Chapter 6
Numerical Optimization for Advanced Turbomachinery Design

René A. Van den Braembussche

Abstract The multilevel-multidisciplinary-multipoint optimization system developed at the von Kármán Institute and its applications to turbomachinery design is presented. To speed up the convergence to the optimum geometry, the method combines an Artificial Neural Network, a Design Of Experiment technique and a Genetic Algorithm. The different components are described, the main requirements are outlined and the basic method is illustrated by the design of an axial turbine blade.

A procedure for multipoint optimization, aiming for optimal performance at more than one operating point, is outlined and applied to the optimization of a low solidity diffuser.

The extension to a multidisciplinary optimization, by combining a Navier-Stokes solver with a Finite Element Analysis, allows an efficient search for a compromise between the sometimes conflicting demands of high efficiency and respect of mechanical constraints. It is shown that a significant reduction of the stresses is possible with only a small penalty on the performance and that this approach may lead to geometries that would normally be excluded when using less sophisticated methods.

6.1 Introduction

Computational Fluid Dynamics (CFD) has seen a very important development over the last 30 years. Navier-Stokes (NS) solvers have reached a high level of reliability at affordable cost. They are now routinely used to ana-

René A. Van den Braembussche
von Kármán Institute for Fluid Dynamics,
Turbomachinery and Propulsion Department,
Waterloose steenweg, 72, 1640 Sint-Genesius-Rode, Belgium
(e-mail: vdb@vki.ac.be)

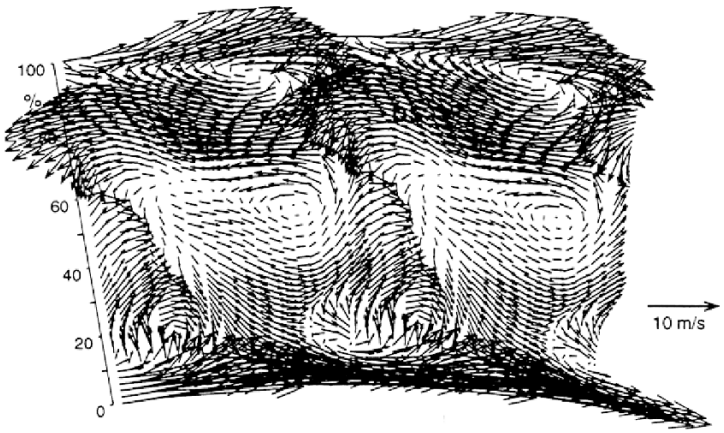

Fig. 6.1 2D view of the 3D flow at the exit of a turbine stage

lyze the fluid flows in the same way Finite Element Analysis (FEA) is used for stress predictions. They provide detailed information about the 3D flow around existing blade shapes and constitute an attractive alternative for detailed flow measurements. Complex flow phenomena can now be studied in what is called "Numerical Laboratories". Although this has resulted in a drastic decrease of the number of prototype testing, there are still two problems that prevent a more efficient use of CFD in the turbomachinery design process.

The first one results from the difficulty to analyze 3D flows on 2D screens or drawings. 2D vector plots are only a poor representation of the reality. They can be very misleading as they may suggest that the flow is penetrating the solid walls (Fig. 6.1). Synthetic environments, also called virtual reality, are very promising in this respect. These techniques are not only applicable to mere computer games but will become part of everyday reality for engineers in the next decade [16]. Designers will walk inside blade rows and diffusers to inspect the complex 3D flow structures by tracing the streamlines and to find out what geometrical changes may be needed to improve the performance.

The second problem relates to the abundance of information provided by the NS calculation. The output of an NS solver contains all the information needed to improve the performance. However, it does not provide any information on what modifications are needed to reach that goal. Three velocity components, the pressure and the temperature in typically 10,000 points (2D flows) or in more than 1,000,000 points (3D flows) are more than what the human brain is able to grasp and fully exploit in new designs. Most of the available information remains unused as the designer will often calculate a global parameter to find out if one geometry performs better than another.

The traditional design procedures in which standard 2D blade sections are selected and scaled up or down to adapt them to the different operating conditions are no longer acceptable. The designer is now faced with the development of new and better performing 2D and 3D blade shapes [10]. He needs new tools to use the available information in a more efficient way than with the traditional trial and error procedure in which the systematic testing of blade shapes has only been replaced by NS calculations. Those manual designs are very time consuming and the outcome depends on the expertise of the designer. This may become problematic since experienced designers are replaced by young engineers, who may have expertise in CFD but limited experience in turbomachinery design. Moreover, they can hardly be expert in all disciplines that interfere with a design (aerodynamics, mechanics, manufacturing etc.). Hence there is a need for automated and computerized design systems.

The main goal when designing turbines or compressors is to achieve light, compact and highly efficient systems while reducing the cost and the duration of the design cycle. Existing computerized design systems are often too expensive in terms of computational effort. Too many design processes have been concluded not because the target has been obtained, but because the deadline has come up. New design systems should therefore aim to be *fast* and affordable.

Turbomachines often operate outside the nominal or design conditions. Compressors for air-conditioning applications must be able to operate efficiently in all seasons, i.e., at different mass flows but constant pressure ratio. Low Solidity Diffusers (LSDs) are specially designed to increase the performance at a large variety of inlet flow conditions. Optimizing those geometries for one operating point is only part of the job. *Multipoint* design systems are needed to find a global optimum, i.e., maximizing the performance at all operating points to minimize the lifetime operating cost of the device.

Optimum performance is of no use if the mechanical integrity of the turbomachine cannot be guaranteed. This requires a stress and/or heat transfer analysis to verify that the stress constraints are not violated. Lower material and manufacturing costs are also important design criteria. Designing turbomachines is therefore a complex *multidisciplinary* exercise.

Inverse design methods define the geometry corresponding to a prescribed pressure or velocity distribution. However specifying the input of such a method, that satisfies mechanical and geometrical constraints and results in high performance in all operating points, is not an easy task. This is particularly difficult for 3D flows where secondary flow phenomena play a dominant role. A lot of insight is required to foresee the mechanical and geometrical consequences of a velocity variation. Adjustment of the target pressure distribution during the optimization process may be needed [7, 20].

Optimization systems searching for the geometry that best satisfies more global requirements in terms of performance, mechanical constraints or any other design criterion are a valuable alternative and have experienced a lot of

attention in recent years. In what follows one will describe the fast, multipoint and multidisciplinary optimization method, developed at the von Kármán Institute, and its application to different turbomachinery designs.

6.2 Optimization Methods

Optimization methods attempt to determine the design variables $X_i (i = 1, n)$ that minimize an objective function $OF(U(X_i), X_i)$ where $U(X_i)$ is the solution of the flow equations $R(U(X_i), X_i) = 0$ and subject to n_A performance constraints $A_j(U(X_i), X_i) \leq 0$ $(j = 1, n_A)$ and n_G geometrical constraints $G_k(X_i) \leq 0$ $(k = 1, n_G)$.

Specifying as Objective Function (OF) the difference between a prescribed and calculated pressure distribution results in an inverse design method. Aiming for the improvement of the overall performance leads to a global optimization technique.

Numerical optimization procedures consist of the following components, described in more detail in the following paragraphs:

- a choice of independent design parameters and the definition of the addressable part of the design space.
- definition of an OF quantifying the performance. Any standard analysis tool can be used to calculate the components such as lift, drag, efficiency, mass flow, manufacturing cost or a combination of all of them.
- a search mechanism to find the optimum combination of the design parameters, i.e., the one corresponding to the minimum of the OF.

6.2.1 Search Mechanisms

There are two main groups of search mechanisms:

- Analytical ones calculate the required geometry changes in a deterministic way from the output of the performance evaluation. A common one is the steepest descent method approaching the area of minimum OF by following the path with the largest negative gradient on the OF surface (Fig. 6.2). This approach requires the calculation of the direction of the largest gradient of the OF and the step length. A comprehensive overview of gradient based optimization techniques is given by Vanderplaats [17].
- Zero-order or stochastic procedures require only function evaluations. They make a random or systematic sweep of the design space or use evolutionary theories such as Genetic Algorithms (GA) or Simulated Annealing (SA).

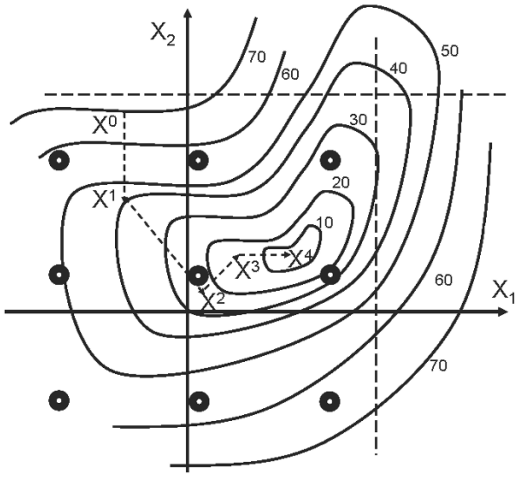

Fig. 6.2 Gradient method (- - -) and zero-order sweep of the design space (o)

To minimize the OF, most numerical algorithms require a large number of performance evaluations and are often very expensive in terms of computer resources. Zero order methods may require even more evaluations than gradient methods but the latter may get stuck in a local minimum. The method presented here uses a zero order search mechanism based on an evolutionary theory.

6.2.1.1 Zero-order Search

A systematic sweep of the design space, defining v values between the minimum and maximum limits of each of the n design parameters, requires v^n function evaluations. Figure 6.2 illustrates how such a sweep, calculating the OF for 3 different values of X_1 and X_2, provides a very good estimation of where the optimum is located with only 9 function evaluations. The risk of converging to a local minimum is low and such a systematic sweep is a valid alternative for analytical search methods for small values of n. However it requires more than 14×10^6 evaluations for $n = 15$.

Evolutionary strategies such as GA and SA can accelerate the procedure by replacing the systematic sweep with a more intelligent selection of new geometries using the information obtained during previous calculations in a stochastic way.

