sampling process has to be performed. How this is done is the first and crucial step when setting up an RBF interpolation. One way to make this choice is by setting a maximum distance that a point can be situated from $\mathbf{s1}$ to be included in the interpolation. This distance, can be interpreted as a radius, R of a circle (2d case) or a sphere (3d case), thereof *radial* basis function. This distance is in general calculated as the euclidean distance. Which simply is the absolute distance in space, $distance = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$

When the points have been chosen, the distances to them from $\mathbf{s1}$ can be computed and gathered in a vector, here called $\mathbf{r}_k = |\mathbf{s1} - \mathbf{S2}_k(R)|$ where $k = 1, 2, \dots, P$ and P is the number of chosen points. These distances are to be weighted against eachother to identify how much each of the points should influence the displacement of $\mathbf{s1}$. Here this distribution is determined by a radial basis function, φ , described in Section 3.5. In RBF interpolation the idea is to obtain the displacement of $\mathbf{s1}$ in all directions from Eq. 3.6.

$$\begin{aligned} F_x(\mathbf{s1}) &= \sum_{k=1}^P \mathbf{a}_k^x \varphi(|\mathbf{s1} - \mathbf{S2}_k(R)|) + c_0^x + c_1^x x + c_2^x y + c_3^x z \\ F_y(\mathbf{s1}) &= \sum_{k=1}^P \mathbf{a}_k^y \varphi(|\mathbf{s1} - \mathbf{S2}_k(R)|) + c_0^y + c_1^y x + c_2^y y + c_3^y z \\ F_z(\mathbf{s1}) &= \sum_{k=1}^P \mathbf{a}_k^z \varphi(|\mathbf{s1} - \mathbf{S2}_k(R)|) + c_0^z + c_1^z x + c_2^z y + c_3^z z \end{aligned} \quad (3.6)$$

where x , y and z are coordinates of $\mathbf{s1}$ and $F(\mathbf{s1})$ are the displacements in this point[4].

This equation is not solvable at this stage, since a and c are unknown. a needs to be solved in all the chosen points $\mathbf{S2}_k(R)$ and represents the size of the displacement in each of these points. If c is to be included in Eq. 3.1 or not is problem dependent. Sometimes a low order polynomial is needed to solve the interpolation problem in an accurate way. Here a polynomial of first order will be solved for. The polynomial represents the displacements of all the chosen points $\mathbf{S2}_k(R)$. The way to solve these unknowns is to use what is already known, i.e. $\mathbf{S2}_j$ and \mathbf{U}_j . The points of $\mathbf{S2}_k$ are chosen using a sphere around $\mathbf{s1}$. Hence, they construct a space that $\mathbf{s1}$ is situated in and a and c can be used to obtain the displacement of $\mathbf{s1}$. Equation 3.6 can be formulated in each point $\mathbf{s2}$ of $\mathbf{S2}_k(R)$, see Eq. 3.7.

$$\begin{aligned} F_x(\mathbf{s2}) &= \sum_{k=1}^P \mathbf{a}_k^x \varphi(|\mathbf{s2} - \mathbf{S2}_k(R)|) + c_0^x + c_1^x x + c_2^x y + c_3^x z \\ F_y(\mathbf{s2}) &= \sum_{k=1}^P \mathbf{a}_k^y \varphi(|\mathbf{s2} - \mathbf{S2}_k(R)|) + c_0^y + c_1^y x + c_2^y y + c_3^y z \\ F_z(\mathbf{s2}) &= \sum_{k=1}^P \mathbf{a}_k^z \varphi(|\mathbf{s2} - \mathbf{S2}_k(R)|) + c_0^z + c_1^z x + c_2^z y + c_3^z z \end{aligned} \quad (3.7)$$

where x , y and z are coordinates of $\mathbf{s2}$ and $F(\mathbf{s2})$ are the displacements in this point[4].

By letting $\mathbf{s2}$ go from $\mathbf{S2}_1(R)$ to $\mathbf{S2}_P(R)$ and setting the conditions of equation 3.8, enough equations are created to solve for the only unknowns in Eq. 3.7, a and c [5].

$$\sum_{k=1}^P a_k = 0, \sum_{k=1}^P a_k x_k = 0, \sum_{k=1}^P a_k y_k = 0 \text{ and } \sum_{k=1}^P a_k z_k = 0 \quad (3.8)$$

where x , y and z are coordinates of $\mathbf{S2}_k(R)$ and a the components of the vectors \mathbf{a}_k^x , \mathbf{a}_k^y and \mathbf{a}_k^z .

The following systems of equations are set up:

$$\mathbf{GA}_x = \mathbf{F}_x, \mathbf{GA}_y = \mathbf{F}_y \text{ and } \mathbf{GA}_z = \mathbf{F}_z \quad (3.9)$$

$$\mathbf{G} = \begin{pmatrix} \varphi_{11} & \varphi_{12} & \varphi_{1P} & \bullet & \bullet & \bullet & 1 & x_1 & y_1 & z_1 \\ \varphi_{21} & \varphi_{22} & \varphi_{2P} & \bullet & \bullet & \bullet & 1 & x_2 & y_2 & z_2 \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ \varphi_{P1} & \varphi_{P2} & \varphi_{PP} & \bullet & \bullet & \bullet & 1 & x_P & y_P & z_P \\ 1 & 1 & \bullet & \bullet & \bullet & 1 & 0 & 0 & 0 & 0 \\ x_1 & x_2 & \bullet & \bullet & \bullet & x_P & 0 & 0 & 0 & 0 \\ y_1 & y_2 & \bullet & \bullet & \bullet & y_P & 0 & 0 & 0 & 0 \\ z_1 & z_2 & \bullet & \bullet & \bullet & z_P & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.10)$$

$$\begin{aligned} \mathbf{A}_x &= (a_1^x, a_2^x, \dots, a_P^x, c_0^x, c_1^x, c_2^x, c_3^x), \mathbf{F}_x = (u_1^x, u_2^x, \dots, u_P^x, 0, 0, 0, 0)^T \\ \mathbf{A}_y &= (a_1^y, a_2^y, \dots, a_P^y, c_0^y, c_1^y, c_2^y, c_3^y), \mathbf{F}_y = (u_1^y, u_2^y, \dots, u_P^y, 0, 0, 0, 0)^T \\ \mathbf{A}_z &= (a_1^z, a_2^z, \dots, a_P^z, c_0^z, c_1^z, c_2^z, c_3^z), \mathbf{F}_z = (u_1^z, u_2^z, \dots, u_P^z, 0, 0, 0, 0)^T \end{aligned} \quad (3.11)$$

where x , y and z are coordinates of $\mathbf{S2}_k(R)$ and u the displacements in these points[4].

The \mathbf{A} vectors in Eq. 3.9 are solved through a matrix inversion of the so called system matrix, \mathbf{G} and three vector-matrix multiplications, giving the a and c coefficients. The displacement in each direction of the point $\mathbf{s1}$ can now be obtained.

Table 3.2: Explanation of variables for RBF interpolation.

$\mathbf{S1}_i$	Points of original geometry that is to be modified.
$\mathbf{s1}$	Arbitrary point of $\mathbf{S1}_i$.
$\mathbf{S2}_j$	Points of original geometry that has known displacements.
\mathbf{U}_j	The known displacements.
$\mathbf{S2}_k(R)$	The chosen points to use for interpolation.
$\mathbf{s2}$	Arbitrary point of $\mathbf{S2}_k(R)$.
φ	Radial basis function.
a	Interpolation coefficients in the points of $\mathbf{S2}_k(R)$.
c	Coefficients of the linear polynomial in the space of $\mathbf{S2}_k(R)$.
\mathbf{G}	System matrix that is inverted to solve the displacements of $\mathbf{s1}$.

3.7 Stability vs accuracy

During RBF interpolation there is a risk that the system matrix becomes difficult to invert due to singularity. A matrix becomes singular when its determinant approaches zero. A simple example of what can cause the determinant of a matrix to be singular follows below.

$$\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$$\det(\mathbf{A}) = 0 \Leftrightarrow ad - bc = 0 \Leftrightarrow \frac{a}{b} = \frac{c}{d} \quad (3.12)$$

In Equation 3.12 it can be seen that if the ratio between the scalars of the two rows are similar, there is a risk that the determinant goes to zero, yielding a singular matrix.

A problem like this would, in the case of the system matrix, occur if the values of the φ 's were very similar and/or if the coordinates of $\mathbf{S2}_k(R)$ have similar ratios between them. This can typically be a problem if the chosen points are equally distributed along a straight line in space, where two or more coordinate components of the chosen points are changing with the same constant value in space. The problem also becomes more noticeable when dealing with small values, as it is easier to end up with a determinant of zero.

One way to ensure that the values of the φ 's are not too similar is to simply scale the RBF with a larger value of b . Since the same scaling is used for all the chosen points, there is no influence on the total weighting of the data. In most cases, data close to a certain point reflects the displacement of this point better than data at larger distances. Hence, at some value the scaling of the RBF will lead to loss of accuracy. The choice of shape factor is known as the *uncertainty principle* and is a matter of compromising between stability and accuracy of the interpolation[2, 6]. There are many strategies of choosing a good shape factor in literature and the "optimal" shape factor is still an open question. Hence, it has so far in most cases been a process where the smallest shape factor that allows the system matrix to be inverted is chosen[7]. In figure 3.7 the Inverse Quadratic RBF is plotted with different values of b .

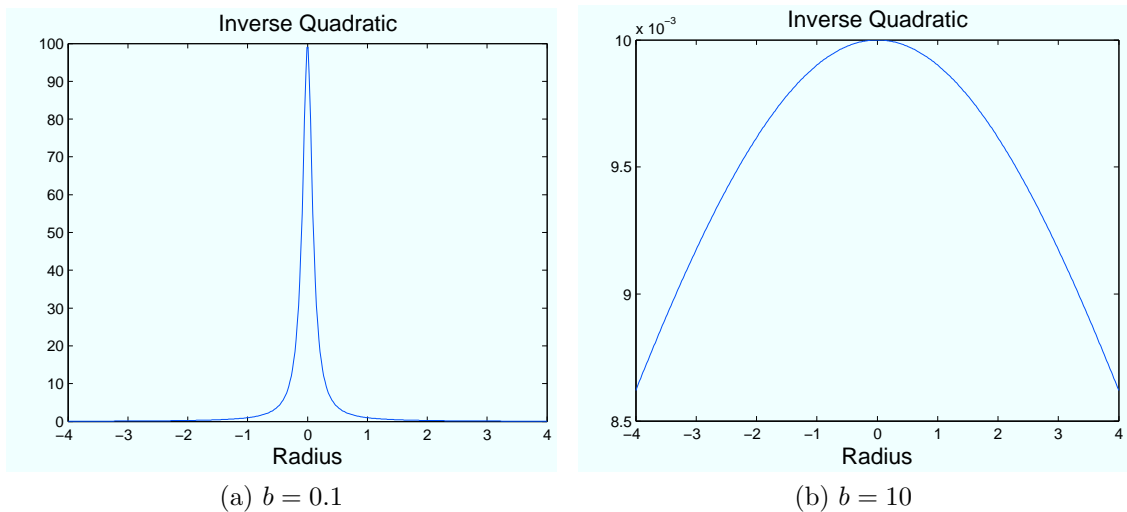


Figure 3.7: Inverse Quadratic RBF with different shape factors

Figure 3.7a shows that a small shape factor generates basically the same value of the RBF for distances larger than 2. Having a lot of data above this distance will give many φ 's of

the same size, leading to the system matrix becoming more difficult to invert. In figure 3.7b it can be seen that a large shape factor gives a smoother RBF for the same distances. This may on the other hand overestimate the importance of data at larger distances, leading to poor accuracy.

The above mentioned problems causing a singular system matrix also has a tendency to occur more often as the number of chosen points increase [3]. A simple remedy for this is of course to reduce the radius used for searching points, this will however imply a risk of decreased accuracy.

If having very dense data in $\mathbf{S2}_j$ the risk of points being too close to each other increases, giving very similar φ 's. In cases such as this a measure can be to remove "duplicates" and points close to each other.[2]

3.8 Simple RBF interpolation

At Volvo Aero Corp, another interpolation method involving RBFs is available. The routine, that is described here, has worked well in other applications. Consider the same situation of two pointsets that are described in section 3.6. The distances from the point to be interpolated, $\mathbf{s1}$ to the data points of $\mathbf{S2}_j$ is calculated. A fixed number then determines how many points that is to be used for interpolating. Typically this number is between ten and twenty. When the points have been gathered, their distances are weighed according to equation 3.13.

$$\mathbf{w}_k = \frac{\varphi_k}{\sum_{k=1}^P \phi_k} \quad (3.13)$$

In the equation φ_k is a vector of RBF's for each distance, ϕ_k is a component of this vector and P is the number of points to use in the interpolation. The displacement in $\mathbf{s1}$ is obtained through equation 3.14, where \mathbf{U}_k is a vector containing the displacements of the sampled points.

$$F(\mathbf{s1}) = \mathbf{w} \cdot \mathbf{U}_k \quad (3.14)$$

3.9 Stall margin

When the pressure at a compressor stage exit becomes too high the flow will eventually reverse direction - a condition called compressor stall/surge. The stall margin at the compressor design point is calculated according to Volvo Aero standard. For a choked flow, the corrected massflow can be expressed as

$$\dot{m}_{corr} = C\pi^{1-\frac{\gamma-1}{2\gamma\eta_p}} \quad (3.15)$$

where C is a constant and π is the total to total pressure ratio. This constant can then be calculated for any point and used to determine the pressure ratio, π_m , at which the design point mass flow would be achieved for a stalling compressor. The relation between this pressure ratio and the design point pressure ratio, π_{ADP} , defines the stall margin

$$SMM = \frac{\pi_m - 1}{\pi_{ADP} - 1} \quad (3.16)$$

π_m is calculated as

$$\pi_m = \left(\frac{\dot{m}_{corr, stall}}{C} \right)^{\frac{1}{1 - \frac{\gamma-1}{2\gamma\eta_p}}} \quad (3.17)$$

where $\dot{m}_{corr, stall}$ is the corrected massflow at which the current speedline reaches stall.

