Process Integration and Automated Multi-Objective Optimization Supporting Aerodynamic Compressor Design

Von der Fakultät für Maschinenbau, Elektrotechnik und
Wirtschaftsingenieurwesen der
Brandenburgischen Technischen Universität Cottbus
zur Erlangung des akademischen Grades eines Doktor-Ingenieurs genehmigte

Dissertation

vorgelegt von

Dipl.-Ing. Akin Keskin

geboren am 11.09.1974 in Berlin

Vorsitzender:	Prof. Dr.-Ing. Arnold Kühhorn
Gutachter:	Prof. Dr.-Ing. habil. Dieter Bestle
Gutachter:	Prof. Dr.-Ing. Christoph Egbers

Tag der mündlichen Prüfung: 30. November 2006

Berichte aus der Luft- und Raumfahrttechnik

Akin Keskin

Process Integration and Automated Multi-Objective Optimization Supporting Aerodynamic Compressor Design

Shaker Verlag
Aachen 2007

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Zugl.: Cottbus, BTU, Diss., 2006

Printed in Germany.

ISBN 978-3-8322-5875-7
ISSN 0945-2214

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen
Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9
Internet: www.shaker.de • e-mail: info@shaker.de

Acknowledgements

This thesis results from a three years work as a research assistant at the Chair of Engineering Mechanics and Vehicle Dynamics at the Brandenburg University of Cottbus within an industrial collaborative research project with the Rolls-Royce Deutschland Company.

First of all, I would like to thank my thesis advisor and reviewer Prof. Dieter Bestle for his guidance and contribution to this research, for the many fruitful discussions, his constant encouragement, and his confidence in my work. Gratitude goes also to Prof. Kühhorn for being the chairman of the examination board of this thesis and many thanks to Prof. Egbers, who kindly agreed to be part of the board of examiners and reviewed the present thesis.

Furthermore, I wish to thank the members of the Chair of Engineering Mechanics and Vehicle Dynamics for the brilliant working atmosphere, the constructive and open discussions in our many seminars. Special thanks to my room mate Dierk Otto and my friend Amit Kumar Dutta for the important teamwork and discussions within our research project and for proof-reading of my thesis.

I would like to thank all colleagues from the Rolls-Royce company for supporting this research project and making this thesis possible. Gratitude goes to Dr. Helmut Richter, who motivated me to do a dissertation, the project leader Dr. Marius Swoboda for his support and guidance, Dr. André Huppertz for the useful discussions and the adaption of the programs *Parablading* and *Mises* for my special purposes.

Finally, I wish to thank my parents and my wife Neslihan for their great support and understanding in all positive and negative aspects of doing this thesis.

Berlin, December 2006 *A. Keskin*

Abstract

Process Integration and Automated Multi-Objective Optimization Supporting Aerodynamic Compressor Design

Akin Keskin

keywords: compressor design, aerodynamics, multi-objective optimization, process integration

Nowadays industrial aerodynamic compressor design is based on mature computer programs developed during several decades. State of the art is to split the complex design process into subsequent design subtasks which are solved by different experts via time-consuming parameter studies. Isolated design of subproblems based on human intuition, however, will result in sub-optimal solutions only. Due to the increasing demand on higher aero engine performance and design cycle time reduction the aspects of process integration and automation as well as numerical optimization become more and more important in today's aerodynamic compressor design.

The intention of this work is to show how process integration and optimization can be used efficiently to support engineering design work in optimal solution finding. Since the aerodynamic compressor design is characterized by many design parameters, multiple constraints and contradicting objectives, multi-objective optimization is used to find Pareto-optimal solutions from which the design engineer can choose trade-offs for his particular design problem. The improvements in terms of process acceleration and design optimization are demonstrated for three selected, but typical industrial engineering design tasks required in three different design phases of the aerodynamic compressor design process, namely preliminary design, throughflow off-design, and blading procedure.

Kurzfassung

Prozessintegration und automatisierte Mehrkriterien-Optimierung zur Unterstützung des aerodynamischen Verdichterentwurfs

Akin Keskin

Schlüsselwörter: Verdichterauslegung, Aerodynamik, Mehrkriterien-Optimierung, Prozessintegration

Der aerodynamische Verdichterentwurf wird heutzutage in der Industrie mit Hilfe von ausgereiften Computerprogrammen durchgeführt, die über Jahrzehnte entwickelten wurden. Stand der Technik ist es, den komplexen Entwurfsprozess in mehrere einzelne Entwurfsaufgaben aufzuteilen, welche durch zeitaufwändige Parameterstudien von unterschiedlichen Experten gelöst werden. Ein isolierter Entwurf basierend auf menschlicher Intuition führt jedoch nur zu sub-optimalen Lösungen. Auf Grund der ansteigenden Anforderung an die Leistung eines Flugtriebwerks und der Reduzierung der Entwicklungszeiten gewinnen die Aspekte der Prozessintegration und -automatisierung als auch der numerischen Optimierung in dem heutigen Verdichterentwurfsprozess an stärkerer Bedeutung.

Die Intention dieser Arbeit ist es, Möglichkeiten aufzuzeigen, wie Prozessintegration und Optimierung effizient genutzt werden können, um die Entwurfsaufgabe des Ingenieurs durch automatische Lösungssuche zu unterstützen. Da der aerodynamische Verdichterentwurfsprozess durch eine Vielzahl von Entwurfsparametern, mehreren Nebenbedingungen und gegensätzlichen Entwurfszielen charakterisiert ist, wird die Mehrkriterien-Optimierung zum Auffinden Pareto-optimaler Lösungen verwendet, von denen der Entwurfsingenieur Kompromisslösungen für seine spezielle Entwurfsaufgabe auswählen kann. Anhand von drei ausgewählten, typisch industriellen Entwurfsaufgaben aus drei unterschiedlichen Entwurfsphasen der aerodynamischen Verdichterauslegung wie der Mittelschnittsrechnung, des Stromlinienkrümmungsverfahrens sowie des Schaufelentwurfs werden verbesserte Ergebnisse in Bezug auf Prozessbeschleunigung und optimierten Entwurf demonstriert.

Contents

Nomenclature

Roman Symbols

A	area
B	Bernstein polynomial
BL	blockage
C	chord length, continuity
c	absolute velocity
C_h	enthalphy-equivalent static pressure rise coefficient
$\Delta\mathcal{D}$	additional exit whirl angle
$\mathcal{F}$	attainable objective space
F	compound function, non-dominated front
f	function
FLF	flow function
g	inequality constraint
H	height, enthalpy, boundary layer shape factor
h	equality constraint
J	number of inequality constraints
K	number of equality constraints
L	camber line length, Lagrange function
l	length parameter
M	Mach number, number of objectives
m	polynomial degree
$\dot{m}$	mass flow
N	B-spline polynomial
n	normal coordinate, polynomial degree, number of parameters
N_b	number of blades

N_s	number of stages
$\mathcal{P}$	admissible design space
P	pressure
$PMXC$	position of maximum thickness
P_t	parent population
Q_t	offspring population
r	radius, radial coordinate
Re	Reynolds number
R_t	overall population
S	pitch
SM	surge margin
T	temperature, thickness
t	tangential coordinate, spline parameter
Tu	turbulence intensity
u	circumferential velocity
w	relative velocity, weighting
WR	working range
x	axial coordinate

Greek Symbols

α	flow angle, step size
β	metal angle
χ	pressure loss coefficient factor
δ	clearance
ε	accuracy
η	efficiency
γ	artificial objective, ratio of specific heat capacities
κ	curvature
λ	Lagrange multiplier
μ	wedge angle, Lagrange multiplier
ω	pressure loss coefficient
Π	pressure ratio
Ψ	stage loading

σ	stiffness, solidity $\sigma = C/S$
τ	tangential angle
ξ	stagger angle

Vectors and Matrices

$\mathbf{B}$	substitute of inverse Hessian matrix
$\mathbf{b}$	vector of control points
$\mathbf{d}$	search direction
$\mathbf{f}$	vector of objective functions
$\mathbf{G}$	matrix for search direction
$\mathbf{g}$	vector of inequality constraints
$\mathbf{H}$	Hessian matrix, $\mathbf{H} = \nabla^2 f$
$\mathbf{h}$	vector of equality constraints
$\mathbf{I}$	identity matrix
$\mathbf{K}$	knot vector
$\mathbf{p}$	vector of design parameters

Subscripts

$_0$	total
$_c$	compressor
$_{datum}$	of datum design
$_E$	exit
$_{eff}$	effective
$_{geom}$	geometric
$_I$	inlet
$_i$	stage
$_{isen}$	isentropic
$_{poly}$	polytropic
$_{WR}$	at working range

Superscripts

^{0}	design point
^{C}	casing

$^{\Delta}$	difference
$^{\cdot}$	derivative
^{H}	hub
^{L}	left
^{l}	lower
^{M}	mid-height
^{PS}	pressure side
^{R}	right
*	optimal
^{SS}	suction side
^{T}	transposed
^{u}	upper

Symbols

$^{-}$	averaged or related
Δ	difference
$^{\circ}$	degree
$^{\hat{}}$	bound
∇	Nabla operator $\nabla \bullet = [\partial \bullet / \partial p_1, \ldots, \partial \bullet / \partial p_n]^T$ $\nabla_f \bullet = [\partial \bullet / \partial f_1, \ldots, \partial \bullet / \partial f_M]^T$
$'$	relative frame
$^{\sim}$	artificial

Acronyms

ACARE	Advisory Council for Aeronautics Research in Europe
BFGS	Broyden-Fletcher-Goldfarb-Shanno Update Scheme
CFD	Computational Fluid Dynamics
DFP	Davidon-Fletcher-Powell Update Scheme
DoE	Design of Experiment
EA	Evolutionary Algorithm
ES	Evolutionary Strategy
GA	Genetic Algorithm
GP	Genetic Programming
HCF	High Cycle Fatigue
LCF	Low Cycle Fatigue
MCS	Monte-Carlo Simulation
MIGA	Multi-Island Genetic Algorithm
NACA	National Advisory Committee for Aeronautics
NLPQL	Nonlinear Programming with Quadratic Line Search
NSGA-II	Non-dominated Sorting Genetic Algorithm II
S1	Blade-to-Blade Surface
S2	Meridional Plane
SA	Simulated Annealing
SQP	Sequential Quadratic Programming

1 Introduction

The increasing globalization and the prosperity of the world population drives the air traffic to become more and more the major means of transportation. The trend shows that passenger kilometers will be doubled in the next 15 years, *Walther et al. (2000)*. However, this demand is in conflict to the global concern for resource preservation and reduction of energy consumption. The *Advisory Council for Aeronautics Research in Europe* (ACARE) published a proposal called "European Aeronautics: A Vision for 2020" pointing out several key elements, including noise and exhaust emission reduction, travel delays, and safer air transport, *ACARE (2001)*. In order to address these issues, aero engine companies need to improve the design technology of their products in terms of higher efficiency and less emissions.

The design of an aero engine is a highly complex and time-consuming multidisciplinary engineering task driven by many different objectives and requirements. Nowadays the overall design process is subdivided into component based subtasks, where different design tools in different disciplines are involved and used in order to fulfill the design targets, *Keskin and Bestle (2004)*. The overall performance that can be achieved for an aero engine is mainly given by the design quality of its components, namely the compressor, the combustor, and the turbine, Figure 1.1. Therefore, efforts are taken to improve the component design and to accelerate the time-consuming highly-iterative design process.

The compressor as one of the most important and challenging components within an aero engine is responsible for 50-60% of the engine length, 40-50% of its weight, and 35-40% of the manufacturing costs, *Steffens and Schäffler (2000)*. The highly complex and multi-disciplinary compressor design process is built up

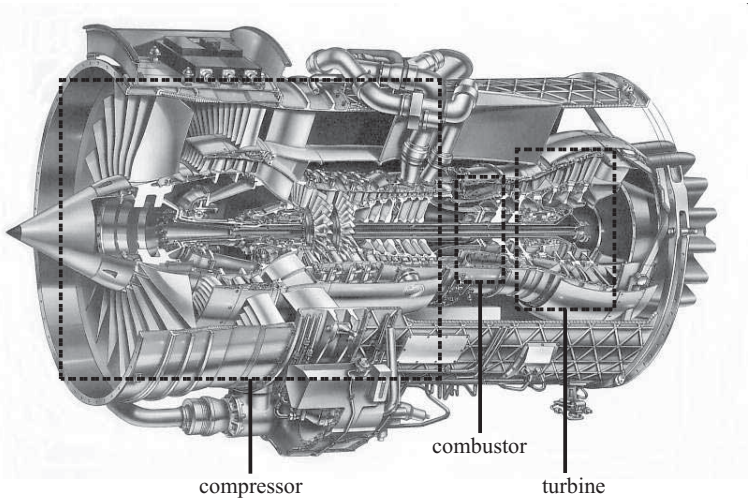


Figure 1.1: Major components of an aero engine demonstrated for Rolls-Royce BR715 (Printed by courtesy of Rolls-Royce Deutschland)

from several separate design phases, Figure 1.2. Starting with a conceptional study according to the market requirements a preliminary analysis of possible compressor designs is performed. The most promising design is chosen for a performance investigation where additional parameters for the subsequent multi-disciplinary design process are prescribed. This is followed by time-consuming inner iterations between the three main disciplines aerodynamics, design, and stress which are performed in order to find the best compressor design fulfilling all constraints and objectives of the design task. Finally, the process ends with the manufacturing process and the assembly with other components of the aero engine.

Aerodynamics plays a significant role in the whole compressor design process since it is the initial step of an iterative multi-disciplinary design procedure where the performance requirements are conducted into the inner design loop. The aerodynamic process itself basically consists of several design steps with increasing complexity and different design tools. The process is rather time-consuming due to many design iterations within and between these individual tools. Therefore,

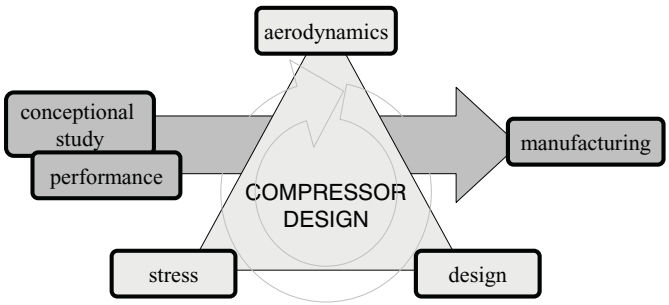


Figure 1.2: Compressor design process

an automation of the process is desired which would have a positive influence to the entire design task leading to shorter design cycles.

The aerodynamic compressor design process is based on technically sophisticated programs developed during several decades. State of the art is to split the individual design processes into several subtasks which are solved by different experts via time-consuming parameter studies. Isolated design of subproblems based on human intuition, however, will result in sub-optimal solutions only. The increasing demand on aero engine performance and design cycle time reduction requires process flow automation and design optimization. Therefore, process integration, process automation and numerical optimization become more and more important.

1.1 Aerodynamic Compressor Design

The aerodynamic compressor design process basically consists of meanline prediction calculation, throughflow calculation, and blading procedures, Figure 1.3. The complexity of the design model, i.e. the number of design parameters, increases during the process flow. In an industrial design process typically a new design always starts on the basis of an existing compressor design.

The meanline prediction is the first step within compressor design. It is a simple one-dimensional calculation of flow parameters along the mid-height line of the compressor where global parameters as the annulus geometry, the number

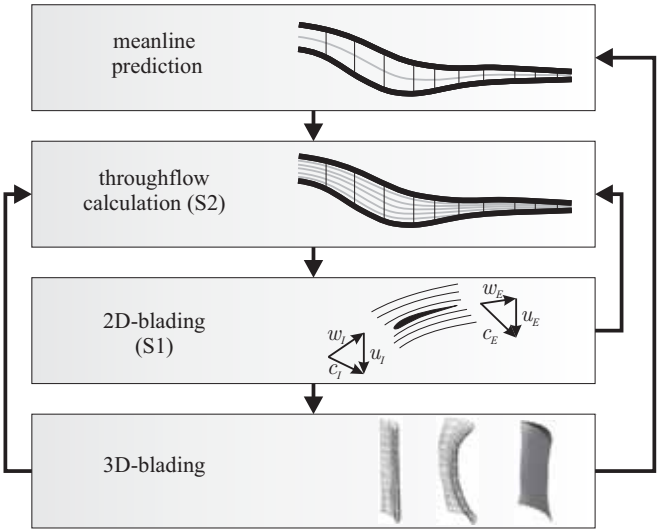


Figure 1.3: Aerodynamic compressor design process

of stages, and the stage pressure ratios are scaled or adapted to the new design problem. Based on the velocity triangles, flow parameters as flow velocities, flow angles, pressure and temperature values are determined at specific axial positions. Mature programs also provide several correlations for loss assumptions and blockage prediction which are important to capture as much flow effects as possible in order to be closer to the real flow field.

The goal at this design phase is to find adequate flow parameter distributions along the one-dimensional mid-height streamline of the compressor which fulfill the performance requirements for the design point and the off-design characteristics as good as possible. The quality of a compressor design is quantified by global parameters as efficiency, surge margin, overall pressure ratio, and design mass flow. The individual stages and sometimes even the single rows are assessed based on design rules and quantities such as the de Haller number, diffusion number, Koch parameter, and stage loading. If design goals are not fulfilled, these parameter distributions may help to detect problems of the compressor design.

The meanline prediction process as it is performed today is a very quick and

reliable method for compressor preliminary design. Beside the assessment for design flow conditions, the low calculation effort makes it efficient to predict also off-design characteristics of the compressor design. By varying shaft speed and compressor exit pressure it is possible to obtain an overview of the compressor performance capability which is summarized in a compressor map. This is an important aspect since it allows to predict a compressor map at such an early design phase and to judge the compressor design according to design and off-design conditions.

Typically, the results of the preliminary process are obtained by time-consuming manual parameter studies based on engineering intuition or experience. The final one-dimensional solution is used as an initial guess for the subsequent design process, e.g. for throughflow calculations. According to *Wu (1952)* the highly three-dimensional flow in compressors can be split into two separate but interrelated two-dimensional surfaces, *Cumpsty (2004)*. Figure 1.4 shows the intersecting surfaces where S1 denotes blade-to-blade surfaces which are surfaces of revolution running from blade suction to blade pressure side, and the meridional plane S2 which is extending from hub to casing. The benefit of this approach is that the complex three-dimensional flow problem can be tackled by less complicated two-dimensional flow analyses. The meanline calculation can be seen as a computation along the intersecting line between the meridional plane S2 and the S1 surface at mid-height radial position.

Hence, the results of the meanline prediction are directly used as input for the subsequent throughflow calculation in the S2 plane, Figure 1.3. Within this design phase the input parameters are extended in radial direction according to some design rules or design experiences, and the two-dimensional flow field is typically solved based on a streamline curvature method. The throughflow design process is a time-consuming and highly iterative process since each parameter modification requires a full flow analysis typically consisting of 21 radially distributed streamlines. If inconsistencies in the design parameters exist, the determination of the streamline distributions often fails which makes the process sometimes more difficult. However, the throughflow calculation is important since it is the point where the initial annulus geometry is smoothed and, if desired, additional contouring features at the hub or casing are introduced. Beside the radial distributions of the design parameters and velocity triangles, i.e. gas inlet and outlet

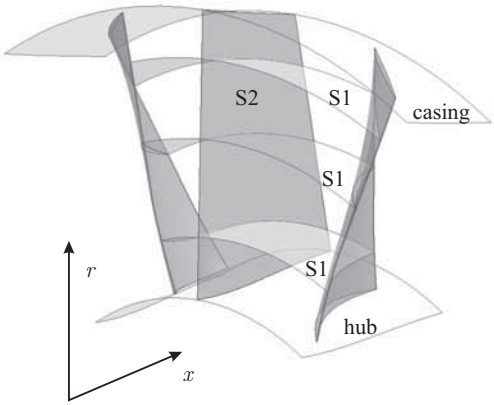


Figure 1.4: Meridional plane (S2) and blade-to-blade surface (S1) definition according to *Wu (1952)*

whirl angles and velocities, streamline geometries are obtained.

Based on the throughflow results, the subsequent 2D-blading process as a next level of design refinement is performed, Figure 1.3. Starting with blade section geometries of an adequate earlier design, the two-dimensional flow field in each corresponding S1 stream surface is solved based on a blade-to-blade flow calculation program. For stationary inlet flow conditions the two-dimensional blade geometries are varied and the flow field around each section is calculated. The results include Mach number and pressure distributions on the section surfaces as well as blade exit whirl angle, velocity, and pressure loss. For an axial compressor the throughflow calculation relates the blade-to-blade flow in radial direction by solving the radial equilibrium equation. The aim of the 2D-blading process is to find blade geometries which match flow angles and flow conditions from the throughflow calculation on each S1 stream surface appropriately. A typical number of radial stream surfaces is 21, however, the 2D-blading is only performed on a smaller number of radial positions of each blade, e.g. 11, 9, or only 7, in order to keep computational costs small. Nevertheless the 2D-blading process remains a very time-consuming task especially for a multi-stage application.

After 2D-blading the resulting blade sections are radially stacked along a specific stacking line of the blade and additional design principles as sweep and

dihedral are introduced to the three-dimensional blade geometry in order to reduce secondary flow losses, *Gümmer (2000)*. As a final step, highly sophisticated three-dimensional CFD calculations are performed for individual blade rows as well as for the whole compressor often including bleed air extraction and shroud leakage flow. The results may be used to update the aerodynamic design models of an earlier phase or to perform the subsequent design and stress analysis.

As can be seen in Figure 1.3, the aerodynamic compressor design process is not straight forward. If design goals cannot be achieved in one of the design tasks it is often required to step back to one of the earlier design phases and to make modifications on design parameters or design assumptions. It is also usual practice to update an earlier model by more accurate results of subsequent design analyses in order to increase the accuracy of the overall design process. Many design iterations within and in between the individual design steps are needed where the final design always reflects a compromise. A good design can only be found by many time-consuming iterations leading to undesired high development times.

1.2 State of the Art in Aerodynamic Optimization

In the last decades a lot of scientific investigations have been carried out dealing with the aspect of aerodynamic optimization in turbo machinery design. This includes compressor design as well as turbine design from which methods and experiences can be reflected to other component designs.

In the field of preliminary design, *Dornberger et al. (2000)* published a multi-objective optimization approach which is applied to a preliminary turbine design process. Based on parameter modification of the flow path and blade chord lengths, an improvement in aerodynamic turbine efficiency and reductions of estimated stresses and costs are pursuit simultaneously. The design problem is solved with a genetic algorithm with the fundamental property of generating Pareto-optimal solutions. A three-dimensional surface of Pareto-optimal points is found which provide trade-off solutions.

The work of *Müller-Töws (2000)* aims to develop a throughflow design process

for multi-stage axial compressors in combination with numerical optimization strategies based on deterministic and stochastic methods. Compressor annulus geometry as well as further design parameters are optimized where multiple design goals are combined in an aggregated function using the weighted-objective approach. *Oyama and Liou (2002)* present an application of multi-objective optimization to a four-stage throughflow compressor design with design modifications made in terms of radial blade parameter variations resulting in 80 design parameters. The design problem is solved as a real multi-objective problem with an evolutionary algorithm resulting in hundreds of reasonable and uniformly distributed Pareto-optimal solutions that outperform the baseline design in both objectives, namely overall compressor efficiency and total pressure rise. A further method for improving axial compressor design in the throughflow design phase is presented by *Ahmed and Lawerenz (2004)*. The optimization problem is solved with a combination of a global single-objective optimization strategy and a surrogate approximation model where design modifications are performed on annulus and blade geometries yielding 625 design parameters. The optimization process requires 23 hours on a cluster with 13 processors for improving the compressor efficiency.

Since the blade design problem is highly complicated and time-consuming due to many design parameters and uncertainties in the function evaluation, a lot of researchers try to tackle the design problem by applying automated optimization strategies. *Trigg et al. (1999)* present an automated process for a single-objective optimization of two-dimensional steam turbine blades on the basis of a genetic algorithm. The geometry is described by Bézier-curves with 17 independent design parameters where the flow around the blade section is calculated by a viscous blade-to-blade solver. The authors report on significant reduction in blade losses of 10-20% compared to the datum designs. The intention of *Köller et al. (2000)* is to develop a new family of subsonic compressor airfoils by using an automated blade design process combining a geometric code for airfoil description based on spline representations, a viscous blade-to-blade flow solver, and a numerical optimization algorithm. In order to solve the multi-objective design problem with a combination of random search and a gradient based optimization method, the objectives are transferred into a scalar utility function using the weighted-objective approach. A reduction in total pressure

loss at design flow conditions and a simultaneous increase in working range of the blade are achieved. Few years later *Sieverding et al. (2004)* solve the multi-objective blade design problem proposed by *Köller et al. (2000)* with a single-objective genetic algorithm. The optimization is performed on a computer with a single processor where the optimization process for one blade section typically requires two weeks for the evaluation of 400 generations. For all investigated blade sections the loss is reduced significantly and the working range is increased compared to the datum NACA 65 blade design.

In some research work the focus lies more in a better flow calculation in order to catch as much flow phenomena as possible in the two-dimensional blading process. In the investigation of *Dennis et al. (2001)* a combination of genetic algorithm and constrained gradient based method is used for optimizing a two-dimensional blade geometry with respect to total pressure loss. The authors use a higher sophisticated Navier-Stokes flow solver with an integrated turbulence model for better flow evaluation where the blade geometry is parameterized using B-splines. A significant reduction of the total pressure loss is reported. In a further investigation of *Sonoda et al. (2003)* a compressor blade section is optimized with respect to the pressure loss at design flow and working range using two different numerical optimization methods, an evolutionary strategy and a multi-objective genetic algorithm. The two-dimensional geometry is described by non-uniform third order rational B-splines leading to 42 design variables and a Navier-Stokes flow solver including a transition and turbulence model is chosen for design evaluation. Both optimization algorithms achieve reasonable improvements in the objective function values. At the same time *Burgubur et al. (2003)* develop a single-objective optimization process to design compressor and turbine blading geometries in turbo machinery using a gradient based method which is coupled to a Navier-Stokes flow solver. Three different investigations considering two- and three-dimensional blade geometries are carried out. As a new approach the geometry variations are described by Spline representations and parameterized deformation functions. In all investigations a loss reduction of 1% at design flow conditions is achieved.

In different scientific investigations it is shown that the experiences in the pure two-dimensional blade optimization process can be used to design a three-dimensional blade geometry. The emphasis of these investigations is to use only

two-dimensional flow solvers in order to design multiple two-dimensional blade sections which are finally stacked in radial direction. *Chung and Lee (2002)* publish a shape optimization approach applied to a transonic compressor blade design (NASA rotor 37) where geometry variations are performed at three preselected blade sections at 30, 50, and 70% blade height. Two objective functions are employed to maximize the blade section efficiencies which are obtained by a quasi-three-dimensional Navier-Stokes solver with an appropriate two-equation turbulence model at design operating conditions. The deterministic optimization requires approximately 8 hours for convergence resulting in 1% efficiency improvement for each investigated approach. *Büche et al. (2003)* propose a method where a complete compressor blade is designed by three individually optimized blade sections. In this work a multi-disciplinary aspect is considered where multiple design criteria as well as aerodynamic and mechanical constraints are aggregated together in an objective function which is minimized by an evolutionary optimization strategy. The blade optimization process requires 12 hours for 4000 designs on a cluster with four processors. The results show a 15% working range improvement compared to the initial design where the loss is not reduced.

In the latter investigations blade optimization is performed on multiple two-dimensional sections which are evaluated by a two-dimensional flow solver. In the work of *Benini (2004)*, however, the quasi-three-dimensional blade design is calculated by a full three-dimensional flow analysis. Based on a transonic compressor rotor, multi-objective geometry optimization is performed with a multi-criterion evolutionary algorithm in order to maximize the isentropic efficiency and the pressure ratio at design point. Geometry modifications are performed on camber line and thickness distributions at three different blade sections with 23 parameters in total, from which a three-dimensional geometry is interpolated. The three-dimensional CFD calculation is performed on a parallel four-processor machine leading to an overall turn around time of about 2000 hours. The final results show trade-off solutions with an efficiency improvement at equal pressure ratio and a higher pressure ratio at a reasonable efficiency level.

The significant increase of computational power and the availability of computer clusters in the last years allow to optimize three-dimensional blade geometries directly. More recently *Sasaki et al. (2006)* publish a multi-objective approach for optimizing a three-dimensional compressor stage which is embedded

in a four-stage axial compressor. In this research the parameterized stage geometry is optimized to improve aerodynamic performance in terms of efficiency, blockage and loss, while satisfying four aerodynamic constraints to maintain the flow similar to a baseline geometry. In order to identify trade-off solutions with a reasonable number of function evaluations, a multi-objective genetic algorithm is adopted as optimizer where only 320 design evaluations are carried out. The final geometries show only slight improvements in the objective values while the computational cost is about 5 hours for one design evaluation on a coarse CFD grid.

These investigations show that there is still a huge demand in process acceleration and automation in turbo machinery design. Especially the benefits of application of multi-objective optimization to the preliminary design phase is not well understood. The investigations in the field of blade design show that there is no standard procedure for solving the design problem. Since it is recognized that multiple goals have to be achieved simultaneously, multi-objective optimization methods have to be used providing trade-off solutions from which the design engineer can finally choose. The pure two-dimensional blade design process is investigated by several researches with different emphasis resulting in good optimization results. However, in terms of process acceleration a lack in the problem definition can be identified which may be solved by a re-definition of the two-dimensional blade design problem. In terms of three-dimensional blade design, a complete blade shape modification and optimization is still too time-consuming due to the numerical flow evaluation. A multi-objective optimization process would require thousands of design evaluations for a proper determination of trade-off solutions. Probably the best compromise between computational cost and design model accuracy is shown by the presented quasi-three-dimensional blade design methods where the blade geometry is obtained by multiple two-dimensional blade section optimizations which make it worth for further investigations.

1.3 Contents and Structure of the Thesis

The emphasis of this work is to apply the aspects of process integration and automation to the industrial aerodynamic compressor design process of the Rolls-Royce Company. The goal is to analyze, evaluate and accelerate the time-

consuming design process and to use validated Rolls-Royce design tools without any modifications in order to be as close as possible to the real engineering work flow. Since the individual design tasks in the complex aerodynamic design process are typically solved by human designers with highly iterative manual parameter studies, it is a further goal to formulate optimization problems which can be solved by numerical optimization. The aim is to use multi-objective optimization methods to find better compressor designs and to support the design engineer in his decision making by providing trade-off solutions between the contradicting design goals. Three typical engineering design tasks in different phases of the aerodynamic compressor design process, namely preliminary design, throughflow calculation, and blading procedure are selected to be analyzed in order to demonstrate the improvements in term of process acceleration and optimization.

This thesis is organized in six chapters. Following the introduction, Chapter 2 gives an overview of the theoretical background for several important aspects used in this work. It contains the principles of process integration as well as an introduction to parameterization methods of design quantities using Bézier-curves and B-splines. Additionally, numerical optimization will be discussed including single- and multi-objective optimization methods, scalarization strategies, and an overview of optimization algorithms used in this thesis. The following three chapters are presenting the application of process integration, automation, and multi-objective optimization to aerodynamic compressor design problems starting with the one-dimensional meanline process in Chapter 3, the throughflow off-design calculation task in Chapter 4, and the time-consuming two- and three-dimensional blading process in Chapter 5. Conclusions and an outlook for future work in the field of aerodynamic compressor optimization will complete the thesis.

2 Theoretical Background

Typical engineering design problems are characterized by many design parameters, multiple constraints and objectives which have to be solved with different design tools running separately on different platforms leading to a time-consuming work flow. This chapter introduces process integration as an efficient method to accelerate the design process flow, and design parameterization which is important to reduce the number of design parameters without reducing the design freedom too much. Finally, an introduction to numerical optimization covering single- and multi-objective optimization is provided for supporting the design engineer in his solution finding, and an overview of the algorithms used in this thesis is given.