SA is derived from the annealing of solids [1]. At a given temperature, the state of the system varies randomly. It is immediately accepted if the new state has a lower energy level. If however, the variation results in a higher state, it is only accepted with a probability Pr that is a function of the temperature.

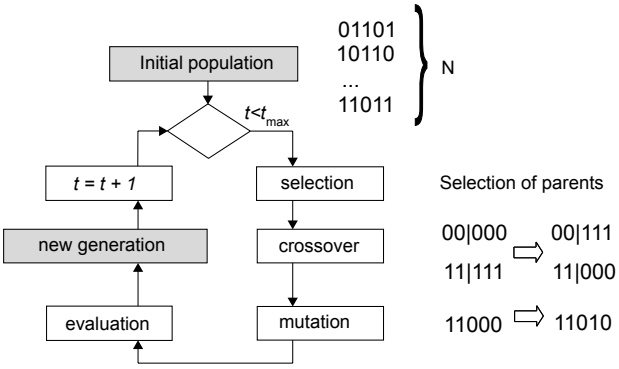

Fig. 6.3 GA operating principle

$$\Pr = \exp \frac{(E_{opt} - E_{act})}{T}$$

As the temperature decreases, the probability of accepting a higher state becomes lower. In a SA algorithm, the design parameters characterize the state of the system whereas the OF characterizes the energy level.

6.2.1.2 Genetic Algorithm (GA)

The method presented here uses a GA to find the optimum. This is a numerical technique that simulates Darwin's evolutionary theory stating that the fittest survives [4]. According to this theory, an individual (a geometry) with favorable genetic characteristics (design variables) is most likely to produce better offsprings. Selecting them as parent, increases the probability that the individuals of next generation will perform better than the previous one.

The operational principle of a standard GA is shown in Fig. 6.3. Pairs of individuals (parents) are selected from an initially random population of N geometries, each represented by a binary coded string of length l. Genetic material is subsequently exchanged between them (crossover) and altered within the offspring (mutation). It is followed by an evaluation of each new individual. This process is used to create the N individuals of the next generation. Such a procedure is repeated for t generations and it is assumed that the best individual of the last generation is the optimum.

Quality of the GA optimizer is measured by:

- the required computational effort, i.e., the number of performance evaluations that are needed to find that optimum (GA efficiency).
- the value of the optimum (GA effectiveness).

The GA software used in the VKI design system is the one developed by David L. Carroll [5]. The optimum parameter setting has been defined

by means of a systematic study on two typical design cases: one geometry defined by 7 parameters and one defined by 27 parameters [8]. Conclusions are based on the solution quality q, i.e., the degree to where the GA optimum approaches the real one within a given effort (5,000 function evaluations). It is defined by:

$$q = \frac{OF_{\mathrm{AV}} - OF_{\mathrm{GA}}}{OF_{\mathrm{AV}} - OF_{\mathrm{min}}} \cdot 100\%$$

where OF_{AV} is the average of the OF over the complete design space.

OF_{min} is the global minimum value of the OF obtained from a systematic (numerically very expensive) scanning of the whole design space.

OF_{GA} is the minimum value of the OF obtained from the GA optimization.

A q value of 100% indicates that the global minimum has been found. The function evaluations for the numerical experiments are made by means of an approximation of the NS solver based on Artificial Neural Networks (ANN, explained in Sect. 6.3.1).

Optimum substring length

In a standard binary-coded GA, the n real-valued design parameters X_i, defining a geometry, are jointly represented by one binary string:

$$\underbrace{1101...0}_{X_1}\ \underbrace{1001...1}_{X_2}\ \underbrace{0011...0}_{X_3}\ \ldots\ \ldots\ \ldots\ \underbrace{0101...1}_{X_n}$$

The substring length, denoted by l (number of bits per variable), determines the total number of values (2^l) that each design parameter can take.

The minimum substring length l_i for the i^{th} design variable depends on the lower and upper bound respectively X_i^{min} and X_i^{max}, as well as on the desired resolution (ε_i) for this variable:

$$l_i = \log_2 \frac{X_i^{\mathrm{min}} - X_i^{\mathrm{max}}}{\varepsilon_i}$$

Very short substrings ($l < 3$) result in a too low resolution and the GA may not be able to accurately locate the minimum. Longer substrings ($3 < l < 10$) enable a higher resolution but cause a larger search space, making it difficult to find the complete optimal binary string. Systematic testing shows that $l = 8$ is the optimum substring length, independent of the number of unknown parameters.

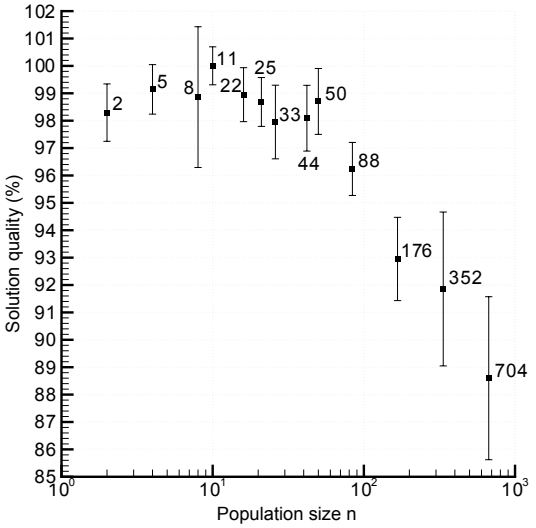

Fig. 6.4 Dependence of GA solution quality on population size for the 27 parameter test case

Selection scheme

Different selection schemes have been proposed. One is the roulette: a system in which the chance that an individual is selected increases proportional with $1/OF$. This scheme favors the best individuals as parents. It is elitist and has larger chances to get stuck in a local optimum.

In the tournament selection, "s" individuals are chosen randomly from the population and the individual with the highest fitness (lowest OF) is selected as parent. The same process is repeated to find the second parent. The parameter s is called the tournament size and can take values between 1 and N (population size). Larger values of s give more chances to the best samples to be selected and to create offsprings. It favors a rapid, although perhaps premature, convergence to a local optimum. Very small values of s result in a more random selection of parents. Tests have shown that a standard value of $s = 2$ gives the best results.

Population size

Fixing the total number of function evaluations at 5,000, the number of generations t is a consequence of the population size N ($N \times t = 5,000$). Figure 6.4 shows the solution quality at the end of the GA run in function of the

population size. The solution quality is maximum for $N = 11$ to 20. Small populations ($N < 10$) converge prematurely to suboptimal solutions, due to a lack of high performing samples in the initial population. Larger populations ($N > 25$) have a sluggish convergence to the optimal geometry because less generations are allowed.

Crossover probability

In a single-point crossover operator, both parent strings are cut at a random place and the right-side portions of both strings are swapped with the probability p_c (Fig. 6.3). In case of uniform crossover, the value of p_c defines the probability that crossover is applied per bit of the complete parent string. High values of p_c increase mixing of string parts but at the same time, increase the disruption of good string parts. Low values limit the search to combinations of samples in the existing design space. Experiments confirm that a single point crossover with probability $p_c = 0.5$ is optimal.

Mutation probability

The mutation operator creates new individuals by changing in the offspring strings a "1" to a "0" or vice versa. The mutation probability p_m is defined as the probability that a bit of a string is flipped. Systematic numerical experiments confirm that the optimum setting for the mutation probability is $p_m = 1/(N \times l)$ for all optimizations. This corresponds to changing on average one bit at every generation.

Figure 6.5 shows how an optimization of the GA parameter settings can lead to an improved and smoother GA convergence.

Creep mutation and Gray coding

Changing one digit in a binary code may result in a large variation of the corresponding digital value: i.e., the small difference between 0111 and 1111 corresponds to a doubling of the digital value. Small variations of the digital value may require a large number of binary digits to be changed: i.e., 0111 and 1000 are adjacent digital values but all four digits are different. This discontinuous relation between the digital value and binary string may confuse the GA optimizer. Creep mutation tries to avoid this by limiting the change of the real value to a binary step length [5]. Gray coding uses an algorithm in which similar binary strings correspond to adjacent digital values. In contrast to what could be expected, no acceleration of convergence was obtained with either one of these approaches.

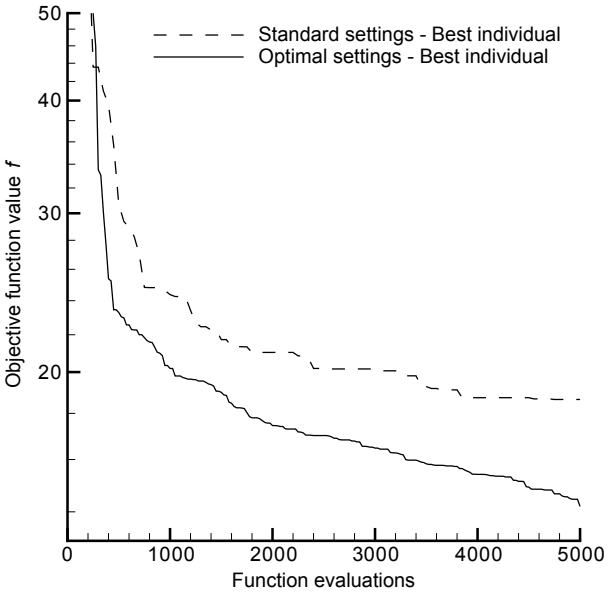

Fig. 6.5 GA convergence for a 27 parameter test case (standard versus optimal parameter setting)

6.2.2 Objective Function

The OF measures how far a geometry satisfies the aero-requirements and if the performance goals that have been set forward are reached.

High aero-performance is not the only objective of an optimization. A good design must also provide good off-design performance (multipoint optimization) and respect the mechanical and manufacturing constraints (multidisciplinary optimization). Some constraints must be satisfied without any compromise (i.e., maximum stress level). They result in an inequality and a more detailed discussion is given in Sect. 6.6.2. Others tolerate some margin (i.e., cost or weight) that can be corrected for after the design is finished (i.e., by adjusting the blade length to achieve the required mass flow). A possible alternative for these inequalities is to add penalty terms to the OF that increase when the constraints are violated [13].