4 Method

Deforming the rotor blade geometry is a big part of the work in this thesis. Unfortunately it is not a straight forward process. It involves some trial and error and the Matlab program written requires certain input data arranged in a specific way. In the following sections the process is explained in detail and can hopefully be used as a tutorial for future calculations. For further information, see the written comments in the Matlab code.

4.1 Preparing the cold CAD geometry

The geometry to be deformed is an ICEM tetin file, consisting of node coordinates and a connectivity list defining how the faces are constructed. In order to read this into the Matlab program it is crucial that it is in the correct format. The geometry file needs to consist of only one surface part. No curves, points or other parts can be present for the script to work. If the geometry consists of several surfaces these need to be merged and converted to facets before imported. If one wants to add nonconformance displacements to the geometry, it is important to make sure that all the surface normals point outwards from the surfaces. This can be done in the **Repair Geometry** menu in ICEM. In this case, also the tip surface needs to be removed. This is to ensure that it does not get deformed in the normal direction when applying the deviations. The tip is added again once the deforming is complete, by creating a new surface in ICEM. The resulting tetin file for input should consist of a few header lines, a list of coordinates, followed by a list of node connectivity and end with a few footer lines, see appendix A. The coordinate and connectivity matrices are read into the Matlab program along with the displacement data.

4.2 FE data

The data from the FEA consist of a node position matrix and an equally constructed displacement matrix. These are imported into the Matlab program and modified to fit the coordinate system and dimensions of the blade geometry. For the interpolation routine to yield periodic boundaries, the hub section of the FE points is duplicated and rotated, as described in section 3.3, creating a continuous hub, see figure 4.1. Finally possible duplicate points in the FE data are removed, since this can cause problems for the RBF interpolation, see section 3.7.

4.3 Manufacturing nonconformances

The displacements due to manufacturing nonconformances are by far the most cumbersome to handle. First off, the data used here is very coarse. Four sections of constant y-coordinate have been measured and are in this case available only in image charts, as in figure 4.2.

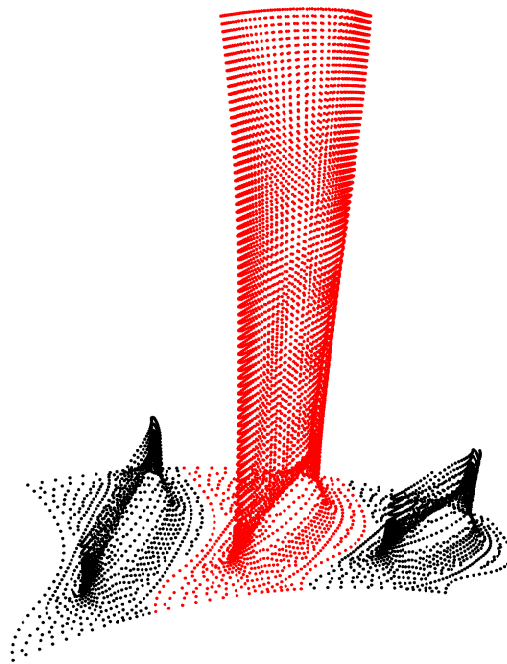


Figure 4.1: The hub section is rotated to each side of itself, ensuring periodic boundaries.

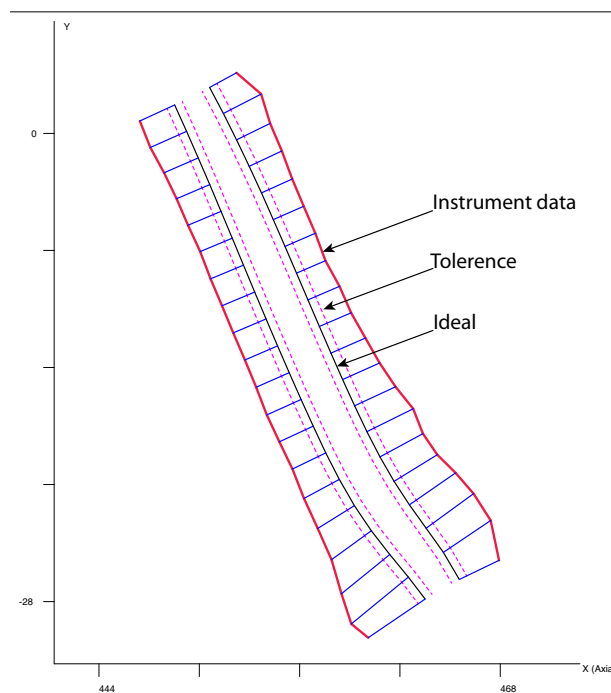


Figure 4.2: Typical representation of the nonconformance data available. 20 points at each side of the blade are available, but in the wrong coordinate system.

Therefore it is required to manually pick each coordinate and transfer into matrices containing coordinates of ideal blade positions and corresponding coordinates with deviations added. This was done using a program called QT Digitizer, reading first the pressure side from leading to trailing edge, followed by the suction side from leading to trailing edge. Each section consist of 40 points; 20 on each side of the blade. The deviations are measured normal to the surface with the blade in a "best fit" position in relation to the measured span, making information about for example stagger angle deviations unavailable. From

the nonconformance charts it is not possible to determine exactly where on the blade the data is measured. Therefore the positions need to be estimated, see section 3.4. Sections from the original ideal geometry, at the y-coordinates in question are extracted as curves from ICEM and imported as a coordinate matrix in Matlab. This makes it possible to relate the profiles to the geometry we want to deform by doing the same cumulative sum calculations on these curves. These curves are split into a pressure and suction side before exported and the start and end points of the curves are also needed for sorting of the points. The result can be seen in figure 4.3.

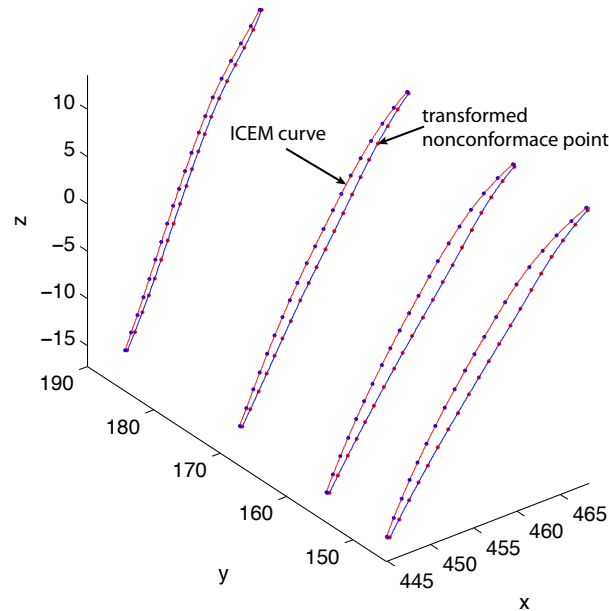


Figure 4.3: Available nonconformance data interpolated onto curves extracted from ICEM

From the charts, e.g. figure 4.2, it is also evident that no deviations are measured at the extreme edges of the blade. It turns out that the interpolation routine works without any point at the edge, but a possibility to add end points is available. The way how is to measure an approximate tip radius at each section in for example ICEM and adding a new coordinate, representing the edge, in the nonconformance point cloud. The added deviation in this point is, in this case, chosen as a mean of the values at the closest points at each side of the blade.

It can be argued that instead of a mean value, one should assume a smaller value at the edges due to the way the blades are manufactured. The nonconformances are due mostly to material flexing while milling, which could be substantially less prominent at the edges. However, setting the deviation to zero in these points created blunt edges when interpolating. While this is not a perfect method, it will suffice to get a general idea of how the deviations could vary across the blade. The code allows the user to add more points to the missing radius, however for the coarse data set present in this case, adding more than one causes instability in the interpolation routine rather than a better representation of the deviations.

Another problem that arises due to lack of data is that no information of nonconformances at the blade tip or hub sections is available. From the trends of existing deviations, seen in figure 4.4, it is safe to assume that the deviations should increase with increasing radius. Since the interpolation routine will not take this into account, but simply use the closest

available data, tip deviations need to be created. Also, since deviations decreases with decreasing radius, it is assumed that the nonconformances at the hub are negligible, and therefore set to zero. The profiles can be acquired by saving curves of the cross sections near the hub and at the tip as ICEM .tin files and extracting the coordinates, just as for the four ordinary profiles. The zero deviations are easily created by adding zeroes to the displacement matrices. The tip profile, on the other hand, requires yet another assumption. Here it is assumed that the growth of deviation magnitude between the last known profile and the newly created tip profile is linear. The slope is calculated from the two highest positioned known profiles and multiplied by the distance to the tip profile, in order to get the displacements at the tip. When all ideal and displaced coordinates have been calculated, the displacements are simply taken as the difference between the two.

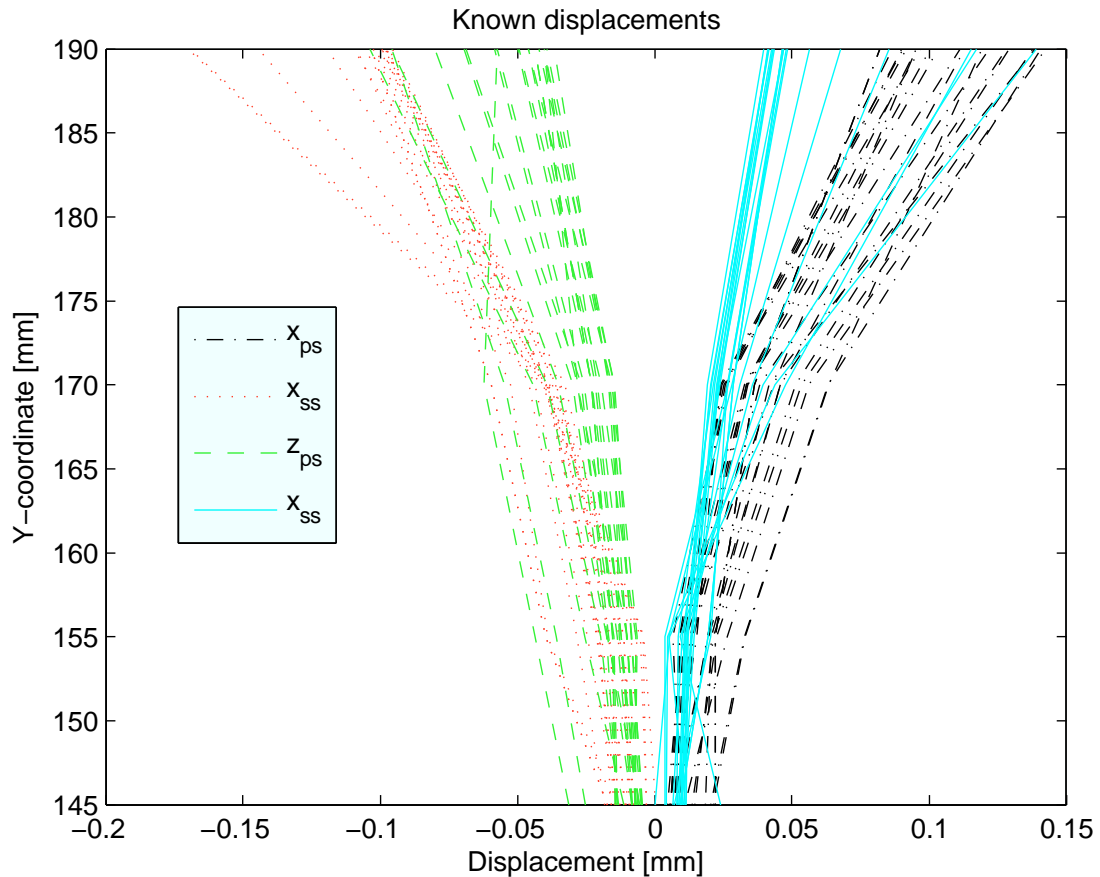


Figure 4.4: Plot of nonconformance data at all given points of a blade. A clear trend can be seen; Deviations increase with increasing radius.