2.1 Process Integration

Today's industrial design process is characterized by a heterogeneous tool set running on different computers with different operating systems. Big companies use in-house codes developed by themselves during several decades for solving particular design tasks. They are essential for the design process since they incorporate the expertise, knowledge, and design rules of the company which are gathered and implemented into such programs over years. The benefits are a good adaption to the computational environment, connection to data bases, and the link to other in-house codes within the process chain. However, developing, supporting, and adapting such codes to new purposes is often too expensive for the companies why more and more commercial applications in different areas are used. It is also a well known fact that commercial codes provide better sup-

port, a more professional software development, and the availability on different operating systems.

The increasing demand on design time reduction drives the requirement of process flow automation by integrating in-house as well as commercial codes on heterogeneous platforms in a common environment. This demand can be addressed by using commercial software packages as *iSight (2004)* or *modeFRONTIER (2006)* which are generic shell software applications for process integration and automation. The thesis is based on *iSight* which basically consists of a task manager, a process integration module, the solution process module, and the solution monitor module, Figure 2.1.

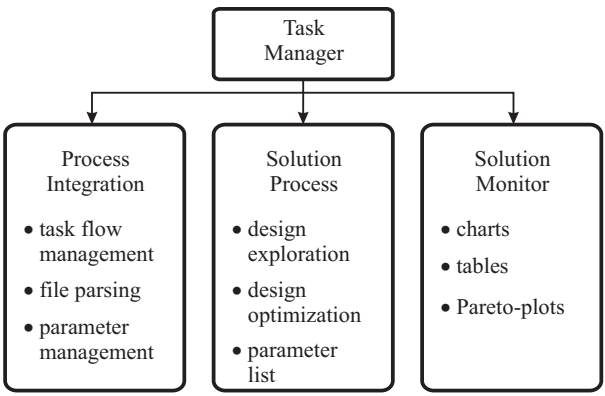


Figure 2.1: *iSight* modules

Once the design process flow is known, the individual programs can be integrated by the process integration module. It provides the possibility to link programs together on heterogeneous platforms and invoke them sequentially or parallelly. The input and output files for each program can be parsed in order to apply parameter modifications as well as to extract results from an analysis code. The data flow is also managed within the process integration module where input and output parameters of the individual programs are defined. Additionally, it is possible to do some simple parameter calculations within a calculation module.

After the process flow is set up in the process integration module, the design problem is solved in the solution process. In this module *iSight* provides design

exploration techniques as *Design of Experiments* (DoE) or *Monte-Carlo Simulation* (MCS) in order to explore the design space for parameter sensitivities. The results can be used either for parameter reduction, design problem approximation based on surrogate models or as an initial step for design optimization. For the latter case, deterministic and stochastic optimization algorithms are provided and can be chosen individually or in combination to define an optimization strategy based on sequential optimization methods. Furthermore, the design parameters, constraints, and objectives have to be selected from the parameter list and required bounds have to be made in this module. As a final step a solution monitor can be started where important design parameters and optimization objectives can be monitored by graphical charts or in a tabulated manner.

2.2 Design Parameterization

In terms of numerical optimization many design parameters, multiple design objectives, and high number of constraints lead to an unmanageable task and raise the computational time dramatically. Hence, appropriate parameterization methods have to be used in order to decrease the number of design parameters without reducing the design freedom and additionally to guarantee technical feasibility of the obtained design. Parameterization smoothes the design problem, reducing the chance to be trapped in a local minimum, and thus increases the possibility of finding a global optimum of the design problem.

Parameterization should be done carefully, since it is imposing implicit constraints on the design problem and could lead to sub-optimal solutions. A trade-off between the maximum design freedom and the minimum parameter number has to be found. The decision on the parameterization method and the number of parameters have a major influence on the final result. In this work two different methods are used in order to parameterize design parameter distributions in 2D-space which will be discussed in the following.

2.2.1 Bézier-Curves

Bézier-curves and -surfaces are one of the most frequently used representations in computer graphics. The theory was independently discovered and developed

by Pierre Bézier in 1962, an engineer for Rénault, and Paul de Casteljau in 1959 working for the Citroën automotive company. Being competitors, both French companies were very secretive about their work, and although de Casteljau's work was slightly earlier than Bézier's, it was never published. Consequently, the field retains Bézier's name. However, the fundamental algorithm which forms the basis for the construction and calculation of Bézier-curves is now credited to de Casteljau, *Farin (1990)*.

The bases of all spline curves are the blending functions. For Bézier-curves they are called Bernstein polynomials B_k^n which are defined as

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{n-k}, \quad k = 0(1)n, \tag{2.1}$$

with the curve parameter $t \in [0,1]$ running along the curve, the curve degree n, the Bernstein index k and the binomial coefficient

$$\binom{n}{k} = \begin{cases} \dfrac{n!}{k!(n-k)!} & \text{for} \quad 0 \leq k \leq n \\ 0 & \text{else.} \end{cases} \tag{2.2}$$

Bernstein polynomials are easy to calculate and are defined for the entire domain of the curve index t. They have the property of nonnegativity, $B_k^n(t) \geq 0 \quad \forall\, k, n$ and partition of unity, $\sum_{k=0}^{n} B_k^n(t) = 1$, *Piegl and Tiller (1997)*.

The Bézier-curve $\mathbf{b}(t)$ in a two-dimensional space can be determined as

$$\mathbf{b}(t) = \begin{bmatrix} b_x^n(t) \\ b_y^n(t) \end{bmatrix} = \sum_{k=0}^{n} \mathbf{b}_k B_k^n(t) \tag{2.3}$$

which is a linear combination of control point positions $\mathbf{b}_k \in \mathbb{R}^2$ and the corresponding Bernstein polynomials (2.1). An example for a Bézier-curve representation is given in Figure 2.2 showing on the left hand side the distribution of the Bernstein polynomials of degree $n = 4$ for a Bézier-curve with five control points and on the right hand side the position of the control points $\mathbf{b}_k$ which determine the resulting Bézier-curve.

In general, Bézier-curves are practicable for geometric representation of complex curves using a low number of parameters. If the complexity of the curve increases, a more flexible Bézier-curve can be created by just adding one or more

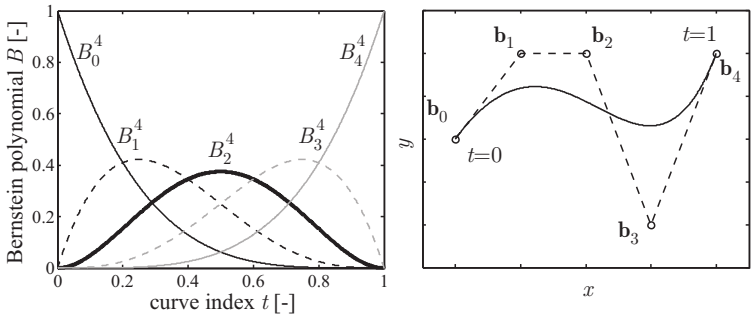


Figure 2.2: Bernstein polynomials of degree $n = 4$ (left) and Bézier-curve defined by five control points $\mathbf{b}_k$, $k = 0(1)4$ (right)

control points. In fact, mathematically an upper limit to this number does not exist. However, due to the dependence of the control point number $n + 1$ on the polynomial degree n, it will be observed that Bézier-curves with many control points tend to oscillate. In practice, it is therefore not recommended to use Bézier-curves with more than 10 control points, *Farin (1990)*. Nevertheless, if complex geometries have to be parameterized, multiple Bézier-curves may be joined together while additional requirements concerning continuity and curvature should be considered at their linking positions in order to guarantee smoothness of the entire curve, *Keskin (2001)*.

It should be noticed that beside their benefits some critical drawbacks of Bézier-curves exist. Typical advantages and disadvantages of Bézier-curves are provided in the following:

Advantages

- Bernstein polynomials are easy to calculate and are defined for the entire curve index t.
- Bézier-curves can be manipulated by modifying their control point positions.
- Bézier-curves lie in the convex hulls of their defining control points.
- The first and last control point coincide with the endpoints of the curve.

- Derivatives of Bézier-curves are also Bézier-curves with a reduction in the polynomial order.

Disadvantages

- Polynomial order is linked to the control point number which may cause numerical instabilities for high order polynomials.
- Modification of one control point influences the whole Bézier-curve, i.e. no local curve control is possible.
- If complex curves or distributions are represented by multiple Bézier-curves, additional conditions at their joining positions have to be used.

Due to these drawbacks of Bézier-curves it is sometimes recommended to use another representation like B-splines which is equipped with more flexibility without loosing generality.

2.2.2 B-Splines

Curves consisting of just a single polynomial segment like Bézier-curves are inadequate if local control is required. A solution to this problem is the usage of B-splines consisting of piecewise polynomial curves. The name B-spline was coined by the Romanian mathematician *Schoenberg (1946)* and is the shortcut for basis spline.

The idea behind B-splines is to use basis polynomials which are defined within a specific curve segment only in order to enable local shape control. The basis polynomials N_k^m can be evaluated by a recursive scheme:

$$N_k^0(t) = \begin{cases} 1 & \text{if} \quad t_k \leq t < t_{k+1} \\ 0 & \text{else} \end{cases}$$

$$N_k^j(t) = \frac{t - t_k}{t_{k+j} - t_k} N_k^{j-1} - \frac{t - t_{k+j+1}}{t_{k+j+1} - t_{k+1}} N_{k+1}^{j-1}, \quad j = 1(1)m \tag{2.4}$$

where the maximum polynomial degree is m, the spline parameter $t \in [0, 1]$ runs along the curve, the index k denotes the corresponding basis polynomial and with

the definition $0/0 =: 0$. The valid segment for each basis polynomial depends on the number of control points $n+1$ and the polynomial order $m+1$ and is defined by a knot vector

$$\mathbf{K} = [\underbrace{a, \ldots, a}_{m+1}, t_{m+1}, \ldots, t_{l-m-2}, \underbrace{b, \ldots, b}_{m+1}]^T \tag{2.5}$$

of length $l = (m+1) + (n+1)$ with monotonically increasing elements. In order to achieve endpoint interpolation of the B-spline curve, the knot vector has to be chosen in such a way that $a = 0$ and $b = 1$. If the inner knots $(t_{m+1}, \ldots, t_{l-m-2})$ are equidistant, the resulting spline is uniform, otherwise it is called non-uniform.

Analogously to the Bézier-curve definition (2.3) the resulting B-spline $\mathbf{b}(t)$ is given by

$$\mathbf{b}(t) = \begin{bmatrix} b_x^n(t) \\ b_y^n(t) \end{bmatrix} = \sum_{k=0}^{n} \mathbf{b}_k N_k^m(t) \tag{2.6}$$

as a linear combination of control point positions $\mathbf{b}_k$ and basis polynomials N_k^m, respectively. Important to distinguish, however, is that the number of control points $n+1$ and the polynomial degree m of the basis functions N_k^m are now independent from each other. This leads to the fact that on the one hand low order basis polynomials can be chosen being more stable in terms of numerical behavior, and on the other hand the number of control points can be increased independently in order to adapt to the complexity of the curve to be represented.

A sample for a distribution of cubic basis polynomials as well as a B-spline defined by five control points is shown in Figure 2.3. As can be seen, not all basis polynomials are non-zero within the whole definition range $[0, 1]$ of curve index t, each of them is defined within its particular segment given by the knot vector $\mathbf{K}$. In this particular case a uniform knot vector $\mathbf{K} = [0, 0, 0, 0, 0.5, 1, 1, 1, 1]^T$ is chosen which drives the first basis polynomial N_0^3 from $t = 0$ to $t = 0.5$ and the last basis polynomial N_4^3 from $t = 0.5$ to $t = 1$. In terms of local control this means that the first and last control point $\mathbf{b}_0$, $\mathbf{b}_4$ are influencing the B-Spline curve only at the first and last 50% of the curve length, respectively. Furthermore, if the distribution of the Bernstein polynomials in Figure 2.2 is compared with the basis polynomials for B-splines in Figure 2.3, it can be observed that the

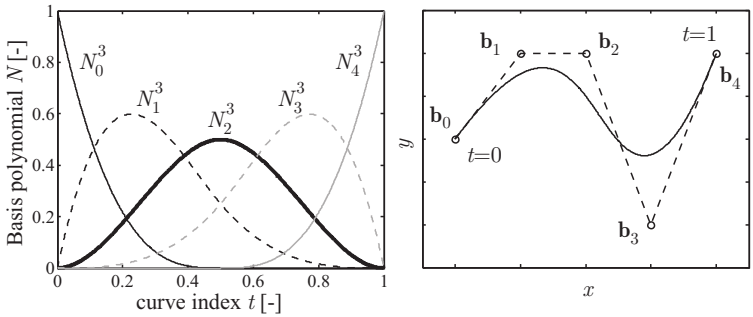


Figure 2.3: Basis polynomials of degree $m = 3$ (left) and cubic B-spline defined by five control points $\mathbf{b}_k$, $k = 0(1)4$ (right)

function values at equal t-positions are different. The higher values of the basis polynomials are responsible for the B-Spline curve being closer to its control points.

Summarizing, some important benefits and drawbacks of B-Splines are the following, *Piegl and Tiller (1997)*:

Advantages

- B-splines are numerically stable, because polynomial degree m of the basis functions is independent of the number $n + 1$ of control points.
- B-splines provide local shape control due to the knot vector $\mathbf{K}$ describing the influencing range of each control point.
- First and last control point coincide with endpoints.
- B-spline lies in the convex hull of defining control points.
- Derivative of a B-spline is also a B-spline with a reduction in the polynomial order.
- If $n = m$ and $a = 0, b = 1$, the B-spline becomes a Bézier-curve.

Disadvantage

- Time consuming determination of the piecewise defined basis functions based on recursive scheme.

In terms of curve parameterization, cubic B-splines with a uniform knot vector **K** should be the first choice, *Farin (2000)*. They are a good compromise between curve smoothness (C^2 continuity, i.e. continuous in first and second derivatives) and computational cost for determination of the cubic basis polynomials, *Harries (1998)*. In general, the design freedom can be varied by adding or subtracting further control points which makes them more flexible and suitable for various applications.

2.3 Numerical Optimization

In the last century, optimization has become more and more popular. Optimization is used within various disciplines and for miscellaneous purposes. Airlines are using mathematical optimization extensively in order to optimize the schedule of the pilots, the flight attendants, and the flight plan itself. In the field of transportation optimization solves logistical problems where the goal is to minimize the time or the costs for transporting freight or passengers from one point to another. Companies are optimizing their products in terms of quality, reliability, costs, and efficiency.

The enormous development in the field of computer technology and the improvements of numerical algorithms in the last decades are responsible for the growing acceptance of numerical optimization. In order to use optimization properly, first the optimization problem has to be analyzed and the design goals have to be identified. They can be profit, time, efficiency, loss, or any other quantity or combination of quantities that can be evaluated by a number. Unknown goals are often used as objectives which have to be minimized or maximized whereas known goals or restrictions are typically treated as equality or inequality constraints during the optimization process.

The objectives, and sometimes the constraints as well, are functions which depend on certain parameters, called design variables or parameters that are to be modified. The purpose of optimization is to find optimal values for these design

parameters which minimize or maximize the objective function values. Often design variables or any other parameters have to be restricted or constrained in some way in order to guarantee feasible solutions, e.g. quantities such as mass or length have to be positive.

The process of identifying objectives, design variables and constraints for a given problem is probably the most important step in setting up the optimization problem. If constraints are ignored or bounds on design parameters are made poorly, optimization will not provide useful insight into the problem. On the other side, if parameters are too restricted, optimization will find only sub-optimal solutions or in the worst case it may become too difficult to solve the optimization problem at all.

Another important point is a proper choice of the optimization algorithm. It depends significantly on the properties of the optimization problem to be solved. Hence, a classification of the optimization problem is extremely useful from the computational point of view since there are many special methods available for solving these particular classes of problems efficiently, *Rao (1996)*. Depending on whether or not constraints exist in the problem, any optimization task can be classified as constrained or unconstrained problem. A further classification can be done based on the mathematical expressions for the objective function and the constraints. According to this, the optimization problem could be distinguished between linear, nonlinear, and quadratic programming problems. An essential question is, however, the number of objective functions involved in the optimization problem. This leads to the point where single- and multi-objective optimization problems have to be distinguished.

2.3.1 Single-Objective Optimization

If the optimization task consists of one objective only, the design problem is called mono- or single-objective problem. Mathematically speaking, single-objective optimization is the minimization or maximization of an objective function f depending on its design variables summarized in the design vector $\mathbf{p}$ subject to K equality constraints $\mathbf{h}$, J inequality constraints $\mathbf{g}$ and bounds on the design parameters or the objective function itself. The optimization problem can be written

as

$$\begin{aligned} & \underset{\mathbf{p}\in\mathbb{R}^n}{opt} \quad f(\mathbf{p}) \\ \text{subject to} \quad & \\ & g_j(\mathbf{p}) \leq 0, \qquad j = 1, 2, \ldots, J, \\ & h_k(\mathbf{p}) = 0, \qquad k = 1, 2, \ldots, K, \\ & \mathbf{p}^l \leq \mathbf{p} \leq \mathbf{p}^u \end{aligned} \tag{2.7}$$

where a maximization of the objective function is equal to the minimization of its negative function value, i.e.

$$min \;\; f(\mathbf{p}) \;=\; - \, max \;\; (-f(\mathbf{p}))\,. \tag{2.8}$$

An optimum to the problem is found if the following optimality conditions are fulfilled:

1. The necessary condition for unconstrained problems is defined as

$$\nabla f(\mathbf{p}^*) = 0 \tag{2.9}$$

   where $\mathbf{p}^*$ is a local minimizer of the function $f(\mathbf{p})$ and ∇f is the gradient of f containing the partial derivatives with respect to the components of the design vector $\mathbf{p}$.

2. The sufficient condition for unconstrained problems requires that additionally the Hessian matrix $\mathbf{H} = \nabla^2 f$ as the matrix of second partial derivatives of the function $f(\mathbf{p})$ evaluated at $\mathbf{p}^*$ is positive definite when $\mathbf{p}^*$ is a minimum point.

For constrained optimization, the Lagrange function

$$L(\mathbf{p}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{p}) - \sum_{k=1}^{K} \lambda_k h_k(\mathbf{p}) - \sum_{j=1}^{J} \mu_j g_j(\mathbf{p}) \tag{2.10}$$

is introduced coupling the objective function f with the equality and inequality constraints to a new function L with the Lagrange multipliers λ and μ. The nec-

essary conditions for constrained optimization have been formulated by Karush and Kuhn-Tucker, *Fletcher (2000)*, and are defined as

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{p}} - \sum_{k=1}^{K} \lambda_k \frac{\partial h_k}{\partial \mathbf{p}} - \sum_{j=1}^{J} \mu_j \frac{\partial g_j}{\partial \mathbf{p}} &= 0, && (2.11)\\ \mathbf{g}(\mathbf{p}) &\leq \mathbf{0}, && (2.12)\\ \mathbf{h}(\mathbf{p}) &= \mathbf{0}, && (2.13)\\ \boldsymbol{\mu} &\leq \mathbf{0}, && (2.14)\\ \mu_j g_j(\mathbf{p}) &= 0 && (2.15) \end{aligned}$$

or with the Lagrange function, *Bestle (1994)*, as

$$\frac{\partial L}{\partial \mathbf{p}} = \mathbf{0}, \quad \frac{\partial L}{\partial \boldsymbol{\lambda}} = \mathbf{0}, \quad \frac{\partial L}{\partial \boldsymbol{\mu}} \geq \mathbf{0}, \quad \boldsymbol{\mu} \leq \mathbf{0}, \quad \mu_j g_j(\mathbf{p}) = 0. \tag{2.16}$$

2.3.2 Multi-Objective Optimization

Most real-world design or decision problems are multi-objective or vector problems which involve simultaneous optimization of multiple objectives. Generally speaking, the goal of a multi-objective optimization problem is to optimize the design parameter vector $\mathbf{p}$ characterized by the minimization or maximization of M objective functions $\mathbf{f}$ and associated equality and inequality constraints. Mathematically the multi-objective optimization problem can be stated in its general form as

$$\begin{aligned} & \underset{\mathbf{p} \in \mathbb{R}^n}{opt} \quad \mathbf{f}(\mathbf{p}) \\ \text{subject to} \quad & \\ & g_j(\mathbf{p}) \leq 0, \qquad j = 1, 2, \ldots, J, \\ & h_k(\mathbf{p}) = 0, \qquad k = 1, 2, \ldots, K, \\ & \mathbf{p}^l \leq \mathbf{p} \leq \mathbf{p}^u \end{aligned} \tag{2.17}$$

Despite the fact that the mathematical formulation of the optimization problem looks quite similar to single-objective optimization, multi-objective optimization is very different. Beside the design space in which each combination of design

parameters is available a second space with the attainable objective function values exist. Figure 2.4 shows the mapping process of a design represented by the design vector $\mathbf{p}$ as part of the feasible design space $\mathcal{P}$ to the attainable objective space $\mathcal{F}$ of a bi-criterion design problem.

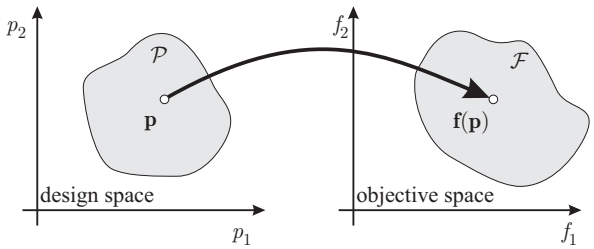


Figure 2.4: Illustration of design space and objective space for a bi-criterion design problem

As described in the previous Section 2.3.1, the aim of a single-objective optimization is the attempt to obtain the best solution to the problem, which is usually the global minimum or the maximum. In case of multiple objectives the contradiction between individual objectives leads to the problem that there may not exist solely one solution which is best with respect to all objectives of the design problem. In a typical multi-objective optimization problem there exists a set of solutions which are superior to the rest of solutions in the search space when all objectives are considered but are inferior to other solutions in the space in at least one objective. Mathematically speaking, in multi-objective optimization a design $\mathbf{p}^1$ is better than $\mathbf{p}^2$ in the case of minimization, if the corresponding objective functions $\mathbf{f}^1 := \mathbf{f}(\mathbf{p}^1)$ and $\mathbf{f}^2 := \mathbf{f}(\mathbf{p}^2)$ are related as

$$\mathbf{f}^1 < \mathbf{f}^2, \text{ i.e. } \quad (f_i^1 \leq f_i^2 \quad \forall \quad i \in M) \wedge (\mathbf{f}^1 \neq \mathbf{f}^2), \tag{2.18}$$

Bestle (1994). These solutions are known as non-dominated solutions or Pareto-optimal solutions, where the rest of the solutions are called dominated solutions.

Figure 2.5 shows an example of an objective space for a bi-criterion minimization problem with conflicting objectives f_1 and f_2. A sorting among the solutions

A to E can be done using the principle of dominance. In this particular case, solution D is dominated by solution A and B, while E is dominated by solution C and B since they are better in both or at least in one objective without being worse in the other. However, since none of the solutions in the non-dominated set A-C is absolutely better than any other, i.e. better in both objectives, each of them is an acceptable solution to the multi-objective minimization problem.

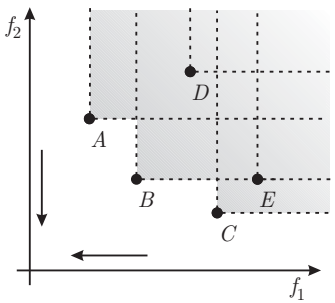


Figure 2.5: Principle of dominance for a bi-criterion minimization problem

In this particular case where both objectives are being minimized, the Pareto-optimal solutions are located at the lower left border of the attainable objective space. For a different combination of minimization and maximization of objectives the Pareto-optimal front varies. In Figure 2.6 four possible borders for two-objective optimization problems are indicated.

The benefit of multi-objective optimization compared to the classical single-objective problem is to provide different solutions to the design problem from which the engineer or designer can choose. The choice of one solution over the other requires problem knowledge or additional decision criteria which are not explicitly formulated in the design task. Thus, one solution selected by a designer may not be acceptable to another designer. Therefore, it may be useful to have knowledge about as much trade-offs as possible within Pareto-optimal solutions, i.e. a wide set of non-dominated solutions from which one or more solutions can be chosen after the optimization process according to some decision-makers.

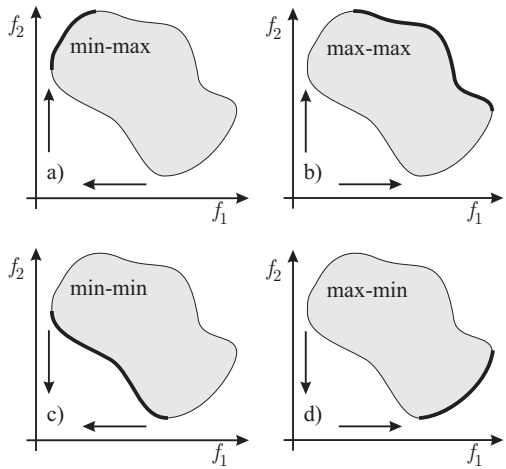


Figure 2.6: Location of the Pareto-optimal solutions for a bi-criterion optimization problem: a) minimizing f_1 and maximizing f_2, b) maximizing both objectives, c) minimizing both objectives, d) maximizing f_1 and minimizing f_2

Hence, in multi-objective optimization two goals are pursued simultaneously, *Deb (2001)*:

1. finding a set of solutions as close as possible to the Pareto-optimal front,

2. finding a set of solutions as diverse as possible.

Figure 2.7 illustrates three different results for the same multi-objective optimization problem. As can be seen for the first case, Figure 2.7a, the fairly good solutions found by the optimizer are placed at the Pareto-front, however, a huge gap between the solutions exists and the diversity is rather poor. The results shown in Figure 2.7b are distributed more homogeneously, but the convergence towards the Pareto-front is rather poor, and therefore these solutions would not be acceptable. The ideal case is shown in Figure 2.7c, where all solutions are located along the Pareto-front and are uniformly distributed.

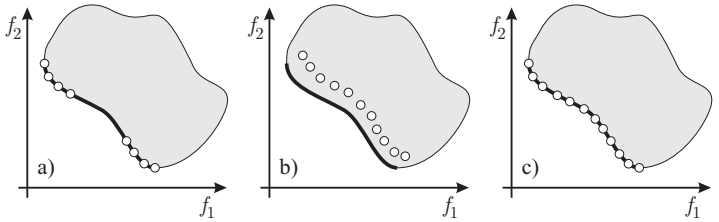


Figure 2.7: Diversity (a) and convergence (b) problems as well as ideal solution (c) for a bi-criterion minimization problem

2.3.3 Classical Scalarization Methods

One way to solve a multi-objective optimization problem is to transform the vector of objectives into a single substitute problem. This method is called scalarization and the purpose of this approach is to create an alternative objective function leading to a new scalar problem which can be solved with a classical optimization algorithm. In the following, different scalarization methods will be introduced and their benefits and drawbacks will be discussed. Since each maximization problem can be transferred to a minimization problem according to (2.8), the case of minimization is considered in the following without loosing generality.

2.3.3.1 Method of Weighted-Objectives

Probably the most common and simplest of all classical scalarization techniques is the method of weighted-objectives which is also known as weighted sum method. This approach is characterized by one composite or utility function F declared by aggregating multiple objective functions f_m with individual weighting factors w_m. The multi-objective optimization problem (2.17) can be re-defined for the case of minimization as

$$\begin{aligned} \underset{\mathbf{p}\in\mathbb{R}^n}{min} \quad & F(\mathbf{p}) \\ \text{with} \quad & F(\mathbf{p}) = \sum_{m=1}^{M} w_m f_m(\mathbf{p}), \quad w_m > 0. \end{aligned} \tag{2.19}$$

The individual objectives are typically normalized, and since the minimum of

the above problem does not change if all weights are multiplied by a constant value, it is usual practice to choose weights such that their sum is equal to one, $\sum_{m=1}^{M} w_m = 1$, *Deb (2001)*.

The application of the weighted-objectives method on a bi-criterion minimization problem is demonstrated in Figure 2.8. Since the composite function F is a linear combination of the objectives f_1 and f_2, its contour lines are straight in the objective space and the slope of the solution levels are defined by the ratio $-w_1/w_2$ of the weighting factors, Figure 2.8a. The task of the minimization procedure is to find the minimum function value of F obtained by the contour line which is tangential to the feasible solution space at the bottom-left corner. Hence, Point A is a Pareto-optimal solution of the minimization problem corresponding to the chosen weighting factors.

It is clear that the preference of an objective can be changed by modifying the corresponding weighting factor which leads to another solution point. This effect can be used in order to find the Pareto-front by obtaining different points on the curve with different combinations of weighting factors, Figure 2.8b. A sequential variation with small incrementing steps of some weighting factors can be used to find as much trade-off solutions as possible. This technique works fine for convex Pareto-fronts only, while for non-convex cases multiple solutions for constant weighting factors could exist and not all points on the Pareto-front can be determined, Figure 2.8c.

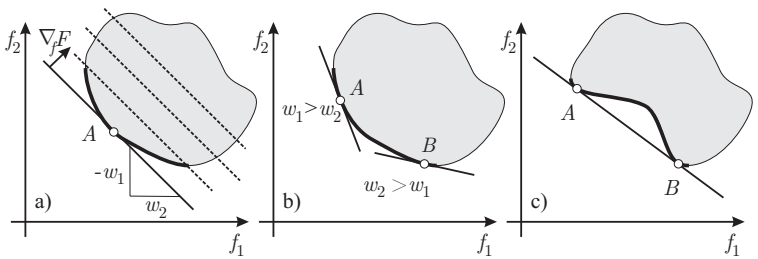


Figure 2.8: Illustration of the method of weighted-objectives: a) contour lines of composite function, b) results for different weighting factors for a convex Pareto-front, c) multiple solutions for a non-convex Pareto-front while solutions in between A and B cannot be determined

2.3.3.2 Distance Method

The distance method is a further scalarization technique where the individual distances between function values $f_m(\mathbf{p})$ and the ideal solution $\mathbf{f}^0$ are used as a new criterion. The minimization problem with the single objective function F derived from multiple objectives is defined as

$$\begin{aligned} &\underset{\mathbf{p}\in\mathbb{R}^n}{min} && F(\mathbf{p}) \\ &\text{with} && F(\mathbf{p}) = \left(\sum_{m=1}^{M} \left| f_m(\mathbf{p}) - f_m^0 \right|^r \right)^{1/r}, \quad 1 \leq r < \infty. \end{aligned} \tag{2.20}$$

The distance depends on the metric r where typically the Taxicab metric $r = 1$, the Euclidean metric $r = 2$, and the Chebyshev metric $r \to \infty$ also known as maximum metric are commonly applied, *Bestle (1994)*.

In Figure 2.9 the principle of the distance method applied to a bi-criterion minimization problem is shown. The Chebyshev metric produces squares for the contour lines of $F(\mathbf{p})$ in the objective space with different diameters depending on the composite function value F, Figure 2.9a. The solution that will be obtained typically lies at the corner point A of the square which is touching the Pareto-front. For the Euclidean metric, function value levels are represented by circles with different radii, Figure 2.9b. Again the solution is given by the touching point A between the Pareto-front and the corresponding circle. It should be noticed that the solution points may vary for different metrics dependent on the Pareto-front distribution. As can be seen in Figure 2.9c, for a non-convex Pareto-front the Euclidean metric can produce several touching points A, B which lead to multiple solutions while solutions in between these points cannot be found.

Generally, the distance method is very similar to the method of objective weighting, but two differences can be figured out:

1. In the distance method the ideal or a target solution for each objective function is required to be known whereas in the method of weighted-objectives the relative importance of each objective is required à priori.