The following lists some contributions to the global OF that are common for the different applications. Each term is multiplied by a weight factor to adjust its relative importance in the optimization procedure.

$$OF_{2D} = w_\eta \cdot P_{\text{perf}} + w_a \cdot P_{\text{aeroBC}} + w_m \cdot P_{\text{mech}} + w_M \cdot P_{\text{Mach}}$$
$$+ w_d \cdot P_{\text{discharge}} + w_G \cdot P_{\text{Geom}} + w_S \cdot P_{\text{Side}}$$

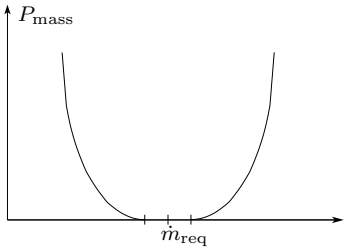

Fig. 6.6 Variation of penalty function for incorrect mass flow

P_{perf} is the penalty for non-optimum performance and increases with decreasing efficiency (η)

$$P_{\text{perf}} = \max\left[\left|\eta_{\text{req}} - \eta\right|, 0.0\right]$$

Minimizing this term corresponds to maximizing the efficiency. The required efficiency η_{req} is set to an unachievable value (for instance 1.0) so that this penalty never goes to zero. The argument is that, after all other requirements are met, the efficiency should still be maximized.

P_{AeroBC} is the penalty for violating the aerodynamic boundary conditions. The purpose of this penalty is to enforce the boundary conditions and requirements at the inlet and outlet of the computational domain that cannot be imposed such as: the outlet flow angle (β_2), the mass flow or pressure ratio, etc. The penalties for not respecting the boundary conditions start increasing when the actual values differ from the target values by more than a predefined tolerance. Following penalty for incorrect mass flow increases when the mass flow differs more than 2% from the required value (Fig. 6.6):

$$P_{\text{mass}} = \left(\max\left[\frac{\left|\dot{m}_{\text{act}} - \dot{m}_{\text{req}}\right|}{\dot{m}_{\text{req}}} - 0.02\ ,\ 0.\right]\right)^2$$

P_{mech} is the penalty for not respecting the mechanical constraints. The latter must be satisfied without compromise because exceeding the maximum stress level cannot be tolerated as it may destroy the device. A rigorous respect of the minimum stress limits requires a Finite Element stress Analysis (FEA) and will be discussed in detail in Sect. 6.6. The computational effort can be drastically reduced if one can replace the mechanical constraints by simpler geometrical ones that are much easier to verify.

The large stresses in the blade root section of radial impellers are a complex function of the blade curvature and lean. The subsequent deformations can reduce the tip clearance to zero which may lead to the destruction of the optimized geometry. Traditional design systems limit the lean (Fig. 6.7) to a maximum value based on experience and simple stress models.

Prescribing the radial variation of the cross section area of a fan blade or low-pressure (LP) turbine blade is a common way to control the centrifu-

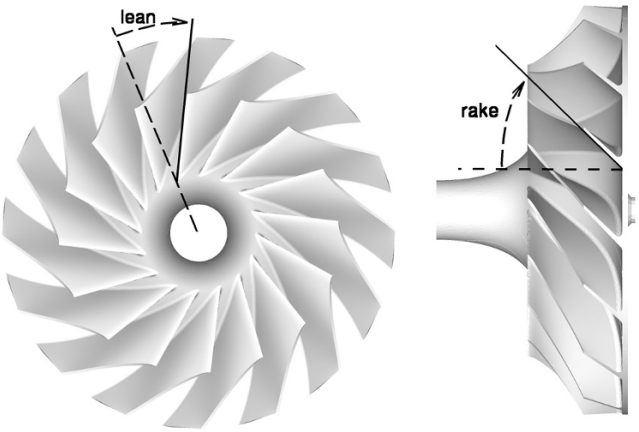

Fig. 6.7 Definition of (**a**) lean and (**b**) rake in radial impellers

gal stresses. The parameters of primary importance in controlling the blade bending due to the static and dynamic load in an axial compressor or turbine blade are: minimum and maximum moment of inertia ($I_{\min}$) and ($I_{\max}$) of the cross sections and the direction κ of its maximum value.

P_{Mach} is the penalty for a non-optimum Mach number distribution. Analyzing the Mach number distribution may help to rank blades that have nearly the same loss coefficient. NS solvers are not always reliable in terms of transition modeling and erroneous penalty function may occur when the transition point is incorrectly located. Transition criteria based on the Mach number distribution may help to relieve this uncertainty.

One is also not interested in designing blades that have very good performance at design point but for which the flow is likely to separate (with large increase in losses) at slightly off-design conditions. A rigorous way of verifying the operating range is discussed in Sect. 6.5. A simpler approach accounts for the changes in the Mach number distribution that can be expected at off-design. It increases the chances for good performance of the blade over a wide range of operating conditions without the cost of extra NS computations.

The Mach number penalties that have been formulated for turbine blade optimizations are presented in Sect. 6.4. They tend to achieve a continuous flow acceleration with minimum deceleration. Mach number penalties for radial compressor impellers are presented in Sect. 6.6.

$P_{\text{discharge}}$ can be used to penalize the spanwise distortion of the flow at the exit. It results from the idea that a more uniform exit flow has a favorable effect on the downstream diffuser or blade row and hence, on the stage effi-

ciency. The distortion penalty quantifies the difference between the average flow angle at 20% (hub) and 80% (shroud) span with the one at midspan

$$P_{\text{dist}} = \left| 1 - \frac{2.\alpha_{\text{midspan}}}{\alpha_{\text{hub}} + \alpha_{\text{shroud}}} \right|$$

The penalty for flow skewness is proportional to the difference between the flow angle, or any other flow quantity, at 20% and 80% span, non dimensionalized by the average value

$$P_{\text{skew}} = \frac{2.(\alpha_{\text{hub}} - \alpha_{\text{shroud}})}{\alpha_{\text{hub}} + \alpha_{\text{shroud}}}$$

P_{Geom} is the penalty for violating the geometrical constraints. They either are related to the mechanical integrity or assure dimensional agreement with other components.

P_{Side} is the penalty for violating the side constraints. Depending on the application, the weight, manufacturing and maintenance cost may be important issues and some geometries can be favored by formulating an appropriate penalty.

The penalties and weight factors are specific for each design. An appropriate choice of the weights can be based on overall design criteria such as energy savings, total lifetime cost, etc. Plotting the different contributions to the OF as a Pareto front in the fitness space (Fig. 6.8) facilitates a trade-off between different counteracting goals. This is particularly useful for problems with only 2 or 3 groups of OF where the Pareto front can easily be visualized. However it is much more cumbersome in higher order problems or when the Pareto front is not convex.

6.2.3 Parameterization

The number of coordinates needed for the complete definition of an arbitrary geometry is infinite and a direct calculation of all of them by a numerical optimization procedure is not feasible. A reduction of the number of variables by an adequate parameterization of the geometry is required. It is important that the parameterization does not exclude any physically-acceptable geometry. A blade shape that can not be generated can of course not be found by the optimizer, even if it would be the optimum one. The parameterization should be sufficiently simple to limit the number of variables that need to be defined. The use of Bézier curves or B-splines to describe the geometry is recommended. This assures smoothness of the surface and facilitates the transfer of the data to Computer-Aided Design and Manufacturing (CAD-CAM) systems. Different parameterizations will be explained in Sects. 6.4 to 6.6.

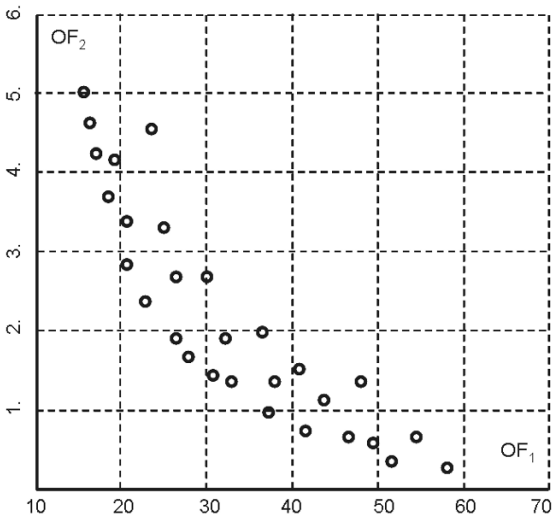

Fig. 6.8 Convex Pareto front

6.3 Two-level Optimization

The system presented here (Fig. 6.9) is developed at the von Kármán Institute [13] and makes use of a GA to minimize the OF. A GA requires a large number of function evaluations. Using an expensive NS solver for all function evaluations is in most cases, prohibitive in terms of computer effort.

One way to reduce the computational effort is by working on different levels of sophistication. Fast but approximate prediction methods can be used to find a near optimum geometry, which is then further verified and refined by a more accurate but also more expensive analysis. Approximations of the NS solver and FEA, called meta-functions, are used for the first level optimization. The more accurate but expensive NS and FEA are used only to verify the accuracy of the meta-function predictions.

Meta-functions not only need to be fast but must also be accurate. The GA can only converge to the real optimum if it receives accurate information about the impact of a geometry change on the performance. Different type of meta-functions have been proposed. The main problem is the risk that the discrepancies between the predictions by the meta-function and the NS results drive the optimizer to a false optimum. Euler and NS solutions on coarse grids are sometimes proposed as meta-functions. They are fast but inaccurate and using them for performance predictions may drive the GA to a non-optimum combination of design parameters. Any further control by an accurate NS solver will reveal the inherent inaccuracy of the fast calculation methods but there is no mechanism to correct for it.