4.4 Simple RBF interpolation

The first interpolation problem that was dealt with, was the cold to hot conversion and the method used was the fast and simple one described in section 3.8. At this stage the Cold geometry was created as a surface mesh, which then was displaced via the interpolation routine. The results appeared to be accurate and smooth as long as the mesh used was somewhat coarser than the FE grid that described the displacements. However when refining the mesh, to be able to describe the blade geometry with a higher resolution, the method started to fail. It could not reconstruct the unknown displacements in a smooth manner between the coarser known points. A typical example of how this could appear is shown in figure 4.5.

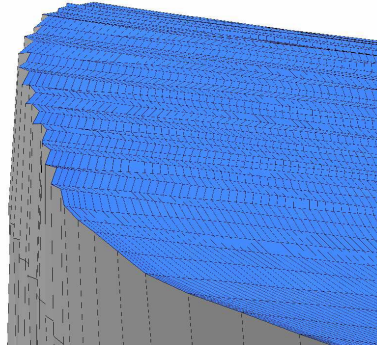


Figure 4.5: Typical result of the simple RBF method when using a finer resolution of the mesh than what is available in the FE points.

Due to the poor accuracy of this method it was abandoned for the more complicated and sophisticated method described in section 3.6. The use of a surface mesh was also replaced by a triangulated surface from a .tin file.

4.5 RBF interpolation cold to hot

The main approach for applying the displacements of the FEA data onto the CAD geometry is to use the theory of RBF interpolation explained in section 3.6. The CAD surface points corresponds to $\mathbf{S1}_i$ and the FEA data to $\mathbf{S2}_j$ and \mathbf{U}_j . Furthermore the RBF that is used is the Inverse Quadratic, see table 3.1. However, the process is not completely straightforward and measures described in section 3.7 have to be taken. In the developed Matlab code a number of parameters can be set for the cold to hot interpolation.

The radius to use for sampling points has to be set according to the nature of the FEA data. To ensure a reasonable interpolation it is desirable to always enclose the CAD surface point with FEA points. A quick way to determine a suitable radius is to pick a FEA point on the blade surface and measure the distances to other FEA points surrounding it and enclosing it in a evenly distributed way, see Figure 4.6. Take the average of these distances and increase it with about 30%. The radius is specified in the parameter `R_cold2hot`.

Choosing the radius like this should ensure that the radius never is too small. However in some situations it may yield too many points being sampled, giving a singular system matrix. The interpolation routine will however not allow a singular matrix and loops the calculation until good conditioning is fulfilled. The first thing done if this problem occurs is that the number of chosen points is decreased until the number of sampled points has reached the value of `min_no_of_points`, which as it sounds is the least number of points that will be used for interpolation. This parameter is by default set to 12 and decreasing it may lead to loss of accuracy, while increasing it gives better accuracy but can cause the routine to get stuck. To get a hint of the maximum value for this parameter, check how many points that would be captured in the procedure described in connection with figure 4.6. If decreasing the number of chosen points is not enough for creating an invertible system matrix, the next step is to increase the shape factor.

The initial and desirable value of the shape factor is by default set to `b = 1/length(FEAdata)`

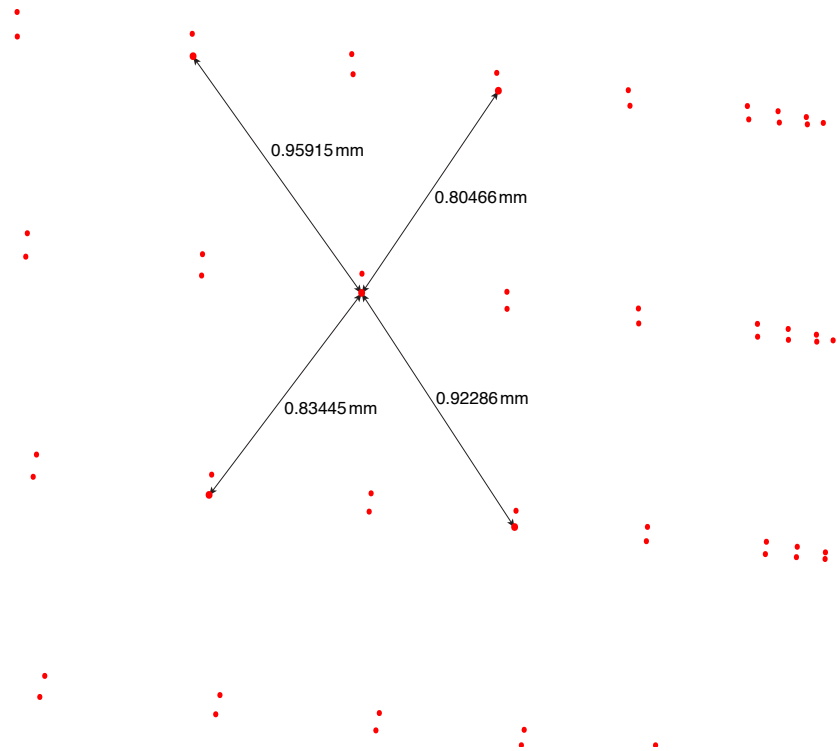


Figure 4.6: The radius to use in the RBF interpolation for cold to hot is determined by measuring distances from one FEA point to other FEA points enclosing it in a evenly distributed way.

but can be scaled by changing `b_scale_c2h`. A small shape factor gives high accuracy but increases the computational time. How fast it is increased in case of singular problems is determined by `b_step` through that `b = b + b_step` for every new try of obtaining an invertible system matrix. This step is by default also set to `b_step = 1/length(FEAdata)` but can be scaled with the parameter `b_step_scale_c2h`. A small increase gives a higher probability of obtaining the best possible accuracy but can also increase the needed computational time substantially. The number of chosen points when increasing the shape factor is fixed to `min_no_of_points`

4.6 RBF interpolation nonconformances

The theory of section 3.6 and 3.7 is, besides from the sampling process, applied also when reconstructing the nonconformances on the hot CAD surface. Due to the very coarse nature of the nonconformance data in the y-direction, the radius needed to always enclose the current CAD surface point becomes so large that points unacceptably far away in the coord-wise direction are included. Instead the fact that the nonconformance data is structured is used.

The sampling starts by finding the nonconformance point closest to the current CAD surface point. After this a number of neighbours on each side, specified by `neighbours`, of this point is found. Finally the corresponding points on the y-section enclosing the CAD point is identified. This gives a fairly good representation of the displacement from such

coarse data. An example of how points are sampled for interpolation can be seen in figure 4.7. As seen in the figure, neighbour points can be located on the opposite blade side, this has to be done to ensure a smooth transition between the sides at the edges.

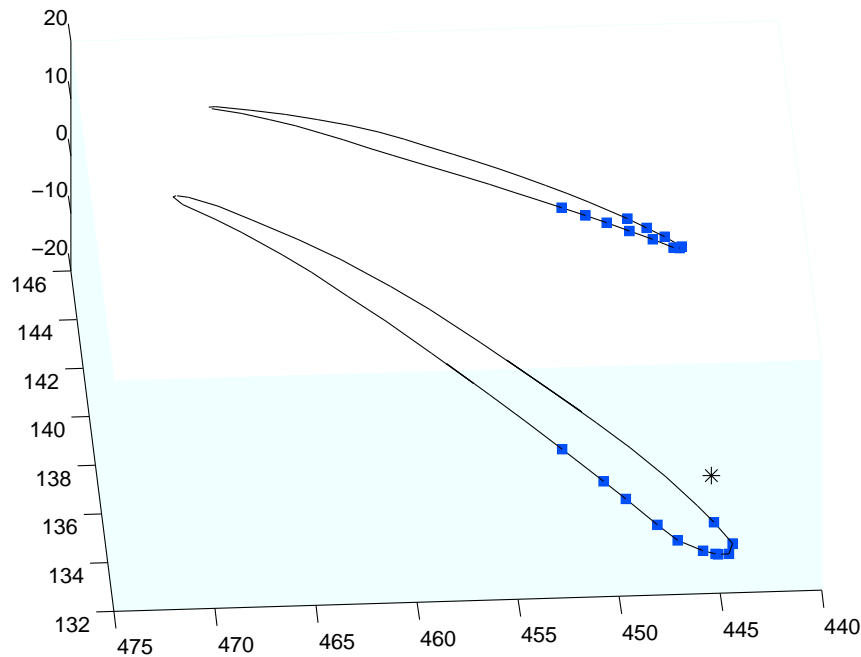


Figure 4.7: The square points will be used for interpolating the displacement of the star point. The hub span and the second lowest span is used for finding points in this case.

If problems with singularity arise, the number of neighbours to include is decreased. The minimum number of neighbours to include can be set in `nb_limit`. The properties of this parameter corresponds to `min_no_of_points` in section 4.5. Furthermore the shape factor parameters `b_scale_NC` and `b_step_scale_NC` also have the same features as their counterparts in section 4.5.

It should also be mentioned that since the displacements all are normal to the surface in the point where they are defined, they are manipulated before being used for interpolation. The surface normal can change quite fast along the blade, especially at the edges. Hence, the displacements of the nonconformance points to be used for solving equation 3.9 are all redirected in the same direction as the normal of the CAD surface point that is to be displaced.

4.7 RBF interpolation strategy

When using this code, it is good to have a strategy for modifying the parameters described in section 4.6 and 4.5. The following recommendations applies for both the cold to hot and the nonconformance interpolation.

- Start with a large shape factor and a small increasing step of it - the computation will be fast and the user will be able to see if the correct geometry is built up. When this is assured the shape factor is lowered for better accuracy.

- If adjusting the shape factor is not enough for desired accuracy - try changing the `R_cold2hot/neighbours`; more points in general gives higher accuracy.
- If still not satisfied - try to increase the `min_no_of_points/nb_limit`.

These recommendations gives the user a starting point when performing the RBF interpolation. However there are some things that should be observed.

- A too small shape factor will yield bad accuracy since it will force the radius/number of neighbours to decrease at an earlier stage.
- Extrapolation of points can cause problems and has not been thoroughly tested.

4.8 Computational domain

Once the blade geometry is defined, the computational domain is constructed in ICEM HEXA. It is likely that the deformed geometry needs to be repaired before starting meshing, since there might be gaps between the displaced surface patches, see figure 4.8.

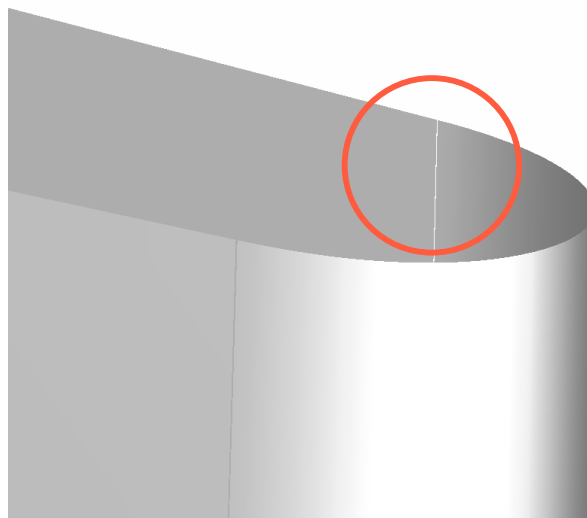


Figure 4.8: Example of surface gaps as a consequence of deforming the geometry.

This is done using **Build topology**. Also, a new tip surface has to be created. The blade geometry is then merged into a domain from [8], replacing the original blade. The script described in [8] is used to read the ideal design of the blade from NX and outputs the corresponding computational domain, containing blade, hub, shroud and rotationally periodic surfaces. Once the topology is built the blade geometry is replaced by the modified version. The mesh is then reprojected onto the new geometry. Since the hub geometry from the FEA and the design geometry are nearly identical and unaffected by nonconformances, it is possible to let the hub surface blend with the domain hub surface. An area where the two geometries cause a problem is in the lower part of the fillet. Here the hub surface from the script and the new fillet surface separates, making associating faces to the correct surface difficult, as illustrated in figure 4.9.

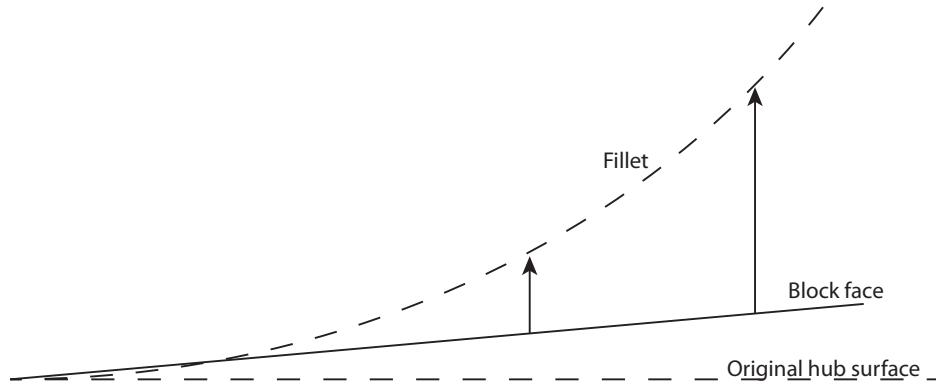


Figure 4.9: Associating face to part becomes important in cases where two surfaces are close. Here the fillet surface is the one to project on.