2. A convex Pareto-front can be found with the distance method by varying $\mathbf{f}^0$ whereas for the method of weighted-objectives a variation of the weighting factors can be used.

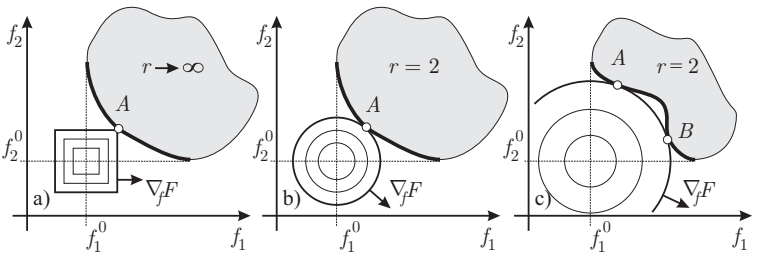


Figure 2.9: Principle of distance method for a bi-criterion minimization problem: a) Chebyshev metric at convex Pareto-front, b) Euclidean metric at convex Pareto-front, c) Euclidean metric at non-convex Pareto-front

2.3.3.3 Compromise Method

The idea of the compromise method, which is also named ϵ-constraint method, is that only one of the original objectives is being optimized whereas the others are taken into account as inequality constraints during the optimization process. The multi-objective minimization problem for a freely chosen objective $f_r(\mathbf{p})$, $r \in [1, \ldots, M]$ and upper bounds $\hat{f}_j$ for the $M-1$ remaining criteria f_j can be re-written as

$$\begin{aligned} & \underset{\mathbf{p} \in \mathbb{R}^n}{min} \;\; f_r(\mathbf{p}) \\ \text{subject to} \quad & \\ & f_j(\mathbf{p}) \leq \hat{f}_j, \quad j \neq r. \end{aligned} \tag{2.21}$$

The compromise method is very useful and can give a good insight to the optimization problem. By relaxing the bounds for the constraints, a further minimization of the objective function is possible whereas more restricted constraints are increasing the objective function value, respectively. An iterative process with sequentially relaxing or restricting the bounds can be used to find additional solutions on the Pareto-front. In order to resolve the whole Pareto-curve, very small step-sizes for the bounds have to be chosen. It is interesting to know that on the one hand this method works for convex as well as for some non-convex solution spaces depending on their complexity, *Deb (2001)*, and on the other hand that

a substitution between the objective and any other constraint is also possible relying on the user's preferences, see Figure 2.10.

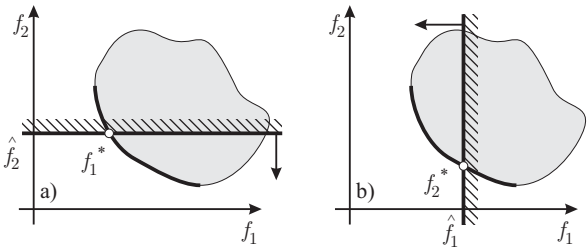


Figure 2.10: Principle of the compromise method for a bi-criterion minimization problem: a) minimization of f_1 with variable bound $\hat{f}_2$, b) minimization of f_2 with variable bound $\hat{f}_1$

2.3.3.4 Min-Max Method

In principle this method is different than the above three methods. Within this scalarization technique an artificial parameter γ is introduced which is the new objective to be minimized and all original objectives are considered as inequality constraints. The parameter γ is both an additional design parameter and the upper bound for all objectives. A suitable mathematical formulation of the min-max method is as follows

$$\begin{aligned} &\underset{(\mathbf{p},\gamma)\in\mathbb{R}^n}{min} \quad \gamma \\ \text{subject to} \quad & \\ & f_j(\mathbf{p}) \leq \gamma, \quad j = 1, 2, \ldots, M. \end{aligned} \tag{2.22}$$

The idea behind this method is to let the optimization algorithm reduce the value of γ during the optimization process, and due to the inequality constraints the original objectives are reduced as well. The optimization stops if no further reduction of γ is possible without violating the constraints, and hence the solution is bounded by the constraint bounds and the Pareto-front, see Figure 2.11.

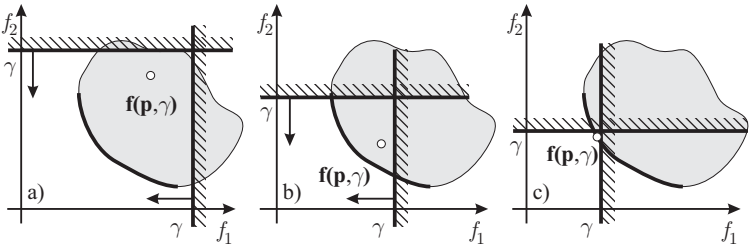


Figure 2.11: Progress of min-max optimization for a bi-criterion minimization problem: a) loose bounds as optimization starts, b) more restricted bounds during optimization, c) final bounds and solution at Pareto-front

2.3.3.5 Discussion about Scalarization Methods

In all of the above mentioned scalarization methods, multiple objectives are combined to form one objective by using some knowledge on the problem being solved. The optimization of the single-objective may guarantee a Pareto-optimal solution for a convex Pareto-front, but results in a single solution point only.

In real-world technical problems, engineers often need different alternatives or trade-offs for decision making. Moreover, if some of the objectives are noisy, have discontinuous variable space, or the Pareto-optimal solution front is non-convex, these methods may not work effectively. The most profound drawbacks of these methods are that

1. they require knowledge of individual optima or the attainable objective space at all prior to starting the optimization,

2. solutions obtained largely depend on the method settings, e.g. underlying weight-vector for the method of weighted-objectives, the chosen metric for distance method, and the step sizes for compromise method,

3. the same problem needs to be solved a number of times with variable parameter settings in order to find as much points on the Pareto-front as possible.

Nevertheless, using scalarization methods is sometimes a good choice if a quick

single solution to the problem is required or a classical single-objective optimization method is available only. It is always the user's responsibility to choose a proper method and proper settings according to the problem definition.

2.3.4 Optimization Algorithms

Once objective, design parameters, and constraints are formulated an optimization algorithm must be chosen for solving the problem. There is no universal optimization algorithm, rather there are numerous algorithms available which are tailored to particular types of optimization problems. The choice of an appropriate algorithm is an important one; it determines whether the problem is solved rapidly or slowly and, indeed, whether the solution is found at all.

Figure 2.12 shows a simple minimization problem with a smooth and continuous function depending on a single variable. In spite of its simplicity, different extrem values for the non-convex objective function f exist, and in terms of numerical optimization, finding the best solution to this particular multi-modal problem is not as trivial as it seems to be. Indeed, various optimization algorithms will behave in a different manner and the right choice of the optimization method is rather important for finding the optimum. Algorithms which are searching based on descend directions would typically find the next local minimum to the starting point which is not necessarily the global minimum, whereas global optimization algorithms have a strategy based on a stochastic approach which makes them able to find the global minimum to the problem.

In the last decades a tremendous number of publications has been published presenting new optimization algorithms, variants of existing methods as well as hybrid methods which combine two or more algorithms. In some classical text books, *Bestle (1994)*, *Coello Coello et al. (2002)*, *Fletcher (2000)*, *Gill et al. (1995)*, *Rao (1996)*, *Moré and Wright (1994)*, classifications and descriptions of optimization algorithms are available which give an overview of existing methods, and which can basically help choosing the right algorithm for a specific type of optimization problem. In the following section one of these classifications will be discussed in more detail.

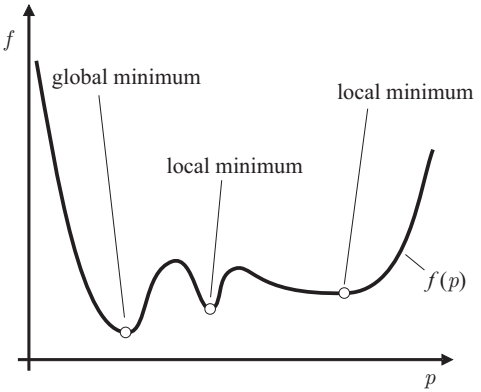


Figure 2.12: Illustration of global and local minima for a simple one-dimensional function

2.3.4.1 Classification of Optimization Algorithms

One possible method of classifying optimization algorithms can be derived from their working principles. An optimization procedure starts typically from an initial design characterized by the design vector $\mathbf{p}^{(1)}$. During optimization a series of designs are created which decrease the objective function value (for the case of minimization) step by step, i.e. $f(\mathbf{p}^{(k+1)}) < f(\mathbf{p}^{(k)})$. The general form of this series is given by

$$\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}, \tag{2.23}$$

with the step size $\alpha^{(k)} \geq 0$ and the search direction $\mathbf{d}^{(k)}$. In general, optimization algorithms differ in how they determine the next design vector $\mathbf{p}^{(k+1)}$. They can be globally distinguished as deterministic or stochastic algorithms, see Figure 2.13.

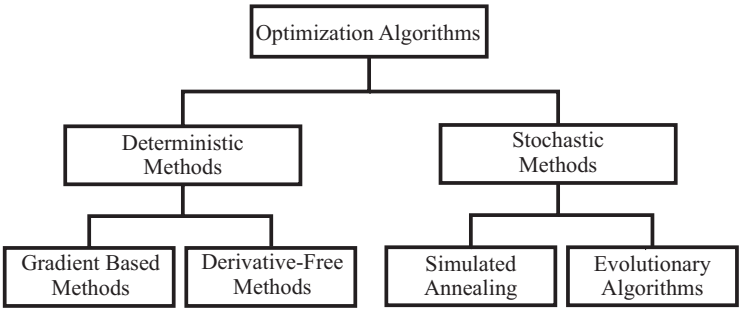


Figure 2.13: Classification of optimization algorithms

2.3.4.2 Deterministic Algorithms

Deterministic algorithms search according to clearly defined rules based on algebraic or analytical schemes. The optimization procedure can be described by the following steps:

1. Start with a given design vector $\mathbf{p}^{(k)}$ and evaluate its corresponding function value $f(\mathbf{p}^{(k)})$.
2. Determine a search direction $\mathbf{d}^{(k)}$.
3. Solve a one-dimensional line search problem in order to find the right step size $\alpha^{(k)}$.
4. Calculate the new design vector by $\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}$ and its function value $f(\mathbf{p}^{(k+1)})$.
5. Repeat steps 2-4 with the new design point until convergence is achieved.

Most deterministic algorithms are working based on this concept. The differences between the individual methods rely on the determination of the search direction and on the line search procedure. In the past, much effort was undertaken for a precise determination of the step size $\alpha^{(k)}$, where the line search procedure has been formulated as an independent minimization task within the optimization step. This time-consuming process is not required in today's algorithms where typically the line search is performed on simple techniques as

the Armijo-rule, the golden section method and sometimes on spline or parabola approximation models, *Bestle (1994)*. It has been shown that a better determination of the search direction is more important than the line search in terms of higher convergence rates and better solution finding. Therefore, it is better to distinguish the different algorithms based on their search direction determination.

In general, deterministic algorithms can be subdivided into *gradient based methods* which compute the search direction on the basis of calculated or approximated gradients and sometimes even on curvature information, and the more simple *search methods* or *derivative-free methods* which determine the search direction by evaluating function values only.

Derivative-Free Methods

The most common *derivative-free methods* are the *Pattern-Search-Methods*, *Lewis et al. (2000)*, the *Simplex-Method* developed by *Nelder and Mead (1965)*, the *Hooke-Jeeves-Method*, *Hooke and Jeeves (1961)*, and the *Conjugate-Directions-Method*, *Fletcher and Reeves (1964)*. Their advantage in comparison with other methods is that they are not requiring any gradient or curvature information. Their search algorithm is based on evaluation of the objective function value only while in some cases even the actual numeric value of the objective function is unimportant, but it is sufficient to know whether a design is better than another or not, *Birk (2003)*. This makes them suitable for optimization problems with expensive and noisy objective functions where an approximation of the gradients is too costly or too inaccurate. These search methods were used in the past and are still used for solving specific types of real-world optimization problems. Today, however, these methods are not the best choice for solving engineering applications which are characterized by a huge number of design variables and constraints, and where the entire optimization time for solving the design task plays an essential role.

Gradient Based Methods

Better approaches in terms of optimization time are the more sophisticated gradient based methods which are superior to derivative-free methods to some extent. They are used to solve a whole bunch of engineering design problems due to their

higher convergence rate which is mainly driven by the knowledge of local properties of the objective function reflected by the gradient and curvature information.

As mentioned before, the main idea of gradient based methods is to determine the search direction $\mathbf{d}$ at each iteration by evaluating the derivatives ∇f of the objective function with respect to the design variables $\mathbf{p}$. It can be shown that many gradient based methods can be specified in the following manner:

$$\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)} - \alpha^{(k)} \mathbf{G}^{(k)} \nabla f^{(k)} \tag{2.24}$$

where $\alpha^{(k)}$ is again a step size parameter, $\mathbf{G}^{(k)}$ is a positive definite matrix, and $\nabla f^{(k)}$ is the gradient for $\mathbf{p}^{(k)}$. First order gradient based methods require only first derivatives of the objective function in order to evaluate Equation (2.24) where second order models additionally have to evaluate second derivatives.

The question that arises at this point is how to calculate the derivatives? If the objective function is given by an analytical function, the derivatives can be calculated exactly. This is the best and most accurate way, however, in many engineering applications an explicit formulation for the objective function is not available, and therefore derivatives have to be approximated numerically. The simplest approach for obtaining derivatives is using finite differences which are based on a Taylor series expansion truncated after a specific term, *Chapra and Canale (2001)*.

The first derivative of the objective function f about the point $\mathbf{p}_0$ can be approximated by forward differences

$$\left.\frac{\partial f}{\partial p_i}\right|_{\mathbf{p}_0} \approx \frac{f\left(\mathbf{p}_0 + \Delta p\, \mathbf{e}_i\right) - f\left(\mathbf{p}_0\right)}{\Delta p}, \quad i = 1(1)n, \tag{2.25}$$

where $\mathbf{e}_i$ is the unit vector in i direction, and Δp is a small parameter perturbation. The approximation error due to truncation of the Taylor series after the first term is linear, i.e. halving the parameter perturbation yields halved truncation error. A better way for approximating the first derivatives is the central difference method

$$\left.\frac{\partial f}{\partial p_i}\right|_{\mathbf{p}_0} \approx \frac{f\left(\mathbf{p}_0 + \Delta p\, \mathbf{e}_i\right) - f\left(\mathbf{p}_0 - \Delta p\, \mathbf{e}_i\right)}{2\, \Delta p}, \quad i = 1(1)n, \tag{2.26}$$

which is more accurate. In contrast to the forward difference approximation the truncation error here would be quartered for halving the perturbation.

Applied to an optimization problem the number of function evaluations increases significantly with the number of design parameters. For a full determination of the gradient ∇f by using forward differences, $n+1$ function evaluations are necessary for a n-dimensional design problem, and for the more accurate central differences $2n+1$ calculations are required at each optimization step. Besides this, it is essential that the objective function exists for all perturbed designs, i.e. that e.g. iterative computations converge and yield results, the objective is not too noisy, and the parameter perturbation is chosen appropriately.

A further important point to consider is that in gradient based methods the objective function as well as constraints are assumed to be smooth which means that they are twice continuously differentiable. The problem in most engineering applications, however, is that this property is often à priori unknown, but nevertheless it is common practice to assume smoothness and to check later if the algorithm converges properly and produces reasonable results.

At this point some of the most common gradient based algorithms should be discussed. A more detailed overview of existing algorithms and comparisons with other methods can be found in classical text books like *Gill et al. (1995)* and *Fletcher (2000)*.

Probably one of the best known first order gradient based algorithms is the method of *Steepest Descent* in which it is assumed that the best search direction is where the objective function value decreases most rapidly. This is given by the negative gradient at each step k, i.e. $\mathbf{d}^{(k)} = -\nabla f^{(k)}$. If the positive definite matrix $\mathbf{G}^{(k)}$ is set equal to the identity matrix $\mathbf{I}$, Equation (2.24) with the step size $\alpha^{(k)}$ becomes

$$\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)} - \alpha^{(k)} \nabla f^{(k)}. \tag{2.27}$$

Despite their linear convergence property these type of methods usually exhibit a quick convergence at the beginning which then leads to oscillatory or zig-zagging behavior, and usually the algorithms terminate far from the solution owing to round-off effects. Therefore in practice they are inefficient and unreliable, *Fletcher (2000)*.

More powerful are the *Newton* or *Newton-Raphson* type second order gradient based methods which are based on the gradient as well as on curvature information. Within these methods, the positive definite matrix $\mathbf{G}^{(k)}$ is set equal to the inverse Hessian matrix $\mathbf{H}^{(k)^{-1}}$, with $\mathbf{H}^{(k)} = \nabla^2 f^{(k)}$, and the next design point is determined by

$$\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)} - \alpha^{(k)} \mathbf{H}^{(k)^{-1}} \nabla f^{(k)}. \tag{2.28}$$

The benefit of these second order algorithms is that they show quadratic convergence, and hence they are the fastest known methods. In contrast to the *Steepest Descent Methods*, the computational costs for the Hessian matrix is a major drawback of these algorithms and, especially if the second order derivatives are also approximated by finite differences, the computational time for evaluating the objective function f increases extremely, *Birk (2003)*.

A good compromise between the first and second order gradient based methods are the *Quasi-Newton Methods* which are characterized by a good convergence rate and relatively low computational costs for the search direction. The idea behind these types of methods is to avoid the expensive determination of the exact Hessian matrix at every step either by calculating the Hessian only every few steps or by approximating the Hessian or its inverse by a symmetric positive definite matrix $\mathbf{B}$. In the latter approach, the matrix $\mathbf{B}$ is initially set equal to the identity matrix $\mathbf{I}$ and while the optimization proceeds it is corrected or updated from iteration to iteration using an appropriate update scheme. The key point is that most update schemes are using the information of first derivatives only which has positive influence on the overall computational time. The *Davidon-Fletcher-Powell* (DFP) and *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) update schemes should be mentioned at this point as common methods within numerous algorithms, *Fletcher (2000)*.

The gradient based algorithms described up to this point are basically developed for unconstrained optimization problems. However, some methods exist to use them also if constraints are introduced. One possibility is by replacing the constrained optimization problem with an unconstrained one by adding a penalty function to the objective function that depends on the value of the violated constraints. This method is called *Penalty-Method* and depending on the

formulation of the penalty function, it can be distinguished between *interior* and *exterior Penalty-Methods*. In general, these methods provide an easy way to consider constraints in an optimization problem which makes them suffice for special purposes. Nevertheless, some major drawbacks exist why they are not suitable for general applications, *Bestle (1994)*.

The current state of the art for solving constrained optimization problems with gradient based methods are the *Lagrange-Newton* type methods which are also known as *Sequential Quadratic Programming* (SQP) algorithms due to their mathematical background. The basic idea of these algorithms is to find an optimal search direction $\mathbf{d}^{(k)}$ by solving a quadratic subproblem considering linear constraints in each iteration. Compared to a simple search method, the determination of the search direction based on solving quadratic subproblems requires significantly more computational effort. However, these methods show a very quick convergence and are highly efficient in solving engineering optimization problems, *Birk (2003)*.

2.3.4.3 Stochastic Algorithms

The idea of stochastic algorithms is generally different compared to deterministic methods. They use a stochastic approach in order to find a better design instead of any gradient information. This property makes them capable to find a global optimum to the design problem and not to be trapped in a local optimal solution. Since the determination of the next design parameters is usually not pure random, these types of algorithms are also called semi-stochastic or heuristic algorithms. Their benefit compared to gradient based methods is basically that they are able to cope with a wide range of features in optimization problems like discrete and continuous design variables, noisy or discontinuous objective functions, and multi-modal problems. However, *Stochastic Algorithms* are also considered to be computationally expensive in terms of the required number of evaluated solutions for convergence, *Büche (2004)*. In the following, *Simulated Annealing* and *Evolutionary Algorithms* as two well known representations of stochastic algorithms will be discussed.

Simulated Annealing

Simulated Annealing methods are stochastic algorithms which mimic the recrystallization of a liquid metal during annealing. Heated up to high temperatures, the atomic structure becomes disordered which makes the atoms capable of moving around in the melt. The melt is kept near its thermodynamic equilibrium when it is slowly cooled down. This drives the individual atoms to reach the minimum energy state and the nearly perfect crystalline structure is formed, i.e. the global minimum is found. The initial situation as well as the cooling process itself are important factors within the annealing procedure. If the initial temperature is too low or the cooling process too fast, the structure becomes frozen before it reaches the desired minimum energy state and a local minimum to the problem is found, *Birk (2003)*.

Transferred to an arbitrary engineering optimization problem, *Simulated Annealing* methods try to find the global optimum to the design problem by a sequence of numerical steps. The algorithm starts with an initial temperature $T^{(k=0)}$ and an initial design point which in the beginning is also the best known solution $\mathbf{p}^* \equiv \mathbf{p}^{(k=0)}$. A new design is created randomly in the neighborhood of the best point while the random disturbance is larger for higher temperatures. The function value of the new design is compared with the best solution and if it is better, i.e. $\Delta = f(\mathbf{p}^{(k+1)}) - f(\mathbf{p}^*) < 0$, then this is taken as new best design and the next iteration can start with a lower temperature $T^{(k+1)} < T^{(k)}$ resulting in a smaller random disturbance. Otherwise the new design is accepted as best design only with a lower probability depending on the temperature. The algorithm proceeds until the minimum temperature level is found, *Kirkpatrick et al. (1983)*. In contrast to deterministic optimization algorithms, new designs with higher function values are accepted with some probability which is the reason why *Simulated Annealing* is able to leave already established local minima to reach the global minimum.

Evolutionary Algorithms

Another group of stochastic methods are called *Evolutionary Algorithms*. In general these algorithms are inspired by the principles of natural evolution to find an optimal solution to a problem. Natural evolution is driven by the principles

of selection, recombination and mutation of genetic information. Individuals in a population which are well adapted to their environment have a higher probability to survive in the nature, known as *'survival of the fittest'*. These individuals are declared with a higher fitness value and are chosen in order to become parents (selection) which produce offsprings for the following generation. The genetic information of the offspring is either a direct copy of the genes of just one single parent, which differs from the natural evolution, or results from the mating process of multiple parents (recombination or crossover). In the latter case the gene of the offspring is arranged form gene sequences of both parents. Additionally, a randomly generated mutation can modify the genetic information of the offsprings (mutation) and the best solutions are selected in the selection process. Figure 2.14 shows the principle work flow of an evolutionary algorithm consisting of crossover, mutation, and selection.

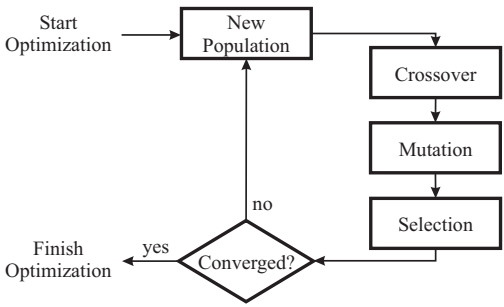


Figure 2.14: Principle work flow of an evolutionary algorithm

Applied to engineering design problems, the genetic information corresponds to the design variables which specify the properties of a solution to the engineering optimization problem, and the fitness of a solution is either determined directly by the objective function or by a combination between the objective function value and the constraints.

It should be mentioned that *Evolutionary Algorithms* (EA) are building an upper class of algorithms containing the subgroups of *Genetic Programming* (GP), *Genetic Algorithms* (GA), and *Evolutionary Strategies* (ES), *Bäck et al. (1997)*. *Genetic Algorithms* were firstly proposed and applied by *Holland (1975)* while *Rechenberg (1973)* developed *Evolutionary Strategies* independently and applied

them to some engineering design problems. Although based on a similar idea, both approaches were different in some aspects at the beginning of their development. In *Genetic Algorithms* an individual is represented by a string of bits and the evolutionary process is based on selection, recombination, and mutation techniques. In contrast to this, in *Evolutionary Strategies* each individual of the population is represented by a vector of real design variables and the evolutionary process is characterized by selection and mutation techniques only. However, nowadays modern algorithms are mixed up with both ideas and hence it is often not simple to classify them anymore.

2.3.4.4 Algorithms Used in this Thesis

Since all of the described strategies have their strengths and weaknesses, different types of algorithms have been chosen to solve the optimization problems within this thesis. Especially for complex design problems like aerodynamic compressor design it is not à priori known which method will work best. Thus, one goal of the thesis is to gather experience in the behavior of different strategies on the various subtasks. The selected optimization algorithms are as follows:

Lagrange-Newton Method NLPQL

The *Nonlinear Programming with Quadratic Line Search* algorithm (NLPQL) was developed by *Schittkowski (1986)*. It is a Lagrange-Newton type algorithm for solving smooth, nonlinear, constrained optimization problems, i.e. minimizing a nonlinear objective function subject to nonlinear equality and inequality constraints. It is assumed that all model functions are continuously differentiable. The internal algorithm is a SQP method with a quadratic approximation of the Lagrange function and a linearization of the constraints. Each iteration step starts with a determination of the search direction by solving a quadratic programming subproblem with an approximation of the Hessian matrix using the BFGS update scheme and a subsequent line search which is performed for determination of the step length α, *Schittkowski (1981)*.

Single-Objective Stochastic Method MIGA

When genetic algorithms are applied to an engineering design application, they require many generations and a large number of individuals in the population in order to obtain good solutions. The idea of the *Multi-Island Genetic Algorithm* (MIGA), *Miki et al. (2000a)*, which is a single-objective optimization algorithm, is to accelerate the solution finding process by dividing the large population into smaller sub-populations like on islands and to execute traditional genetic operations on each sub-population separately. The algorithm then periodically selects individuals from each sub-population and moves them to other sub-populations in an exchange called migration. Two parameters are basically driving the migration process: the migration interval which is the number of generations between each migration process, and the migration rate which is the percentage of individuals selected for migration from each sub-population at the time migration occurs. The emigrants are selected randomly in their sub-populations and the migration topology is typically selected as a ring with random destinations where each sub-population has one destination and the destinations are determined randomly at every migration period, *Kaneko et al. (2000)*.

In general it can be shown that genetic algorithms with a distributed population show better performance in terms of convergence and fitness values than a single population GA. Furthermore, this approach is suitable to be implemented on parallel computers, because the communication between the processors occurs only in the migration phase. Therefore, if each sub-population is assigned to one processor of a parallel computer, a nearly linear reduction in speed can be expected, *Miki et al. (2000b)*.

Multi-Objective Stochastic Method NSGA-II

The *Non-dominated Sorting Genetic Algorithm II* (NSGA-II) is a multi-objective genetic algorithm developed by *Deb et al. (2000)* and is a revised version of NSGA, *Srinivas and Deb (1995)*. In most aspects, the algorithm does not have much similarity with the original NSGA, but the authors kept the name NSGA-II to highlight its genesis and place of origin.

Figure 2.15 shows the basic idea of the algorithm. In NSGA-II the offspring population Q_t which is created by the parents and the parent population P_t

itself are combined together to form an overall population R_t. A non-dominated sorting algorithm based on the vector criterion of each individual is performed on R_t. The new population P_{t+1} is filled by solutions of different non-dominated fronts F_i starting with the best non-dominated front F_1 and continuing with the next ones until the generation size is maintained. It may happen that the number of individuals in the last considered solution front is bigger than the available slot size. Instead of arbitrarily discarding some members from the last front, a crowding algorithm is performed which estimates the distance of these solutions to each other. The solutions of the last front with the highest diversity, i.e. widely spread in the objective space, are included in the new population P_{t+1} whereby the rest of them are rejected. The next offsprings Q_{t+1} are created based on the new population P_{t+1} by tournament selection, recombination and mutation operators, *Deb (2001)*.

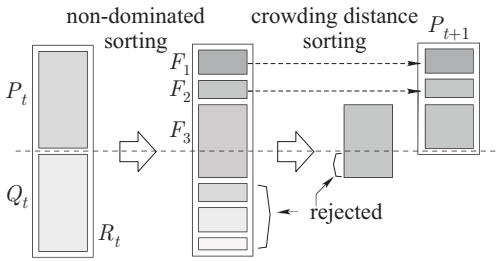


Figure 2.15: Illustration of the NSGA-II procedure, *Deb (2001)*

The NSGA-II is currently one of the most popular multi-objective genetic algorithms. It basically consists of two major aspects: an *elite-preserving* operation and a crowding algorithm. The non-dominated sorting algorithm carries over the best solutions denoted as the elite of a generation to the next generation. Hence, the fitness and the corresponding values of the objective functions of the best solutions do not deteriorate and furthermore the best solutions found early in the optimization process will never be lost unless a better solution is discovered. The crowding algorithm is important to drive the required diversity of non-dominated solutions and to distribute the solutions on the Pareto-front. Both aspects together makes the NSGA-II robust, reliable and applicable to many engineering design problems.

3 Optimization Based Preliminary Design

The aerodynamic compressor design process starts with the preliminary design phase, where new design ideas or philosophies are investigated in terms of feasibility and improvement. The calculation results, which are generally based on analytical equations, give a rough overview on performance and stability criteria that can be achieved for a specific compressor design. The calculation time is rather short, however, a complete investigation of different design studies is manually too expensive due to the huge number of design parameters and constraints.

On the basis of a given Rolls-Royce preliminary design tool, process integration and automation is applied in order to accelerate the meanline prediction process and to support the design engineer in time-consuming parameter studies. Numerical optimization is performed with the MIGA, NLPQL, and NSGA-II algorithms to improve a given design of a 9-stage high pressure research compressor, *Klinger (2004)*, with respect to conflicting design goals as efficiency, surge margin and overall pressure ratio. Based on design conditions, design changes are made with respect to the annulus geometry and the stage pressure ratio distribution.

3.1 Introduction

Meanline prediction is the first step of the complex aerodynamic compressor design process. The goal is to provide a robust and reliable instrument for a first guess and proper choice of design parameters for design and off-design conditions. The performance that can be achieved by a well-designed multi-stage axial

compressor is mainly determined by the choices made for a number of global parameters. No amount of subsequent development effort can correct a poor design where the basic selection of global parameters is inconsistent with the design objectives. On account of this, meanline prediction as a preliminary design procedure setting the values of these global parameters is one of the very essential steps of the entire aerodynamic compressor design process, *Keskin and Bestle (2005)*.

Within the preliminary design phase, calculations are performed along the mid-height line of the compressor and the main aerodynamic and geometric parameters are determined, Figure 3.1. The process typically starts based on an old initial axial compressor design and new performance requirements. Operating conditions and geometric constraints in term of minimum or maximum radii of the annulus or even sometime the maximum compressor length are à priori given or restricted and have to be adjusted or adapted to the new design problem. Basically, the goal is to find appropriate design parameter distributions along the one-dimensional mid-height streamline of the compressor which fulfill the design requirements and constraints as good as possible.

The entire suite of available parameters describing the whole compressor model can be split into design variables which are assumed to be adjustable within the design process in order to fulfill design requirements, and system constants which are invariant during the design calculation. The selection of the design variables and the system constants depends on the design goals, the sensitivities of the objective functions on parameter variations, and obviously on the knowledge and expertise of the design engineer.

If a manual search for a good design is performed, only few parameters are taken into account and small variations on the design variables are applied. Obviously, it is too complicated for a human design engineer to understand the dependence of the outputs on the inputs and if a large number of design parameters are changed simultaneously and many criteria and constraints have to be considered. This manual search technique is rather time-consuming and the success depends significantly on the experience of the engineer.

Therefore, it should be the overall aim to implement the preliminary design process into a common design environment and to accelerate the design process by automation of the engineering work flow. Due to the rather complex design

process with a huge number of design parameters and a high number of design criteria for design and off-design conditions, it is more realistic at present to improve the design process by partial automation of a specific work flow. When the design process runs automated and the design parameters, objectives, and constraints are defined properly, numerical optimization can be applied to support the design engineer in solution finding and decision making. Especially if conflicting criteria exist, multi-objective optimization methods can find Pareto-optimal trade-offs from which the design engineer can choose one or more solutions for the particular design problem afterwards.