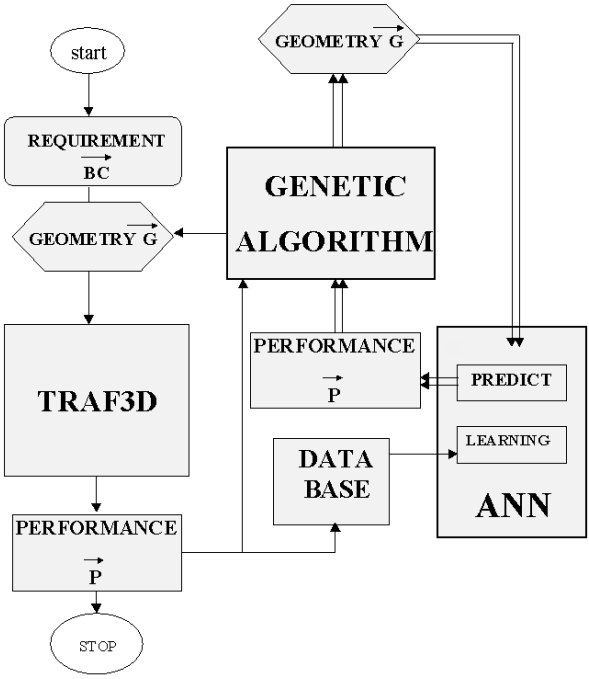

Fig. 6.9 Flowchart of optimization system

The meta-function used in the present method is an Artificial Neural Network (ANN). This interpolator uses the information contained in the database to correlate the performance to the geometry, similar to what is done by an NS solver. However, an ANN is a very fast predictor and allows the evaluation of the numerous geometries generated by the GA with much less effort than an NS solver. Unfortunately, a verification by means of a more accurate but time consuming NS solver indicates that such a fast prediction is not always very accurate. The results (geometry and performance) of this verification are added to the database and a new optimization cycle is started. It is expected that the new learning on an extended database will result in a more accurate ANN. This procedure is repeated until the ANN predictions are in agreement with the NS calculations, i.e., once the GA optimization has been made with an accurate performance predictor. In this way, there will be no discrepancy between the optimum found by a GA, driven by the meta-function or by the results of NS analyses. However, the number of time-consuming NS analyses is much smaller than what would have been required by a GA and NS combination.

The TRAF3D NS solver [3] is used to predict the aerodynamic performance. Similar grids with the same number of cells are used for all computations to guarantee a comparable accuracy for all the predictions.

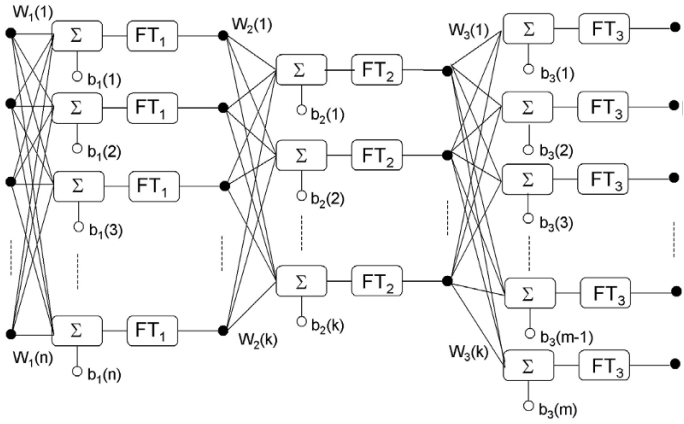

Fig. 6.10 Architecture of a three-layer ANN

6.3.1 Artificial Neural Networks

Artificial Neural Networks (ANN) are used to predict the performance of a new geometry by means of the information contained in the database. This requires the learning of the relation between the n input data (geometry parameters) of a process (NS solver) and the m outputs of the process (mass flow, efficiency, local pressures and temperatures, velocities, etc.). The use of an exact ANN predictor could reduce the effort to one design cycle by the GA. Hence, improving the accuracy of the ANN will shorten the design process.

An ANN (Fig. 6.10) is composed of n, k, m elementary processing units called neurons or nodes. These nodes are organized in layers and joined with connections (synapses) of different intensity, called the connection weight (W) to form a parallel architecture. Each node performs two operations: the first one is the summation of all the incoming signals and a bias b_i, the second one is the transformation of the signal by using a transfer function (FT). For the first layer this corresponds to:

$$a_1(i) = FT_1\left(\sum_{j=1}^{n} W_1(i,j).n(j) + b_1(i)\right)$$

A network is generally composed of several layers: an input layer, zero, one or more hidden layers and one output layer. The coefficients are defined by a learning procedure relating the output to the input data.

The main purpose of ANN is not to reproduce the existing database with maximum accuracy but to predict the performance of new geometries it has not seen before, i.e., to generalize. A well-trained ANN may show a less

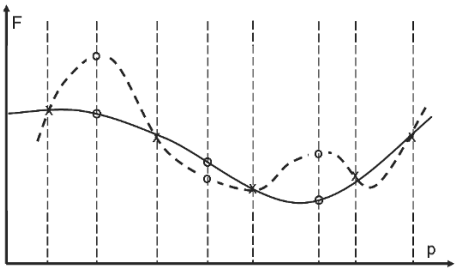

Fig. 6.11 Comparison between well trained —— and over-trained - - - - ANN (x database samples, o new geometries)

accurate reproduction of the database samples but predicts more realistic values for new geometries. This is illustrated in Fig. 6.11 comparing an over-trained ANN function with a well trained one. The first one reproduces the database samples exactly but the large oscillations of the function between the database values result in unrealistic predictions of the function at the intermediate locations.

Three conditions are necessary, although not sufficient, for a good generalization.

The first one is that the inputs to the ANN contain sufficient information pertaining to the target, so that one can define a mathematical function relating correctly outputs to inputs with the desired degree of accuracy. Hence the designer should select design parameters that are relevant, i.e., that have an influence on performance.

The second one is that the function to be learned (relating inputs to the outputs) is smooth. Small changes in the input should produce a small change in the outputs. Most physical problems are well defined in this respect. However, the appearance of large separation zones or large changes in shock position and strength for small geometrical changes may result in discontinuous changes of the output and complicate the problem.

The third one requires that the training set is sufficiently large and contains representative samples of all cases that one wants to generalize (the "population" in statistical terminology). It is difficult to define the minimum size of the training set that is required. Experience has shown that there is no advantage in creating a very large database because the design system itself will generate new geometries until the ANN achieves the required accuracy.

The standard back-propagation technique is the most widely used algorithm for ANN training. The available samples are normally subdivided into "training", "test" and "validation" sets. Each of them has its own purpose.

- the training set contains the samples used for the training; that is to define the parameters (weights and bias).
- the test set contains the samples used to assess the generalization capacity of a fully-specified ANN with given weights and architecture.

- the validation set, if used, contains the samples used to tune the ANN architecture (not the weights), for example to choose the number of hidden unit layers and nodes.

6.3.2 Database

The main purpose of the database is to provide input to the ANN, i.e., information about the relation between the geometry and performance. The more general and complete this information, the more accurate the ANN can be and the closer the results of the GA optimization will be to the real optimum. Hence, a good database may considerably speed up the convergence to the optimum.

Making a database is an expensive operation because it requires a large number of 3D NS calculations. One therefore aims for the smallest possible database containing the maximum amount of information about the available design space. This means a maximum of relevant information with minimum redundancy so that the impact of every design parameter is included but only once.

Any information missing in the database may result in an erroneous ANN that could drive the GA towards a non optimum geometry. This will slow down the convergence but will not lead to an incorrect final result since the NS analysis of that geometry will provide the missing information when added to the database.

A more risky situation is the one where an incomplete database results in an erroneous extrapolation by the ANN, predicting a low performance (large OF) in that part of the design space where in reality the OF is low. As a consequence, the corresponding geometry will never be selected by the GA and no information will be generated to correct this error. This is an additional argument to assure that the initial database covers the whole design space.

Design Of Experiment (DOE) refers to the process of planning an experiment so that the appropriate data, when analyzed by statistical methods, result in valid and objective conclusions. It is used in the optimization process to select the most significant geometries to be stored in the database. The theory of DOE is explained in many excellent textbooks [11]. The advantages of using DOE, to construct the database for the optimization program, have been evaluated in detail by Kostrewa et al. [9].

Factorial designs make a systematic scan of the design space. The common one is where each of the n design parameters has only two values corresponding to the "high" or "low" level of the design variable. A complete coverage of such a design requires 2^n observations and is called full factorial DOE. Fractional factorial DOE requires 2^{n-p} analyses where p defines the fraction of lower order combinations that are not analyzed. A typical initial database

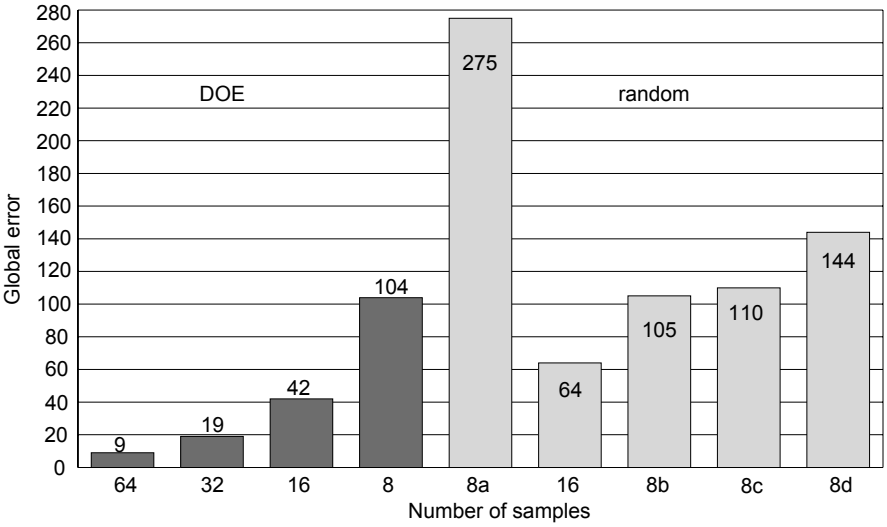

Fig. 6.12 Global error of ANN as a function of Database samples

contains a total of $2^6 = 64$ samples. The following evaluates the loss of information for a test function with 6 variables.

$$R = 1 - 0.001(A-D)^3 + 0.002(C+E)(F-B) - 0.06(A-F)^2(F+C)(E+A)$$

The results of the ANN predictions, trained on the databases defined by DOE, are compared to those of databases in which the variables are randomly generated between the prescribed boundaries.