Here it is necessary to associate the block faces to the part containing the fillet. Therefore it is favourable if as much of the hub consist of the new surfaces. Some handiwork might be necessary, including creating new surfaces for mixing planes, closing small gaps between the new and old geometries and generating new support curves for ogrid edges. Also, the block topology needs to be modified in order for the fillet of the blade to be accounted for. Figures 4.10-4.12 show an overview and some features of the rotor domain mesh. It is a block structured hexahedra mesh with O-grids around the blade, which in turn is surrounded by an H-grid. The node distribution at the periodic boundaries is 1:1.

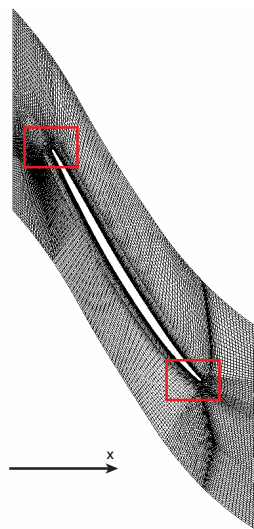


Figure 4.10: Scan plane at the midsection of the mesh.

4.9 Numerical simulation

When the mesh of rotor 2 is complete, meshes of the stator vanes before and after the rotor is added and the CFD simulations is performed in ANSYS CFX 12.1.

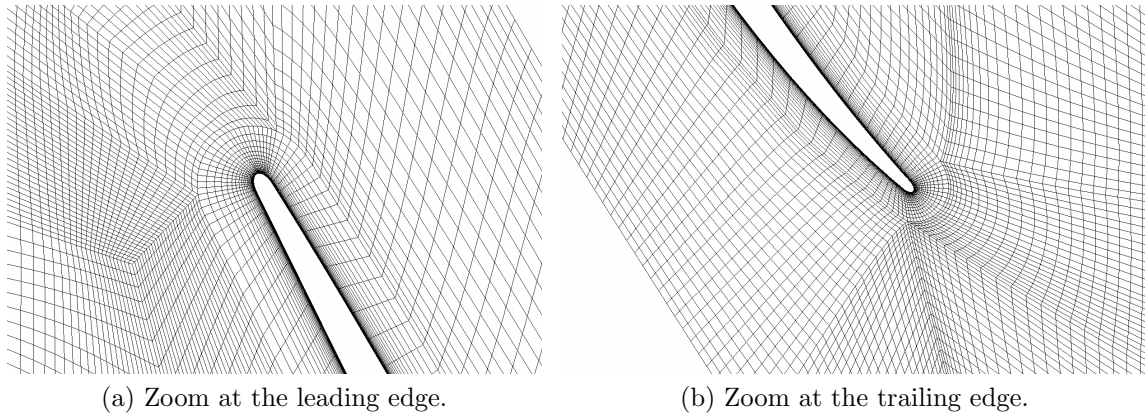


Figure 4.11: Mesh at the marked areas of the midsection seen in figure 4.10.

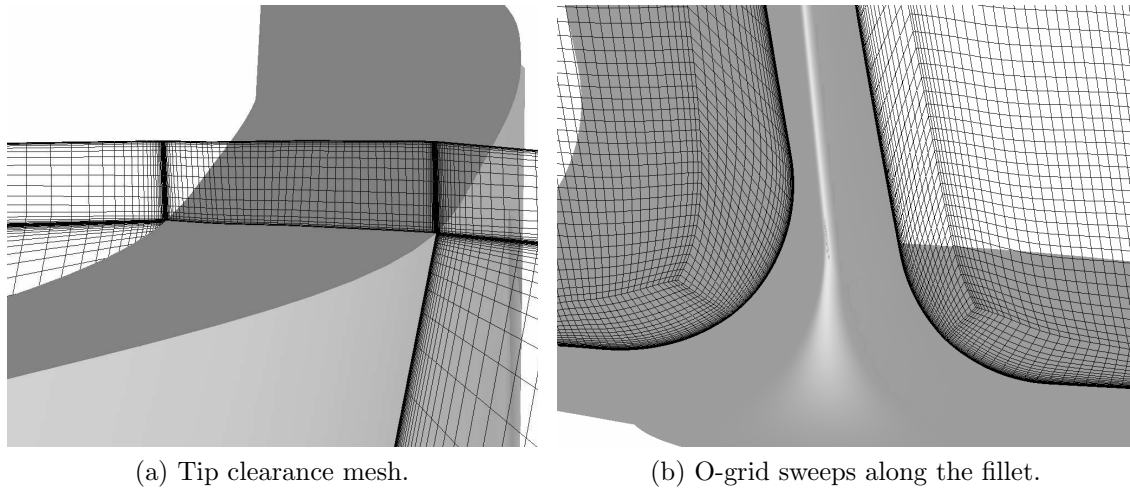


Figure 4.12: Mesh details.

4.9.1 Mesh configurations

Four runs are to be compared: The two deformed blades using the $k - \omega$ SST model and Low Reynolds number (Low Re) meshes and the Design geometry using a mesh with a first cell size suitable for scalable $k - \varepsilon$ with wall functions. The latter is what is currently being used in the design optimization process. It is therefore interesting to run the simulations using these particular meshes to see if the results differ substantially from the, probably more accurate, Low Re meshes. The main difference between them, apart from the cell count, is that there is no fillet or tip clearance in the design mesh. In addition, a Wall Function mesh of the Hot geometry (Hot WF) was used in a series of $k - \varepsilon$ simulations in order to get an estimate of the speedline characteristics and a good initial guess when running the Low Re Simulations. It is also compared to the Design geometry. The number of cells in the boundary layer was 37 and 8 for the Low Re and Wall Function meshes respectively. Similarly, the first node distance for each was 0.00015 mm and 0.1 mm.

The setup used in all calculations consist of three subdomains; the second rotor as well as the first and second stator, see figure 4.13. Due to time limitations, both design and Low Re stator meshes were meshed in G3dmesh and supplied to the project. These are based on aero definitions, thus not including the fillet at the shroud or any gap in the hub area. They serve merely as "dummy" blocks in the configuration in the sense of allowing for the boundary conditions to be set in a way that satisfy the known experimental values.

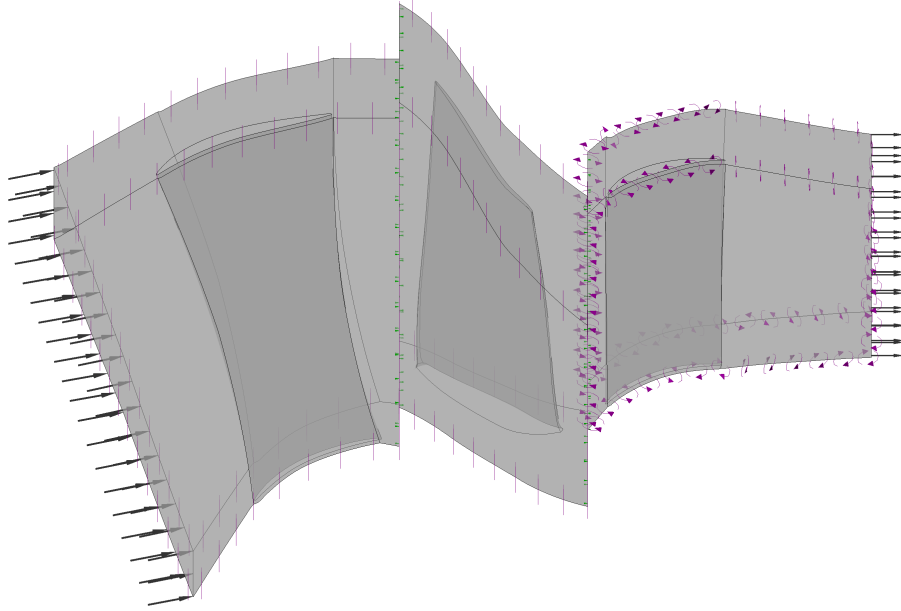


Figure 4.13: Computational domain, consisting of stator 1, rotor 2 and stator 2.

4.9.2 Simulation settings

A mixed pressure boundary condition is used at the inlet with specified profiles of total pressure, total temperature, flow angles, turbulent kinetic energy and turbulent dissipation. Experimental data of P_0 and T_0 before the first stator exist. The remaining properties needed for the inlet boundary condition are extracted from a previous CFD simulation of a complete compressor configuration [9]. This simulation shows a slightly lower value than the measured T_0 in the position where the experimental equipment was placed. To compensate for this difference, the flow angle from the CFD results is altered. This is required as an inlet boundary condition in the simulations and therefore needs to be estimated. Assuming all of the temperature difference is due to less work being done in the first rotor, a new whirl velocity for the simulation can be estimated from the Euler equation[10]

$$C_p(T_{02CFD} - T_{01CFD} - \Delta T_{02exp,CFD}) = U\Delta C_w \Rightarrow \quad (4.1)$$

$$\Delta C_w = C_p(T_{02CFD} - T_{01CFD} - \Delta T_{02exp,CFD})/U \quad (4.2)$$

The shroud, hub, blade and vane are set to no slip walls. The rotor domain is set to rotate, while the rotor shroud is set as a counter rotating no slip wall. Between each subdomain a so called mixing plane, that averages the flow properties circumferentially between the moving and stationary mesh regions [11], is placed. An absorbing boundary condition is used at the outlet where the static pressure is increased while the rotational speed is constant, producing a back pressure sweep ranging from a choked operating point to an operating point close to stall. These sweeps create speedlines, which are to be compared between the different geometries. For a complete list of simulation parameters, see appendix B.

5 Results and discussion

In this section the performance of the developed interpolation method is presented, as well as the results from the CFD simulations performed using the different geometries.

5.1 Interpolation method

5.1.1 Hot to cold interpolation

To confirm that the interpolation routine works as desired, comparisons between cold and hot geometries of CAD and FE points are done. Figure 5.1 shows that the Cold geometry and the cold FE points overall represents the same geometry.

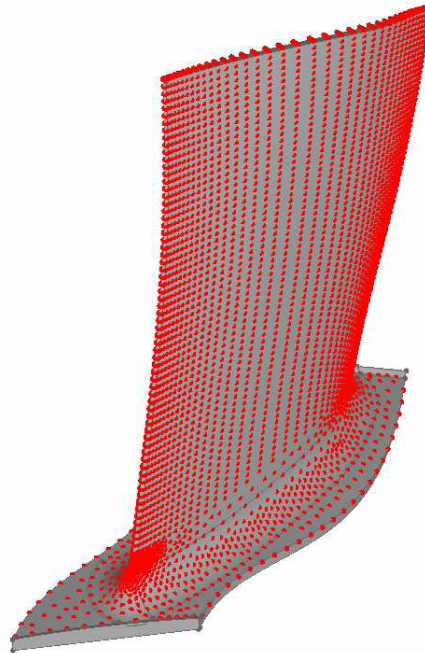


Figure 5.1: Cold geometry and cold FE points.

However, if one looks closely the two data sets are not completely the same. In an ideal situation these two geometries would be the same, as the flow chart in figure 2.2 implies. The reason for that they are different in this case has not been fully investigated, but most likely an approximation error has been introduced when going from the CAD geometry to the FE mesh. The difference should however be so small that the RBF interpolation is unaffected, the closest points are the same as if the geometries were matched exactly. To quantify the small differences between them, spans (hereafter called top and lower) according to figure 5.2 are zoomed in at the leading and trailing edge where the differences are most prominent. Comparisons of the differences when going from cold to hot is done at the edges at the two spans.

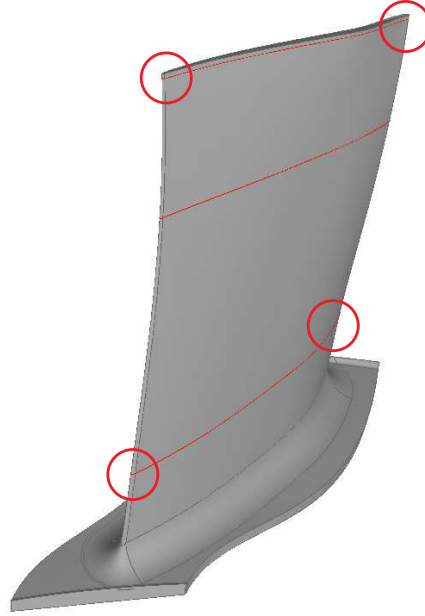


Figure 5.2: Spans were the differences between the CAD geometry and the FE points are studied in hot and cold state.

If the RBF interpolation works properly the difference between the Cold geometry and the cold FE points should be the same as the difference between the Hot geometry and the hot FE points. In figure 5.3 and figure 5.4 the differences between the CAD geometry and the FE points before and after the RBF interpolation are shown. The points where the distances are measured, are indicated in the figures and are situated at corresponding locations for the Hot and Cold geometries.