3.2 Design Problem

As already mentioned, many different design goals and a huge number of constraints have to be pursued in parallel in order to achieve a good preliminary aerodynamic compressor design. It depends on the problem formulation if parameters as the compressor length, the number of stages, the overall pressure ratio, and many more have to be considered as constraints or as objectives to be optimized in this design task. The most important aerodynamic parameters, however, are the efficiency and the stability of the compressor reflected by the surge margin value which always have to be maximized at reasonable pressure ratio levels.

Efficiency, surge margin, and overall compressor pressure ratio are conflicting criteria in the design process and an optimization based on one objective only would typically reduce the others, *Keskin and Bestle (2005)*. In terms of aerodynamic compressor design it is better to consider all three criteria in a multi-objective manner or at least to introduce lower bounds to one or two of them in order to achieve acceptable objective levels.

Based on a Rolls-Royce preliminary design tool a 9-stage high pressure compressor is being optimized. The three major design criteria in the following investigations are

- overall polytropic efficiency $\eta_{c,poly}$ (A.5),
- overall compressor pressure ratio Π_c (A.8), and

- surge margin SM (A.10).

It should be mentioned that all three objectives are direct results of the Rolls-Royce meanline prediction calculation program and that especially the surge margin calculation is based on correlations.

Beside these criteria, which are maximized, some constraints have to be considered with respect to blade loading, flow conditions and stability measures in order to achieve reasonable solutions. More precisely, the

- stage loadings (A.9) have to be kept below a user-defined value $\hat{\Psi}$ for all $N_s = 9$ stages, i.e.

$$\Psi_i \leq \hat{\Psi}, \quad i = 1(1)N_s, \tag{3.1}$$

- relative rotor and absolute stator inlet Mach numbers should not be too high in order to avoid shock losses, i.e.

$$M_{I,i}^{\prime R} \leq \hat{M}_I^{\prime R}, \quad i = 1(1)N_s, \tag{3.2}$$

$$M_{I,i}^{S} \leq \hat{M}_I^{S}, \quad i = 1(1)N_s, \tag{3.3}$$

- compressor exit Mach number has to be limited to avoid flame-out in the combustion chamber, i.e.

$$M_{E,N_s}^{S} \leq \hat{M}_E, \tag{3.4}$$

- the Koch parameter (A.6) should not exceed a user defined stability margin, i.e.

$$C_{h,i} \leq \hat{C}_h, \quad i = 1(1)N_s, \tag{3.5}$$

- rotor and stator diffusion numbers (A.3) have to be bounded by a constant value guaranteeing flow stability, i.e.

$$DF_i^R \leq \widehat{DF}, \quad i = 1(1)N_s, \tag{3.6}$$

$$DF_i^S \leq \widehat{DF}, \quad i = 1(1)N_s, \tag{3.7}$$

- and rotor and stator de Haller numbers (A.1) have to be higher than a specific constant value required for flow stability, i.e.

$$DH_i^R \geq \widehat{DH}, \quad i = 1(1)N_s, \tag{3.8}$$

$$DH_i^S \geq \widehat{DH}, \quad i = 1(1)N_s. \tag{3.9}$$

For a 9-stage compressor this would sum up to 73 inequality constraints which have to be taken into account by the optimization routine. Depending on the optimization algorithm applied to this problem alternative definitions of the constraints may be used. If the optimizer uses an active set strategy, which is the case with most SQP algorithms, the constraints can be used as defined, because the algorithm will concentrate on active and violated constraints automatically. Else, it is better to group the constraints by minimization and maximization over all stages which means that only the maximum or minimum value of a parameter is considered, respectively. This reduces the number of constraints dramatically and may help the optimizer to find the solution to the problem quicker. The possible non-smoothness of such an approach does not hurt especially if genetic algorithms are used for optimization.

The entire multi-objective optimization problem can then be defined by summarizing the objectives and constraints as

$$\underset{\mathbf{p} \in \mathbb{R}^n}{max} \begin{bmatrix} \eta_{c,poly} \\ SM \\ \Pi_c \end{bmatrix} \tag{3.10}$$

subject to

$$
\begin{aligned}
&\max_i \Psi_i \leq \hat{\Psi} && \max_i DF_i^R \leq \widehat{DF} \\
&\max_i M_{I,i}^{tR} \leq \hat{M}_I^{tR} && \max_i DF_i^S \leq \widehat{DF} \\
&\max_i M_{I,i}^S \leq \hat{M}_I^S && \min_i DH_i^R \geq \widehat{DH} \\
&\max_i C_{h,i} \leq \hat{C}_h && \min_i DH_i^S \geq \widehat{DH} \\
&M_{E,N_s}^S \leq \hat{M}_E
\end{aligned}
$$

where the vector $\mathbf{p}$ contains the free design parameters within the optimization problem which can be modified by the optimization algorithm.

3.3 Parameterization

As already mentioned two categories of input parameters exist, the design parameters which are changeable by the optimization algorithm in order to achieve the design goals, and system constants which are invariant parameters or settings required for the calculation. Typical design variables within the preliminary design process are the

- coordinates (x, r) of the annulus geometry,
- axial distribution of the stage pressure ratio Π_i of each stage,
- number of blades $N_{b,i}$ for each rotor and stator,
- corresponding blade aspect ratios H_i/C_i for each blade row,
- blade solidities σ_i for each blade row,
- maximum thickness to chord ratios T_i/C_i for each blade row and
- tip or hub clearances δ_i for each blade row, respectively.

Some of these variables are redundant, e.g. the solidity of a given blade can be determined by its corresponding mid-height radius and the number of blades and vice versa. The blade aspect ratios are typical design variables, however, they could also be calculated by the blade inlet and exit positions, the mean annulus height, and the clearance values. The design engineer can choose which

parameters should be used as variable based on his experience or the availability of the data.

The system constants are parameters describing the compressor inlet flow conditions, i.e.

- inlet mass flow $\dot{m}_{c,I}$,
- inlet total pressure $P_{0,c,I}$,
- inlet total temperature $T_{0,c,I}$,
- inlet whirl angle $\alpha_{c,I}$,

and additional parameters regarding the operating conditions of the compressor:

- shaft speed,
- number of bleeds,
- position of bleeds,
- bleed mass flow.

These values are à priori given and are characterizing the design point conditions of the compressor.

The Rolls-Royce preliminary design program provides different models for the surge and the loss assumptions which can be chosen through the model parameter settings within the input file. Depending on the selection, appropriate equations and correlations are used and applied to the aerodynamic compressor design problem within the meanline prediction calculation process.

Figure 3.1 shows compressor design parameters for the meanline prediction calculation. In the upper part of the figure important parameters for the annulus geometry definition as well as geometric parameters such as blade height H and blade clearances δ are demonstrated for a single stage of a multi-stage compressor design. In the lower part velocity triangles for the same single stage are shown for each blade inlet and outlet position which describe the velocity relation in the relative and absolut frame, where α and α' are the velocity angles in the absolute and relative frame, respectively, c and w are the corresponding absolute and relative velocities, and u is the circumferential velocity. Furthermore, some important

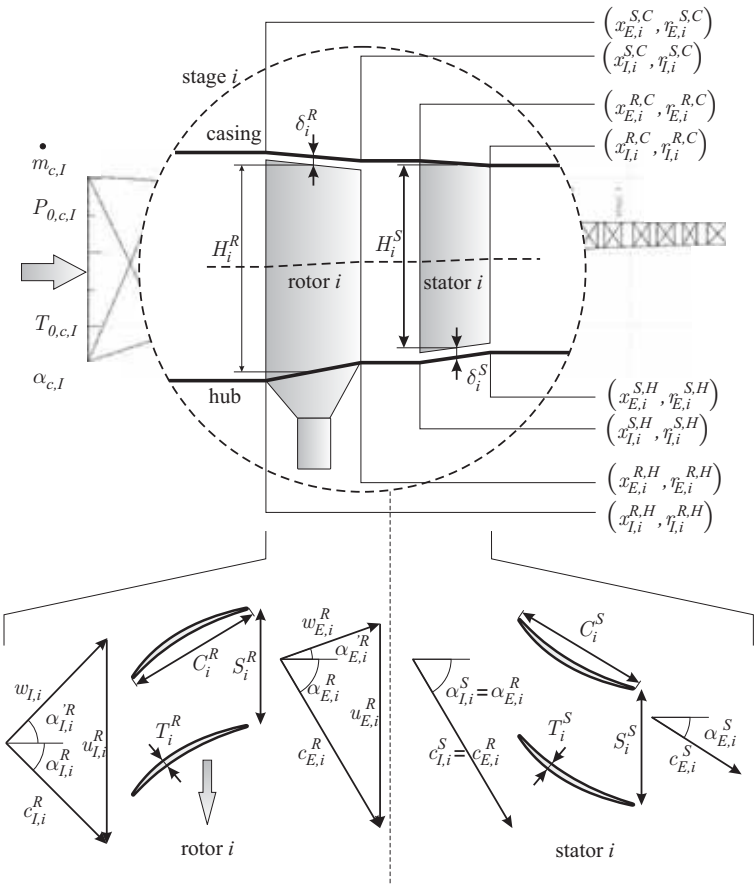


Figure 3.1: Input parameters for meanline prediction design

geometric parameters such as chord length C, maximum blade thickness T and blade pitch S are shown which are considered for different calculations.

One of the most time-consuming design tasks within the preliminary design procedure is the annulus geometry design. The annulus shape is described by coordinates for the inner hub $\left(x^H, r^H\right)$ and outer casing $\left(x^C, r^C\right)$ geometry line, respectively. The number of coordinates depends on the number of blade rows

while each row consists of four coordinates describing the blade corner points. The annulus geometry for a compressor with N_s stages would thus be described by $8N_s$ corner points or in 2D-space by $16N_s$ design parameters.

Some restrictions on the preliminary calculation program reduce the number of parameters describing the annulus geometry. For the used calculation program contractions of the annulus inner and outer lines are possible within blade inlet and exit positions only. Thus, the radial coordinates of the downstream inlet blade row are equal to the exit radial coordinates of the previous row, i.e. $r^H_{E,i} = r^H_{I,i+1}$ and $r^C_{E,i} = r^C_{I,i+1}$, see Figure 3.1. Additionally, tapering of the blades is not considered within the calculation which means that the axial coordinates of the outer annulus line are equal to the corresponding inner coordinates at each axial position, $x^C_i = x^H_i$. Altogether these restrictions reduce the number of design parameters to $10N_s$ which is, however, still a rather large number and may cause problems.

Firstly, in terms of numerical optimization many design parameters, multiple design objectives, and high number of constraints lead to an unmanageable task and rise the computational time dramatically. Secondly, free choice of the x and r coordinates may cause technically infeasible or rapidly changing annulus geometries. Hence, it is better to use smoothing parameterization methods in order to decrease the number of design parameters without reducing the design freedom too much, and additionally to guarantee technical feasibility of the resulting design. This will also reduce the chance to be trapped in a local minimum and thus increases the possibility of finding a global optimum to the design problem.

Parameterization should be done carefully, since it is imposing implicit constraints on the design problem and thus could lead to sub-optimal solutions. Therefore, a trade-off between the maximum design freedom and the minimum number of parameters has to be found.

For the current investigation Bézier-curves (2.3) with $n+1$ control points are chosen to parameterize the annulus geometry and the total pressure ratio distribution which are curves in 2D-space. The mathematical simplicity and the properties regarding continuity and differentiability of Bézier-curves guarantee smooth distributions which are desired in particular for the annulus geometry. As already mentioned, the annulus geometry is determined by two curves, the inner hub line $r^H(x)$ and the outer casing line $r^C(x)$. The two curves may be parameterized

in different ways: the easiest way is to use independent spline representations for the inner and outer line, respectively. The drawback of this method is that constraints have to be introduced to avoid intersection of the two lines without restricting their degree of freedom. As a second approach a parameterization of either the inner or outer annulus line and a further spline describing the thickness distribution $\Delta r(x) = r^C(x) - r^H(x)$ of the annulus would be possible. For this approach a constraint is also needed which ensures that the thickness does not become negative, however, this is much easier to implement. The same constraint is required for the third approach which is associated with classical blade profile parametrization, where the annulus geometry is determined by a superposition of a parametric annulus mid-height line $r^M(x) = \left(r^C(x) + r^H(x)\right)/2$ and a parametric annulus thickness distribution $\Delta r(x)$, see Figure 3.2. If we compare the last two, the benefit of latter approach is that a aerodynamic engineer is usually more familiar with such kind of description.

In the following investigation the mid-height and thickness lines are described by

$$\mathbf{b}^M(t) = \begin{bmatrix} x(t) \\ r^M(t) \end{bmatrix} = \sum_{k=0}^{n} \begin{bmatrix} b^M_{x,k} \\ b^M_{y,k} \end{bmatrix} B^n_k(t) \tag{3.11}$$

and

$$\mathbf{b}^\Delta(t) = \begin{bmatrix} x(t) \\ \Delta r(t) \end{bmatrix} = \sum_{k=0}^{n} \begin{bmatrix} b^\Delta_{x,k} \\ b^\Delta_{y,k} \end{bmatrix} B^n_k(t) \tag{3.12}$$

with $B^n_k(t)$ according to (2.1) and $t \in [0, 1]$. The x-coordinates of the control points for $\mathbf{b}^M(t)$ and $\mathbf{b}^\Delta(t)$ are chosen independently in order to have full design freedom. This approach results in $4(n+1)$ design parameters describing the whole annulus geometry. If restrictions are imposed on the inlet and outlet position and height of the compressor annulus for compatibility with the low pressure compressor and combustion chamber, this can be easily handled by fixing the very left and right control point positions, i.e.

$$\begin{bmatrix} x^M_0 \\ r^M_0 \end{bmatrix} = \begin{bmatrix} x_I \\ \dfrac{r^C(x_I) + r^H(x_I)}{2} \end{bmatrix} \tag{3.13}$$

$$\begin{bmatrix} x_n^M \\ r_n^M \end{bmatrix} = \begin{bmatrix} x_E \\ \dfrac{r^C(x_E) + r^H(x_E)}{2} \end{bmatrix} \tag{3.14}$$

$$\begin{bmatrix} x_0^\Delta \\ r_0^\Delta \end{bmatrix} = \begin{bmatrix} x_I \\ r^C(x_I) - r^H(x_I) \end{bmatrix} \tag{3.15}$$

$$\begin{bmatrix} x_n^\Delta \\ r_n^\Delta \end{bmatrix} = \begin{bmatrix} x_E \\ r^C(x_E) - r^H(x_E) \end{bmatrix} \tag{3.16}$$

where x_I and x_E are given inlet and outlet axial coordinates of the compressor and $r^H(x_I), r^C(x_I), r^H(x_E), r^C(x_E)$ are given radial coordinates.

As a result of this approach, Figure 3.2 shows the parametric annulus mid-height (a) and thickness (b) distributions each parameterized by a Bézier-curve with $n+1=5$ control points. The resulting smooth annulus geometry and its meanline distribution is demonstrated in Figure 3.2c.

This annulus geometry, however, cannot be used directly with the Rolls-Royce preliminary design program. The hub and casing annulus lines have to be transferred into discrete point string data indicating the blade inlet and outlet positions. The problem here is that for given axial positions x the required radial positions r have to be determined from the parameterized curves $x(t), r(t)$.

This problem can be solved using a Newton method. Based on a given x-coordinate $\bar{x}$ of a blade inlet or exit position, the corresponding t-parameters $\bar{t}^M$, $\bar{t}^\Delta$ of the parameterized mid-height and thickness curves are found from $x(\bar{t}^{M,\Delta}) \stackrel{!}{=} \bar{x}$, where $x(t)$ is the first coordinate of the Bézier-curves (3.11) and (3.12), respectively. The corresponding radii are computed from the second coordinates of the Bézier-curves by

$$r^C(\bar{x}) = r^M(\bar{t}^M) + \Delta r(\bar{t}^\Delta)/2 \tag{3.17}$$

$$r^H(\bar{x}) = r^M(\bar{t}^M) - \Delta r(\bar{t}^\Delta)/2. \tag{3.18}$$

Figure 3.2d shows the final annulus geometry as it is used as input to the meanline prediction program.

Beside the annulus geometry other parameters as the overall pressure ratio play an important role in aerodynamic compressor design. However, the design task of the engineer within the aerodynamic design process is not only to guarantee the achievement of the required overall value, but it is more or less the

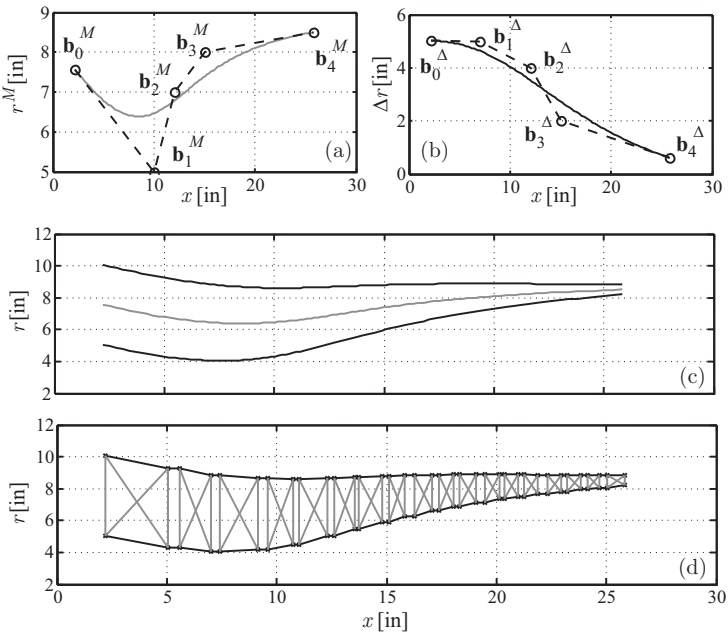


Figure 3.2: Annulus geometry definition by annulus mid-height line (a) and annulus thickness (b), resulting annulus geometry (c) and point string data (d) for a 9-stage compressor

search for the best distribution of the pressure rise without violating the loading constraints for each row.

For the given 9-stage, high pressure compressor application nine individual stage pressure ratio values have to be taken into account which leads to nine additional design parameters. If Equation (A.8) for a given overall pressure ratio is taken into account, the number of design parameters can be reduced to eight. An individual definition of each stage pressure ratio, however, does not guarantee the required smooth parameter distribution. Hence, a parameterization of this non-geometric curve based on the same kind of Bézier-curve as for the annulus line definition can be used which implies smooth and continues properties of the parameter distribution, Figure 3.3. The parametric stage pressure ratio

distribution can be determined by

$$\mathbf{b}^{\Pi}(t) = \begin{bmatrix} s(t) \\ \Pi(t) \end{bmatrix} = \sum_{k=0}^{n} \begin{bmatrix} b_{x,k}^{\Pi} \\ b_{y,k}^{\Pi} \end{bmatrix} B_k^n(t) \tag{3.19}$$

where $s \in [1, N_s] \subset \mathbb{R}$ is a continuous stage variable. For getting a complete distribution running through the whole compressor, the first and last control point has to satisfy the conditions

$$s(0) = 1, \quad s(1) = N_s. \tag{3.20}$$

Figure 3.3a shows the stage pressure ratio distribution of the 9-stage high pressure compressor parameterized with a Bézier-curve given by five control points. In order to define the pressure ratios of the individual stages, the values of $\Pi(t)$ have to be taken at discrete points $t = t_i$ where $s(t_i) = i$, $i = 1(1)N_s$, Figure 3.3b.

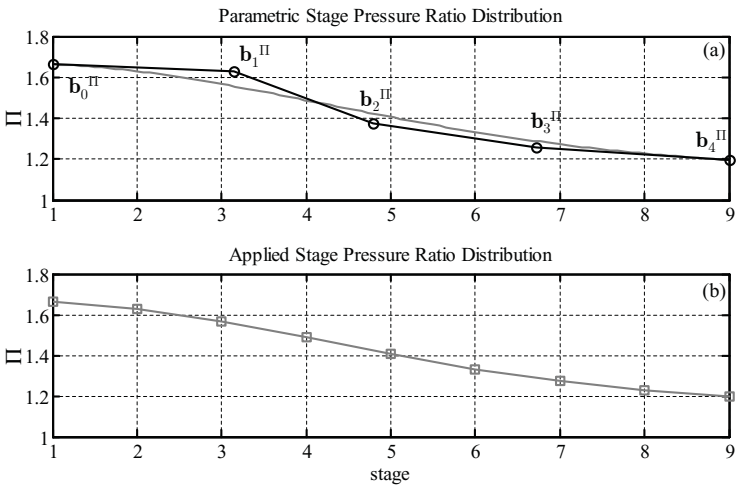


Figure 3.3: Stage pressure ratio parametrization (a) and distribution (b)

3.4 Process Integration

The Rolls-Royce meanline prediction process is a typical example for a computer-aided engineering design task. The preliminary design tool has been developed and validated by the company over the last decades supporting the engineer in his preliminary aerodynamic design activities. The classical engineering work flow starts with initial design parameters of an existing compressor provided by an input file. The preliminary design program is manually executed and the results are gathered in an output file from which the design criteria and constraints are extracted. If the design goals are missed, adjustments on the input parameters are made and the evaluation process is repeated until the design objectives are fulfilled.

In order to speed up this human driven design process, the meanline program has changed from a batch-program on a mainframe to an interactive analysis and design tool guiding the engineer through graphical user interfaces. In terms of process integration and automation, however, it is necessary to step back to batch-capable versions of the analysis program. If different design tools are involved within the design process, it is fundamental to define interfaces in order to convert different data formats and to execute the evaluation tools without time-consuming user interactions. The aim is to map the user actions onto transparent, flexible, and well-defined flow charts. Therefore, the commercial process integration and automation program *iSight* is used enabling to integrate the given Rolls-Royce meanline prediction program into a common design environment.

Figure 3.4 shows how the Rolls-Royce meanline prediction process is integrated in *iSight*. The whole process can be split into the process flow, the design evaluation, and the external programs used within the process. The automated engineering work flow starts with an initialization where required model and design parameters are prepared for the following optimization procedure. If the whole analysis process is done manually, many iterations my be necessary and the design loop has to be run several times with different adjusted design parameters. In the current integrated approach the process flow has to be run only once, while the optimization routine interacts with the design evaluation process several times in order to find an optimal design. The process flow as well as the design evaluation flow are defined and controlled by the *iSight* process integration

module. When the optimization algorithm requires a new design evaluation, a *Matlab* script is started automatically by *iSight* which calculates the new annulus geometry and stage pressure ratio distribution based on the new design parameters. This information is transferred into the meanline prediction input file and the Rolls-Royce meanline prediction program is invoked. After the calculation has converged, the results are bundled in an output file, where *iSight* extracts basic information required for further parameter evaluation. Based on the results, a second *Matlab* script is used for post-processing and calculation of the criteria and constraints. The optimization loop proceeds with a new design parameter vector until the desired optimized design is found.

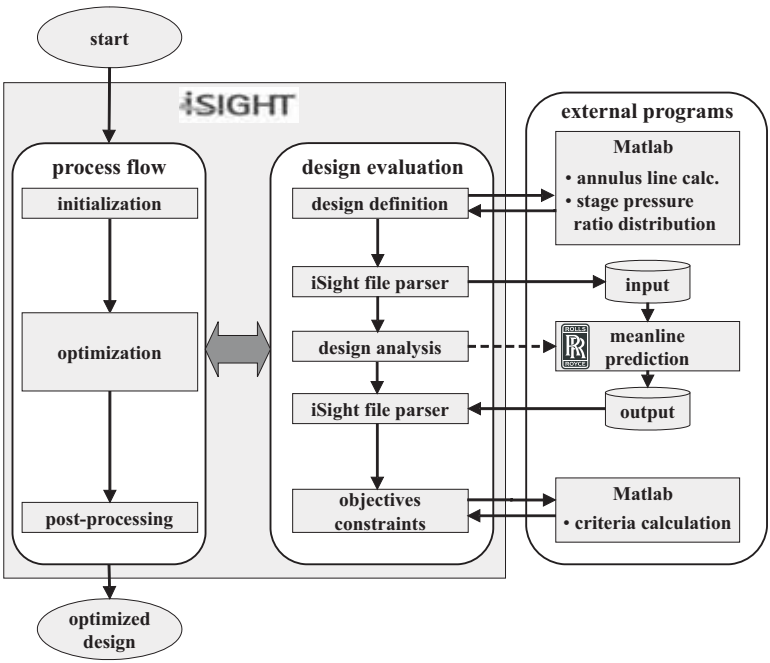


Figure 3.4: Integration of the meanline prediction process flow

3.5 Results and Discussion

In order to demonstrate the potential of process integration, automation and multi-objective optimization on preliminary aerodynamic compressor design, the annulus geometry of a given 9-stage high pressure compressor design is optimized. The goal of this investigation is to find an improved compressor design by modifying the annulus geometry only, but keeping the inlet and outlet annulus coordinates constant. In a first investigation the annulus geometry is parameterized according to Equations (3.11) and (3.12) with $n+1=4$ control points, respectively, leading to a design vector

$$\mathbf{p} = \left[b^M_{x,1}, b^M_{x,2}, b^M_{y,1}, b^M_{y,2}, b^\Delta_{x,1}, b^\Delta_{x,2}, b^\Delta_{y,1}, b^\Delta_{y,2}\right]^T.$$

The multi-objective optimization problem (3.10) originally consisting of three individual objectives is transferred by the compromise method (2.21) to a scalar optimization problem where the main objective to be maximized is the overall compressor polytropic efficiency $\eta_{c,poly}$. The surge margin at design point SM is considered as an inequality constraint with an user-defined lower bound $\widehat{SM}$, and the overall pressure ratio Π_c is kept constant since it is determined uniquely by the given pressure ratio distribution. In order to create feasible solutions, all design constraints in (3.10) are taken into account resulting in

$$\underset{\mathbf{p}\in\mathbb{R}^n}{max} \quad \eta_{c,poly} \tag{3.21}$$

subject to

$$\begin{aligned}
&\underset{i}{max}\,\Psi_i \le \hat{\Psi} && \underset{i}{max}\,DF^R_i \le \widehat{DF}\\
&\underset{i}{max}\,M'^R_{I,i} \le \hat{M}'^R_I && \underset{i}{max}\,DF^S_i \le \widehat{DF}\\
&\underset{i}{max}\,M^S_{I,i} \le \hat{M}^S_I && \underset{i}{min}\,DH^R_i \ge \widehat{DH}\\
&\underset{i}{max}\,C_{h,i} \le \hat{C}_h && \underset{i}{min}\,DH^S_i \ge \widehat{DH}\\
&M^S_{E,N_s} \le \hat{M}_E && SM \ge \widehat{SM}\\
&\text{with } i = 1(1)9
\end{aligned}$$

The scalar optimization problem is solved with the Multi-Island Genetic Algorithm, see Section 2.3.4.4, with a population size of 20 individuals on each

of the 20 islands and the evolution process proceeds over 20 generations. The results of this first investigation are shown in Figure 3.5. Despite the fact that this investigation is a scalar optimization only, the resulting feasible solutions are plotted in the bi-criterion space in order to provide some insight in the original vector optimization problem. The non-dominated solutions in the upper right corner of the cloud of feasible solutions are obtained by a non-dominated sorting algorithm and can be interpreted as a rough estimation of the Pareto-front. A comparison of the best solution found by MIGA with the baseline design which is indicated as *Human Design* shows that this approach is not able to outperform the initial design in terms of polytropic efficiency.

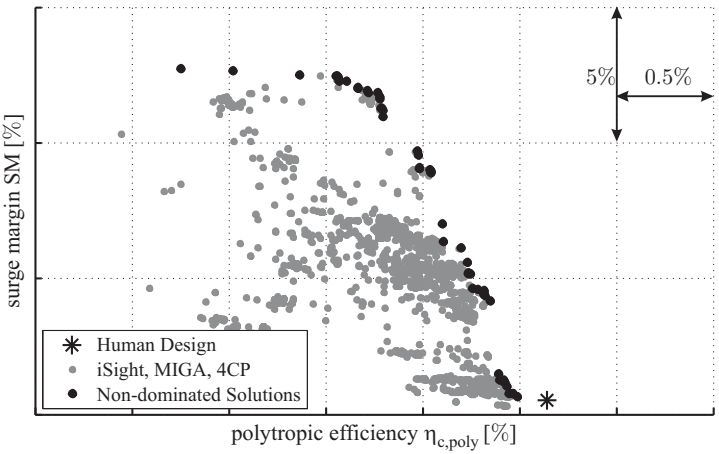


Figure 3.5: Annulus line optimization using Bézier-splines with 4 control points

The reason for this phenomenon lies in the parameterization of the annulus geometry. A detailed investigation shows that four control points are not sufficient to provide the required flexibility to the annulus line geometry. In application of parameterization, the desired minimum number of design parameters is always in conflict with the requirement of maximum flexibility. In this particular case the parameterization was to restrictive implying constraints to the optimization problem which drive the numerical optimization to produce sub-optimal solutions.

The optimization task (3.21) is solved again where the design freedom is in-

creased by introducing an additional control point to both parametric annulus definition curves (3.11) and (3.12), i.e.

$$\mathbf{p} = \left[b^M_{x,1}, b^M_{x,2}, b^M_{x,3}, b^M_{y,1}, b^M_{y,2}, b^M_{y,3}, b^\Delta_{x,1}, b^\Delta_{x,2}, b^\Delta_{x,3}, b^\Delta_{y,1}, b^\Delta_{y,2}, b^\Delta_{y,3}\right]^T .$$

The results of this investigation are shown in Figure 3.6 where the number of generations is increased to 40 due to the increased design freedom.

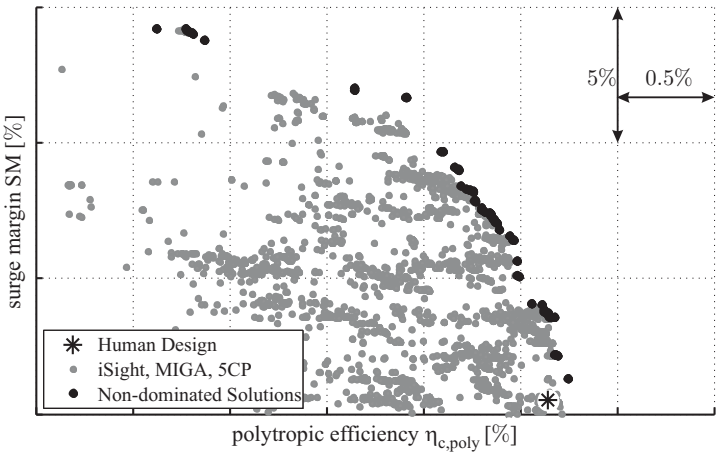


Figure 3.6: Annulus line optimization using Bézier-splines with 5 control points

Compared with the previous result, the increased design freedom leads to clearly better solutions. The front of non-dominated results becomes more dense and the overall number of feasible solutions is increased due to the higher generation count. If the pseudo Pareto-solutions are compared with the initial design, superior solutions can be observed. A closer look in the region of interest of Figure 3.6 shows improved designs with respect to both efficiency and surge margin, Figure 3.7. If the efficiency plays an important role, a solution with absolutely 0.11% points improvement can be found without loosing any surge margin, and on the other side if more surge margin is desired, an improvement of absolutely 3.2% points is possible at a constant efficiency level. These two solutions represent individually best solutions in terms of efficiency and surge margin, respectively.

Further solutions exist lying in between of these two points which are slightly better in both objectives compared to the datum design.