The full factorial design requires $2^6 = 64$ runs to estimate all possible parameter combinations. The loss of information with fractional designs is measured by:

$$\text{Global error} = \sum_{i=1}^{2^6} \frac{\text{exact value} - \text{predicted value}}{\text{exact value}} \cdot 100$$

Comparing the error obtained by means of the DOE technique and by means of randomly selected samples (Fig. 6.12), clearly shows that for the same number of NS results in the database, the DOE based predictions are consistently more accurate than the ones based on the randomly generated samples.

Randomly generated databases are all different and so is the accuracy of the ANN predictions. The four randomly generated cases with 8 samples in the database, show an error that varies between 105 (8b), equal to the one obtained with the DOE defined database, up to an error that is almost 3 times larger (8a).

Only 2-level designs (every variable can take two values) and one central-point run (every value is at the center of the allowed range) are considered for the database used in following examples. The high and low value of each design variable are located at 75% and 25% of the parameter range, respectively. The central point is the mid value (50%). The range is defined by the designer based on his experience about feasible geometries and mechanical constraints.

6.4 Single Point Optimization of Turbine Blade

The optimizing design procedure has been successfully tested on a large number of designs and will be illustrated here by the design of a 2D highly loaded axial turbine blade section.

6.4.1 2D Blade Geometry Definition

The geometry definition, described here, makes use of Bézier curves to define the camber line and the blade suction and pressure side relative to the camber line.

The camber line is defined by 3 control points (Fig. 6.13.a). The first one coincides with the leading edge. The second one is at the trailing edge. Its position relative to the leading edge is defined by the stagger angle (γ) and axial chord length Lref. A third control point is located at the intersection between the tangent to the camber at leading and trailing edge. It is defined by the angles $\beta_{1\mathrm{m}}$ and $\beta_{2\mathrm{m}}$ with the axial direction.

The suction side is defined by a Bézier curve starting at the leading edge and ending at the tangent to a circle of radius Rte, defining the trailing edge thickness. The camber line is divided in a number of intervals which can be of equal or variable length using a stretching factor (Fig. 6.13.b). The lengths Tss(1), Tss(2), Tss(3), Tss(4), Tss(5), Tss(6) and Tss(7) measured in the direction perpendicular to the camber line determine the position of the 7 Bézier control points of the suction side.

The first Bézier control point coincides with the leading edge (start of the camber line) (Fig. 6.14.a) and the third point is the one defined by the parameter Tss(1). The second Bézier control point is defined by imposing that the suction side starts perpendicularly to the camber line at the leading edge and that the suction side curvature radius at the leading edge equals Rle. This condition defines the length Tss(0).

δ_{te} is the wedge angle between suction and pressure side at the trailing edge (Fig. 6.14.b). A Bézier control point is defined as the intersection of the

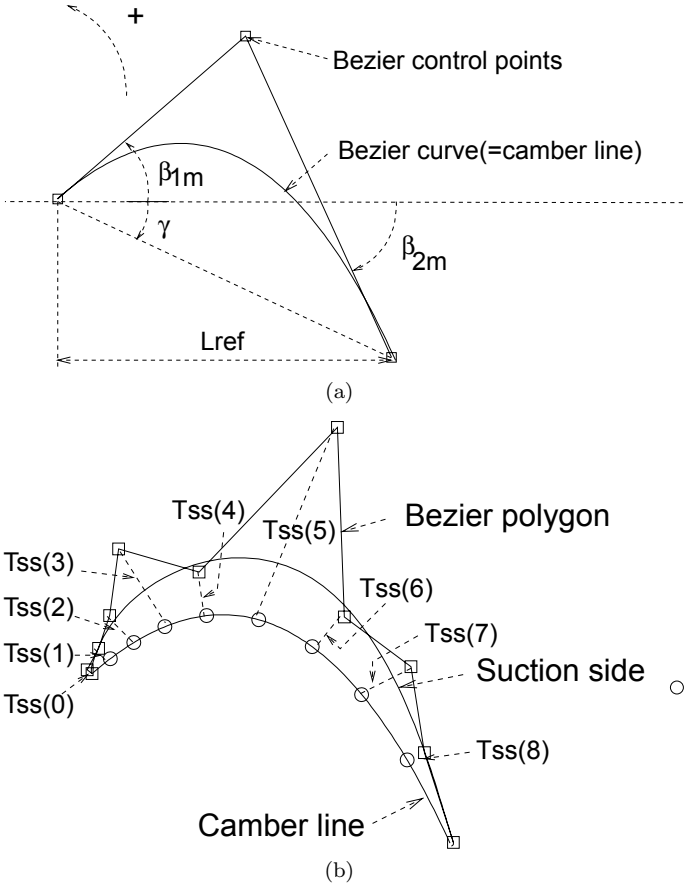

Fig. 6.13 Geometry model: (**a**) definition of the camber line and (**b**) the suction side

tangent at the trailing edge circle and the line perpendicular to the camber line at point 8.

The pressure side is defined in a way similar to the suction side. Imposing the same Rle as on the suction side assures a continuous curvature at the leading edge. Figure 6.15 demonstrates the capabilities of the method to represent the large variety of blades encountered in axial turbines.

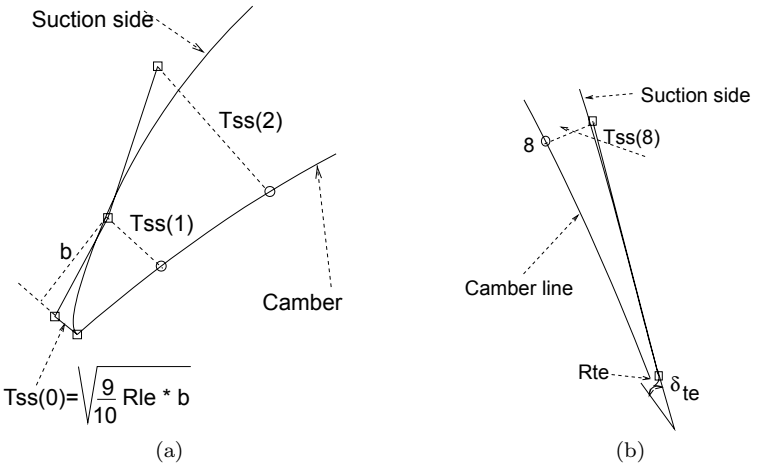

Fig. 6.14 Geometry model: (**a**) detailed view of the leading edge (**b**) and trailing edge

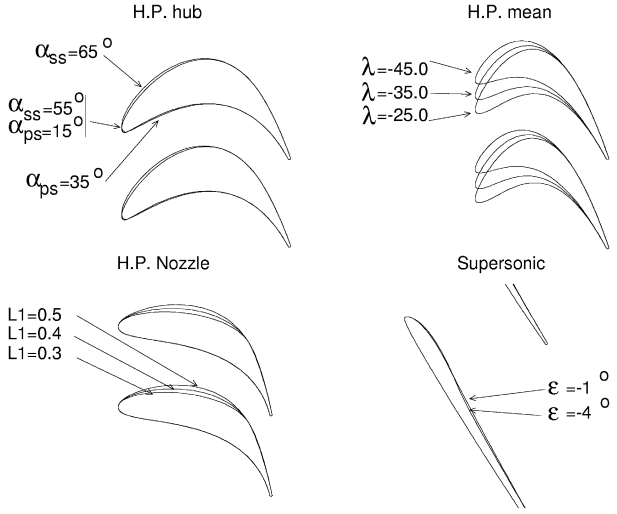

Fig. 6.15 Typical turbine blade sections generated by a 17-parameter model

6.4.2 Penalty for Non-optimum Mach Number Distribution $P_{\mathbf{Mach}}$

The characteristics of the optimum Mach number M distribution for 2D turbine blades are well known and allow the definition of penalties for a non optimum Mach number distribution. These penalties can also be derived from experimental correlations.

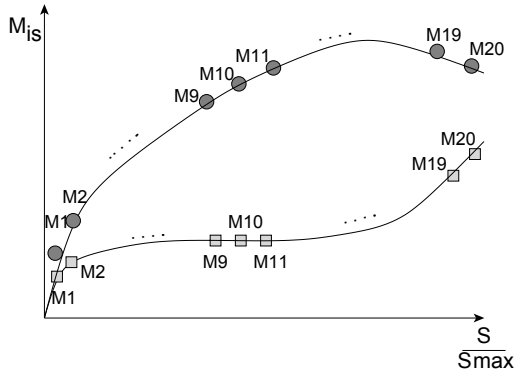

Fig. 6.16 Parametric representation of the Mach number distribution

The OF used in a turbine blade optimization [15] increases when there is a high probability of early transition, laminar or turbulent separation or poor off-design performances. It is a sum of penalties based on the Mach number in 20 points on each side of the blade (Fig. 6.16). The values may be defined from an NS calculation or predicted by the ANN.