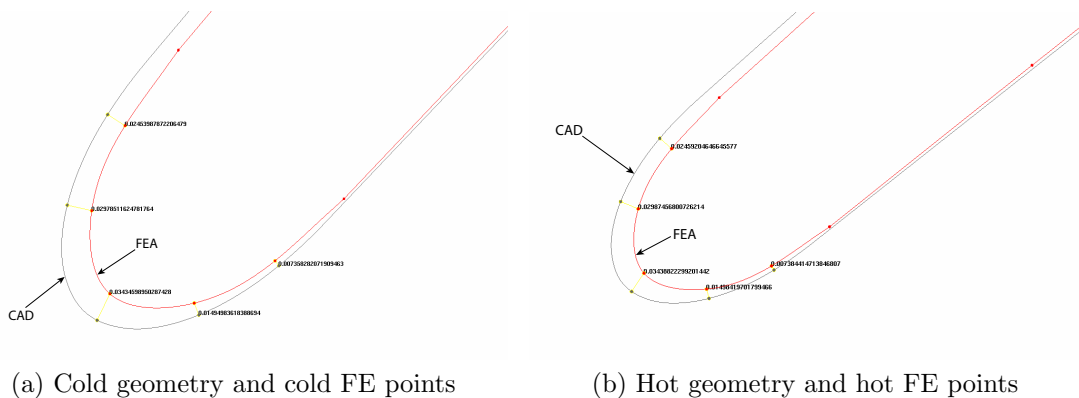


Figure 5.3: Difference between cold and hot geometries of CAD and FE points at the leading edge of the top span.

The distances measured show that the difference between the geometries is well preserved through the RBF interpolation. The largest change due to the interpolation of the measured distances is in fact 0.7% of the smallest displacement of all FE points. Based on the validation of the cold to hot step that has been made here, the RBF interpolation of this transformation is very accurate. To fully trust this validation it would of course be preferred that the CAD geometry and the FE points represents the exact same geometry. It would also be good to ensure that the FEA is performed on the correct geometry, giving the exactly correct displacements. To speculate in how much this small difference influences the FEA results is outside the scope of this project.

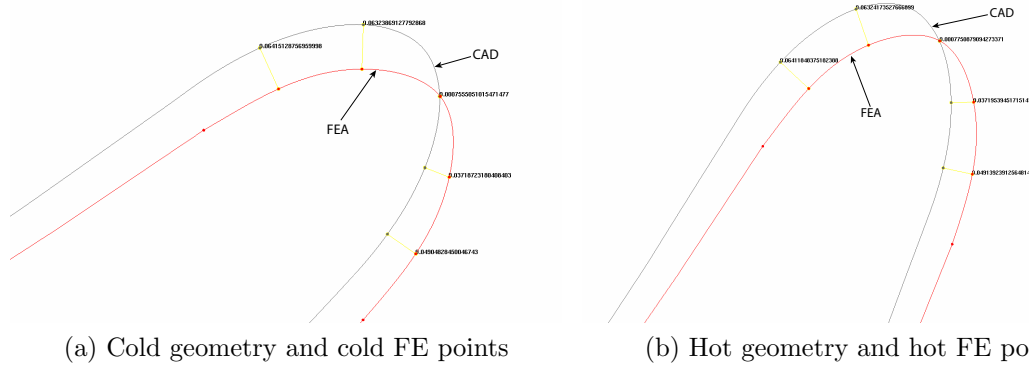


Figure 5.4: Difference between cold and hot geometries of CAD and FE points at the trailing edge of the lower span.

5.1.2 Nonconformance interpolation

Validation of the interpolation to the Hot Nonconf geometry is obtained through that the geometry is compared with the known nonconformance data at the two sections, $y = 145\text{mm}$ and $y = 190\text{mm}$, see figure 5.5.

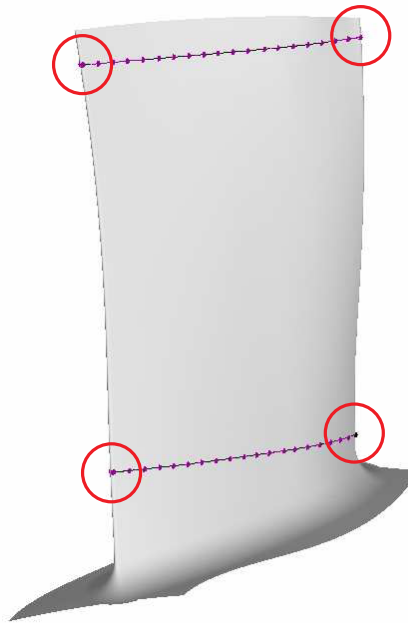


Figure 5.5: Spans were the accuracy of RBF interpolation of the nonconformances is studied.

In the middle of the pressure and suction sides, the difference between the curves is small. Closer to the edges, where there is lack of enclosing data, there is a more noticeable difference. Figure 5.6 and figure 5.7 show the points at the leading and trailing edge of the two sections marked in figure 5.5. The maximum difference is in the point closest to the trailing edge on the pressure side of the 190-section and is 35% (0.05 mm) of the displacement in this point. The point closest to the trailing edge on the suction side deviates 0.02 mm from the real nonconformance. This is more than the displacement itself in this point!

To reduce the error in these points the shape factor of the RBF was lowered. This reduces the error in some of the points shown in the figures, however the error is increased in other

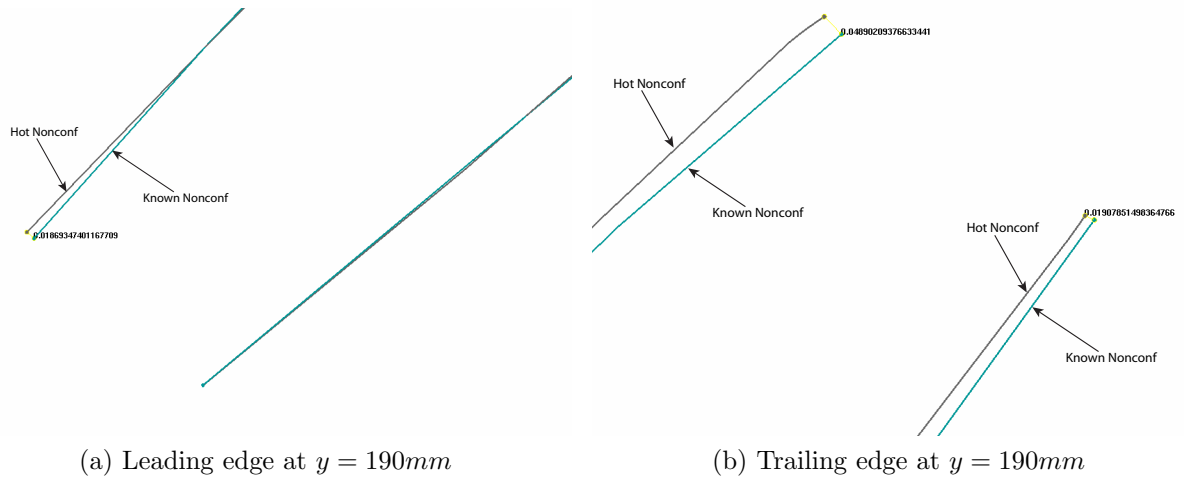


Figure 5.6: Difference between the obtained Hot Nonconf geometry and known nonconformance points at $y = 190mm$.

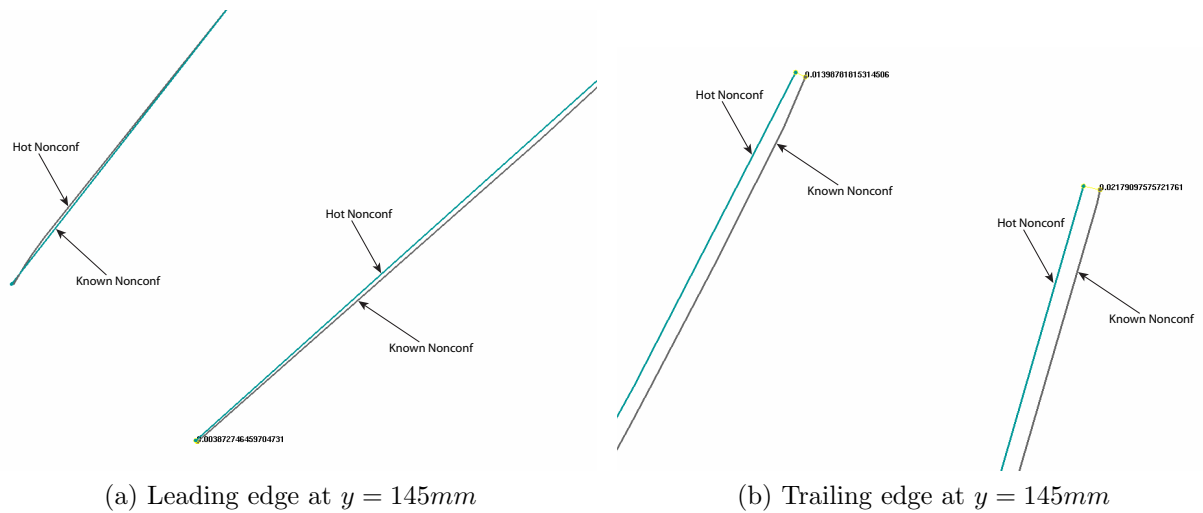


Figure 5.7: Difference between the obtained Hot Nonconf geometry and known nonconformance points at $y = 145mm$.

points and the surface of the blade is not built up in such a smooth way as in the first interpolation. In figure 5.8 this is demonstrated in the trailing edge of the 190-section. Compared to figure 5.6b the error is decreased in the known points, but a clear defect can be noticed between them on the pressure side. Figure 5.9 also shows the lack of surface smoothness when a smaller shape factor is used.

The fact that there are such large errors of the nonconformance interpolation close to the edges is worrying if aiming to reconstruct a geometry in a precise way. It can be argued that the coarse data provided of the nonconformances contributed to these errors, especially since there is no data of the deviations at the edges. Another possible reason, can be that when interpolating near the edges, some neighbours are picked from the opposite side of the blade, which in some cases represent displacements that differ quite much from the ones on the "correct" side. On the other hand this is to some extent needed to get a smooth transition across the radius.

Anyhow it is probably possible to achieve a more exact interpolation from the provided

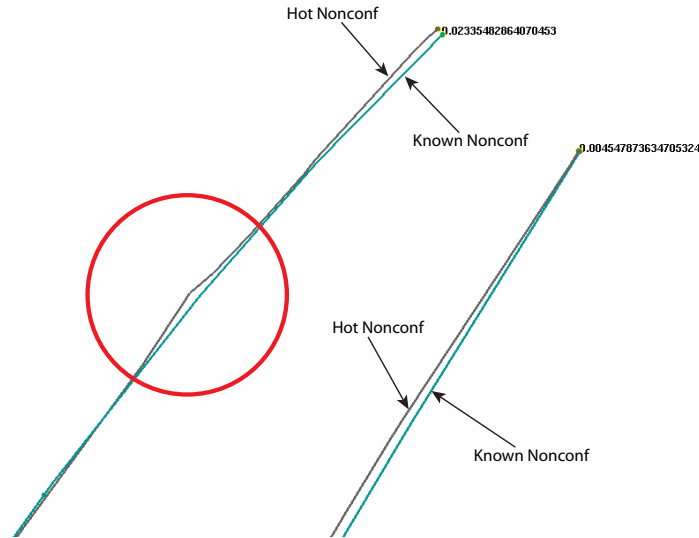


Figure 5.8: Difference between the obtained Hot Nonconf geometry and known nonconformance points at trailing edge and $y = 190\text{mm}$ when using a smaller shape factor.

data through further development of the RBF interpolation routine, by for example a more sophisticated sampling process or by trying other RBF's. Testing ideas in RBF interpolation is very time consuming and no further work has been put in to minimize the errors.

As stated above the errors are quite concerning if aiming for exact interpolation results. The other point of view that needs attention here is how the errors affects the goal to reconstruct a real geometry including manufacturing nonconformances. Measurements of four different rotor blades have been available and this weak supplement shows great variations of the nonconformances on the components. In figure 5.10 the nonconformances of the four available blades are plotted at three different coord-wise positions (leading edge, mid-coord and trailing edge), confirming the large spread of data.

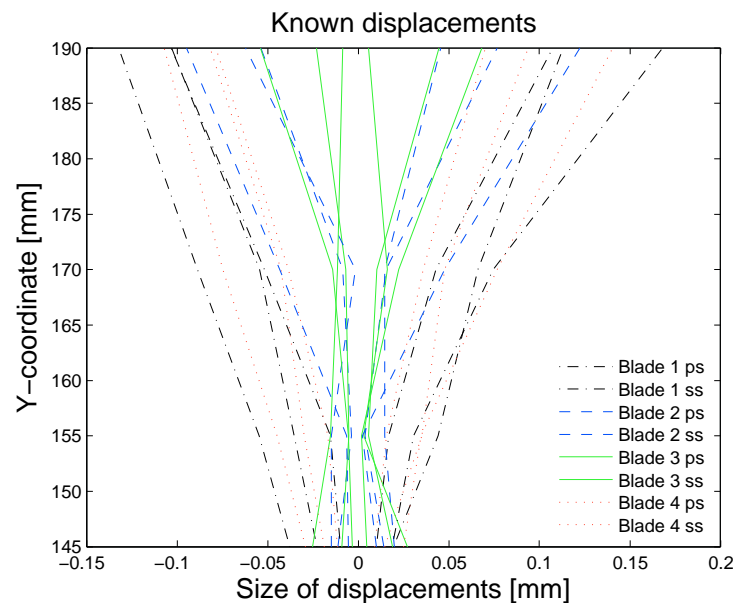


Figure 5.10: Size of nonconformance at leading edge, mid-coord and trailing edge on both sides of the four measured blades available. Pressure side to the left and suction side to the right.