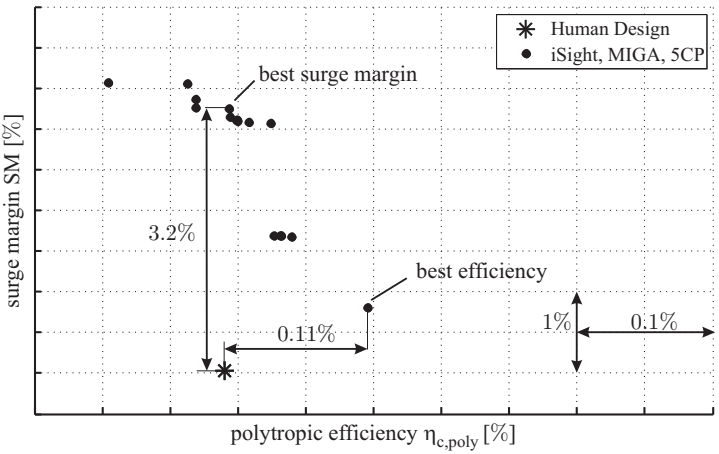


Figure 3.7: Optimal trade-off results with MIGA

The result of this investigation looks very promising. The front of non-dominated solutions offers the design engineer a good basis to discuss on trade-offs between polytropic efficiency and surge margin. However, it should be reemphasized that this investigation is done based on a single-objective optimization algorithm only and obviously it can not be expected that the non-dominated solutions are really representing the Pareto-front. Beside this major problem two more issues can be addressed. Firstly, the result shown in Figure 3.6 contains undesired gaps in the non-dominated solution front, and secondly a huge number of solutions are generated in a region which is not really of interest. Therefore, it should be the goal to find more solutions at the Pareto-front, to close the gaps and to avoid undesired solutions.

In order to address these issues, another investigation based on the same annulus line parameterization is done. The optimization problem (3.21) is solved again using the gradient based algorithm NLPQL, see Section 2.3.4.4. The difference between this approach and the previous one is that 40 individual optimizations are performed with an increasing lower bound on the surge margin $\widehat{SM}$ in order

to create a more dense non-dominated solution front. All individual optimization runs are started from the same initial human design point. The overall compressor pressure ratio, as the third objective, is kept invariant due to a constant stage pressure ratio distribution which makes this investigation comparable with the previous one.

Figure 3.8 shows the feasible solutions obtained by all 40 individual NLPQL optimization runs. It can be seen that the gradient based algorithm produces solutions closer to the Pareto-front. Compared with the MIGA investigation the solutions are significantly improved and due to the application of the compromise method with moving bound a clearly better Pareto-front can be observed. Furthermore, the gaps within the frontier are reduced which is basically driven by the small step sizes of the inequality surge margin constraint. This is basically an important point in terms of discussing trade-off solutions.

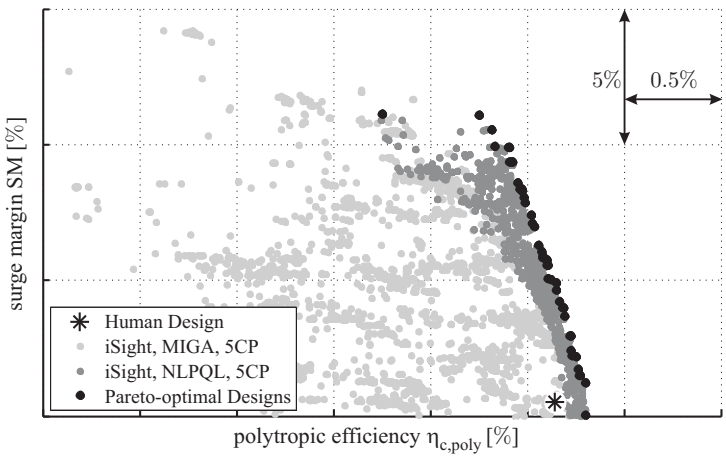


Figure 3.8: NLPQL results in comparison with MIGA

The improvement of the annulus design can be better seen in Figure 3.9 where the design engineer can choose between a compressor design with an increased polytropic efficiency by 0.16% points at constant surge margin, a design which has a 4.4% points higher surge margin at the equal efficiency level, or one of the well distributed trade-offs with better surge margin and efficiency.

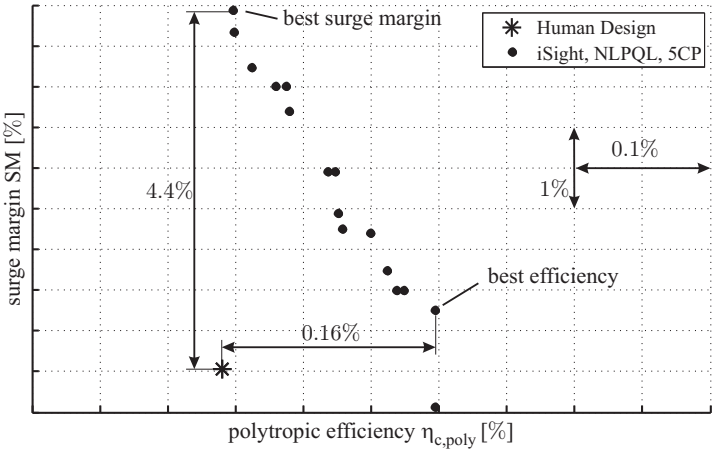


Figure 3.9: Pareto-optimal results with NLPQL

Summarizing, Figure 3.10 compares the non-dominated solution fronts for the presented three investigations. In general one can say that the optimal solutions found by the deterministic method are obviously more reliable than results from the stochastic approach and that the solution depends significantly on the design freedom. It is obvious that additional control points to the annuls line parameterization lead to more flexibility and thus to better designs, however, from the practical point of view a trade-off between flexibility and computational costs has to be found.

It should be mentioned that more competitive Pareto-fronts would be determined by the stochastic approach if the MIGA algorithm would be repeated by means of the compromise method with variable surge margin steps. However, in general genetic algorithms need too many function evaluations and the determination of the Pareto-front would be too expensive in term of overall computational time.

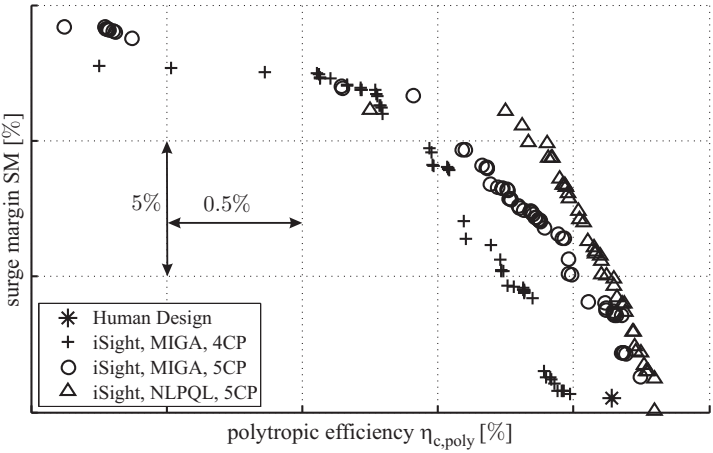


Figure 3.10: Comparison of non-dominated solutions

As mentioned before, the annulus geometry optimization is obviously one of the key elements in compressor preliminary design especially due to the number of design parameters and the requirements for smoothness which always have to be considered. Nevertheless, the performance of the compressor depends also on other parameter values and especially on their appropriate distribution along the compressor mid-height. Hence, in a further investigation the stage pressure ratio distribution as an additional design quantity beside the annulus geometry is introduced. In order to guarantee required smoothness, the stage pressure ratio distribution is also parameterized by a Bézier-curve with five control points according to Equation (3.19) which can be also seen in Figure 3.3. The number of design parameters rises up to 20 consisting of 12 for the parametric annulus description and 8 for the stage pressure ratio distribution, i.e.

$$\begin{aligned}\mathbf{p} \;=\; \big[&b^M_{x,1}, b^M_{x,2}, b^M_{x,3}, b^M_{y,1}, b^M_{y,2}, b^M_{y,3}, b^\Delta_{x,1}, b^\Delta_{x,2}, b^\Delta_{x,3}, b^\Delta_{y,1}, b^\Delta_{y,2}, b^\Delta_{y,3}, \dots \\ &\dots, b^\Pi_{x,1}, b^\Pi_{x,2}, b^\Pi_{x,3}, b^\Pi_{y,0}, b^\Pi_{y,1}, b^\Pi_{y,2}, b^\Pi_{y,3}, b^\Pi_{y,4}\big]^T .\end{aligned}$$

The multi-objective optimization problem (3.10) is solved by using the overall polytropic efficiency as solely objective and transferring the surge margin to an

inequality constraint with a single lower bound $\widehat{SM}$. In this particular case the overall compressor pressure ratio Π_c is not restricted to the desired quantity rather it is just calculated by Equation (A.8) depending on the individual stage pressure values according to their distribution. However, restrictions of the control point positions for the stage pressure ratio parameterization is necessary in order to obtain feasible solutions. The multi-objective optimization problem (3.10) is then transferred by the compromise method as

$$\max_{\mathbf{p}\in\mathbb{R}^n} \quad \eta_{c,poly} \tag{3.22}$$

subject to

$$\begin{aligned}
&\max_i \Psi_i \leq \hat{\Psi} && \max_i DF_i^R \leq \widehat{DF}\\
&\max_i M'^R_{I,i} \leq \hat{M}'^R_I && \max_i DF_i^S \leq \widehat{DF}\\
&\max_i M^S_{I,i} \leq \hat{M}^S_I && \min_i DH_i^R \geq \widehat{DH}\\
&\max_i C_{h,i} \leq \hat{C}_h && \min_i DH_i^S \geq \widehat{DH}\\
&M^S_{E,N_s} \leq \hat{M}_E && SM \geq \widehat{SM}\\
&2 \geq b^{\Pi}_{y,k} \geq 1, && \text{with } i = 1(1)9,\ k = 1(1)5
\end{aligned}$$

Figure 3.11 shows the feasible solutions in the criterion space obtained by the genetic algorithm MIGA. Due to the increased design freedom the number of function evaluations is adopted to a population size of 30 individuals distributed on each of the 30 islands and evolved over 60 generations. The results of the three-criterion problem are shown in a two-dimensional plot where the overall pressure ratio as the third criterion is indicated by the color.

The amount of function evaluations leads to a high number of feasible solutions which are distributed within the criterion space. The improvement of overall pressure ratio for reduced efficiency levels is a confirmation for the contradiction of the objectives. A further interesting point here is that compressor designs exist where this improvement is correlated with higher surge margin values. These results can be used to understand the design problem better and to find new innovative solutions. In order to quantify the goodness of this investigation, the solutions may be compared with the datum design. A closer look around the datum design and a filtering of the results with respect to surge margin and

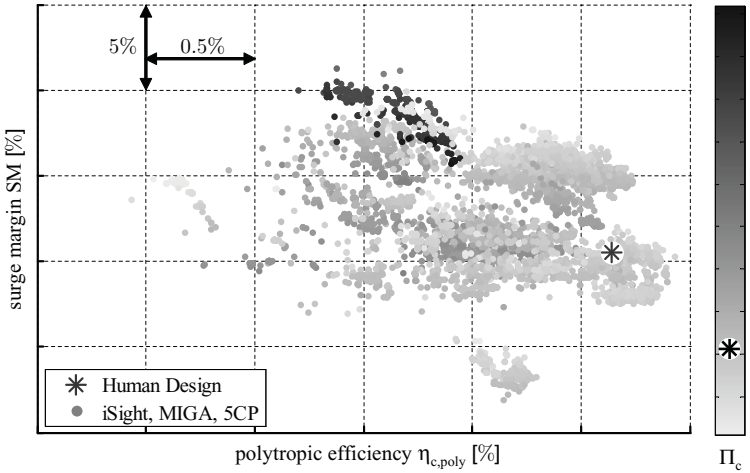


Figure 3.11: Multi-criterion optimization with respect to efficiency, surge margin and overall pressure ratio

overall pressure ratio based on the datum design values shows that improved designs with respect to all three objectives can be found, Figure 3.12.

In Figure 3.13 the annulus geometries as well as the stage pressure ratio distributions for three extreme designs from Figure 3.12 are shown. The maximum efficiency design shows an unload of the first stage which is covered by the last three stages. The overall pressure ratio and surge margin are quite similar to the initial design and the efficiency is improved by 0.16% points. For the second case the surge margin is increased by 4.9% points at reasonable efficiency values and the overall pressure ratio is also slightly improved. In the last case a reduction of the stage pressure ratio at the three front stages and an increase of other stages can be observed which leads to an improvement of 2% in overall pressure ratio in comparison to the datum design without loosing surge margin and efficiency. In comparison with the initial design it can be recognized that all annulus geometries differ form each other. In particular the front region is characterized by a higher annulus height and the rear part by an increased radial coordinate of the mid-height line. From the corresponding stage pressure ratio plots in general a redistribution of the loading is visible.

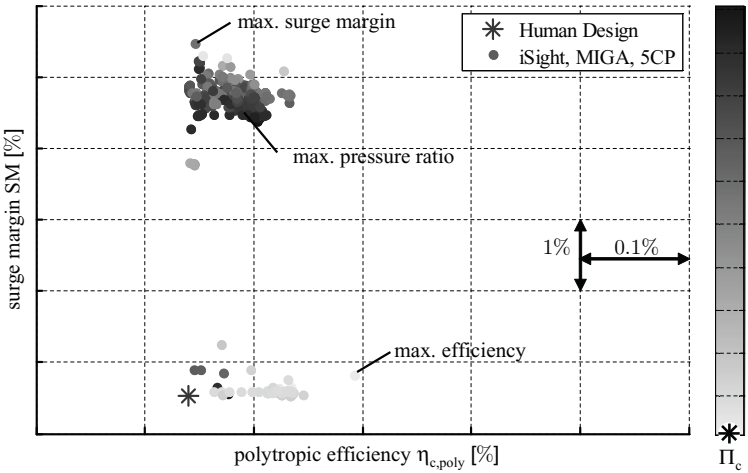


Figure 3.12: Results of multi-criterion optimization with respect to three objectives

Due to the increased design freedom, new designs can be found. An interesting point here is that a lot of new designs exist with higher overall pressure ratio and surge margin values at the cost of polytropic efficiency. This is typical for multi-criterion optimization problems with contradicting objectives.

From the industrial point of view it is interesting to notice that apart from the improved designs the process integration and automation leads to an acceleration of the design process. This fact can be proved in Table 3.1 where an overview of the calculated cases in terms of function evaluations, convergence, feasibility rate, and overall optimization time is given. It can be seen that the number of function evaluations is increased with the design freedom and that evolutionary algorithms require more function evaluations than deterministic methods. The genetic algorithm is rather robust against non-convergent performance evaluation where the high percentage of defective designs results from the stochastic nature of the search strategy. Gradient based methods would not work with such a high failure rate resulting in non-smooth performance evaluations. Therefore, the NLPQL algorithm was forced to use small parameter variations for the numerical gradient calculation resulting in a rather high convergence rate.

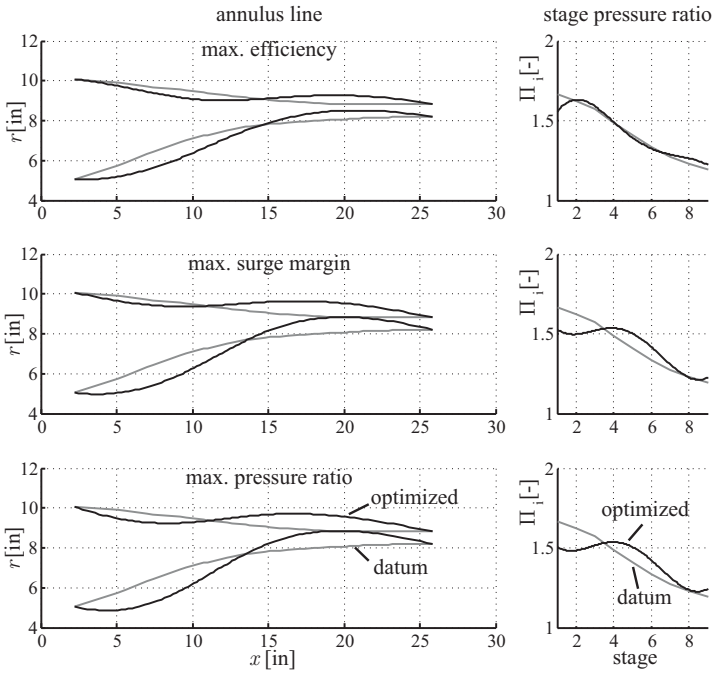


Figure 3.13: Annulus geometry (left) and stage pressure ratio distribution (right) of multi-criterion optimization with respect to efficiency, surge margin and overall pressure ratio

For a better comparison between NLPQL and the corresponding MIGA case the resulting numbers for one step of the compromise method are additionally shown in the lower row. The total time for finding the best efficiency value on a specific surge margin level can be reduced if only one point of the Pareto-curve is desired. Finally, it should be mentioned that a human designer requires approximately 2 minutes for a function evaluation based on the interaction with the slow graphical user interface. The time for one function evaluation of the automated process is less than 10 seconds, resulting in a process speed-up factor of 12.

Table 3.1: Function evaluation and optimization time for meanline process

	function evals	converged solutions	feasible solutions	total time [h]
MIGA (annulus, 4CP)	8000	90%	58%	19.3
MIGA (annulus, 5CP)	16000	86%	40%	30.5
NLPQL (annulus, 5CP)	10768	99%	8%	18.2
MIGA (annulus+Π, 5CP)	54000	79%	28%	94.1
NLPQL (annulus, 5CP)				
1 compromise step	151	100%	24%	0.28

During post-processing a surprising issue with the surge margin prediction of the meanline analysis tool was detected. It turned out that the correlation based prediction of the design point surge margin is not valid for geometry variations as performed in the presented investigations. Therefore, the preliminary design process was extended by an additional evaluation step. For each design multiple meanline prediction calculations with variable compressor exit pressure values are performed in order to determine the design speed characteristic of the compressor from which the surge margin value is extracted according to Equation (A.12). The determination of the compressor design point efficiency as an objective of the optimization process is also based on the results of the characteristic calculation.

In a first investigation the multi-objective optimization design problem (3.21) is solved based on the NLPQL algorithm as part of the compromise method consisting of 25 inequality surge margin steps where design variations are made by annulus geometry modifications only. Figure 3.14 shows all feasible results in the criterion space and furthermore the obtained non-dominated solutions after a sorting algorithm. The values for surge margin and efficiency of the datum design are shown which are also determined with the same new calculation approach in order to make the solutions comparable with each other. It can be seen that NLPQL produces a lot of better solutions compared with the datum design in term of polytropic efficiency as well as for surge margin values and that trade-offs between the two objectives can be made.

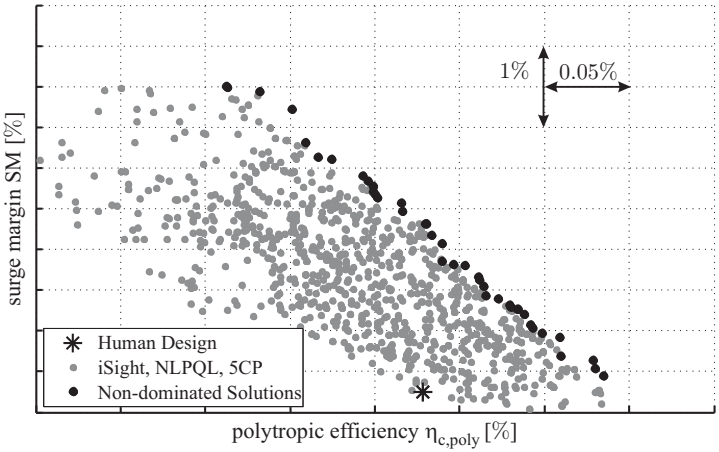


Figure 3.14: NLPQL results for extended process in comparison with datum design

A closer look around the human design is given in Figure 3.15 where an improvement in efficiency of about 0.12% points is achieved without loosing any surge margin or an increase in surge margin of 2.1% points at an equal efficiency level. A comparison with the previous investigation based on the former surge margin definition, Figure 3.9, shows that the relative improvement in efficiency is fairly similar whereas the improvement in surge margin is reduced. The reason for this is due to the new definition of the surge margin, however, the trend in the improvement is also repeated for both objectives in this investigation.

Obviously, the drawback of the extended process is that the overall evaluation time is slightly higher due to additional calculations required for the surge margin. However, the new definition is not correlation based anymore and the whole design speed characteristic is available for postprocessing. Figure 3.16 shows a comparison of the design speed characteristics between the datum design with the best efficiency and best surge margin points from Figure 3.15. If we concentrate on the pressure ratio distributions, Figure 3.16b, it can be observed that all three curves are intersecting each other at the design point which means that the flow function and the pressure ratio at the design point is equal as it is

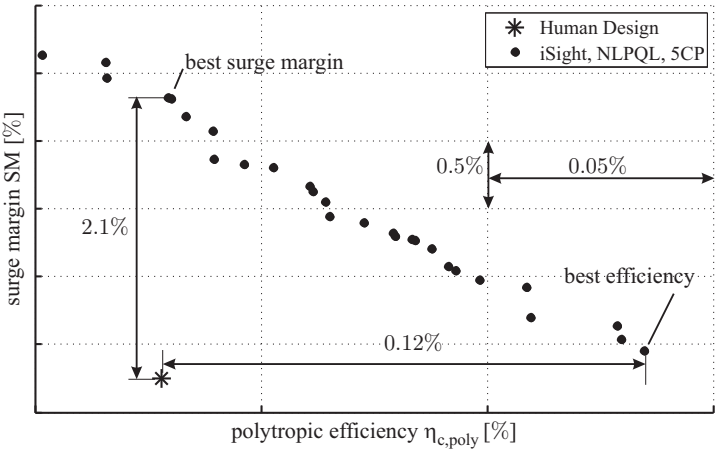


Figure 3.15: Non-dominated solutions of NLPQL for extended process

required from the pressure constraint. The improvement of the best surge margin design is mainly driven by a higher total pressure ratio at low flow function values, whereas the comparison of the best efficiency design with the datum design shows no significant deflection. In terms of polytropic efficiency, Figure 3.16a, the distributions show that the improvement of the best efficiency design is reflected for the whole flow function range which means that this solution is absolutely better for the entire design speed characteristic. Additionally, it is interesting to notice that the best surge margin design also produces higher efficiency values for lower flow functions.

In order to be sure that the obtained non-dominated solutions by the gradient based method are good approximations of the real Pareto-front, an investigation based on the multi-objective genetic algorithm NSGA-II, see Section 2.3.4.4, is performed. Since the aim is to capture the Pareto-front as good as possible, the diversity of the solution is driven by a high number of individuals in the population, namely 100, whereas the convergence towards the Pareto-optimal front is obtained by an evolution process over 200 generations.

The resulting non-dominated solutions of both investigations for the annulus line optimization are compared in Figure 3.17. As can be seen, the resulting

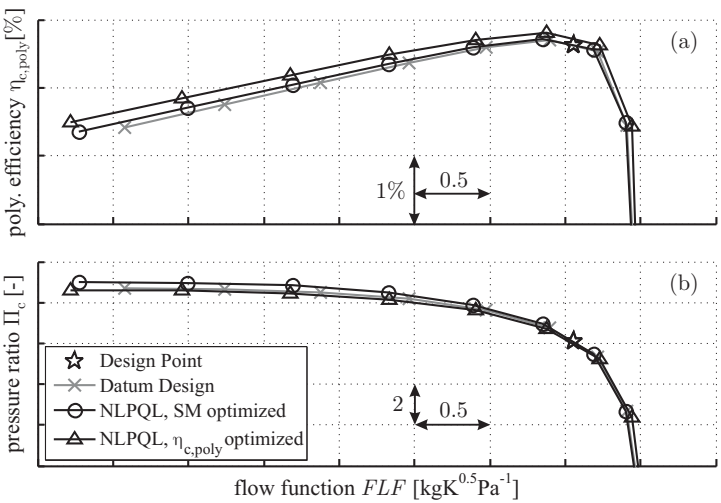


Figure 3.16: Comparison of polytropic efficiency (a) and pressure ratio (b) distributions

points are well distributed and both non-dominated frontiers are lying very close to each other. The biggest deflection can be found for low efficiencies where the genetic algorithm obtains about 0.5% points higher surge margin values compared to the gradient based method. It can be generally observed that the NSGA-II produces better solutions and hence it outperforms NLPQL. This holds true for the optimization performance, however, if the optimization time and the number of required function evaluations are considered, the trend turns towards the gradient based method which is shown in Table 3.2.

Comparing the number of function evaluations shows that the gradient based approach is much quicker reflected by only 28% function evaluations of the genetic algorithm. The number of converged solutions are quite similar whereas NLPQL produces less feasible solutions as known from the previous investigation. From the industrial point of view the process performance is of major importance which is reflected by the total optimization time which shows that the gradient based method is faster than the genetic algorithm by a factor of 3.6. Hence, if the analysis tools are robust, i.e their convergence rate is high, and the design space

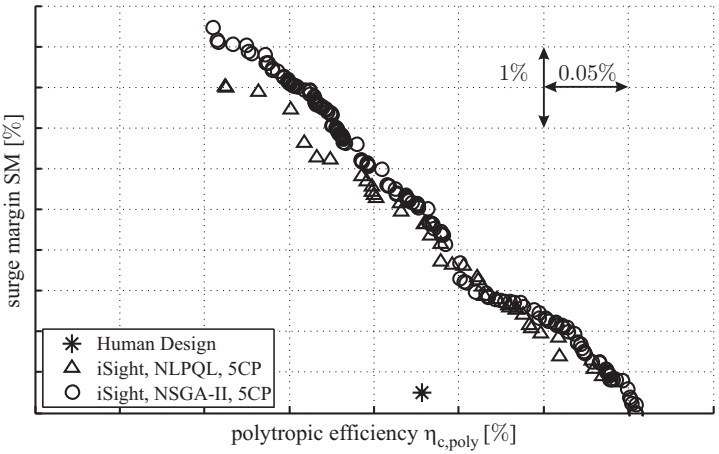


Figure 3.17: Comparison of non-dominated solutions based on alternative surge margin definition

is smooth, gradient based methods with an appropriate scalarization technique are always preferable for the Pareto-front determination.

Table 3.2: Function evaluation and optimization time for extended meanline process

	function evals	converged solutions	feasible solutions	total time [h]
NLPQL (annulus, 5CP)	5655	99%	12%	9.6
NSGA-II (annulus, 5CP)	20100	98%	58%	34.5

4 Optimization Applied to Throughflow Calculation

The throughflow calculation is an extension of the previous meanline calculation in radial direction. It requires more calculation time and is too complex to be fully integrated in an optimization process. Therefore, the throughflow off-design calculation as a subproblem of the throughflow process is taken as a demonstration how process integration and optimization can be applied to industrial design problems for solving typical time-consuming design tasks. The emphasis is to accelerate the throughflow off-design process and to provide automated optimization methods and techniques for finding solutions which support the human engineer.

4.1 Introduction

The throughflow calculation is the second step within the aerodynamic compressor design process. In this design phase appropriate radial distributions of flow parameters in the meridional plane have to be found which fulfill the overall compressor requirements. For this reason the results of the meanline prediction calculation are used as initial solution to the throughflow calculation. Parameters along the mid-height line as blade solidities, losses, stage pressure ratios, and flow angles are expanded at specific axial positions in radial direction based on engineering expertise and design rules. The annulus geometry as well as the predicted blockage distribution are also transferred into the new design model.

Throughflow equations for the flow in a compressor or turbine can be solved

with a streamline curvature method. The central part of this method involves the radial equilibrium equation which is a non-linear ordinary differential equation relating the radial static pressure gradient of a general swirling flow to the streamline geometry. In the early use of this equation simplifying assumptions had to be made in order to integrate it analytically. However, with the development of high speed computers it has become possible to solve the equation using a streamline curvature method without the very restrictive simplifications needed before. In the streamline curvature method the radial equilibrium equation is solved iteratively once the inlet mass flow, the blockage assumption, and the annulus geometry is given. Basically, the throughflow calculation starts with an approximation of the streamline geometry on 7 to 21 radial positions. The static pressure is obtained by integrating the radial equilibrium equation which is then used to derive a new approximation to the streamline geometry. This iterative procedure is repeated until the result converges and the flow field in the S2 plane is obtained. The results of the throughflow calculation process are on one hand radially distributed aerodynamic parameters at each axial blade inlet and outlet position and on the other hand the streamline distribution itself within the annulus geometry.

Once, the design throughflow calculation is performed, operating phases reflecting take-off or climb situations have to be investigated in order to judge the feasibility and reliability of the compressor design at off-design flight conditions. The overall performance requirements for the compressor at these conditions are à priori given and have to be fulfilled by making assumptions on blade losses and blade exit whirl angle deflections.

Figure 4.1 shows the general work flow for obtaining an off-design throughflow solution. The process starts with a necessary design point throughflow calculation which is used as an initial solution to the off-design problem. Thereupon the engineer has to apply the off-design conditions which include the off-design shaft speed, inlet total temperature, inlet total pressure, and inlet mass flow. If bleeds and variable stators exist in the compressor design, the bleed air mass flow as well as the variable stator angles have to be adapted to the design problem too. The next step is to run the throughflow calculation with these off-design parameter settings. Unfortunately, in some cases where the off-design is far away from the initial design point solution, it may happen that the throughflow calculation does

not converge. The reason for this is that the blade loss and exit whirl angle assumptions are not set properly to the problem.

The problem in such a case is that no solution file exists from which objective function values could be extracted. Therefore, it is necessary to use the inlet mass flow as a temporary design parameter in order to achieve some intermediate converged off-design solution. Since the calculation converges, objective values can be extracted and compared with the required compressor performance data. During the off-design solution finding process, obviously, the inlet mass flow has to be driven back to the correct value.

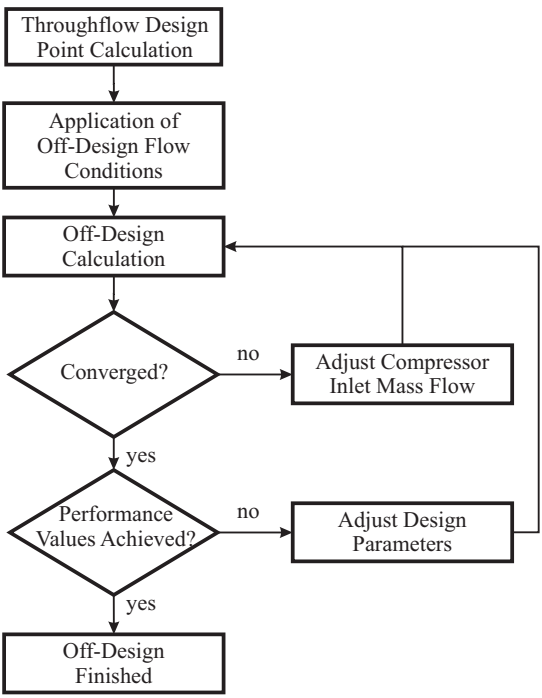


Figure 4.1: Off-design preparation

The throughflow off-design calculation is a highly iterative process which can be accelerated by means of process integration and automation. In order to fulfill the multiple performance requirements, the throughflow off-design problem can be formulated as a multi-objective design problem.