- Penalty on the local slope of the Mach number distribution: a minimum positive slope of the Mach number distribution is required on the front part of the suction side in order to avoid deterioration of the performance at off-design incidence angle.
- Penalty on the second derivative of the Mach number distribution: the optimization process may result in a wavy Mach number distribution because of local changes in curvature radius of the blade surface. This may have a small impact on blade losses if the boundary layer is already turbulent but can deteriorate the off-design performances of the blade. One therefore penalizes the Mach number distribution for which the second derivative changes sign, i.e., with an inflection point in the suction side Mach number distribution.
- Penalty on the deceleration: it is important to limit the deceleration on the front part of the blade pressure side in order to avoid separation at negative incidence angles. This penalty is proportional to the difference between the first maximum Mach number found on the pressure side (starting from the stagnation point) and the minimum Mach number along the pressure side. It is also well known that the deceleration on the second half of the suction side has an important influence on the losses and in case of low Reynolds number may lead to flow separation.

Table 6.1 Imposed parameters

β_1^{flow} (deg.)	18.0
M_2^{is}	0.90
Re	5.8×10^5
$\gamma = Cp/Cv$	1.4
Tu (%)	4
Lref (m)	0.0520
Pitch/Lref	1.0393
TE thick. (m)	1.2×10^{-3}

Table 6.2 Mechanical and aerodynamical requirements

	Imposed		After
	Min.	Max.	18 modif.
Surface (m^2)	5.20×10^{-4}	6.80×10^{-4}	5.36×10^{-4}
$I_{\min}(m^4)$	7.50×10^{-9}	1.20×10^{-8}	7.45×10^{-9}
$I_{\max}(m^4)$	1.25×10^{-7}	2.20×10^{-7}	1.28×10^{-7}
$\kappa_{I_{\max}}$ (deg.)	-50.0	-30.0	-37.5
β_2^{flow} (deg.)	-57.8	-57.8	-57.62
Loss coef. (%)	0.0	0.0	1.90

6.4.3 Design of a Transonic Turbine Blade

The method is illustrated by the redesign of a transonic turbine blade with an outlet isentropic Mach number of 0.9. The design requirements imposed for this example are displayed in Tables 6.1 and 6.2.

The best blade of the initial database is used as starting geometry. The Mach number distribution has a shock at mid-chord (Fig. 6.17.a). The small constant velocity region on the suction side close to the leading edge and the low velocity on the pressure side close to the leading edge indicate that the incidence angle on the initial blade is too large. After the first modification (one GA and NS verification), this incidence angle has been partially reduced by decreasing the stagger angle (Fig. 6.17.b). The shock intensity is also smaller but the suction side Mach number distribution is still wavy. The shock completely disappeared after 13 design iterations. The stagger angle has decreased in order to adapt the blade geometry to the prescribed inlet flow angle. The smooth shock-free Mach number distribution is reflected in the low loss coefficient.

Figure 6.18 compares the value of the OF predicted by the ANN with the one predicted by the NS solver during the design process. The value of the OF computed by the approximate model decreases until iteration 13 after which only very small improvements are found. The value predicted by the NS solver shows large discrepancies between both predictions at iteration 2, 5 and

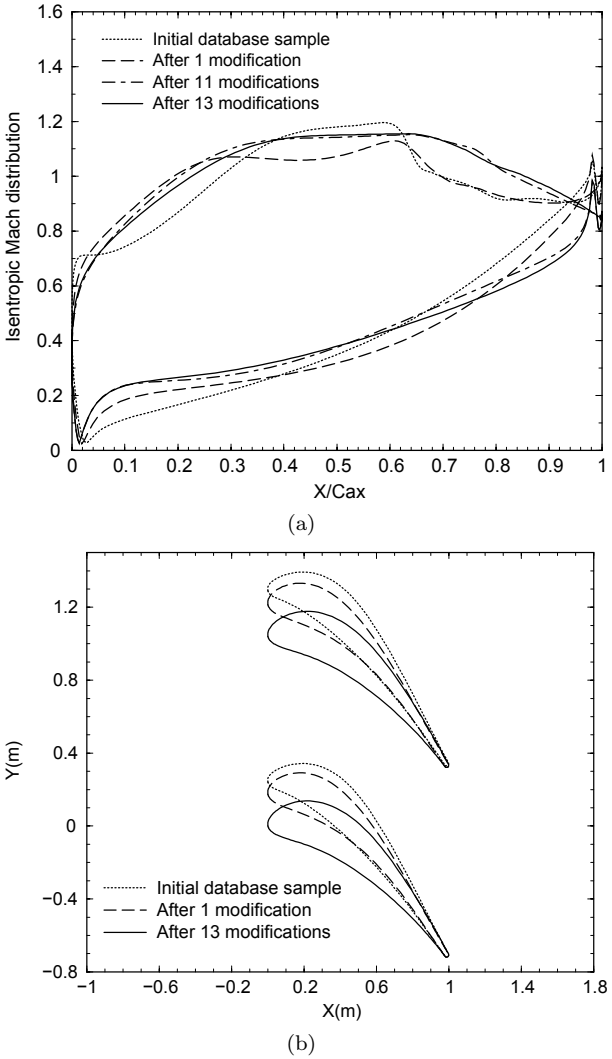

Fig. 6.17 Variation of (**a**) Mach number and (**b**) blade geometry during convergence

9. It indicates that during the first design iterations, the ANN predictions are not very accurate because the database does not sufficiently cover the relevant design space. However this shortcoming is remediated by adding new geometries to the database. Since these blades are close to the desired operating point they provide very valuable information and the ANN becomes more and more accurate. Starting from iteration 13, the ANN predictions are very reliable. This illustrates the self-learning capacity of the proposed

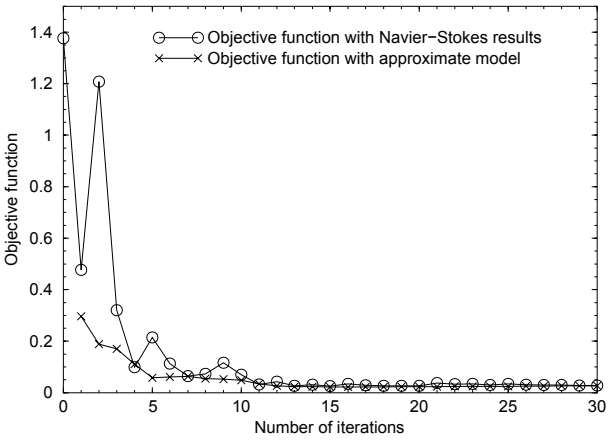

Fig. 6.18 Convergence history

Table 6.3 Inlet conditions at the 3 operating points

	Low $\dot{m}$	Medium $\dot{m}$	Large $\dot{m}$
α(deg)	68.	60.	54.
P_2°(Pa)	218395.	213447.	187666.
P_3/P_2°	0.9157	0.9276	0.9430
T_2° (K)	365.4	360.3	350.4

procedure. The whole procedure could have been stopped after 15 iterations but has been continued to verify good convergence.

6.5 Multipoint Optimization of a Low Solidity Diffuser

The radial compressor vaned diffusers provide a higher pressure recovery and efficiency than vaneless ones but the operating range is limited by stall, at positive incidence, and diffuser throat choking, at negative incidence. Low Solidity Diffusers (LSD) are characterized by a small number of short vanes and do not show a well defined throat section. They intend to stabilize the flow at low mass flow (avoiding diffuser stall) without limiting the maximum mass flow by choking. The solidity (chord/pitch) is typically on the order of 1 or less (Fig. 6.19). A multipoint optimization is mandatory for the LSD design because a wide operating range is the major purpose of these devices.

The optimization of the LSD [12] is done for the 3 operating points listed in Table 6.3. Inlet conditions are different for each operating point and defined by the impeller exit flow at the corresponding mass flows.

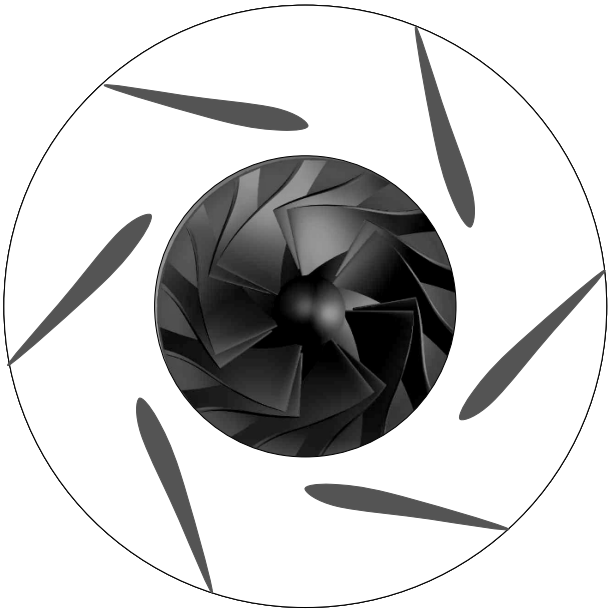

Fig. 6.19 Low Solidity Diffusor

The blade geometry is defined by a NACA 65 thickness distribution superposed on a camber line defined by a 4-parameter Bézier curve. A 5^{th} design parameter is the scale factor for the thickness distribution (between 0.7 and 1.3). The 6^{th} parameter is the number of blades (between 6 and 12).

The performance criteria are the static pressure rise and total pressure loss, non-dimensionalized by the diffuser inlet dynamic pressure

$$Cp = \frac{\bar{P}_3 - \bar{P}_2}{\bar{P}_2^\circ - \bar{P}_2} \qquad , \qquad \omega = \frac{\bar{P}_2^\circ - \bar{P}_3^\circ}{\bar{P}_2^\circ - \bar{P}_2}$$

Making the database is quite costly because it requires an NS analysis of each geometry at three operating points. The initial database is therefore limited to only 10 geometries requiring 30 NS calculations on a grid with 400,000 cells.

One wants to maximize what the diffuser is supposed to do: i.e., to increase the static pressure with minimum total pressure losses. The latter is the difference between the real pressure rise and the isentropic one.

$$Cp + \omega = Cp_{\text{isentropic}}$$

The maximum value $Cp_{\text{isentropic}}$ depends on the inlet conditions and diffuser geometry.