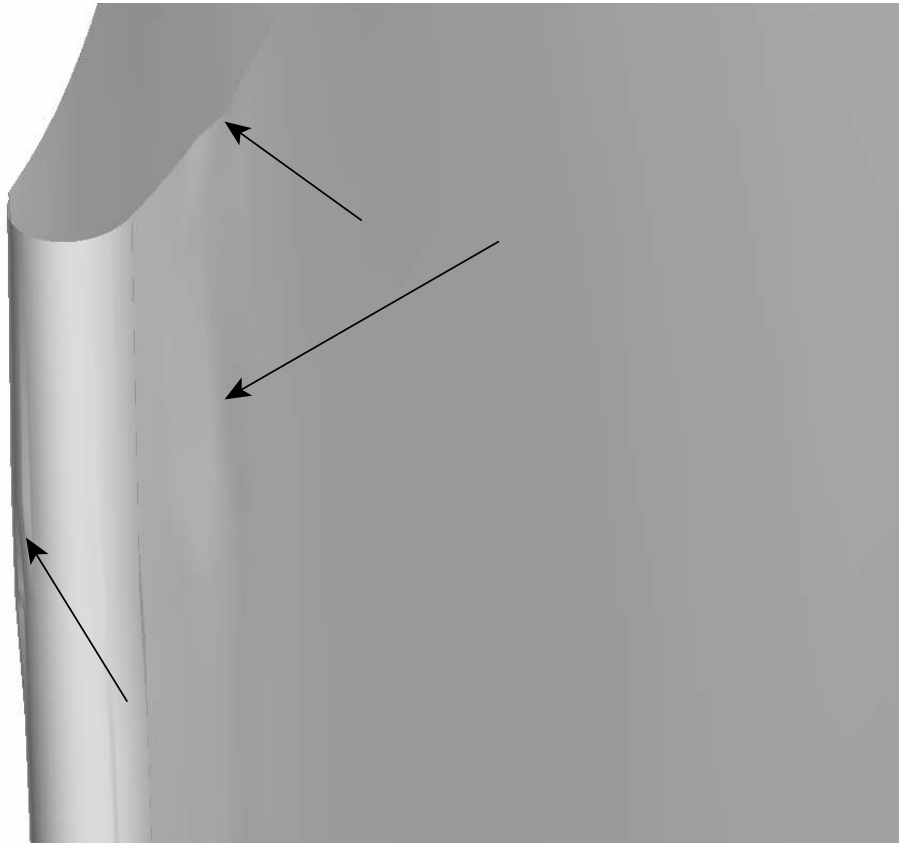


Figure 5.9: The smoothness of the blade surface is lost when aiming for a better accuracy through a smaller shape factor.

The nonconformance data that has been used to construct the Hot Nonconf geometry is the one denoted Blade 1. Seen in figure 5.10 is that this blade has the largest nonconformances of the four blades at the shown positions. This means that the Hot Nonconf geometry, in areas where the interpolation induces a larger deviation than the provided data, is overpredicting the size of the nonconformances. Taking this into account, the nonconformance interpolation has been sufficient to reconstruct the characteristics of a manufactured blade in most areas, but is overpredicting the effect particularly at the edges. It is also worth noticing, that even if the errors are large in percentage of the displacement in the point, the absolute size of the error is in the magnitude of $1 \cdot 10^{-2}$ to $5 \cdot 10^{-2}$ mm. This is of course not sustainable in the long run and preferably some of the earlier mentioned measures would be taken.

5.2 Difference between Design and Hot geometry

The Cold geometry has been obtained through transforming the Design geometry in a hot to cold process. Here this process has through FEA been performed in the other direction, i.e. cold to hot. Assuming that the RBF interpolation of the cold to hot step is accurate, it is possible to determine if any differences of the blade geometry are introduced in the hot to cold \rightarrow cold to hot computations. In the overall view of figure 5.11 it can be seen that the Design and Hot geometry in general represents the same surface. However, when taking a closer look it is possible to see that differences occur, especially at the edges. The differences are studied at the spans (hereafter called top and mid) shown in figure 5.12. Figure 5.13 and 5.14 show the Design and Hot geometries at the leading and trailing edge on these spans.

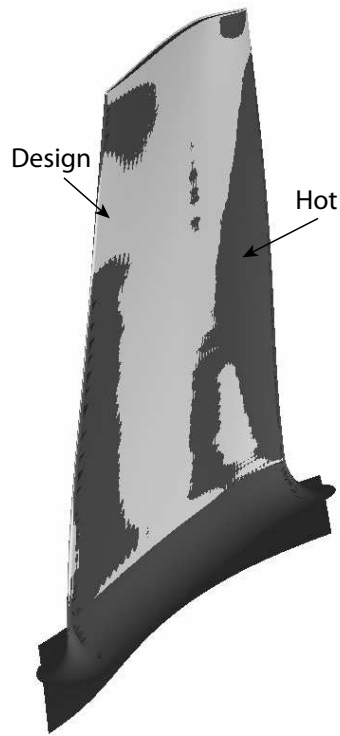


Figure 5.11: An overview of the Design and Hot geometries shows that they in general represent the same blade surface.

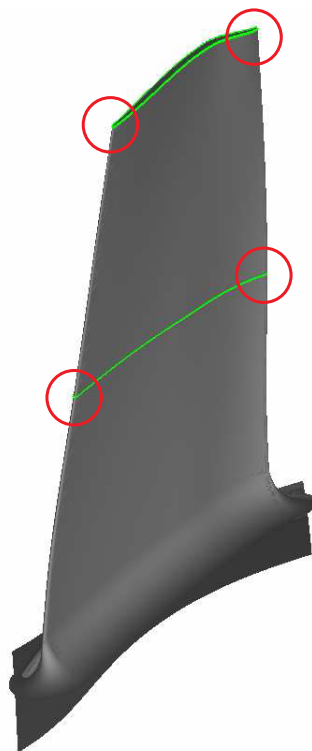


Figure 5.12: Differences between the Design and Hot geometries are investigated at marked locations.

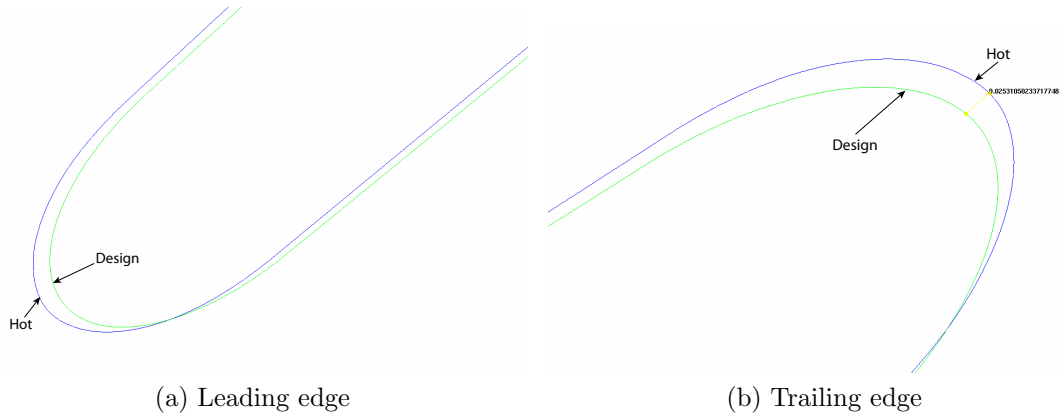


Figure 5.13: Difference between Hot geometry and Design geometry at the top span.

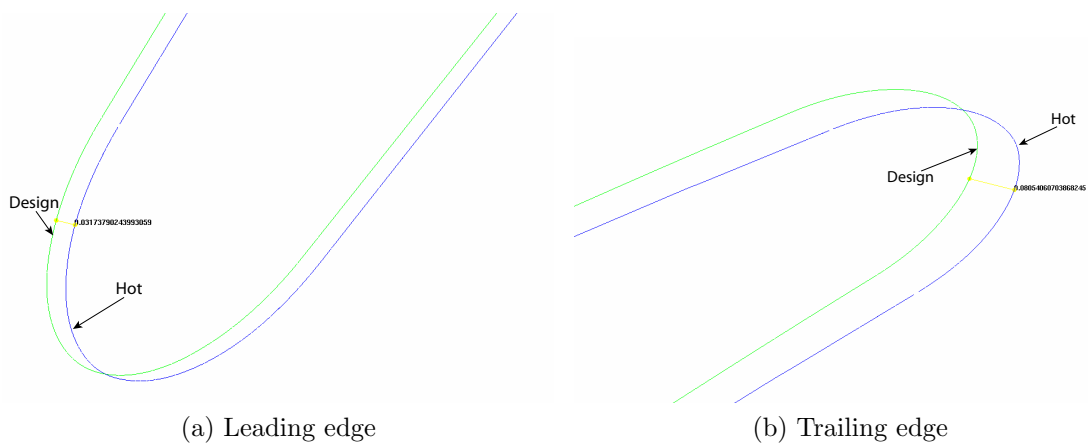


Figure 5.14: Difference between Hot geometry and Design geometry at the mid span.

The figures show that at the top, the Hot geometry is slightly less displaced, while this relation is reversed at the mid span. It also seems that the Hot geometry has a bit longer coord. It is hard to draw any overall conclusion of the relationship between the geometries from these figures or where their differences originate from. However, the size of the differences at the edges are in the magnitude of $1 \cdot 10^{-2}$ to $1 \cdot 10^{-1}$ mm, which is almost in the same region as the size of the largest nonconformances. It is difficult to draw any conclusions about changes of the overall performance of the rotor from the differences at the investigated spans. To further try to establish the differences between the two blade geometries, approximate stagger and blade angles are measured at the mid span using ICEM. The results can be seen in 5.1 and show that the hot to cold \rightarrow cold to hot process can introduce changes of these angles. It is reasonable to assume that this affects the area and thus mass flow through the blade passage. However the uncertainties of the measurements are large and no overall radial trends can be found.

Table 5.1: Stagger and blade angles of Hot and Design geometry at mid span

Geometry	Stagger, ξ	β at leading edge	β at trailing edge
Design	54.085	60.67	38.33
Hot	54.019	61.60	37.30

5.3 Passage area comparison

A geometry of a cold rotor blade was transformed using the method described in section 4, resulting in one Hot and one Hot Nonconf geometry. An interesting aspect of the transformation is to investigate how the passage area changes with changing blade surface. Using the equations in section 3.2, with a fillet radius of 4 mm, the area change from Design to Hot geometry due to fillets and tip clearance was estimated to 1.82 %. The change in area between Hot and Nonconf was calculated differently. The shortest distance between a number of spans for each geometry was calculated in Matlab and the corresponding position coordinates were imported into ICEM. These formed a surface between the blades from which the area could be calculated, see figure 5.15. This showed a 0.84 % reduction in area from Hot to Hot Nonconf.

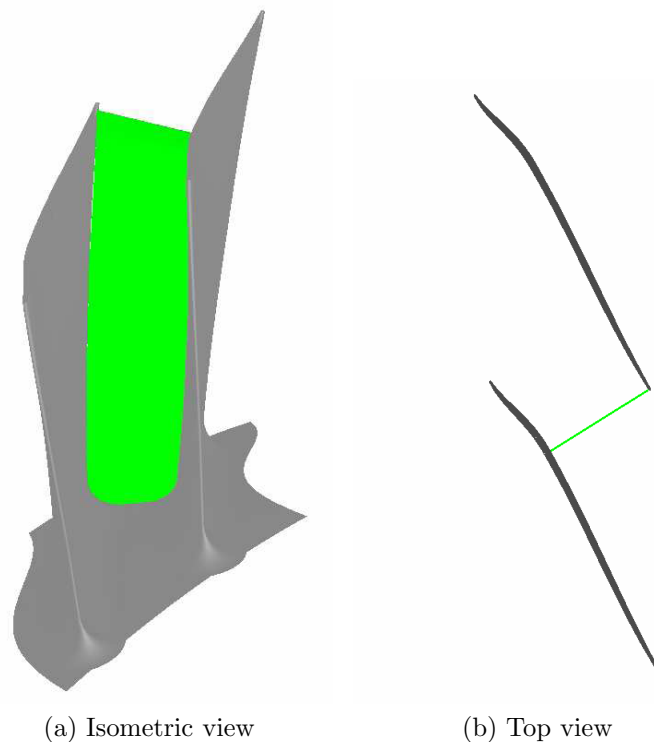


Figure 5.15: Surface placed in between two blades

5.4 Results from CFD simulations

5.4.1 Overall compressor characteristics.

Speedlines were obtained by varying the total to total pressure ratio $\frac{P_{02}}{P_{01}}$ at constant rotational speed, and plotting against corrected massflow, $\dot{m}_{corr} = \dot{m} \frac{P_{0,ref}}{P_0} \sqrt{\frac{T_0}{T_{0,ref}}}$. Such lines from the different simulations can be seen in figure 5.16, together with the compressor working line. The point where a speedline intersects with the working line is the design point and is where the compressor operates. From the figure it is clear that the Design geometry at its design point allows more mass flow through the blade passage, at a higher pressure ratio, than the Hot geometry. Since the mesh size and the turbulence model is the same for these cases, the difference is believed to be due to the change in area between the

blades caused by the fillet and tip clearance, see section 3.2. An interesting observation is that the simulation performed on the Hot geometry when using a Low Re mesh and k- ω SST predicts a lower mass flow capability. It is thus evident that the results are relatively sensitive to turbulence model and mesh density. The Hot Nonconf geometry has an even narrower passage than the Hot, explaining why the mass flow at the design point is lower than in the case of the Hot geometry. Corrected mass flow, pressure ratio and polytropic efficiency ($\eta_p = \frac{\gamma-1}{\gamma} \frac{\ln(P_{02}/P_{01})}{\ln(T_{02}/T_{01})}$) for each design point is presented in table 5.2

Table 5.2: Comparison of mass flow, pressure ratio and polytropic efficiency.