4.2 Off-Design Optimization Problem

The throughflow off-design problem can be transferred into a rather small optimization problem consisting of two design variables and three objectives. The first design parameter reflects the increase or in some specific cases the decrease of the pressure losses at each blade row at off-design flow conditions. It is defined as an additional loss assumption factor $\chi \in \mathbb{R}$ which is multiplied on all rotor and stator blade radial loss distributions, respectively. The second design parameter describes the change in exit flow angle also caused by off-design flow conditions. The parameter is denoted by $\Delta\mathcal{D} \in \mathbb{R}$ and it is assumed that the deflection is radially constant and equal for all rotor and stator blade rows. Both design parameters are typically restricted by user defined bounds in order to avoid convergence problems of the throughflow calculation program due to physically unrealistic values. Summarizing, the design vector for the throughflow off-design calculation process can be expressed by

$$\mathbf{p} = [\chi, \Delta\mathcal{D}]^T .$$

As already mentioned, the aim of the throughflow off-design calculation process is to match required performance parameters. These parameters are the overall compressor pressure ratio Π_c, the isentropic compressor efficiency $\eta_{c,isen}$ and the compressor exit total temperature $T_{0,c,E}$. Due to the fact that it is often not possible to hit all three values exactly, an off-design calculation is being accepted when the difference in polytropic efficiency and overall compressor pressure ratio is less or equal 1% and the deflection in compressor exit total temperature is less or equal 1K compared to the given performance values. The objectives of the throughflow off-design calculation can be formulated in such a way that all three terms are in the same order of magnitude. The first objective

$$\widehat{\Pi}_c = \frac{\Delta\Pi_c}{\Pi_c} \cdot 100\% \tag{4.1}$$

describes the relative deviation of the overall compressor pressure ratio from the required value, the second objective

$$\widehat{\eta}_{c,isen} = \frac{\Delta\eta_{c,isen}}{\eta_{c,isen}} \cdot 100\% \tag{4.2}$$

describes the relative deviation of the isentropic compressor efficiency, and the third objective

$$\widehat{T}_{0,c,E} = \Delta T_{0,c,E} \tag{4.3}$$

uses the absolute deviation of the compressor total exit temperature. Summarizing, the multi-objective throughflow off-design problem can be expressed by

$$\underset{\mathbf{p}\in\mathbb{R}^n}{min} \begin{bmatrix} \widehat{\Pi}_c \\ \widehat{\eta}_{c,isen} \\ \widehat{T}_{0,c,E} \end{bmatrix} \tag{4.4}$$

subject to

$$\chi^l \leq \chi \leq \chi^u$$
$$\Delta\mathcal{D}^l \leq \Delta\mathcal{D} \leq \Delta\mathcal{D}^u$$

with upper and lower bounds for the design parameters.

4.3 Throughflow Off-Design Process Integration

The Rolls-Royce off-design calculation process is also integrated in an automated environment using *iSight* as process integration and automation tool, Figure 4.2.

The process starts based on a design throughflow solution where the new design conditions have to be applied à priori according to the off-design case in an initialization process. The chosen optimization algorithm provides a set of design parameters **p** which are parsed in the throughflow input file and the throughflow calculation program is invoked. The results are written in an output file from which the objective function values are extracted. If the required performance values are not achieved, the next optimization step starts with a new design vector **p**.

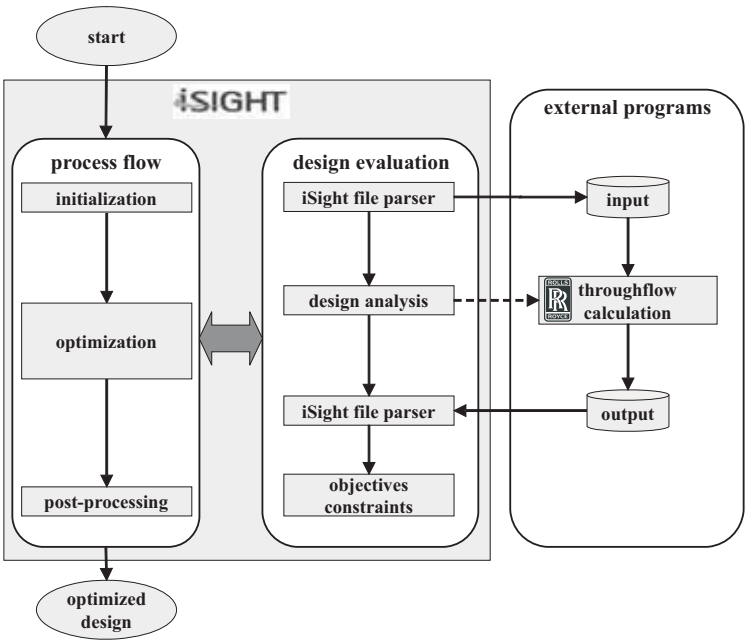


Figure 4.2: Integration of the throughflow off-design process flow

For some off-design cases where the convergence rate of the throughflow calculation is very low, it is required to firstly use a pattern method, which explores the design space for a promising point from where the overall optimization process can start. Especially, if the calculations do not converge around the starting point this step becomes essential for gradient based algorithms, because non-converged

calculations do not provide any confident information and hence the determination of gradient information becomes very problematic. Furthermore, if a good starting point is used the chance of finding the global optimum increases for any deterministic method. If a pattern method is desired in the automated throughflow off-design calculation process, it can be simply chosen in the process flow environment of *iSight*.

4.4 Results and Discussion

In this section results of a throughflow off-design problem based on a 6-stage high pressure compressor model with five bleed positions and two variable stator blade rows are presented. All investigations are run at maximum take-off operating conditions and differ basically with respect to the objective function definition. Due to the fact that the off-design case converges properly in a wide range of the design space, the gradient based algorithm NLPQL is chosen and the required gradients for the search direction are calculated numerically based on forward differences with an appropriate small parameter perturbation. The pressure loss coefficient factor χ and the additional exit whirl angle $\Delta\mathcal{D}$ are bounded by

$$0.5 \leq \chi \leq 1.5$$
$$1^\circ \leq \Delta\mathcal{D} \leq 2^\circ$$

In the first investigation the off-design throughflow problem is solved by transferring the multi-objective problem (4.4) to a single-objective problem using the weighted-objective scalarization approach (2.19). Since the individual objectives may become negative, absolute values are considered which makes the approach equivalent to the distance method with the metric $r = 1$, see Section 2.3.3.2. The new compound function F to be minimized reads as

$$\begin{aligned} &\underset{\mathbf{p}\in\mathbb{R}^2}{min} && F(\mathbf{p}) \\ &\text{with} && F(\mathbf{p}) = \left|\widehat{\Pi}_c\right| + \left|\widehat{\eta}_{c,isen}\right| + \left|\widehat{T}_{0,c,E}\right|. \end{aligned} \tag{4.5}$$

Taking absolute values implies non-smoothness in the objective function which can lead to convergence problems of the optimization algorithm. Hence, a second

investigation with a smooth objective function F is run by means of the distance method (2.20) with the metric $r = 2$, i.e.

$$\begin{aligned} &\underset{\mathbf{p}\in\mathbb{R}^2}{min} && F(\mathbf{p}) \\ &\text{with} && F(\mathbf{p}) = \sqrt{\left(\widehat{\Pi}_c\right)^2 + (\widehat{\eta}_{c,isen})^2 + \left(\widehat{T}_{0,c,E}\right)^2}. \end{aligned} \tag{4.6}$$

In order to find the best approach for the throughflow off-design optimization, a third investigation based on the min-max method (2.22) is performed. An artificial parameter γ is introduced as new objective and the original ones are considered as inequality constraints bounded by γ. Since it is also a design variable, the design parameter space is enlarged, i.e. $\mathbf{p} = [\chi, \Delta\mathcal{D}, \gamma]^T$, leading to the min-max optimization problem

$$\begin{aligned} &\underset{\mathbf{p}\in\mathbb{R}^3}{min} && \gamma \\ &\text{subject to} \\ & && \left|\widehat{\Pi}_c\right| \leq \gamma \\ & && |\widehat{\eta}_{c,isen}| \leq \gamma \\ & && \left|\widehat{T}_{0,c,E}\right| \leq \gamma. \end{aligned} \tag{4.7}$$

The results of all three investigations, denoted by case 1 to case 3, are summarized in Table 4.1. For a general comparison two major aspects can be distinguished: the process performance described by the required number of function evaluations, feasible solutions and overall optimization time, and the optimization performance indicated by the obtained optimization solutions.

It can be seen, that all three formulations are able to solve the throughflow off-design problem properly. If we compare the process performance of the three cases, only small differences are identifiable. The number of required function evaluations is fairly equal, but case 3 produces less feasible solutions during the optimization procedure which is mainly driven by the additional constraints of the min-max formulation, and has no influence on the overall optimization time. If we compare the process performance, no favorite can be declared. The optimization performance which is reflected by the best obtained solution shows that

the min-max formulation yields very low values for all three objective functions. Compared to the human design, it outperforms the manual search procedure significantly. The weighted-objective method as well as the distance method show improvements in the efficiency and temperature criteria compared to the human design, while the total pressure ratio could not be improved. Hence, both approaches provide trade-off solutions to the multi-objective throughflow off-design problem.

In order to judge the process acceleration and the quality of the final solutions, results achieved by a manual search procedure of a human design engineer are also shown in the table. It can be seen that the min-max formulation is significantly better in terms of process performance and optimization performance. Despite the fact that the automated optimization processes require twice more function evaluations as the human engineer, the overall process time is reduced for all three cases by more than 80%. The reason for this lies in the acceleration of the time consuming throughflow input file preparation by the automated parsing functionality of *iSight*.

Table 4.1: Function evaluation and optimization time for throughflow off-design calculation

	human engineer	case 1 (weighted-obj.)	case 2 (distance)	case 3 (min-max)	case 4
function evals	23	46	44	43	21
feasible runs	-	44	42	19	19
total time [h]	8.0	1.52	1.42	1.40	0.67
time/eval [min]	20.87	1.98	1.94	1.95	1.91
$\left\|\widehat{\Pi}_c\right\|$ [%]	0.220	0.653	0.334	0.048	0.446
$\|\widehat{\eta}_{c,isen}\|$ [%]	0.530	0.336	0.035	0.069	0.069
$\left\|\widehat{T}_{0,c,E}\right\|$ [K]	0.830	0.030	0.230	0.060	0.810
optimal χ	1.020	0.945	0.964	0.970	0.972
design $\Delta\mathcal{D}$	-0.590	-0.633	-0.590	-0.582	-0.567

At this point it should be mentioned that this comparison is not really fair. In the three investigated cases the optimization algorithm is seeking for the best

solution to the problem where the human engineer usually stops his search at the point where all three objective values are below the required bounds. Hence, a fourth investigation is performed on the basis of the distance method (4.6) where the optimization procedure is stopped according to the same bounds being used by the human engineer. The results of case 4 shows that the number of required function evaluations can be reduced which has a positiv influence on the process performance. A comparison with the original case 2 shows that using the stopping criteria saves 53% computational time. The more interesting comparison, however, is between the solutions of the human engineer and case 4. Here, the process performance is increased and the overall computational time is reduced by 92% which is equivalent to a process acceleration factor of 12.

A further interesting point is the number of required optimization steps where each consists of search direction determination and the line search procedure. Figure 4.3 shows that the weighted-objective method requires seven, the distance method and the min-max formulation six, and the truncated distance method only four iteration steps, respectively.

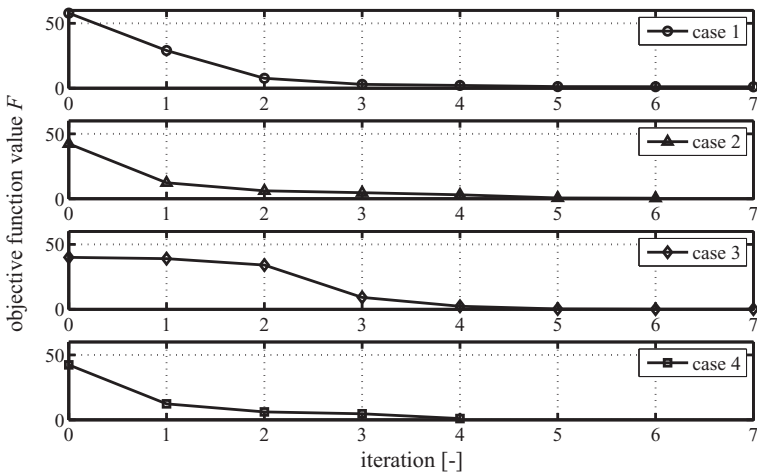


Figure 4.3: Convergence distribution for throughflow off-design cases

In order to analyze the complexity of this particular off-design problem and

to demonstrate how close the solutions are to each other, a design space matrix is calculated where both design parameters are modified within their feasible bounds in 21 equidistant steps. This results in 441 throughflow calculations illustrated in Figure 4.4 where a converged solution is indicated by a single point and the corresponding objective function value according to (4.5) by the gray scale. Non-converged solutions in the right lower corner are indicated by the hatched region. It can be seen that the design space is pretty smooth with increasing values for the objective function F in the upper left and lower right corner. Furthermore, a flat valley can be found with just a single optimum indicating the desired minimum of the objective function. The solutions of all cases are in the vicinity of this optimum, and in particular the design vectors for case 2, 3, and 4 are very close to each other.

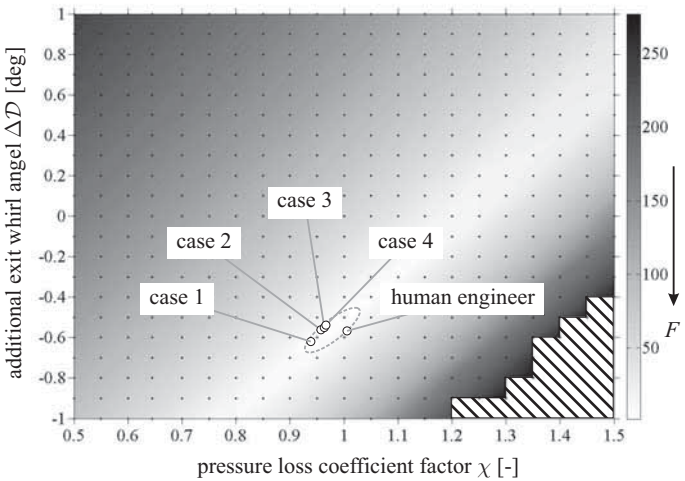


Figure 4.4: Throughflow off-design results according to objective (4.5) in the design space

From the optimization point of view it can be summarized that this throughflow off-design problem is not very challenging. The obtained results show that this smooth problem can be solved with elementary scalarization techniques and the best solution is obtained with the min-max formulation while the computa-

tional costs for seeking the best solution is equivalent for all three techniques. From an industrial point of view, however, it is interesting to notice that the engineering work is accelerated significantly by means of process integration and automation. If additional stopping constraints are introduced to the process in order to come closer to the real engineering work flow, a further improvement of the process performance can be observed and the overall process is accelerated by a factor of 12.

5 Blade Design

The blading process consist of two-dimensional blade section design as well as the more complex three-dimensional blade geometry generation. From the industrial point of view it is a rather complex process since it involves a geometry generation tool and a numerical flow analysis for the determination of the blade geometry. The emphasis of this chapter is to show possibilities in industrial process integration and automation for accelerating the time-consuming blading process. Since in blade design multiple goals have to be pursued simultaneously, the application of multi-objective optimization is demonstrated based on the stochastic NSGA-II optimization method. Furthermore, two alternative problem definitions for the two-dimensional blade design problem will be introduced and discussed which consider aerodynamic and geometric constraints for the optimization process. Finally, a method for three-dimensional blade design will be presented which is based on multiple two-dimensional optimized blade sections.

5.1 Introduction

The blade design process is the final step within the aerodynamic compressor design. The goal is to find three-dimensional blade geometries for all rotor and stator blades within the compressor which fulfill the requirements of the previous throughflow calculation in terms of flow turning with a minimum in loss production for defined aerodynamic design and off-design flow conditions. This is basically one of the most time-consuming and challenging design steps within the aerodynamic compressor design.

Nowadays, there is no straightforward approach to design three-dimensional

blade geometries directly, since the design freedom is too high and the flow in an aero engine is too complicated. Even on today's computers 3D-flow analysis based on Navier-Stokes equations is too expensive why the three-dimensional blade design problem is approximated according to *Wu (1952)* by a set of two-dimensional design problems defined on several radially distributed axis-symmetric stream surfaces S1. In these subtasks two-dimensional blade section geometries have to be found iteratively which are finally stacked along a specific stacking line in order to obtain the three-dimensional blade geometry.

It is obvious that the aerodynamic performance of the final airfoils, and therefore of the whole compressor design, depends significantly on the design quality of the individual blade sections. Hence, a major part in today's engineering work is related to the iterative two-dimensional blade profile design.

It can be shown that the industrial requirement for automated determination of blade section geometries can be overcome by implementing the individual design programs into a common environment. Since the engineering work flow is integrated and automated, numerical optimization can be used in order to find appropriate blade geometries with respect to the design targets. Furthermore, the application of multi-objective optimization can support the engineer in his decision making by providing trade-off solutions between conflicting objectives.

5.2 Blade Design Problem

The goal of the two-dimensional blade design process is to find blade section geometries which fulfill the flow turning requirements from the previous design point throughflow calculation with a minimum loss production at given inlet flow conditions as inlet velocity c_I, inlet flow angle α_I, and inlet pressure P_I, *Keskin et al. (2006b)*. The loss that a blade section produces depends on its geometry and the inlet flow conditions. As demonstrated in Figure 5.1, the total pressure loss coefficient ω defined in Equation (A.13) increases for a given geometry for a deviated inlet flow angle α_I as it occurs for different operating conditions. At design flow conditions α_I^0 the flow around the blade section is smooth and the loss production is typically rather small. If the flow angle is reduced the stagnation point moves towards the suction side and the flow around the leading edge gets highly accelerated. A separating bubble on the pressure side

may occur and a further reduction of the flow angle drives the flow to separate on the pressure side leading to higher losses. On the other side an increase of the flow angle also produces higher losses due to suction side flow separation.

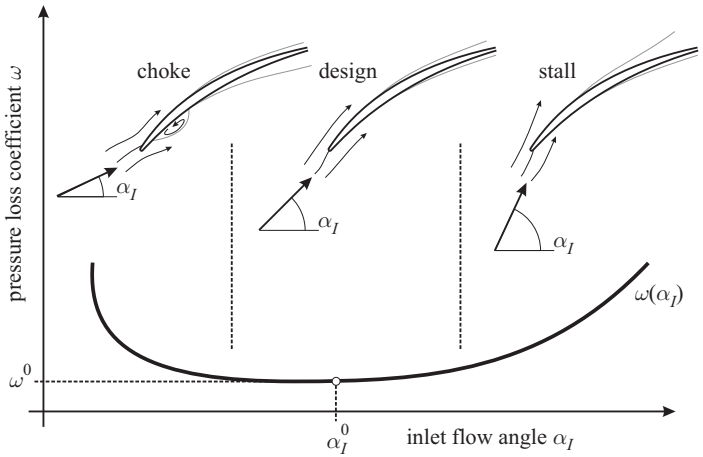


Figure 5.1: Pressure loss coefficient distribution for inlet flow angle variation

Hence, blade geometry design pursues multiple design objectives simultaneously. The first objective is to minimize the loss $\omega^0 = \omega(\alpha_I^0)$ at design flow conditions, where the other objective is to maximize the blade working range WR defined in such a way that the loss does not exceed a prescribed loss level ω_{WR} due to inlet flow angle variation, Figure 5.2. This level is typically set to $\omega_{WR} = 2\omega^0$.

In order to guarantee feasible blade designs, geometric and aerodynamic constraints have to be taken into account during the optimization process. It is of major importance for the compressor design process that the blade design fulfills elementary geometric requirements in order to be accepted from the subsequent stress analysis process and to keep the design iteration with the aerodynamics as small as possible. Thus, the position of maximum blade thickness $PMXC$ is bounded by reasonable values, and inequality constraints are defined to keep the cross section area A and the thickness to chord ratio T/C defined in Figure 5.5 above the values of the datum design, respectively.

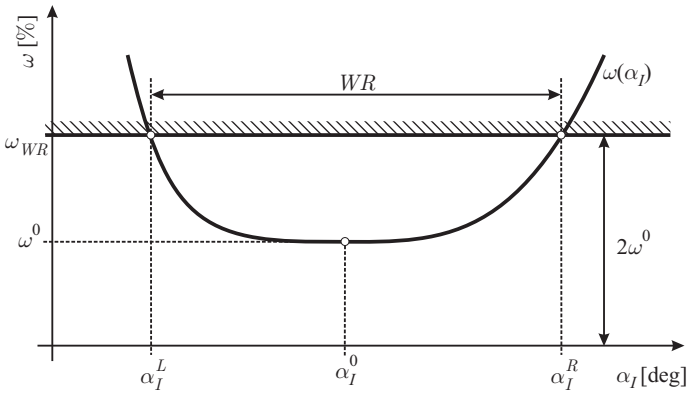


Figure 5.2: Definition of design point loss ω^0 and working range WR

From the aerodynamic point of view further design constraints have to be considered. In order to guarantee constant performance of the following blade rows, the exit flow angle α_E has to be preserved by the new design within an accuracy of $\varepsilon = 0.25°$, and the separation that may occur on the blade suction side has to be avoided. In industrial design the boundary layer shape factor H which describes the relation between the displacement thickness to the momentum thickness of the boundary layer is used as a criteria for flow separation where it is required that the shape factor H_E^{SS} on the suction side at the trailing edge is smaller than 2.5, *Castillo et al. (2004)*.

Summarizing, the multi-objective blade design problem can be formulated as the minimization of the pressure loss coefficient at design flow conditions ω^0 and the maximization of the working range WR which is equivalent to the minimization of $(-WR)$. This has to be achieved by variation of the blade geometry parameters summarized in the design vector $\mathbf{p}$ taking into account geometric and aerodynamic constraints:

$$\min_{\mathbf{p}\in\mathbb{R}^n} \begin{bmatrix} \omega^0 \\ -WR \end{bmatrix} \tag{5.1}$$

subject to

$$
\begin{aligned}
&20\% \leq PMXC \leq 60\% \\
&A \geq A_{datum} \\
&T/C \geq [T/C]_{datum} \\
&H_E^{SS} \leq 2.5 \\
&\alpha_E - \alpha_{E,datum} \leq 0.25
\end{aligned}
$$

A big issue of solving the multi-objective two-dimensional blade design problem (5.1) is the determination of the working range value WR. The standard procedure is to vary the inlet flow angle α_I and to determine the loss curve in order to find the two intersection points $\omega^L = \omega\left(\alpha_I^L\right) = \omega_{WR}$ and $\omega^R = \omega\left(\alpha_I^R\right) = \omega_{WR}$. This, however, requires small changes of the inlet flow angle in order to determine the loss curve accurately which is leading to a rather time-consuming procedure.

In order to release design evaluation from the need of computing the whole loss curve and searching for the two intersection points, two alternative methods for solving the two-dimensional blade design problem can be formulated, *Keskin et al. (2006a)*. In both formulations only three points on the loss curve are required. The first objective for all cases is the minimization of the loss ω^0 at design inlet flow conditions α_I^0 which is obtained by one single flow analysis.

In the first alternative method the inlet flow angle deviations $\Delta\alpha_I^L$ and $\Delta\alpha_I^R$ are introduced as additional artificial design variables defining the flow angles $\alpha_I^L = \alpha_I^0 - \Delta\alpha_I^L$ and $\alpha_I^R = \alpha_I^0 + \Delta\alpha_I^R$, respectively. The sum $\Delta\alpha_I^L + \Delta\alpha_I^R$ then declares the working range value $\widehat{WR}$. During optimization $\widehat{WR}$ is not necessarily corresponding with the real working range value of the blade section design, Figure 5.3. However, a maximization of the working range $\widehat{WR}$ will force the optimization algorithm to increase the distance between the two points α_I^L and α_I^R and hence pushes the losses $\omega^L = \omega\left(\alpha_I^0 - \Delta\alpha_I^L\right)$ and $\omega^R = \omega\left(\alpha_I^0 + \Delta\alpha_I^R\right)$ towards the required upper bound ω_{WR}. Thus, for the first approach the blade design problem (5.1) is redefined as

$$
\underset{\mathbf{p},\Delta\alpha_I^R,\Delta\alpha_I^L}{min}
\begin{bmatrix}
\omega\left(\alpha_I^0\right) \\
-\left(\Delta\alpha_I^L + \Delta\alpha_I^R\right)
\end{bmatrix}
\tag{5.2}
$$

subject to

$$
\begin{array}{ll}
20\% \leq PMXC \leq 60\% & \Delta\alpha_I^L \geq 0 \\
A \geq A_{datum} & \Delta\alpha_I^R \geq 0 \\
T/C \geq [T/C]_{datum} & \omega^L \leq \omega_{WR} \\
H_E^{SS} \leq 2.5 & \omega^R \leq \omega_{WR} \\
\alpha_E - \alpha_{E,datum} \leq 0.25 &
\end{array}
$$

where $\Delta\alpha_I^L$ and $\Delta\alpha_I^R$ have to be bounded in order to guarantee positiv values and ω^L and ω^R have to be kept below an user defined tolerance for the upper working range loss ω_{WR}.

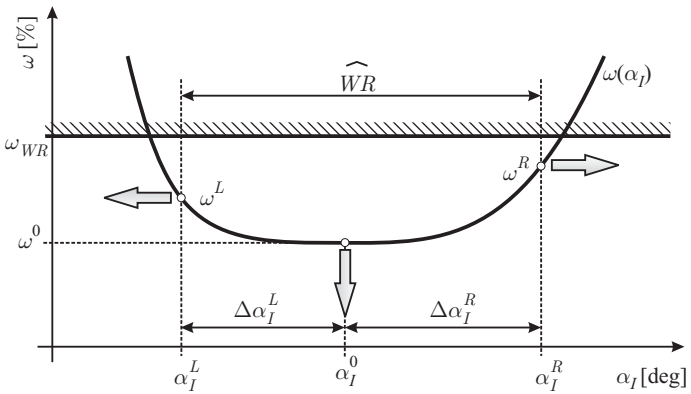


Figure 5.3: First approach for solving the blade design problem by maximizing $\widehat{WR} = \Delta\alpha_I^L + \Delta\alpha_I^R$ at constant working range loss level ω_{WR}

As a second alternative method the blade design problem can be solved by pre-defining the two inlet flow angle deviations $\Delta\alpha_I^L$ and $\Delta\alpha_I^R$ according to the needs on the working range, i.e. $\widehat{WR} = \Delta\alpha_I^L + \Delta\alpha_I^R = const$. The design goal is then to minimize the two losses ω^L and ω^R as demonstrated in Figure 5.4. This problem definition has now three objectives which have to be minimized simultaneously and may be solved by applying the min-max formulation (2.22) to the minimization of the two outer losses ω^L and ω^R by introducing γ as an

artificial objective:

$$\min_{\mathbf{p},\gamma} \begin{bmatrix} \omega\left(\alpha_I^0\right) \\ \gamma \end{bmatrix} \tag{5.3}$$

subject to

$$\begin{aligned} &20\% \leq PMXC \leq 60\% && \omega^L \leq \gamma \\ &A \geq A_{datum} && \omega^R \leq \gamma \\ &T/C \geq [T/C]_{datum} \\ &H_E^{SS} \leq 2.5 \\ &\alpha_E - \alpha_{E,datum} \leq 0.25 \end{aligned}$$

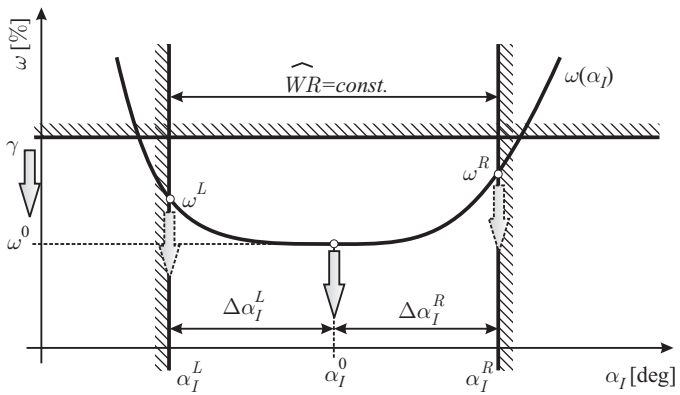


Figure 5.4: Second approach for solving the blade design problem by minimizing ω^L and ω^R for given working range $\widehat{WR}$

It should be noticed that in both alternative methods some drawbacks can be pointed out. Firstly, the number of design parameters is increased by artificial parameters which typically increases the number of required function evaluations. This, however, is not a real drawback since the significant reduction to only three required loss point calculations in both definitions accelerates the determination process of the objective function values tremendously. Moreover,

few more design parameters have a very low influence on the overall optimization time, particularly if genetic algorithms are used. It may be also a drawback that in the alternative definitions the working range values are artificial quantities which make them not directly comparable with the original definition of the blade design problem. However, with supplementary effort during the postprocessing phase this drawback can be covered and comparable working range values can be provided afterwards. Hence, they are chosen for all following investigations as both definitions are showing promising properties.

5.3 Blade Parameterization

For the blade parameterization and modification the Rolls-Royce program *Parablading* is used, *Gräsel et al. (2004)*. This is a blade geometry generation tool for describing three-dimensional blade shapes on the basis of radially stacked two-dimensional blade sections. The blade sections are parameterized on S1 stream surfaces according to the previous throughflow calculation. The parameterization is flexible and offers enough design freedom for a large variety of section geometries by a low number of design parameters. *Parablading* provides an interactive graphical use interface, but can also be run in batch mode which is important in terms of process integration and automation.

Each individual blade section is split into four segments consisting of blade leading and trailing edges plus blade suction and pressure sides. The segments are joined together at the blend points using tangency conditions, Figure 5.5. In total 28 parameters are required for describing a complete blade section geometry with circular edges where the set of parameters can be split into 3 leading edge, 3 trailing edge, 10 suction side, 10 pressure side, and 2 global parameters.

The section build up process starts with the determination of the tangential angles τ_I^{SS}, τ_I^{PS} at the blend points $\mathbf{b}_0^{SS}$, $\mathbf{b}_0^{PS}$ by trigonometrical relations between the metal angle β_I, wedge angle μ_I, and the radius r_I of the leading edge circle. As a next step the global parameters, i.e. stagger angle ξ and chord length C, in combination with the blade exit metal angle β_E, exit wedge angle μ_E, and trailing edge radius r_E are used for the determination of the blend points $\mathbf{b}_7^{SS}$, $\mathbf{b}_7^{PS}$ and the tangential angles τ_E^{SS}, τ_E^{PS} at the trailing edge. Suction and pressure sides of the blade sections are parameterized each by a cubic B-spline (2.6) with 8 control

points $\mathbf{b}_i$, $i = 0(1)7$, demonstrated for the suction side in Figure 5.6. The control point $\mathbf{b}_0^{SS}$ and $\mathbf{b}_7^{SS}$ are the starting and end points of the curve corresponding to the B-spline parameters $t = 0$ and $t = 1$.