The best results have been obtained with the following OF

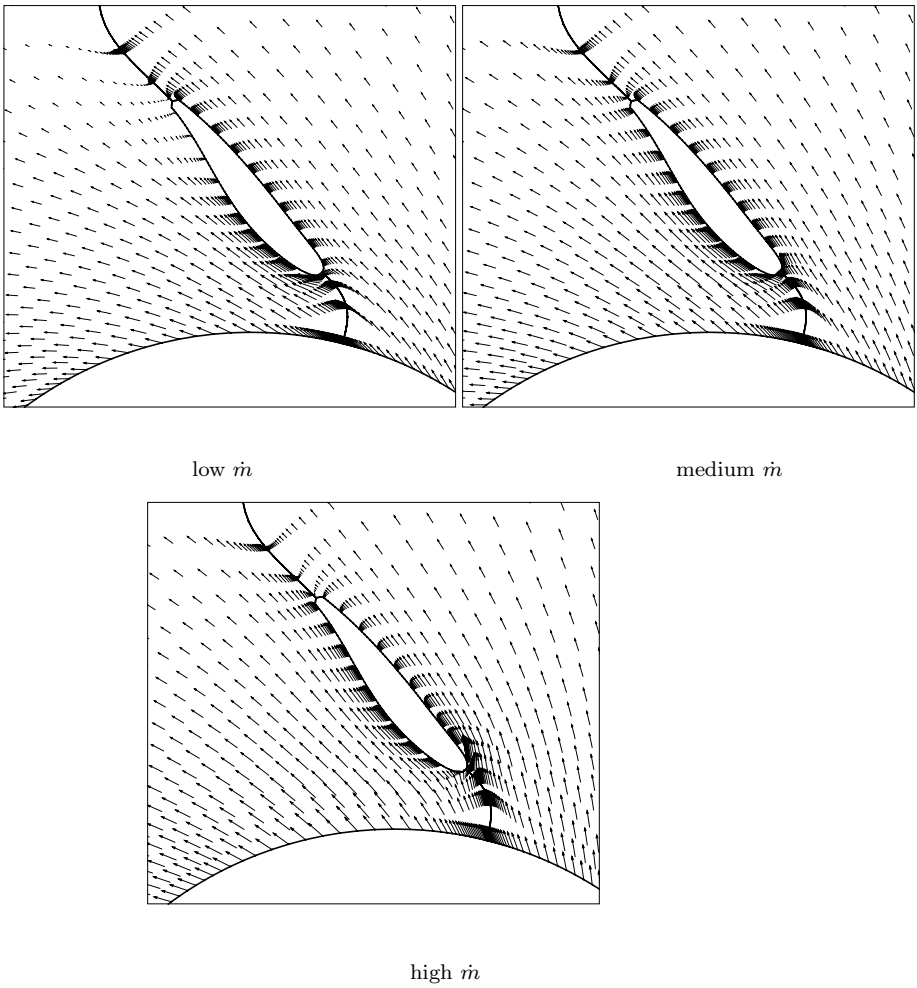

Fig. 6.20 Midspan velocity vector of optimized diffuser

$$OF = (1 - (w_{\text{low}} \cdot Cp_{\text{low}} + w_{\text{med}} \cdot Cp_{\text{med}} + w_{\text{high}} \cdot Cp_{\text{high}}))$$

$$+w_{\text{low}} \cdot \omega_{\text{low}} + w_{\text{med}} \cdot \omega_{\text{med}} + w_{\text{high}} \cdot \omega_{\text{high}}$$

$$w_{\text{low}} = 0.25 \quad , \quad w_{\text{med}} = 0.5 \quad , \quad w_{\text{high}} = 0.25$$

Using three different ANN, dedicated to the performance prediction at the three operating points, improves the convergence.

The velocity vectors at midspan, shown in Fig. 6.20, indicate attached flow at all three operating points and hence stable operation at low mass flow as well as a large pressure rise at high mass flow.

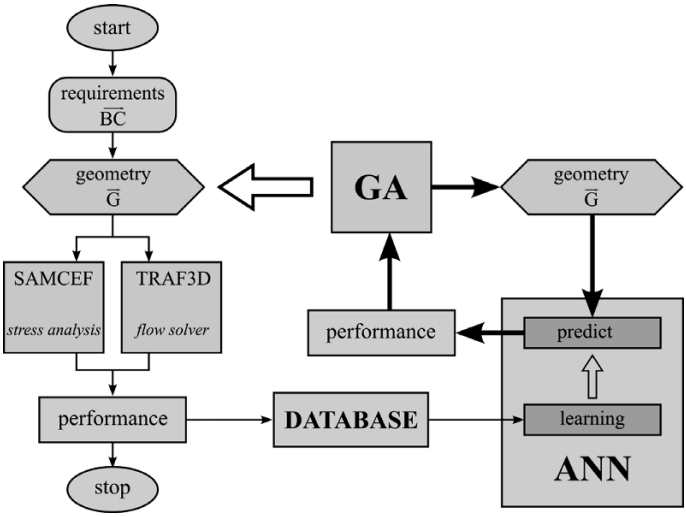

Fig. 6.21 Multidisciplinary optimization flow chart

6.6 Multidisciplinary Optimization

Mechanical constraints such as maximum stress and deformation have a direct impact on the turbomachinery integrity and must therefore be rigorously respected. Some of them can be guaranteed by a simple limitation of a design parameter and do not require any further analysis. Bird ingestion resistance may be accounted for by specifying a minimum leading edge radius (Rle) for fan blades. Corrosion may define the minimum trailing edge radius (Rte) and blade thickness.

However most of the mechanically unacceptable geometries result from a combination of different design parameters and cannot be avoided by restricting the individual parameters. A rigorous approach is the verification of the stress level by an FEA of the $N \times t$ geometries generated by the GA. This is possible by extending the two-level design method to more than one discipline [18]. The GA, searching for the optimum geometry, gets its input from the FEA as well as from the NS flow analysis (Fig. 6.21). The same type of extension can also be made to verify the constraints related to aero-acoustics or weight limitations.

The multidisciplinary optimization method requires the following extensions: stress predictions by the ANN, FEA stress analysis in parallel with the NS calculations, an extension of the OF to account for mechanical targets and the specification of additional design parameters that allow stress reductions. The main advantages of this approach (Fig. 6.21) are:

- The existence of only one "master" geometry, i.e., the one defined by the geometrical parameters used in the GA optimizer. This eliminates

possible approximations and errors when transmitting the geometry from one discipline to another.
- The existence of a global OF accounting for all design criteria. This allows a more direct convergence to the optimum geometry without iterations between the aerodynamically optimum geometry and the mechanically acceptable one.
- The possibility to do parallel calculations. The different analyses can be made in parallel if each discipline is independent, i.e., if stress calculations do not need the pressure distribution on the vanes and flow calculations are not influenced by geometrical deformations.

The multidisciplinary optimization method is illustrated by the design of a 20 mm diameter radial compressor for a micro-gas turbine rotating at 500,000 rpm [18]. The corresponding tip speed of 523.6 m/s results in very high centrifugal stresses. Titanium TI-6AL-4V has been selected for its high yield stress over mass density ratio (σ_{yield}/ρ). The characteristics used in the calculation are: Elasticity modulus = 113.8×10^9 Pa, Poisson modulus = 0.342 and mass density = 4.42×10^3 kg/m^3.

6.6.1 3D Geometry Definition

The hub and shroud meridional contour of radial impellers are defined by third-order Bézier curves, between the leading edge and trailing edge section (Fig. 6.22) [2, 6]. They are fully defined by the control points (X0,R0) to (X3,R3) at hub and shroud. The axial length (X3-X0) and outlet radius R3 are prescribed. They are the result of a preliminary 1D design where one can also account for the off-design operation.

The radius R1 should be larger than R0 at the shroud because otherwise, the unshrouded impeller cannot be mounted. One imposes that X2 is smaller than or equal to X3 at hub and shroud in order to avoid that the impeller exit bends forward. Restricting the possible variations of the design parameters to realistic values also accelerates the convergence.

Second-order curves are used for the upstream and downstream extensions. The points A and B at hub and shroud are automatically adjusted to obtain a smooth transition between the impeller and the radial inlet section. The 6 unknowns that need to be defined during the optimization process are indicated by arrows in Fig. 6.22.

The blade camber line β distribution at hub and shroud are defined by cubic Bézier curves in Bernstein polynomial form

$$\beta = \beta_0(1-u)^3 + \beta_1(1-u)^2 u + \beta_2(1-u)u^2 + \beta_3 u^3$$

β_0 is the blade camberline angle at the leading edge hub or shroud ($u = 0$) and β_3 is the blade trailing edge camber angle ($u = 1$). Similar curves are defined for the splitter vane β distributions.

The coordinates θ of the blade camber line are computed by integrating the β distribution along hub and shroud (Fig. 6.23):

$$d\theta = \frac{dm \tan \beta}{R}$$

The leading and trailing edge blade angles β_0 and β_3 can vary over $\pm 5°$ around a first estimate of the optimum value. Values of β_1 and β_2 are not constrained but the values of θ_3, obtained by integrating β along hub and shroud, should not be too different at the training edge, i.e., the trailing edge rake angle (Fig. 6.7) should be less than 45°.

The splitter trailing edge blade angles are equal to the full blade values at hub and shroud. This results in 14 design variables for the full and splitter blade camber line definition.

The impeller blade definition is completed by a parameterized thickness distribution (Fig. 6.24). Blade thickness distributions at hub and shroud are function of one parameter: the thickness LE of the ellipse defining the leading edge. Trailing edge thickness TE is related to LE. The blade thickness is fixed at the shroud (LE=TE=0.3 mm). The parameter defining the blade thickness at the hub can vary between 0.3 and 0.6 mm. The same value is used for the main and splitter blades. This increases the number of design parameters by 1.