Geometry	\dot{m}_{corr}	Π_{TT}	η_p
Design	11.208	1.640	0.886
Hot WF	10.971	1.600	0.867
Hot	11.036	1.611	0.877
Hot Nonconf	10.978	1.601	0.870

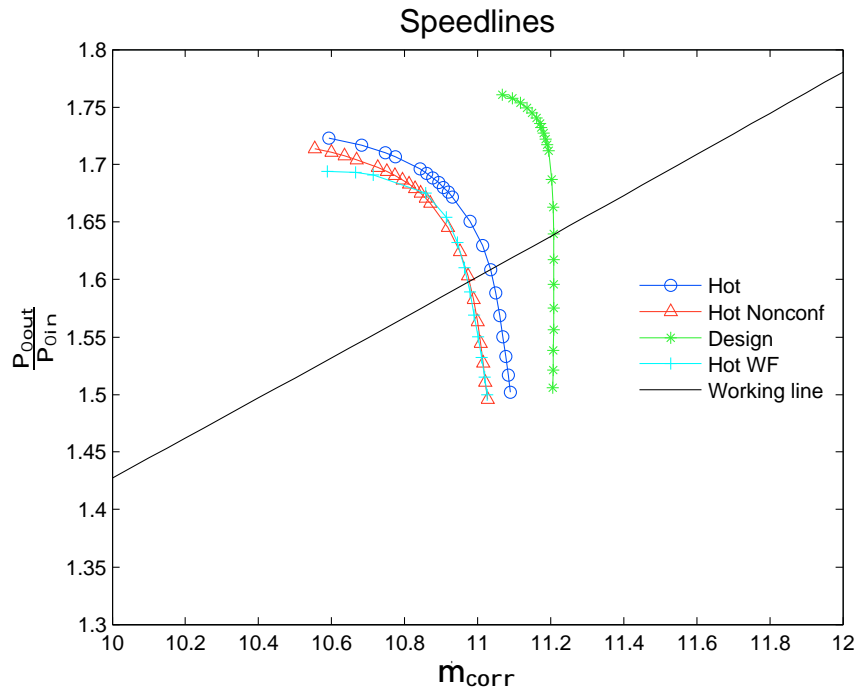


Figure 5.16: Speedlines for the different simulations.

The fractional reduction in corrected massflow between the different geometries was calculated and is in table 5.3 compared to the values of passage area reduction calculated in section 5.3. The numbers do not exactly correlate, but it is safe to say that the largest part

Table 5.3: Change in massflow compared to area reduction.

Mass flow reduction	Area reduction
$\frac{\dot{m}_{corr,Design} - \dot{m}_{corr,Hot WF}}{\dot{m}_{corr,Design}} = 2.26\%$	$\frac{A_{Design} - A_{Hot}}{A_{Design}} = 1.8\%$
$\frac{\dot{m}_{corr,Hot} - \dot{m}_{corr,Hot Nonconf}}{\dot{m}_{corr,Hot}} = 0.52\%$	$\frac{A_{Hot} - A_{Hot Nonconf}}{A_{Hot}} = 0.8\%$

of the mass flow reduction could be explained by the corresponding change in passage area. The resources available do not allow for an exact calculation of the reduced blade passage

area. However, as mentioned in section 5.2 stagger and blade angles also can have an effect on the area, possibly explaining the remaining mass flow reduction. Clearly there is a difference in results when switching between turbulence models, but the two comparisons are performed on cases with identical simulation settings.

5.4.2 Stall margins

Each speedline was achieved by ramping up the outlet static pressure until the simulations diverged and the compressor considered to be stalling. The stall margin for each point was calculated according to section 3.9 and displayed in table 5.4. It can be seen that the largest difference is between the Design and the Hot WF geometry, where the geometry changes causes an almost 5 % increase in stall margin. The Hot and Hot Nonconf geometries shows similar values.

Table 5.4: Stall margin for each design point.

Geometry	Stall margin
Design	23.08 %
Hot WF	27.99 %
Hot	33.00 %
Hot Nonconf	33.17 %

5.4.3 Profiles from rake positions

Profiles of design point total pressure and total temperature at the positions of experimental measurements after the rotor (rake 2) can be seen in figure 5.17. Since the geometries are evaluated at different pressure ratios, naturally the pressure profiles are not coinciding. However, the radial shape distribution seems to be unchanged between the Hot and Hot Nonconf blade. Values of mass flow average of total pressure and total temperature taken at the cross section of rake 2 can be seen in table 5.5. The fact that more losses occur in the Hot Nonconf case than in the Hot is clear since T_0 is almost identical but P_0 is considerably lower for the Hot Nonconf geometry, which could also be seen in figure 5.17.

Table 5.5: Mass flow averaged total pressure and total temperature at rake 2

Geometry	P_0	T_0
Design	2.605×10^5 Pa	409.2 K
Hot	2.562×10^5 Pa	407.5 K
Hot Nonconf	2.545×10^5 Pa	407.1 K

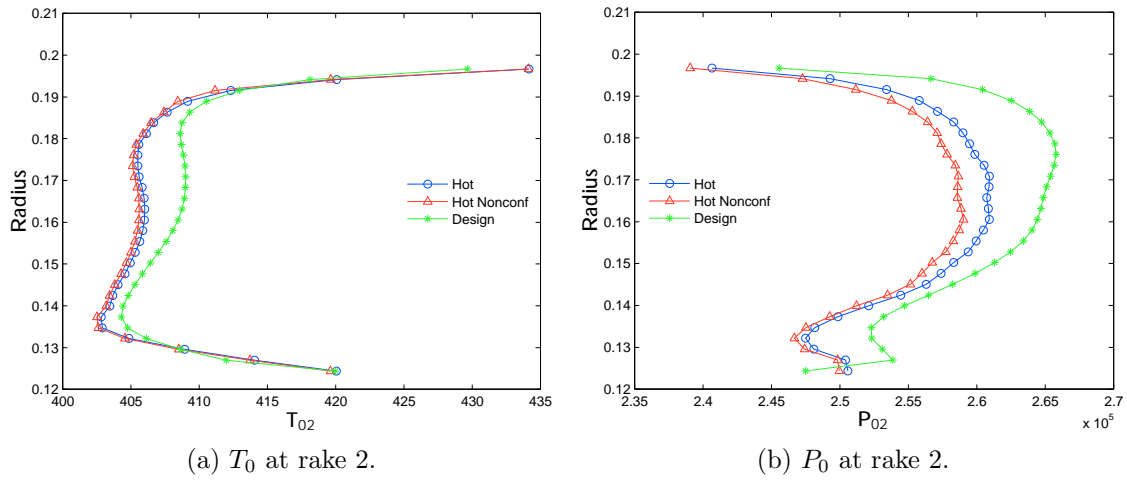


Figure 5.17: Profiles of total temperature and total pressure at the positions of the measurement rakes.

6 Conclusions and recommendations

With the available data in this project, the developed routine performs well in the cold to hot transformation of the rotor blade. However, the results are less reliable for the nonconformance case. Experience acquired during this work indicate that nonconformance data of finer resolution would help in obtaining a more reliable result.

It is clear from section 5.4 that several factors come into play when moving from Design geometry $k - \varepsilon$ simulations to Low Re $k - \omega$ simulations of blades with real geometry effects. The lack of fillets and tip clearance in the Design geometry seems to play a big role in why there is a difference, especially when looking at the speedlines in section 5.4.1. The pressure rise is lower for the Hot geometry, giving a lower polytropic efficiency, which has been seen in previous studies regarding the effects of fillets [12]. The further reduction of passage area due to nonconformances yield a further decrease in mass flow. However, the difference seen in section 5.2 between the Design surface and the Hot surface might also contribute to the results. It is reasonable to believe that the net effect of these discrepancies is a change of choke area and thus will also affect the mass flow. Further studies of this need to be done to be able to quantify the impact. A $k - \omega$ SST simulation of the Design geometry using a Low Re mesh would be of interest to see how much the turbulence model affects the outcome.

Further work will be performed to be able to identify the reasons why the preliminary CFD analysis greatly overestimates the efficiency through the second stage of Blenda.

References

- [1] Tatari, M., 2010, “A localized interpolation method using radial basis functions”, *World Academy of Science, Engineering and Technology*, Vol. 69, pp. 498–503.
- [2] Schaback, R., 2007, A practical guide to radial basis functions. <http://num.math.uni-goettingen.de/schaback/teaching/sc.pdf>. Chapter of future book.
- [3] Carr, J., Beatson, R., Cherrie, J., Mitchell, T., Fright, W., McCallun, B., and Evans, T., 2001, “Reconstruction and representation of 3d objects with radial basis functions”, *SIGGRAPH '01 Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, Vol. 1, pp. 67–78.
- [4] Østergaard Noe, K., 2009, Noe’s tutorial on deforming 3d geometry using rbf’s. <http://cg.alexandra.dk/tag/radial-basis-function/>.
- [5] Buhmann, M., 2010, “Radial basis function”, *Scholarpedia*, 5(5), p. 9837.
- [6] Sturgill, D., 2009, “Variable shape parameter strategies in radial basis function methods”. Master’s thesis, Marshall University, Huntington, West Virginia, USA.
- [7] Sarra, S. A., 2006, “Integrated multiquadric radial basis function approximation methods”, *Computers and Mathematics with Applications*, Vol. 51(8), p. 1283–1296.
- [8] Instruction for use of icem-scripts for blade-mesh generation, volvo report vols:10038042.
- [9] Ellbrant, L., Eriksson, L.-E., and Mårtensson, H., 2011, “Cfd validation of a high speed transonic 3.5 stage axial compressor”, *ISABE Conference(ISABE-2011-1226)*.
- [10] Cohen, H., and Saravanamuttoo, H., 2001, *Gas turbine Theory, Fifth edition*. Pearson Education Limited, Harlow.
- [11] Ansys, 2009, Ansys cfx, release 12.1, *ANSYS Inc*.
- [12] Engel, K., Nürnberger, D., Kügeler, E., and Weber, A., 2008, “Influence of blade fillets on the performance of a 15 stage gas turbine compressor”, *ASME paper no. GT2008-50748*.