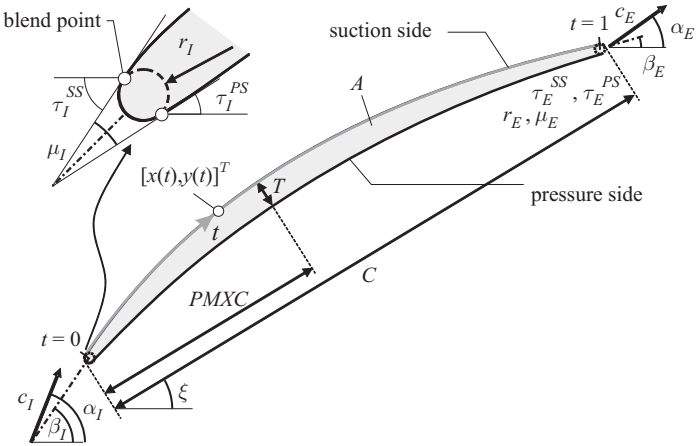


Figure 5.5: Blade section parameters

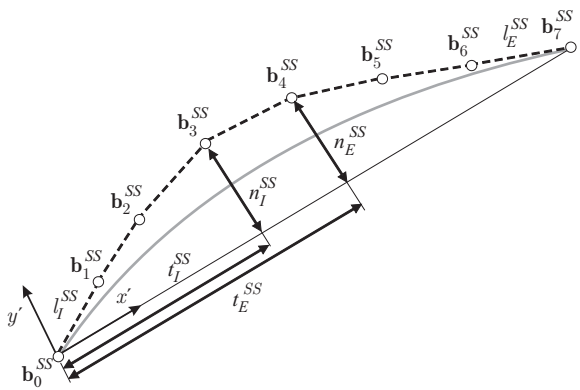


Figure 5.6: Blade suction side parameterization within *Parablading*

The points $\mathbf{b}_0^{SS}$, $\mathbf{b}_1^{SS}$, and $\mathbf{b}_2^{SS}$ are related. Point $\mathbf{b}_0^{SS}$ is the given blend point connecting the suction side with the leading edge circle while $\mathbf{b}_1^{SS}$ is determined by a free parameter l_I^{SS} describing the distance along the tangential direction τ_I^{SS}. The control point $\mathbf{b}_2^{SS}$ can be determined by the position of the first two points and additionally the curvature $\kappa_I^{SS} = \kappa^{SS}(0)$ and the stiffness $\sigma_I^{SS} = \sigma^{SS}(0)$ at the blend point $\mathbf{b}_0^{SS}$ where the curvature and the stiffness at each curve position t can be determined by

$$\kappa(t) = \frac{\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)}{\sqrt{[\dot{x}(t)^2 + \dot{y}(t)^2]^3}} \tag{5.4}$$

$$\sigma(t) = \frac{\dot{x}(t)\ddot{x}(t) + \dot{y}(t)\ddot{y}(t)}{\sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}}, \tag{5.5}$$

Bronstein et al. (2005). In the same way the points $\mathbf{b}_5^{SS}$, $\mathbf{b}_6^{SS}$, and $\mathbf{b}_7^{SS}$ are determined by the trailing edge parameters l_E^{SS}, τ_E^{SS}, $\kappa_E^{SS} = \kappa(1)$, and $\sigma_E^{SS} = \sigma(1)$, respectively. The inner points $\mathbf{b}_3^{SS}$ and $\mathbf{b}_4^{SS}$ can be chosen freely by tangential coordinates t_I^{SS}, t_E^{SS} and normal coordinates n_I^{SS}, n_E^{SS} in a local x'-y'-coordinate system. The pressure side control points $\mathbf{b}_i^{PS}$, $i = 0(1)7$, are determined analogously to the suction side procedure.

5.4 Blade Design Process

The two-dimensional blade design process consists basically of two steps, geometry generation and design evaluation. Generally these steps are separated and performed manually leading to a rather time-consuming design task. As a first goal of this investigation the bade design process is accelerated and automated by the use of *iSight* as a front end and control tool to integrate the two design programs into one common blading process.

Figure 5.7 illustrates the blading process as it is integrated into the *iSight* environment. When the optimization algorithm requires a new design evaluation, the file parsing functionality in *iSight* is used in order to substitute values of the design parameters in the input file of the blade geometry generation program

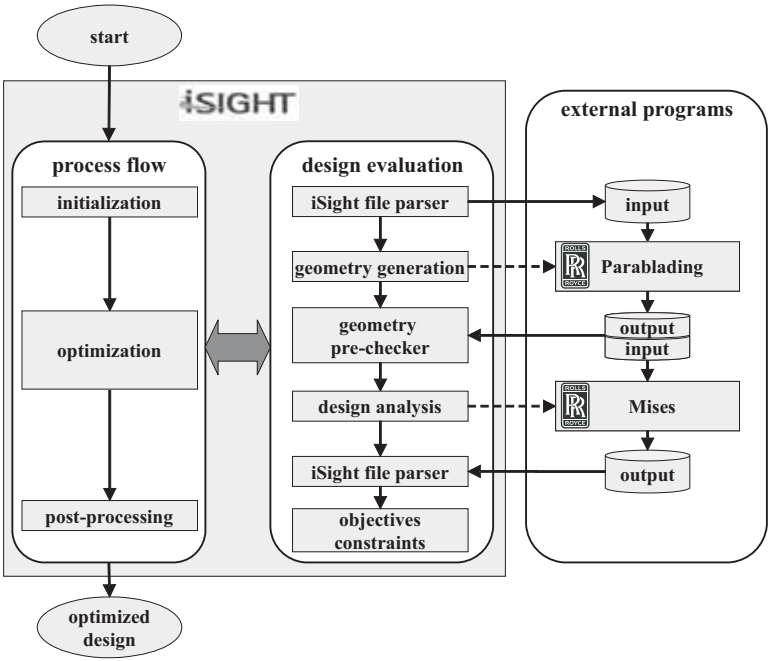


Figure 5.7: Illustration of the automated 2D-blade optimization process

Parablading. After the program is invoked, a new geometry is generated according to the input parameters which is stored in an output file needed for the subsequent two-dimensional flow simulation code *Mises*.

Before the time-consuming flow analysis is started, basic geometric parameters are extracted from the *Parablading* output file and checked if the new blade geometry fulfills sufficient geometric constraints as for example minimum cross section area or thickness to chord ratio. If at least one of the geometric constraints is violated the design process starts again from the beginning with a new blade section design. This is an important step, since genetic algorithms randomly generate blade profiles which may violate even elementary properties required for a smooth flow. Furthermore, this step helps to keep the number of flow simulations

as low as possible which has an positive impact to the overall optimization time.

In case of a well behaving geometry the flow domain is discretized with a structured H-type computational grid which along the flow consists of 60 grid lines in the inlet region, 70 grid lines on the blade section surface, and 45 lines in the outlet region, Figure 5.8. In order to resolve high gradient regions the stream lines are clustered in the computational domain, i.e. the spacing between grid lines varies. Across the flow 16 grid lines are used which are clustered towards the blade surface, and at the leading edge the streamlines are more dense due to the high local curvature.

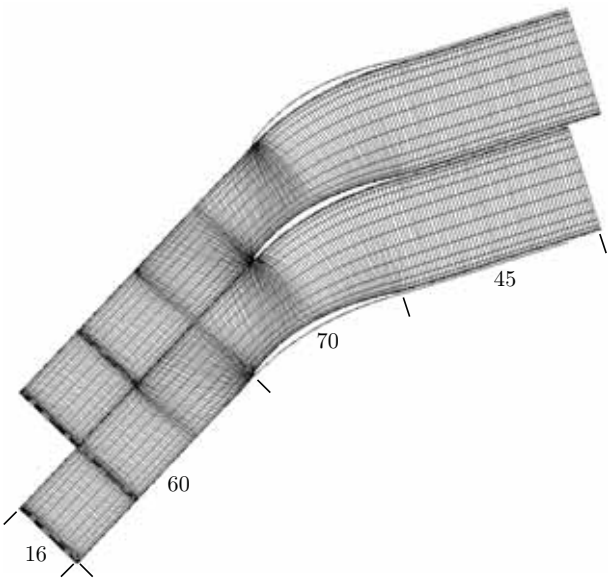


Figure 5.8: Computational grid used for the numerical flow analysis

The flow around the blade section is calculated with the flow solver *Mises*, *Drela (1986)*. It is a zonal approach coupling an inviscid outer flow with viscous boundary layers over blades and in wakes. The inviscid flow is modeled by the steady state Euler equations which are solved on a streamline grid of nodes. Viscous and inviscid flow equations are solved simultaneously with an iterative Newton-Raphson technique. The turbulence intensity is set to $Tu = 4\%$ in

order to simulate turbo machinery environment, and the Abu-Ghannam/Shaw transition model, *Abu-Ghannam and Shaw (1980)*, is used to calculate the free transition position on suction and pressure side, *Youngren and Drela (1991)*.

The results of the *Mises* calculation are stored into an output file from which *iSight* extracts the required information to evaluate objectives and constraints. The overall optimization proceeds until an optimal design is found or the maximum number of function evaluations is achieved. Finally, the results and trade-offs may be visualized in both the design and the criterion space in order to support the design engineer in his decision making.

5.5 Results and Discussion

In a first investigation a comparison of the two-dimensional blade design problem definitions (5.2) and (5.3) is performed according to the process flow in Figure 5.7. On the basis of an initial Rolls-Royce stator blade mid-section design, geometry modifications are performed at leading edge and suction side of the blade section geometry only, resulting in a vector of 12 free design variables

$$\mathbf{p} = \left[\beta_I, \mu_I, l_I^{SS}, \kappa_I^{SS}, \sigma_I^{SS}, t_I^{SS}, n_I^{SS}, t_E^{SS}, n_E^{SS}, l_E^{SS}, \kappa_E^{SS}, \sigma_E^{SS}\right]^T$$

where the leading edge circle radius r_I is kept fixed at a reasonable minimum value. The leading edge and the suction side parameters are chosen due to their major influence on the blade performance according to engineering experience. Additionally, parameter reduction as it is performed here keeps the design freedom small and the required number of function evaluations low. The calculations are performed with constant inlet flow conditions for both investigated definitions, Table 5.1, where the outlet parameters are results of the flow analysis which have to be matched to the predefined outlet flow conditions given by a throughflow calculation.

The *Non-dominated Sorting Genetic Algorithm* (NSGA-II), Section 2.3.4.4, is used with an initial population size of 50 individuals running over 124 generations which is equivalent to 6200 function evaluations. Each evaluation step requires 3 flow computations with *Mises* according to the loss evaluation at the three inlet

Table 5.1: Inlet flow conditions for 2D-blading

	M_I	α_I	Re_I	$P_{0,I}$	P_I	c_I
value	0.776	46.83	1.1E06	438.7	295.2	310.1
unit	[-]	[deg]	[-]	[kPa]	[kPa]	[m/s]

flow condition points.

Figure 5.9 shows the admissible designs in the criterion space for the first design problem (5.2). Altogether these are 40.7% feasible solutions of the 6200 possible design modifications generated by the genetic algorithm which fulfill all constraints. According to problem definition (5.2) the optimal trade-offs lie at the upper left border which are indicated as non-dominated solutions. As can be seen, most of the the designs generated by the optimization algorithm are clearly better than the datum design, and especially the non-dominated solutions show clearly better performance with respect to both objectives, loss at design point and working range.

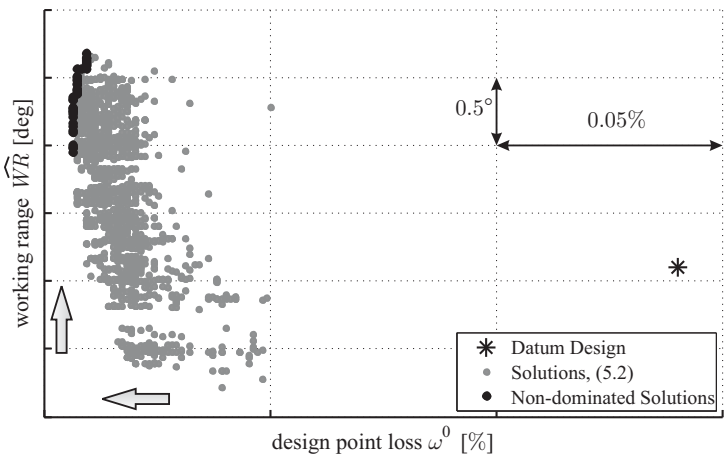


Figure 5.9: Criterion space and non-dominated solutions for blade design problem (5.2)

In Figure 5.10 the criterion space for the second problem definition (5.3) is shown. Each point is indicating a feasible solution while according to the problem

definition the optimal trade-offs can be found in the lower left corner indicated as non-dominated solutions. In this case 94.9% of 6200 function evaluations converged and 51.4% feasible solutions are obtained. It can be seen that the trade-offs are concentrated in the lower left corner and that the differences are rather small. Due to the fact that no real Pareto-curve exists it can be assumed that this problem definition leads to a single optimal solution which corresponds to the point in the lower left corner. Unfortunately, a direct comparison of the solutions with the datum design or the results of problem (5.2) is not possible, because γ is a problem specific value which is not contained in the other definitions. However, a comparison of the real working range values is possible from which the improvements in the design can be extracted.

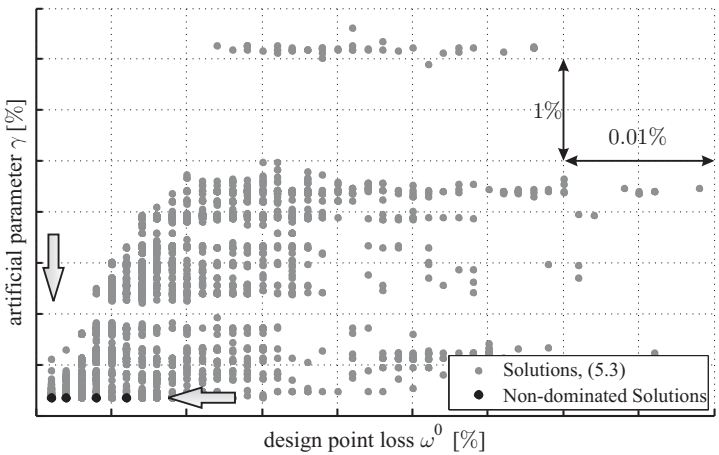


Figure 5.10: Criterion space and non-dominated solutions for blade design problem (5.3)

Thus, for the non-dominated solution of design problem (5.2) which provides the highest artificial working range value $\widehat{WR}$ and the solution of (5.3) which provides the minimum losses, full loss curve calculations are performed. In Figure 5.11b a comparison of the loss curves of these two non-dominated solutions with the datum design is shown. It can be seen that both optimized designs show similar loss distributions with an improvement in working range. At a reasonable

loss level of 4% a working range rise of about 1.0° with respect to the datum design is achieved. Additionally, a reduction of the loss at design point is observable reflected by approximately 6% for both optimized designs. The increased performance can also be concluded from the higher static pressure rise coefficient

$$c_p = \frac{P_E - P_I}{P_{0,I} - P_I} \tag{5.6}$$

with the exit and inlet static pressure P_E, P_I and the inlet total pressure $P_{0,I}$ in Figure 5.11a at stall and choke region. A comparison of the blade section geometries confirms that both optimized designs are very close to each other and only tiny changes in the geometry can be found, Figure 5.12.

In general, the optimized designs show more front loading reflected by a wider inlet wedge angle and a position of maximum thickness which is moved towards the leading edge. If we summarize the results of this first investigation, an improvement of a two-dimensional blade section design with respect to working range and pressure loss has been achieved. Both alternative formulations of the original blade optimization problem definition show promising results whereby no clear favorite can be pointed out.

In Table 5.2 an overview of the process performance related parameters is given. For an equal number of function evaluations, problem definition (5.2) shows a higher convergence rate which is important in terms of process time. The reason for this is that each non-converged solution requires more process time since the flow solver is running until the maximum number of its inner iterations is reached or for the worst case the flow solver or even the grid generator is crashed. This effect is substantiated by the lower overall optimization time for problem definition (5.2) due to its higher convergence rate. Hence, if the process time is considered in a comparison, definition (5.2) shows slightly better performance.

Table 5.2: Comparison of process performance parameters

	function evals	converged solutions	feasible solutions	total time [h]
definition (5.2)	6200	97.5%	40.7%	88.1
definition (5.3)	6200	94.9%	51.4%	99.7

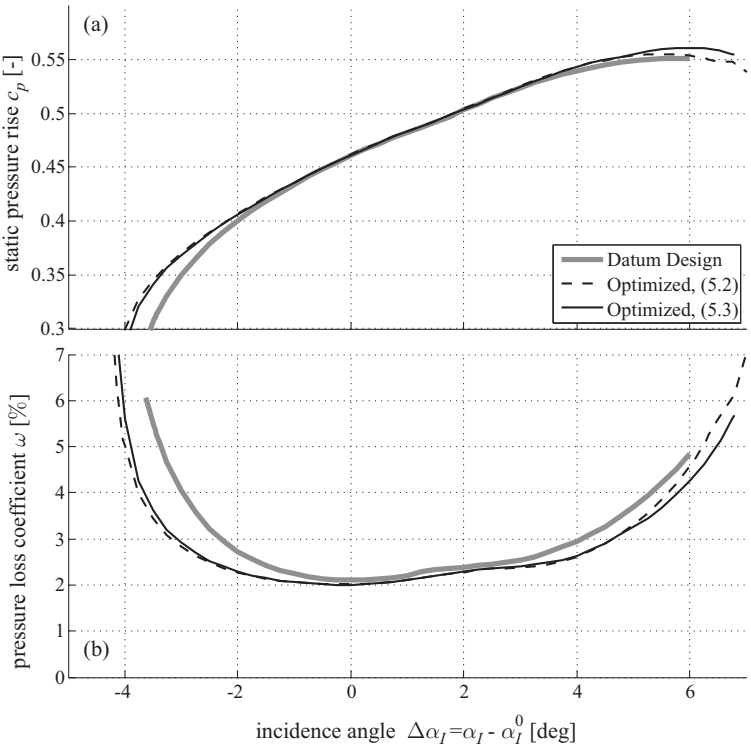


Figure 5.11: Comparison of static pressure rise (a) and pressure loss coefficient (b) between datum design and selected optimized solutions of problem definition (5.2) and (5.3)

In a second investigation the two-dimensional blade section optimization problem is solved with an increased number of design freedom. Beside the leading edge and suctions side parameters, geometry modifications on trailing edge and pressure side are considered. Furthermore, the stagger angle ξ as a global parameter is part of the new design vector

$$\begin{aligned}\mathbf{p} \;=\; & \big[\beta_I, \mu_I, l_I^{SS}, \kappa_I^{SS}, \sigma_I^{SS}, t_I^{SS}, n_I^{SS}, t_E^{SS}, n_E^{SS}, l_E^{SS}, \kappa_E^{SS}, \sigma_E^{SS}, \dots \\ & \dots, l_I^{PS}, \kappa_I^{PS}, \sigma_I^{PS}, t_I^{PS}, n_I^{PS}, t_E^{PS}, n_E^{PS}, l_E^{PS}, \kappa_E^{PS}, \sigma_E^{PS}, \beta_E, \mu_E, \xi\big]^T\end{aligned}$$

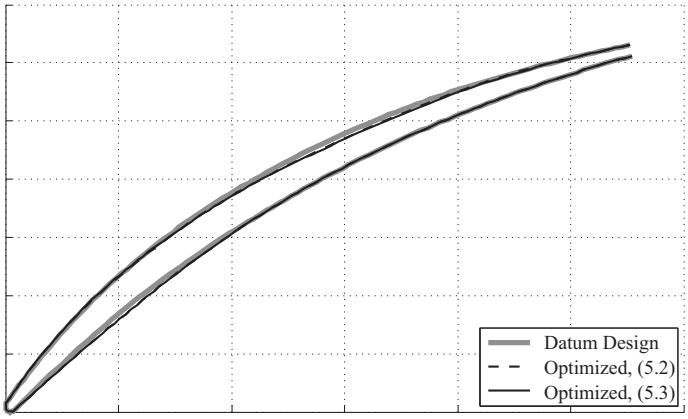


Figure 5.12: Comparison of datum design with optimized blade section geometries

which consists of 25 design parameters. Again, leading and trailing edge radii r_I, r_E are kept fixed during the design process which is performed according to Figure 5.7. The NSGA-II optimization algorithm is chosen with 50 individuals in each population while the number of generations is increased to 200 due to the extended design parameter vector. Optimization problem definition (5.2) is chosen due to the promising results from the previous investigation.

The results of this investigation are shown in the criterion space, Figure 5.13, where beside the feasible solutions trade-offs are indicated for comparison with the datum design. In general, it can be observed that the distribution of the solution points as well as the position and expansion of the non-dominated solution front looks very similar to the previous investigation where leading edge and suction side parameters are modified only, Figure 5.9. A detailed comparison of the non-dominated solutions of both cases shows that the increased design freedom on blade modification results in a slightly higher value for the potential working range.

In order to validate and compare the results of the non-dominated solutions which provide the highest working range value, the corresponding designs of the previous and the current investigation are selected and loss curve calculations are performed within the range of $\pm 10^\circ$ deflection to the datum inlet flow an-

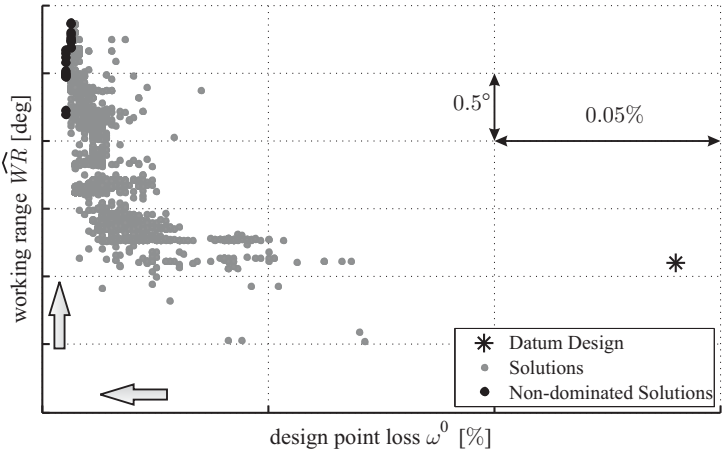


Figure 5.13: Criterion space and non-dominated solutions for blade section optimization with 25 parameters

gle. The resulting pressure rise coefficient and loss distributions are shown in Figure 5.14 where additionally the results of the datum design are included for better comparison.

As can be seen, the optimized designs are better with respect to both objectives, working range and loss at the design flow angle, compared with the datum design. If all blade section parameters can be changed, an increase in static pressure for higher inlet flow angles, i.e. in the stall region, can be achieved. This is reflected in a gain of working range whereas in the choke region the improvement with respect to the datum design is rather small. In case of leading edge and suction side geometry modification only, the improvement in working range is mainly obtained for lower relative inlet flow angles driven by a higher static pressure rise. In general, the loss curves of both optimized designs are slightly shifted to each other, but the level of improvement is rather equal. In terms of loss at design flow angle both show a reduction of about 6% compared to the datum design whereas the improvement in working range is approximately 1.0° at a reasonable working range loss level of $\omega_{WR} = 4\%$.

If the resulting blade section geometries are compared, Figure 5.15, only tiny

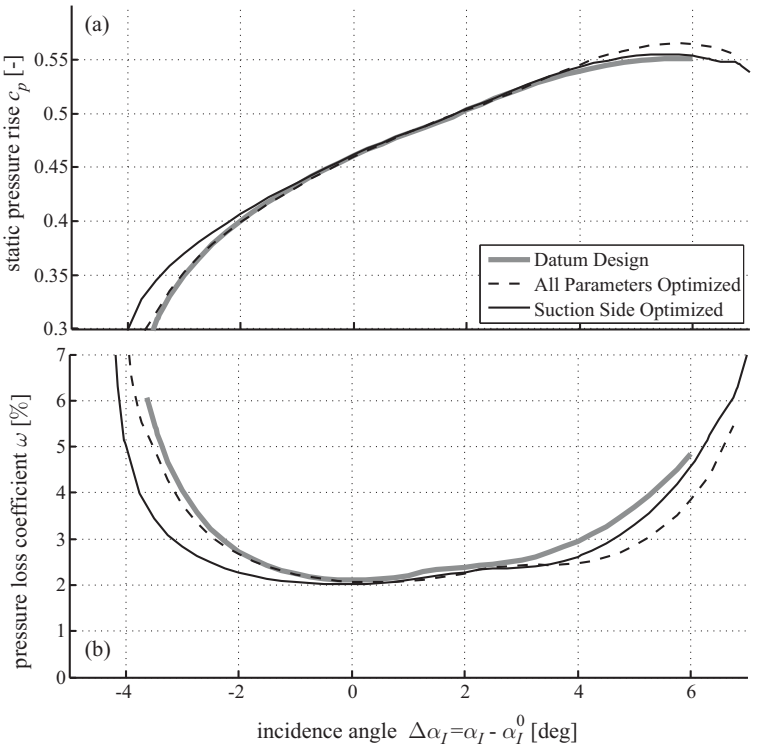


Figure 5.14: Comparison of static pressure rise (a) and pressure loss coefficient (b) between datum and optimized designs

deflections in the geometry can be found. Both optimized designs are very close to each other, in particular the redistribution of the cross section area caused by a higher inlet wedge angle and the movement of the maximum blade thickness towards the leading is equivalent. In other words, compared with the datum design both optimized sections show a more front loaded behavior which is the reason for the improvement in working range.

This effect can be better seen from a comparison of the isentropic Mach number distributions, Figure 5.16, which are calculated based on the pressure relation

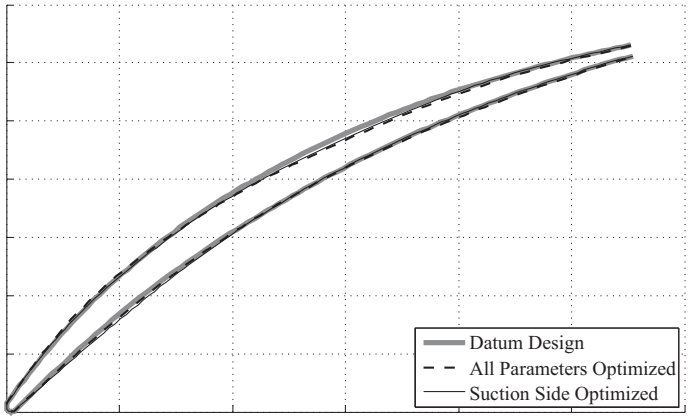


Figure 5.15: Comparison of optimized blade section geometry

$$M_{isen} = \sqrt{\frac{2}{\gamma - 1}\left(\left(\frac{P_0}{P}\right)^{\frac{\gamma - 1}{\gamma}} - 1\right)} \tag{5.7}$$

where P is the static pressure on the blade surface, P_0 the total pressure and γ the ratio of specific heat capacities. Each Mach number distribution shows two peaks close to the blade leading edge which are results of the curvature discontinuity at the two leading edge blend points, and increasing Mach number values due to flow acceleration at the blade suction side. It can be observed that both optimized designs show a stronger flow acceleration on the suction side and hence a higher Mach number compared to the datum design. It is also fact that the axial position of this maximum Mach number is located more at the front of the blade section according to the described geometry modifications. Furthermore, the loading near the leading edge is increased and decreased in the middle part, while it appears almost unchanged at the trailing edge. It is interesting to notice that in spite of the fact that the higher supersonic Mach numbers of both optimized cases cause more shock losses the overall loss is reduced for both. Generally this investigation confirms that front-loaded blade designs typically lead to wider working ranges.

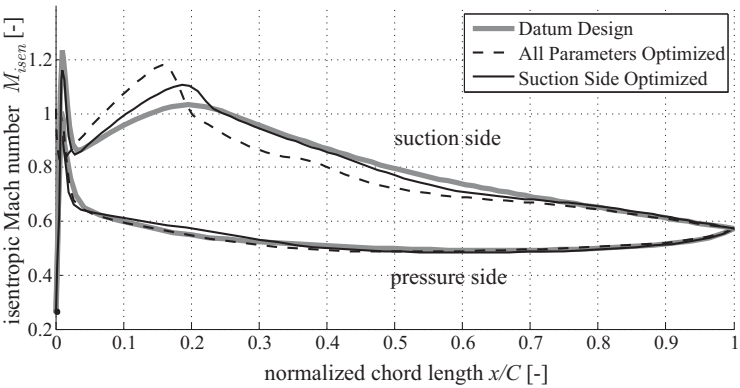


Figure 5.16: Isentropic Mach number distributions for datum and optimized blade sections

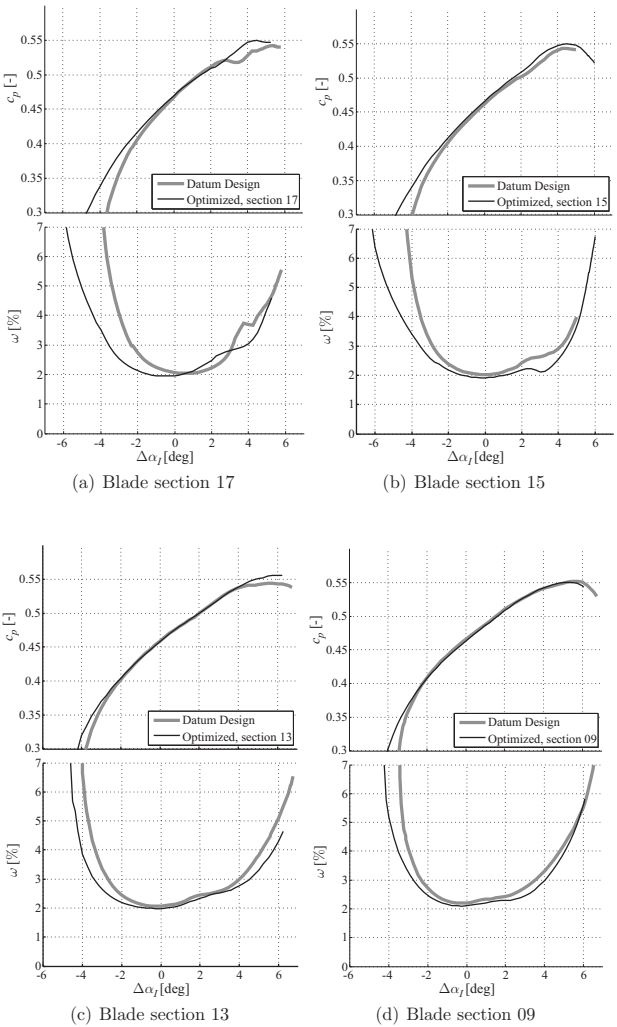
Summarizing, the results of these two investigations have shown that a compressor blade section can be designed automatically and design improvements with respect to multiple objectives can be made. The application of a multi-objective optimization strategy additionally provides multiple solutions which can be used to discuss trade-offs and from which the design engineer can choose a solution depending on additional decision criteria.

A complete three-dimensional blade geometry can be found by designing several two-dimensional blade sections which are radially stacked. Therefore, it is obvious to use the demonstrated automated blade design and optimization process also for finding the other desired blade section geometries. In order to demonstrate the general feasibility, the presented approach for automated blade design is applied to further blade sections of the same Rolls-Royce high pressure compressor stator blade design. In this particular case, the blade is divided into 21 sections starting with section 01 at the hub and ending with section 21 at the casing, where section 11 describes the mid-section of the blade. Based on the already optimized mid-section, six further sections are selected for improvement, three between mid-section and hub and three towards the casing where sections close to hub and casing are avoided due to possible flow solver instabilities. The multi-objective design problem is solved based on problem definition

(5.2) with the NSGA-II algorithm where 50 individuals in each population are treated over 200 generations. All parameters for leading and trailing edge, suction and pressure side as well as the stagger angle are variable for each section during the optimization process where the bounds for the design parameters are selected according to the minimum and maximum values of the datum design. The boundary conditions for the flow solver are also adapted for each section optimization.

Figures 5.17 and 5.18 show the resulting static pressure rise coefficient and loss distributions for the upper sections 17, 15, and 13 as well as for the lower sections 09, 07, and 05, respectively. In general, an improvement in working range and loss at design flow conditions can be observed for all optimized section designs. The increase in working range is mainly driven by better choke properties and in some particular cases even by higher pressure rise at stall. Section 17 shows the highest working range improvement of approximately 2° at a reasonable working range loss level $\omega_{WR} = 4\%$ whereas the improvements for the other sections lie between 0.5° and 1.5°. Furthermore, it can be seen that almost each loss curve is symmetric around the design inlet flow angle which is an additional property of the problem definition. Hence, beside the improvement in loss at design flow conditions, in almost each optimized section a decrease in loss production at off-design conditions can be observed.

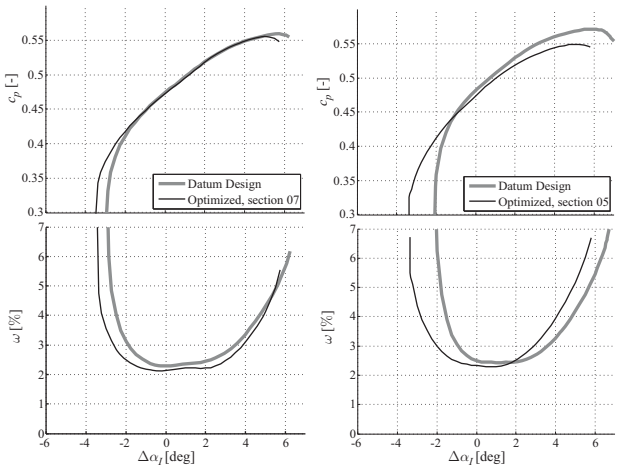
The biggest differences in the loss curves and static pressure rise distribution can be found on section 05 where the datum design is shifted in horizontal direction, Figure 5.18b. The intention of the design engineer is to gain more stall margin by a re-definition of the design inlet flow angle in choke direction. This is a common and very easy method for achieving more stall margin in a manual compressor blade design process. However, this increases also the losses at design inlet flow conditions and is not an optimal solution to the design problem. The optimized section 05 which shows a symmetrical loss distribution around the design inlet flow angle can be interpreted as a trade-off solution between lower losses and lower stall margin. If the loss reduction is mainly pursuit, the current optimized design would be the choice. If more stall margin is desired, the optimization problem has to be adapted. This, however, is not a big issue since a simple extension of the optimization problem definition by an additional constraint would ensure the required value.



(a) Blade section 17 (b) Blade section 15

(c) Blade section 13 (d) Blade section 09

Figure 5.17: Static pressure rise coefficient and loss distribution for selected blade sections - part 1



(a) Blade section 07 (b) Blade section 05

Figure 5.18: Static pressure rise coefficient and loss distribution for selected blade sections - part 2

Figure 5.19 presents a summary of the radial distributions of the design point loss ω^0 and the working range WR at a comparable working range loss level of $\omega_{WR} = 4\%$ for the datum and the seven optimized blade sections. The comparison shows that the application of automated two-dimensional blade design optimization to all blade sections produces promising results. In general, a reduction in pressure loss at design inlet flow conditions and an increase in working range can be observed for each optimized blade section design. It should be noticed that all these results are taken just as one out of a set of non-dominated solutions which provide various optimal solutions in terms of multi-objective optimization. The optimal designs in Figure 5.19 represent the best non-dominated solutions with respect to the working range criteria, however, if the pressure loss at design point is more important, other optimal design points can be chosen from the set of available non-dominated solutions resulting in a different radial distribution for the objective functions.

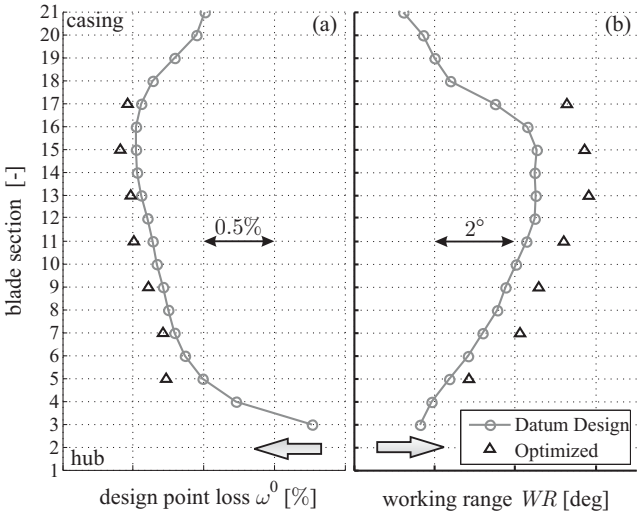


Figure 5.19: Comparison of radial distributions of pressure loss coefficient (a) and working range (b) at $\omega_{WR} = 4\%$ for seven optimized blade sections with datum design

If the optimized blade sections from Figure 5.19 are stacked along a radial stacking line, a three-dimensional blade geometry is obtained which can be seen in Figure 5.20. The comparison with the datum design shows that the new geometry differs significantly. The sections 17, 13, 11, 09, and 05 look rather smooth whereas sections 15 and 07 produce undesired wiggles in the three-dimensional blade geometry. In particular, the trailing edge shows the biggest deflection which is mainly caused by the corresponding stagger angles.

In terms of industrial design, this three-dimensional blade geometry would not be acceptable since the required radial smoothness is not fulfilled. The reason for this is that each optimized section is chosen independently based on its maximum working range value only. Hence, it would be better to select the blade sections from their set of non-dominated solutions depending on an additional criterion which considers the neighboring section geometries in order to obtain a smooth three-dimensional blade design. This is just one simple method which can be implemented easily in the current process, however, the better approach would

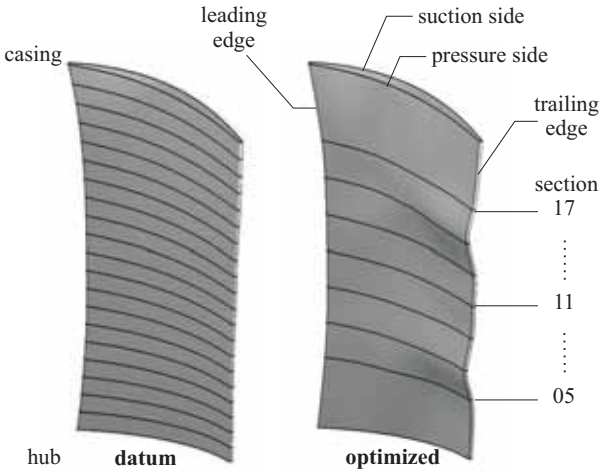


Figure 5.20: Comparison of three-dimensional blade geometry for datum design (left) and optimized design (right) obtained by seven individual multi-objective blade section optimization runs

be to consider the required radial smoothness of the blade geometry during the multi-objective blade optimization process.

6 Conclusions and Outlook

This thesis illustrates some possibilities of process integration and automated optimization for supporting the aerodynamic compressor design process. It demonstrates that time consuming design tasks can be automated and significant reductions in the overall design cycle time are achievable. In most aerodynamic design problems multiple contradicting design goals have to be pursuit simultaneously and many constraints have to be considered. The multi-objective optimization strategy is a method for solving these kind of problems in an accurate way and provides trade-off solutions from which the design engineer can choose depending on his experience and his final decision. It is clear that setting-up the design process and choosing appropriate optimization algorithms are prerequisite for the success, and expertise in process definition, problem formulation and optimization algorithms is essential.

As a first demonstration for the benefits that can be achieved by process integration and optimization, the highly iterative annulus design task which is part of the preliminary design process was automated and numerical optimization methods were performed on the basis of stochastic and deterministic methods. In order to achieve feasible solutions, constraints on typical compressor design parameters have been considered. Based on a Rolls-Royce 9-stage high pressure compressor design, a significant increase in conflicting design objectives, namely efficiency, surge margin, and overall compressor pressure ratio, as well as an improvement in process acceleration, resulting in a speed-up factor of 12, are shown. Beside the process improvements, automated design can find limits and admissibilities of the problem definition and of the design tools. As an important result of this investigation, an issue in the reliability of the preliminary analysis tool is pointed

out. Since the process integration is flexible, an adaption of the automated preliminary design process by including an additional analysis for the determination of the surge margin value was performed. Non-dominated solutions are achieved with a deterministic optimization method using the compromise approach and a stochastic multi-objective optimization algorithm where improvements in compressor efficiency and surge margin can be observed.

The benefits of process integration and automation are presented also on the throughflow off-design calculation which is a subtask of the throughflow calculation process. Based on a Rolls-Royce 6-stage high pressure compressor design, a gradient based optimization algorithm is used to find solutions to the multi-objective throughflow off-design problem where three different scalarization techniques are applied in order to transfer the multiple goals into a single-objective design problem. Improved solutions to the design problem are found and the required design time is accelerated by a factor of 12 compared to the manual solution finding process.

In the last part of this work the compressor blade design problem was conducted. As an important step the two-dimensional blade design problem was re-formulated and two alternative problem definitions were provided which are characterized by a lower number of required function evaluations compared to the standard procedure. Multi-objective optimizations based on a Rolls-Royce stator blade mid-height section show promising results with significant improvements in pressure loss at design flow conditions and working range with a multi-objective genetic optimizer considering aerodynamic and geometric constraints. The design process is accelerated by factor of at least 10 since the manual procedure of finding the working range by at least 30 individual flow calculations is reduced to only three points. In a final step a method for three-dimensional blade design optimization was introduced based on multiple individual blade section designs. The automated blade design process was applied to further radially distributed blade sections and improved designs are selected to be compound to a three-dimensional blade shape.

It is obvious that the aerodynamic compressor design process is too complicated to cover all aspects in this work. From the industrial point of view it is required to implement more tools and more processes in order to come closer to the real engineering work flow.

In the demonstrated preliminary design process improvements for design point conditions are achieved only. However, a compressor has to work also for different operating conditions which makes the design process more complicated. As an outlook for further work an extension of the presented design tasks by introducing more design parameters is possible. This increases the design freedom and the chance to find better solutions to the design problems, but also requires a longer optimization time or an adaption of the optimization algorithms and strategies. Furthermore, an extension of the design problems by considering more objectives and constraints can be done in order to cover additional design aspects in the preliminary design phase.

The proposed three-dimensional blade design approach based on several radially stacked optimized blade sections is one possible design method. However, flow around a blade is highly three-dimensional and very sensitive in term of surface smoothness which was not considered in the presented approach directly. As an possible extension of the blading process an additional design criteria for the blade section determination could be used to guarantee radial smoothness of the blade geometry. It is also worth to investigate new parameterization methods describing three-dimensional blade shapes which imply parameter reduction and geometry smoothness with a maximum of design freedom. In order to catch as much flow phenomena as possible, in particular at the blade hub and casing region, the usage of a more sophisticated CFD method with an appropriate turbulence model is recommended.

Appendix: Aerodynamic Compressor Design Parameters

The appendix presents brief descriptions and definitions of important aerodynamic compressor design parameters which are used in this thesis.

A.1 De Haller Number

The de Haller number DH is a simple indicator for endwall loading, i.e. separation of the boundary layer at the endwall, *de Haller (1953)*. It is defined as fraction of the outlet velocity w_E to the inlet velocity w_I:

$$DH = \frac{w_E}{w_I}. \tag{A.1}$$

De Haller deduced that the velocity out of a blade row should not be less than about 0.75 times the inlet velocity if the performance is to be satisfactory. This limit is determined empirically and therefore it is not really fix. Due to different design philosophies and new technologies it is possible to reduce this value down to 0.6.

A.2 Blockage

The blockage BL is a result of viscous flow phenomena within the compressor. It is defined as the relation between the effective cross section area A_{eff} due to annulus wall boundary layer evolution and the geometrically available flow area A_{geom}:

$$BL = \left(1 - \frac{A_{eff}}{A_{geom}}\right) \cdot 100\%. \tag{A.2}$$

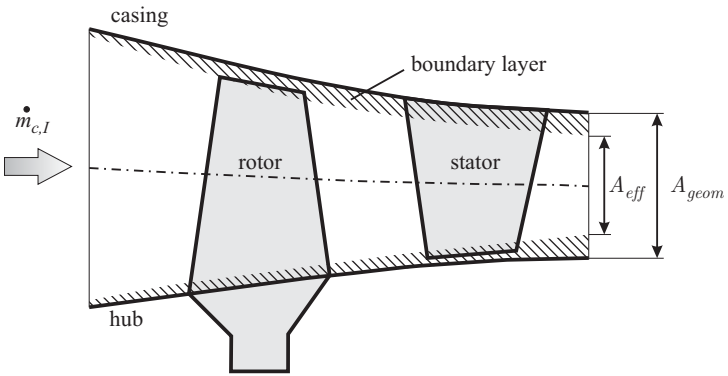


Figure A.1: Definition of blockage

A.3 Diffusion Factor

The diffusion factor DF was derived by *Lieblein et al. (1953)* and is an assessment for the blade loading. It relates empirically the peak velocity on the suction side of the blade to the velocity at the trailing edge:

$$DF = 1 - \frac{w_E}{w_I} + \frac{\Delta w_u}{2\,\sigma\,w_I} \tag{A.3}$$

The first term is the one-dimensional deceleration of the flow equivalent to the de Haller number (A.1) whereas the second term describes the loading due to flow turning correlated with the blade solidity where Δw_u is the change in the circumferential component of the relative velocity and σ is the solidity. The flow is critical for $DF > 0.5$ and values more then 0.6 indicate blade stall, *Cumpsty (2004)*.

A.4 Efficiency

In compressor design two efficiency values are distinguished. The compressor isentropic efficiency $\eta_{c,isen}$ is a value where it is assumed that the process of flow compression is performed adiabatic and reversible. It can be calculated by

$$\eta_{c,isen} = \frac{\left(\frac{P_{0,c,E}}{P_{0,c,I}}\right)^{\frac{\gamma-1}{\gamma}} - 1}{\frac{T_{0,c,E}}{T_{0,c,I}} - 1} \tag{A.4}$$

where γ is the ratio of specific heat capacities, $P_{0,c,E}$ and $P_{0,c,I}$ are the compressor exit an inlet total pressures while $T_{0,c,E}$ and $T_{0,c,I}$ are the exit and inlet total temperatures.

The isentropic efficiency is not a real indicator to compare the aerodynamic quality of two compressor designs. Due to definition (A.4) the isentropic efficiency of compressors with identical aerodynamic quality gets lower when the overall pressure ratio is increased and this trend may be confusing or misleading, *Cumpsty (2004)*. This can be avoided by using the polytropic efficiency which removes the penalty for higher pressure ratio so that compressors of equal aerodynamic quality but significantly different pressure ratio would have the same polytropic efficiency though a different isentropic efficiency. The compressor polytropic efficiency can be determined by

$$\eta_{c,poly} = \frac{\gamma-1}{\gamma} \cdot \frac{\ln\left(\frac{P_{0,c,E}}{P_{0,c,I}}\right)}{\ln\left(\frac{T_{0,c,E}}{T_{0,c,I}}\right)}. \tag{A.5}$$

A.5 Koch Parameter

A prediction of compressor instabilities in the early design phase is absolutely essential. *Koch (1981)* published a reliable method for assessing the stall margin capability of a new compressor design during the early preliminary design phase by estimating the pressure rise at stall. Based on the use of the large General

Electric data base of measurements on compressors he defined the Koch parameter as an enthalpy-equivalent static pressure rise coefficient of a stage i as

$$C_{h,i} = \frac{\Delta H_{0,i} - \frac{1}{2}\left(\left(u^R_{E,i}\right)^2 - \left(u^R_{I,i}\right)^2\right)}{\frac{1}{2}\left(\left(w^R_{I,i}\right)^2 + \left(c^S_{I,i}\right)^2\right)}. \tag{A.6}$$

The major idea is to correlate the basic rectilinear diffusor parameters length L and exit height H_2 with the geometry of a compressor blade passage described by the camber line length L and the exit passage width H_2, Figure A.2. The enthalpy-equivalent static pressure rise coefficient C_h is compared with the maximum static pressure rise of two-dimensional diffusors based on the correlations found by *Sovran and Klomp (1967)*. In axial compressor design it is required to keep the actual Koch parameter C_h below the maximum static pressure rise coefficient of the rectilinear diffusor in order to avoid stall.

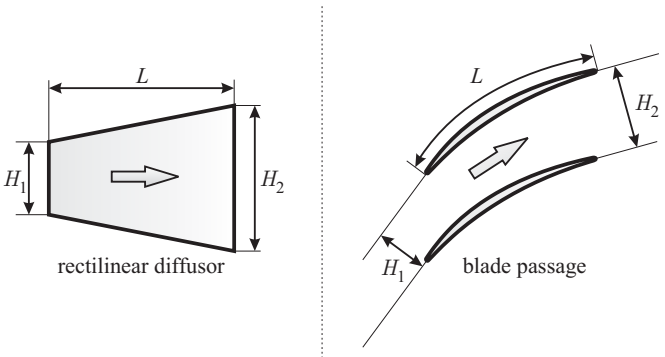


Figure A.2: Correlation between rectilinear diffusor and compressor blade row

A.6 Stage Pressure Ratio

The stage pressure ratio Π_i is defined as the total pressure at stage exit (stator exit) related to the total pressure at the stage inlet (rotor inlet) position:

$$\Pi_i = \frac{P^S_{0,E,i}}{P^R_{0,I,i}}. \tag{A.7}$$

A.7 Overall Pressure Ratio

The overall pressure ratio Π_c is defined as the product of all stage pressure ratios Π_i where N_s denotes the total number of stages:

$$\Pi_c = \prod_{i=1}^{N_s} \Pi_i. \tag{A.8}$$

A.8 Stage Loading

The stage loading Ψ is defined as ratio of the total enthalpy rise ΔH_0 of a stage with respect to the square of the circumferential speed u. Since the total enthalpy can be calculated by the total temperature T_0 and the specific heat capacity c_p according to $H_0 = c_p T_0$, the stage loading becomes

$$\Psi = \frac{\Delta H_0}{u^2} = \frac{c_p\, T_{0,E} - c_p\, T_{0,I}}{u^2}. \tag{A.9}$$

It should be noticed that u can either be taken as the blade tip speed or the speed at mid-height radius which is more common, and the enthalpy change can be the static or total enthalpy depending on the context where the total is more common, *Cumpsty (2004)*.

A.9 Surge Margin

There are many different ways of defining surge margin, but one of the most simple is illustrated in Figure A.3:

$$SM = \frac{\Pi_s - \Pi_c}{\Pi_c} \cdot 100\% \tag{A.10}$$

where Π_c is the compressor design point pressure ratio on the working line for the given shaft speed and Π_s is the predicted pressure ratio on the surge line for the same mass flow function which is generally defined as

$$FLF = \frac{\dot{m}_{c,I}\sqrt{T_{0,c,I}}}{P_{0,c,I}}. \tag{A.11}$$

In a multistage compressor for use in a turbojet engine, it is normal to insist on a surge margin of about 25%, *Cumpsty (2004)*. If the complete compressor map is not available, it is more appropriate to determine the surge margin from parameters obtained from a single shaft speed. This leads to another very common surge margin definition for the design point based on the maximum and working line pressure ratios and the corresponding flow function values at this single speed line given by

$$SM = \left(\frac{\frac{\widetilde{\Pi}_s}{FLF_s}}{\frac{\Pi_c}{FLF_c}} - 1 \right) \cdot 100\%. \tag{A.12}$$

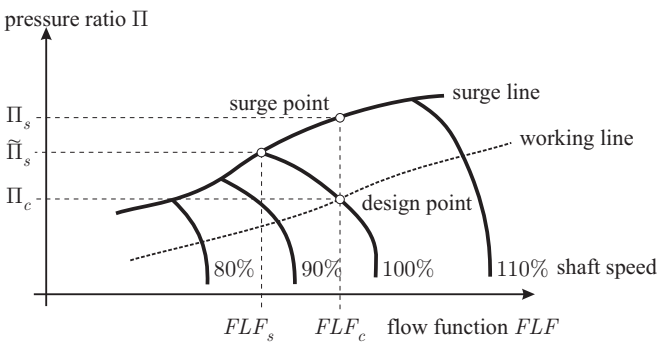


Figure A.3: Definition of compressor surge margin

A.10 Total Pressure Loss Coefficient

In the field of turbo machinery, different possible blade loss definitions exist which describe the pressure reduction between blade exit and inlet position. A common definition is given as

$$\omega = \frac{P_{0,E,isen} - \bar{P}_{0,E}}{P_{0,I} - P_I} \tag{A.13}$$

where $P_{0,E,isen}$ is the exit total pressure for an isentropic case, $\bar{P}_{0,E}$ the averaged exit total pressure, and $P_{0,I}$, P_I are the inlet total and static pressures, respectively.

List of Figures

List of Tables

References

Abu-Ghannam, B. J. and R. Shaw (1980). Natural Transition of Boundary Layers - The Effects of Turbulence, Pressure Gradient, and Flow History. *Journal of Mechanical Engineering Science 22*(5), pp. 213–228.

ACARE (2001). *European Aeronautics: A Vision for 2020*. Advisory Council for Aeronautics Research in Europe. http://www.acare4europe.org.

Ahmed, R. and M. Lawerenz (2004). Auslegung vielstufiger Axialverdichter mit parallelen Optimierungsstrategien und neuronalen Netzen. *DGLR-Jahrestagung, DGLR-2004-263*.

Bäck, T., U. Hammel, and H.-P. Schwefel (1997). Evolutionary Computation: Comments on the History and Current State. *IEEE Trans. Evolutionary Computation 1*(1), pp. 3–17.

Benini, E. (2004). Three-Dimensional Multi-Objective Design Optimization of a Transonic Compressor Rotor. *Journal of Propulsion and Power 20*(3), pp. 559–565.

Bestle, D. (1994). *Analyse und Optimierung von Mehrkörpersystemen*. Berlin: Springer.

Birk, L. (2003). Introduction to Nonlinear Programming. In L. Birk and S. Harries (Eds.), *Optimistic. Optimization in Marine Design*, Berlin: Mensch & Buch, pp. 53–82.

Bronstein, I. N., K. A. Semendjajew, and G. Musiol (2005). *Taschenbuch der Mathematik*. Frankfurt am Main: Harri Deutsch.

Büche, D. (2004). *Multi-Objective Evolutionary Optimization of Gas Turbine Components*. Aachen: Shaker.

Büche, D., G. Guidati, and P. Stoll (2003). Automated Design Optimization of Compressor Blades for Stationary, Large-Scale Turbomachinery. *Proceedings of ASME Turbo Expo 2003, GT2003-38421*.

Burgubur, S., C. Toussaint, C. Bonhomme, and G. Leroy (2003). Numerical Optimization of Turbomachinery Bladings. *Proceedings of ASME Turbo Expo 2003, GT2003-38310*.

Castillo, L., X. Wang, and W. K. George (2004). Separation Criterion for Turbulent Boundary Layers Via Similarity Analysis. *ASME Journal of Fluids Engineering 126*(3), pp. 297–304.

Chapra, S. C. and R. P. Canale (2001). *Numerical Methods for Engineers: With Software and Programming Applications*. New York: McGraw-Hill.

Chung, J. and K. D. Lee (2002). Shape Optimization of Transonic Compressor Blade Using Quasi-Three-Dimensional Flow Physics. *AIAA Journal 40*(2), pp. 389–391.

Coello Coello, C. A., D. A. Van Veldhuizen, and G. B. Lamont (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York: Kluwer Academic Publishers.

Cumpsty, N. (2004). *Compressor Aerodynamics*. Malabar: Krieger Publishing Company.

de Haller, P. (1953). Das Verhalten von Tragflügelgittern in Axialverdichtern und im Windkanal. *Brennstoff-Wärme-Kraft, Band 5, Heft 10*, pp. 333–336.

Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester: John Wiley & Sons.

Deb, K., S. Agrawal, A. Pratap, and T. Meyarivan (2000). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. Merelo, and H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature – PPSN VI*, Berlin: Springer, pp. 849–858.

Dennis, B. H., G. S. Dulikravich, and Z.-X. Han (2001). Optimization of Turbomachinery Airfoils with Genetic/Sequential-Quadratic-Programming Algorithm. *AIAA Journal of Propulsion and Power 17*(5), pp. 1123–1128.

Dornberger, R., D. Büche, and P. Stoll (2000). Multidisciplinary Optimization in Turbomachinery Design. *European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS*.

Drela, M. (1986). Two Dimensional Transonic Aerodynamic Design and Analysis Using the Euler and Boundary Layer Equations. *Gas Turbin Laboratory at MIT Report* (187).

Farin, G. (1990). *Curves and Surfaces for Computer Aided Geometric Design. A Practical Guide*. San Diego: Academic Press Inc.

Farin, G. (2000). *The Essentials of CAGD*. Natick: AK Peters, Ltd.

Fletcher, R. (2000). *Practical Methods of Optimization* (2nd ed.). Chichester: Wiley.

Fletcher, R. and C. Reeves (1964). Function Minimization by Conjugate Gradients. *Computer Journal 7*, pp. 149–154.

Gill, P. E., W. Murray, and M. H. Wright (1995). *Practical Optimization*. New York: Academic Press.

Gräsel, J., A. Keskin, M. Swoboda, H. Przewozny, and A. Saxer (2004). A Full Parametric Model for Turbomachinery Blade Design and Optimisation. *ASME Proceedings of DETC, DETC2004-57467*.

Gümmer, V. (2000). *Pfeilung und V-Stellung zur Beeinflussung der dreidimensionalen Strömung in Leiträdern transsonischer Axialverdichter*. Fortschritt-Berichte VDI, Reihe 7, Nr. 384. Düsseldorf: VDI.

Harries, S. (1998). *Parametric Design and Hydrodynamic Optimization of Ship Hill Forms*. Berlin: Mensch & Buch.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Ann Arbor: University of Michigan Press.

Hooke, R. and T. A. Jeeves (1961). Direct Search Solution of Numerical and Statistical Problems. *Journal of the ACM 8*, pp. 212–229.

iSight (2004). *iSight Reference Guide 9.0*. Engenious Software, Inc. http://www.engineous.com.

Kaneko, M., M. Miki, and T. Hiroyasu (2000). A Parallel Genetic Algorithm with Distributed Environment Scheme. In H. R. Arabnia (Ed.), *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, Volume 2, Las Vegas: CSREA Press, pp. 667–674.

Keskin, A. (2001). Paramterische Beschreibung von Schaufelprofilen unter Berücksichtigung der Krümmungsmanipulation. Diplomarbeit, Technische Universität Berlin.

Keskin, A. and D. Bestle (2004). Verbesserter Designprozess für moderne Hochdruckverdichter. *Forum der Forschung 17*, pp. 37–42.

Keskin, A. and D. Bestle (2005). Application of Multi-Objective Optimization to Axial Compressor Preliminary Design. *DGLR Jahrestagung, DGLR-2005-192*.

Keskin, A., A. K. Dutta, and D. Bestle (2006a). Alternative Approach for Solving a Multi-Objective Optimization Problem in Aerodynamic Compressor Blade Design. In G. Winter, W. Haase, and J. Periaux (Eds.), *Design Optimisation: Methods & Application. Proceedings of ERCOFTAC 2006*, Las Palmas: University of Las Palmas de Gran Canaria, pp. 227–230.

Keskin, A., A. K. Dutta, and D. Bestle (2006b). Modern Compressor Aerodynamic Blading Process Using Multi-Objective Optimization. *Proceedings of ASME Turbo Expo 2006, GT2006-90206*.

Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi (1983). Optimization by Simulated Annealing. *Science 220*(4598), pp. 671–680.

Klinger, H. (2004). Das Engine 3E Kerntriebwerk von Rolls-Royce Deutschland. *DGLR Jahrestagung 2004, DGLR-2004-178*.

Koch, C. C. (1981). Stalling Pressure Rise Capability of Axial Flow Compressor Stages. *ASME Journal of Engineering for Power 103*, pp. 645–656.

Köller, U., R. Mönig, B. Küsters, and H.-A. Schreiber (2000). Development of Advanced Compressor Airfoils for Heavy-Duty Gas Turbines - Part I: Design and Opimization. *ASME Journal of Turbomachinery 122*, pp. 397–405.

Lewis, R., V. Torczon, and M. Trosset (2000). Direct Search Methods: Then and Now. *Journal of Computational and Applied Mathematics 124*(1), pp. 191–207.

Lieblein, S., F. C. Schwenk, and R. L. Broderick (1953). Diffusion Factor for Estimating Losses and Limiting Blade Loadings in Axial-Flow-Compressor Blade Elements. *NACA RM E53D01*.

Miki, M., T. Hiroyasu, and M. Kaneko (2000a). A Parallel Genetic Algorithm with Distributed Environment Scheme. In L. D. Whitley, D. E. Goldberg, E. Cantú-Paz, L. Spector, I. C. Parmee, and H.-G. Beyer (Eds.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, San Francisco: Morgan Kaufmann, pp. 376.

Miki, M., T. Hiroyasu, and M. Kaneko (2000b). New Crossover Scheme for Parallel Distributed Genetic Algorithms. *Proceeding of the IASTED International Conference, Parallel and Distributed Computing And Systems 1*, pp. 145–150.

modeFRONTIER (2006). *modeFRONTIER 3.0 User Manual*. Esteco Srl. http://www.esteco.com.

Moré, J. J. and S. J. Wright (1994). *Optimisation Software Guide - Frontiers in Applied Mathematics*. Philadelphia: Society for Industrial & Applied Mathematics.

Müller-Töws, J. (2000). *Aerothermodynamische Auslegung der Meridianströmung mehrstufiger Axialverdichter mit Hilfe von Optimierungsstrategien*. Dissertation, Universität Gesamthochschule Kassel.

Nelder, J. A. and R. Mead (1965). A Simplex Method for Function Minimization. *Computer Journal 8*, pp. 308–313.

Oyama, A. and M.-S. Liou (2002). Multiobjective Optimization of a Multi-Stage Compressor Using Evolutionary Algorithm. *AIAA Journal of Propulsion and Power, AIAA-2002-3535*.

Piegl, L. and W. Tiller (1997). *The NURBS Book* (2nd ed.). Berlin: Springer.

Rao, S. S. (1996). *Engineering Optimization: Theory and Practice* (3rd ed.). New York: Wiley-Interscience.

Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Stuttgart: Frommann-Holzboog.

Sasaki, D., A. J. Keane, and S. Shahpar (2006). Multiobjective Evolutionary Optimization of a Compressor Stage Using a Grid-Enabled Environment. *44^{th} AIAA Aerospace Science Meeting and Exhibition, AIAA-2006-340*.

Schittkowski, K. (1981). The Nonlinear Programming Method of Wilson, Han, and Powell with an Augmented Lagrangian Type Line Search Function. Part I: Convergence Analysis. *Numerische Mathematik 38*(1), pp. 83–114.

Schittkowski, K. (1986). NLPQL: A Fortran Subroutine Solving Constrained Nonlinear Programming Problems. *Annals of Operations Research 5*, pp. 485–500.

Schoenberg, I. (1946). Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions. *Qart. Appl. Math. 4*, pp. 45–99 and 112–141.

Sieverding, F., B. Ribi, M. Casey, and M. Meyer (2004). Design of Industrial Axial Compressor Blade Sections for Optimal Range and Performance. *ASME Journal of Turbomachinery 126*, pp. 323–331.

Sonoda, T., Y. Yamaguchi, T. Arima, M. Olhofer, B. Sendhoff, and H.-A. Schreiber (2003). Advanced High Turning Compressor Airfolis for Low Reynolds Number Conditions, Part 1: Design and Optimization. *Proceedings of ASME Turbo Expo 2003, GT2003-38458*.

Sovran, G. and E. D. Klomp (1967). Experimentally Determined Optimum Geometries for Rectilinear Diffusers with Rectangular, Conical or Annular Cross Sectins. In G. Sovran (Ed.), *Fluid Mechanics of Internal Flow*, New York: Elsevier Publishing Co., pp. 270–319.

Srinivas, N. and K. Deb (1995). Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation 2*(3), pp. 221–248.

Steffens, K. and A. Schäffler (2000). Triebwerksverdichter - Schlüsseltechnologie für den Erfolg bei Luftfahrtantrieben. *DGLR Jahrestagung, DGLR-JT2000-001*.

Trigg, M. A., G. R. Tubby, and A. G. Shreard (1999). Automatic Genetic Optimization Approach to Two-Dimensional Blade Profile Design for Steam Turbines. *ASME Journal of Turbomachinery 121*, pp. 11–17.

Walther, R., J. Frischbier, and R. Selmeier (2000). Aeromechanische Gestaltung fortschrittlicher Triebwerksverdichter. *DGLR Jahrestagung, DGLR-JT2000-105*.

Wu, C.-H. (1952). A General Theory of Three-Dimensional Flow in Subsonic and Supersonic Turbomachines of Axial-, Radial-, and Mixed-Flow Types. *NACA TN-2604*.

Youngren, H. and M. Drela (1991). Viscous/Inviscid Method for Preliminary Design of Transsonic Cascades. *27th AIAA Joint Propulsion Conference, AIAA-91-2364*.