Appendix A

Structure of the tin file used.

```
// tetin file version 1.1
// Tetin file generated by pstotin version 12.1 (Build 9.11.2.8) on Linux 2.6.36-02063602-generic
// Parasolid Kernel Version 21.0 (Build 256)
//
// written by ICEM CFD - version 12.1 on Fri Mar 11 13:09:15 2011
// Tetin file generated by pstotin version 12.1 (Build 9.11.2.8) on Linux 2.6.18-194.11.4.el5
// Parasolid Kernel Version 21.0 (Build 256)
//
// written by ICEM CFD - version 12.1 on Tue Apr 26 11:18:32 2011

set_triangulation_tolerance 0.001
define_family SURFACE color 5962547

define_surface name F_1081.35 family SURFACE_DEFORMED tetra_size 1e+10
unstruct_mesh n_points 28678 n_triangles 54654
445.40909161 137.02604776 -14.45103680
445.42827419 136.90807275 -14.42451814
445.40584962 136.85631831 -14.45781460
471.49623898 141.31088289 7.88148539
471.39843793 141.28513722 7.82190118
471.29888936 141.25851472 7.76505037
471.19769802 141.23095500 7.71104513
.
.
.
.
0 1 2
0 2 3
3 2 4
3 4 5
5 4 6
5 6 7
.
.
.
.
affix 0
# the periodic angle is 360/42 = 8.5714282989502
periodic 0 0 0 1 0 0 per_n=42
define_model 1e+10 reference_size 1 units mm
define_prism_meshing_parameters law 0 nlay 3 hratio 1.2
return
```

Appendix B

1. Mesh Report

Table 1. Mesh Information for Configuration07_001

Domain Nodes Elements
ROTOR2 62700 56832
STATOR1 70356 64000
STATOR2 70356 64000
All Domains 203412 184832

2. Physics Report

Table 2. Domain Physics for Configuration07_001

Domain - ROTOR2
Type Fluid
Location FLUID
Materials
Air Ideal Gas
 Fluid Definition Material Library
 Morphology Continuous Fluid
Settings
Buoyancy Model Non Buoyant
Domain Motion Rotating
 Alternate Rotation Model Off
 Angular Velocity -2.2456e+04 [rev min⁻¹]
 Axis Definition Coordinate Axis
 Rotation Axis Coord 0.1
Reference Pressure 0.0000e+00 [atm]
Heat Transfer Model Total Energy
 Include Viscous Work Term On
Turbulence Model SST
 Production Limiter Kato Launder
Turbulent Wall Functions Automatic
Domain - STATOR1
Type Fluid
Location Primitive 3D 2
Materials
Air Ideal Gas
 Fluid Definition Material Library
 Morphology Continuous Fluid
Settings
Buoyancy Model Non Buoyant
Domain Motion Stationary
Reference Pressure 0.0000e+00 [atm]
Heat Transfer Model Total Energy
 Include Viscous Work Term On
Turbulence Model SST
 Production Limiter Kato Launder
Turbulent Wall Functions Automatic
Domain - STATOR2
Type Fluid
Location Primitive 3D
Materials
Air Ideal Gas
 Fluid Definition Material Library
 Morphology Continuous Fluid
Settings
Buoyancy Model Non Buoyant
Domain Motion Stationary
Reference Pressure 0.0000e+00 [atm]
Heat Transfer Model Total Energy
 Include Viscous Work Term On
Turbulence Model SST
 Production Limiter Kato Launder
Turbulent Wall Functions Automatic
Domain Interface - InterfaceR2Per
Boundary List1 InterfaceR2Per Side 1
Boundary List2 InterfaceR2Per Side 2
Interface Type Fluid Fluid
Settings
Interface Models Rotational Periodicity
 Axis Definition Coordinate Axis
 Rotation Axis Coord 0.1
Mesh Connection Direct
Domain Interface - InterfaceR2S2
Boundary List1 InterfaceR2S2 Side 1
Boundary List2 InterfaceR2S2 Side 2
Interface Type Fluid Fluid
Settings
Interface Models General Connection
 Frame Change Stage
 Pressure Profile Decay 5.0000e-02
Pitch Change Specified Pitch Angles
 Pitch Angle Side1 8.5714e+00 [degree]
 Pitch Angle Side2 4.5000e+00 [degree]
Mesh Connection GGI
Domain Interface - InterfaceS1Per
Boundary List1 InterfaceS1Per Side 1
Boundary List2 InterfaceS1Per Side 2
Interface Type Fluid Fluid
Settings
Interface Models Rotational Periodicity
 Axis Definition Coordinate Axis
 Rotation Axis Coord 0.1
Mesh Connection Automatic
Domain Interface - InterfaceS1R2
Boundary List1 InterfaceS1R2 Side 1
Boundary List2 InterfaceS1R2 Side 2
Interface Type Fluid Fluid

```

Settings
Interface Models General Connection
  Frame Change Stage
  Pressure Profile Decay 5.0000e-02
Pitch Change Specified Pitch Angles
  Pitch Angle Side1 4.7368e+00 [degree]
  Pitch Angle Side2 8.5714e+00 [degree]
Mesh Connection GGI
Domain Interface - InterfaceS2Per
Boundary List1 InterfaceS2Per Side 1
Boundary List2 InterfaceS2Per Side 2
Interface Type Fluid Fluid
Settings
Interface Models Rotational Periodicity
  Axis Definition Coordinate Axis
  Rotation Axis Coord 0.1
Mesh Connection Automatic

```

Table 3. Boundary Physics for Configuration07_001

```

Domain Boundaries
R0TOR2 Boundary - InterfaceR2Per Side 1
Type INTERFACE
Location periodic_sh
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceR2Per Side 2
Type INTERFACE
Location PER 1
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceR2S2 Side 1
Type INTERFACE
Location MIX 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS1R2 Side 2
Type INTERFACE
Location MIX 1
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - Blade
Type WALL
Location SURFACE_HOT
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - Hub
Type WALL
Location HUB
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - Shroud
Type WALL
Location SHROUD
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Velocity Counter Rotating Wall
Wall Roughness Smooth Wall
STATOR1 Boundary - Inlet
Type INLET
Location inlet
Settings
Flow Direction Cylindrical Components
  Unit Vector Axial Component Bcexpmatched.vx(radii)
  Unit Vector Theta Component Bcexpmatched.vt(radii)
  Unit Vector r Component Bcexpmatched.vr(radii)
  Axis Definition Coordinate Axis
  Rotation Axis Coord 0.1
Flow Regime Subsonic
Heat Transfer Total Temperature
  Total Temperature Bcexpmatched.t0(radii)
Mass And Momentum Total Pressure
  Relative Pressure Bcexpmatched.p0(radii)
Turbulence k and Epsilon
  Epsilon Bcexpmatched.epsilon(radii)
  k Bcexpmatched.k(radii)
Boundary - InterfaceS1Per Side 1
Type INTERFACE
Location Primitive 2D 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS1Per Side 2
Type INTERFACE
Location Primitive 2D A 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux

```



```

Boundary - InterfaceS1R2 Side 1
Type INTERFACE
Location outlet
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - BladeS1
Type WALL
Location blade
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - HubS1
Type WALL
Location hub
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - ShroudS1
Type WALL
Location shroud
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
STATOR2 Boundary - InterfaceR2S2 Side 2
Type INTERFACE
Location inlet 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS2Per Side 1
Type INTERFACE
Location Primitive 2D 3
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS2Per Side 2
Type INTERFACE
Location Primitive 2D A 3
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - Outlet
Type OUTLET
Location outlet 2
Settings
Flow Regime Subsonic
Mass And Momentum Average Static Pressure
    Pressure Profile Blend 5.0000e-02
    Relative Pressure 1.9000e+05 [Pa]
Pressure Averaging Average Over Whole Outlet
Boundary - BladeS2
Type WALL
Location blade 2
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - HubS2
Type WALL
Location hub 2
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - ShroudS2
Type WALL
Location shroud 2
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall

```

CFX settings for a LowRe simulation

1. Mesh Report

Table 1. Mesh Information for Configuration07_001

```

Domain Nodes Elements
ROTOR2 2136331 2080516
STATOR1 1174808 1137760
STATOR2 1174808 1137760
All Domains 4485947 4356036

```

2. Physics Report

Table 2. Domain Physics for Configuration07_001

```

Domain - ROTOR2
Type Fluid

```

```

Location FLUID
Materials
Air Ideal Gas
    Fluid Definition Material Library
    Morphology Continuous Fluid
Settings
Buoyancy Model Non Buoyant
Domain Motion Rotating
    Alternate Rotation Model Off
    Angular Velocity -2.2456e+04 [rev min^-1]
    Axis Definition Coordinate Axis
    Rotation Axis Coord 0.1
Reference Pressure 0.0000e+00 [atm]
Heat Transfer Model Total Energy
    Include Viscous Work Term On
Turbulence Model SST
    Production Limiter Kato Launder
Turbulent Wall Functions Automatic
Domain - STATOR1
Type Fluid
Location Primitive 3D 2
Materials
Air Ideal Gas
    Fluid Definition Material Library
    Morphology Continuous Fluid
Settings
Buoyancy Model Non Buoyant
Domain Motion Stationary
Reference Pressure 0.0000e+00 [atm]
Heat Transfer Model Total Energy
    Include Viscous Work Term On
Turbulence Model SST
    Production Limiter Kato Launder
Turbulent Wall Functions Automatic
Domain - STATOR2
Type Fluid
Location Primitive 3D
Materials
Air Ideal Gas
    Fluid Definition Material Library
    Morphology Continuous Fluid
Settings
Buoyancy Model Non Buoyant
Domain Motion Stationary
Reference Pressure 0.0000e+00 [atm]
Heat Transfer Model Total Energy
    Include Viscous Work Term On
Turbulence Model SST
    Production Limiter Kato Launder
Turbulent Wall Functions Automatic
Domain Interface - InterfaceR2Per
Boundary List1 InterfaceR2Per Side 1
Boundary List2 InterfaceR2Per Side 2
Interface Type Fluid Fluid
Settings
Interface Models Rotational Periodicity
    Axis Definition Coordinate Axis
    Rotation Axis Coord 0.1
Mesh Connection Direct
Domain Interface - InterfaceR2S2
Boundary List1 InterfaceR2S2 Side 1
Boundary List2 InterfaceR2S2 Side 2
Interface Type Fluid Fluid
Settings
Interface Models General Connection
    Frame Change Stage
    Pressure Profile Decay 5.0000e-02
Pitch Change Specified Pitch Angles
    Pitch Angle Side1 8.5714e+00 [degree]
    Pitch Angle Side2 4.5000e+00 [degree]
Mesh Connection GGI
Domain Interface - InterfaceS1Per
Boundary List1 InterfaceS1Per Side 1
Boundary List2 InterfaceS1Per Side 2
Interface Type Fluid Fluid
Settings
Interface Models Rotational Periodicity
    Axis Definition Coordinate Axis
    Rotation Axis Coord 0.1
Mesh Connection Automatic
Domain Interface - InterfaceS1R2
Boundary List1 InterfaceS1R2 Side 1
Boundary List2 InterfaceS1R2 Side 2
Interface Type Fluid Fluid
Settings
Interface Models General Connection
    Frame Change Stage
    Pressure Profile Decay 5.0000e-02
Pitch Change Specified Pitch Angles
    Pitch Angle Side1 4.7368e+00 [degree]
    Pitch Angle Side2 8.5714e+00 [degree]
Mesh Connection GGI
Domain Interface - InterfaceS2Per
Boundary List1 InterfaceS2Per Side 1
Boundary List2 InterfaceS2Per Side 2
Interface Type Fluid Fluid
Settings
Interface Models Rotational Periodicity
    Axis Definition Coordinate Axis
    Rotation Axis Coord 0.1
Mesh Connection Automatic

```

Table 3. Boundary Physics for Configuration07_001

```

Domain Boundaries
ROTOR2 Boundary - InterfaceR2Per Side 1
Type INTERFACE
Location periodic_sh
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceR2Per Side 2
Type INTERFACE
Location PER 1
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceR2S2 Side 1
Type INTERFACE
Location MIX 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS1R2 Side 2
Type INTERFACE
Location MIX 1
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - Blade
Type WALL
Location "SURFACE_HOT, TIP"
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - Hub
Type WALL
Location HUB
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - Shroud
Type WALL
Location SHROUD
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Velocity Counter Rotating Wall
Wall Roughness Smooth Wall
STATOR1 Boundary - Inlet
Type INLET
Location inlet
Settings
Flow Direction Cylindrical Components
Unit Vector Axial Component Bcexpmatched.vx(radii)
Unit Vector Theta Component Bcexpmatched.vt(radii)
Unit Vector r Component Bcexpmatched.vr(radii)
Axis Definition Coordinate Axis
Rotation Axis Coord 0.1
Flow Regime Subsonic
Heat Transfer Total Temperature
Total Temperature Bcexpmatched.t0(radii)
Mass And Momentum Total Pressure
Relative Pressure Bcexpmatched.p0(radii)
Turbulence k and Epsilon
Epsilon Bcexpmatched.epsilon(radii)
k Bcexpmatched.k(radii)
Boundary - InterfaceS1Per Side 1
Type INTERFACE
Location Primitive 2D 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS1Per Side 2
Type INTERFACE
Location Primitive 2D A 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS1R2 Side 1
Type INTERFACE
Location outlet
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - BladeS1
Type WALL
Location blade
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - HubS1
Type WALL
Location hub
Settings
Heat Transfer Adiabatic

```

```

Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - ShroudS1
Type WALL
Location shroud
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
STATOR2 Boundary - InterfaceR2S2 Side 2
Type INTERFACE
Location inlet 2
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS2Per Side 1
Type INTERFACE
Location Primitive 2D 3
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - InterfaceS2Per Side 2
Type INTERFACE
Location Primitive 2D A 3
Settings
Heat Transfer Conservative Interface Flux
Mass And Momentum Conservative Interface Flux
Turbulence Conservative Interface Flux
Boundary - Outlet
Type OUTLET
Location outlet 2
Settings
Flow Regime Subsonic
Mass And Momentum Average Static Pressure
    Pressure Profile Blend 5.0000e-02
    Relative Pressure 1.9000e+05 [Pa]
Pressure Averaging Average Over Whole Outlet
Boundary - BladeS2
Type WALL
Location blade 2
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - HubS2
Type WALL
Location hub 2
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall
Boundary - ShroudS2
Type WALL
Location shroud 2
Settings
Heat Transfer Adiabatic
Mass And Momentum No Slip Wall
Wall Roughness Smooth Wall

```