The streamwise position of the splitter blade leading edge is also a design parameter. It is defined as a percentage of the main blade camber length and

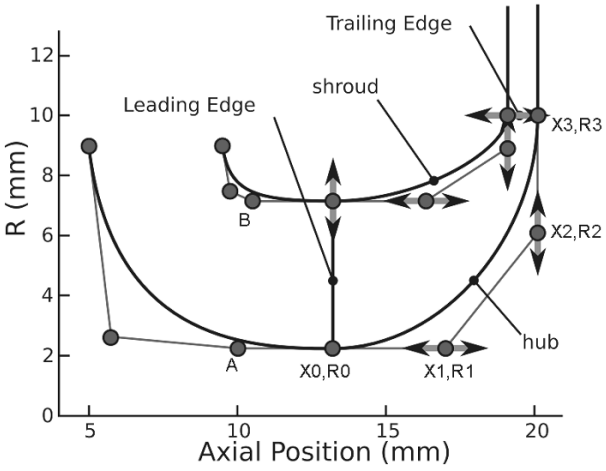

Fig. 6.22 Parameterization of meridional contour. Meridional contour defined by Bézier control points

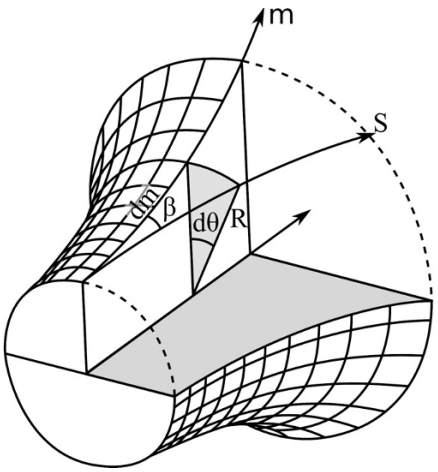

Fig. 6.23 Definition of θ distribution

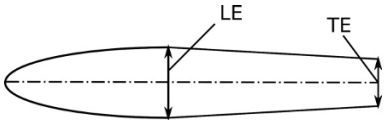

Fig. 6.24 Parameterized thickness distribution perpendicular to the blade camber line (not to scale)

can vary between 20% and 35% at hub and shroud (2 extra design parameters).

The number of blades could also be a design parameter to be optimized, but has been fixed to 7 for manufacturing reasons. This brings the total number of design parameters that need to be defined by the optimizer to 23.

6.6.2 Multidisciplinary Objective Function

The OF driving the multidisciplinary optimization increases with decreasing aero performance and when the aero and mechanical requirements are not met

$$OF = \ w_m \cdot P_{\text{mech}} + \ w_\eta \cdot P_{\text{perf}} + w_m \cdot P_{\text{mass}} + \ w_M \cdot P_{\text{Mach}}$$

The first penalty concerns the mechanical stresses. Limiting the maximum stress results in an inequality and geometries that do not satisfy this condition should be eliminated. However, this information about undesired geometries can also be used to guide the optimization algorithm towards acceptable ones. This is achieved by adding to the OF an extra term that increases

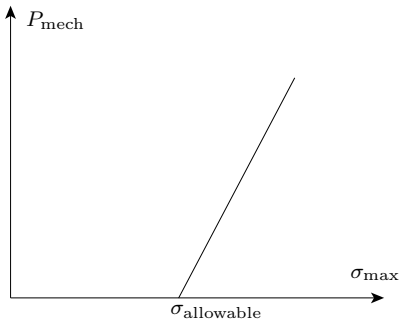

Fig. 6.25 Penalty function for not respecting stress limits

when the stress exceeds the prescribed maximum allowable value. Adding those geometries to the database makes this information available to the ANN which in turn informs the GA about what part of the design space is unacceptable in terms of stress level. Those geometries will therefore not be proposed for further investigation by the NS and FEA.

The penalty increases linearly when a prescribed value $\sigma_{\text{allowable}}$ is exceeded and is zero when the inequality $\sigma_{\text{max}} < \sigma_{\text{allowable}}$ is true (Fig. 6.25)

$$P_{\text{mech}} = \max\left[\frac{\sigma_{\text{max}} - \sigma_{\text{allowable}}}{\sigma_{\text{allowable}}}, 0\right]$$

This weak formulation of the stress constraint does not guarantee that the proposed geometry satisfies the stress limit in a strict way. It can not be excluded that an increase of the stress penalty is compensated by an equivalent decrease of another penalty term. However this risk can be minimized by increasing the weight factor of the stress penalty. The final selection of the optimum geometry is anyway made by the designer, on the basis of the NS and FEA results of all geometries produced by the optimization algorithm.

The second penalty term concerns the efficiency and has been already explained in Sect. 6.2.2

The third penalty verifies the mass flow and has now two contributions. The first one, explained in Sect. 6.2.2, increases when the total mass flow differs from the required one by more than a prescribed value. The second one penalizes the difference in mass flow on both sides of the splitter blade

$$P_{\text{massdiff}} = \left(\frac{\dot{m}_{\text{upper}} - \dot{m}_{\text{lower}}}{\dot{m}_{\text{upper}} + \dot{m}_{\text{lower}}}\right)^2$$

Having the same mass flow in every flow channel will result in a more uniform impeller exit flow and favors a more uniform distribution of blade loading.

The penalty on the Mach number is different from the one used for turbine blades (Sect. 6.4.2) and has two contributions. The first one penalizes negative

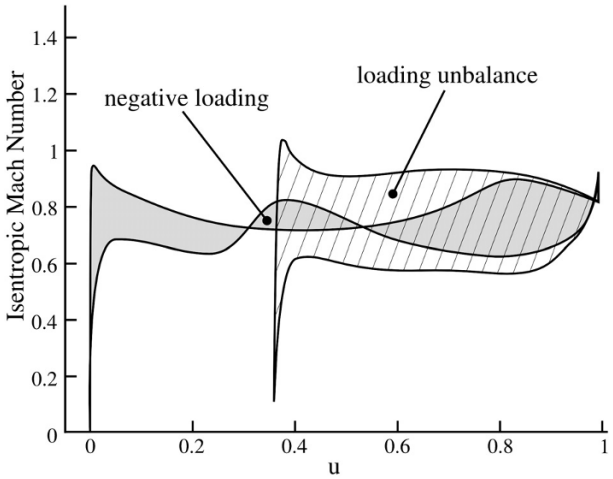

Fig. 6.26 Penalty function for negative loading and loading unbalance in a compressor with splitter vanes

loading and is proportional to the area between the suction and pressure side when the pressure side Mach number is higher than the suction side one (Fig. 6.26). Areas of negative loading are penalized because they result in extra friction losses without contribution to the pressure rise

$$P_{\text{negativeloading}} = \int_0^1 \max(M_{ps}(s) - M_{ss}(s), 0.0) \cdot ds$$

The second contribution to the Mach penalty increases with the load unbalance between main blade and splitter blade. This penalty compares the area between the suction and pressure side Mach number distribution of main blade A_{bl} and splitter blade A_{sp}, corrected for the difference in blade length (Fig. 6.26):

$$P_{\text{load unbalance}} = \left(\frac{A_{bl} - A_{sp}}{A_{bl} + A_{sp}} \right)^2$$

6.6.3 Design Conditions and Results

The computational domain starts at constant radius in the radial inlet and ends in the parallel vaneless diffuser at $R/\text{R2} = 1.5$ (Fig. 6.22). Part of the hub surface at the inlet (for $R < 4$ mm) rotates because it connects the compressor shaft to the electric generator. The total inlet temperature is 293 K and the total inlet pressure is 1.013×10^5 Pa.

Table 6.4 List of penalty parameters and weight factors

stress penalty weight	24.0	mass flow difference weight	20.0
efficiency required	82.5	Mach negative loading weight hub	15.0
efficiency penalty weight	40.	Mach negative loading weight shroud	24.0
mass flow required	20.0 g/s	Mach loading unbalance weight	1.0
mass flow penalty weight	18.0	Mach loading unbalance weight	1.5

The disk thickness at the compressor rim is 1 mm. It is connected to a 8 mm diameter shaft with a fillet of 2 mm radius, all made of one piece. A fillet radius of 0.25 mm is applied at the blade hub to limit the local stress concentrations. The unshrouded impeller has a tip clearance of 0.1 mm, which is 10% of the exit blade height. This is typical for these small impellers and one of the reasons for the moderate efficiencies.

The weights in the OF depend on the application and allow emphasis on performance or on mechanical integrity. They have been determined from the knowledge gained in previous optimizations and are listed in Table 6.4. Taking into account the difference in weight factors, an efficiency drop of 1% is as penalizing as an excess in stress limit of $40/24 = 1.66\%$ (or 6.66 MPa).

The optimization starts from a "baseline" impeller which is the result of a simple aerodynamic optimization without stress analysis. Although this geometry has a good efficiency, it cannot be used because an FEA predicts von Mises stresses in excess of 750 MPa. It serves as a reference for further optimizations.

The TRAF3D solver [3] with an extension to calculate impellers with splitters is used to predict the aerodynamic performance of the radial impellers. Structured H-grids with $2 \times 216 \times 48 \times 52$ (1,090,000 cells) are used for all computations to guarantee a comparable accuracy for all the samples stored in the database. All computations are non-adiabatic with wall temperature fixed at 400 K, as found in a previous study on the heat transfer from the turbine to the compressor [19].

The commercial code SAMCEF [14] is used for the stress calculation. Quadratic tetrahedral elements are used as a compromise between element quality and automatic meshing. Similar grids with 250,000 nodes and 160,000 elements are used for all samples. The grid is refined in areas of stress concentrations. Periodic boundary conditions are applied, such that only a $1/7^{\text{th}}$ part of the geometry needs to be analyzed. The maximum allowable stress is a function of the material temperature. A large safety margin, to account for vibrations and possible local temperature peaks, results in a maximum allowable value of 400 MPa.

Eight individual ANNs are used: one to predict the efficiency, two to predict the mass flow in the channels on each side of the splitter, 4 ANNs predict the Mach number distribution respectively at hub and shroud of the full and splitter blades and one predicts the maximum stress in the geometry. This

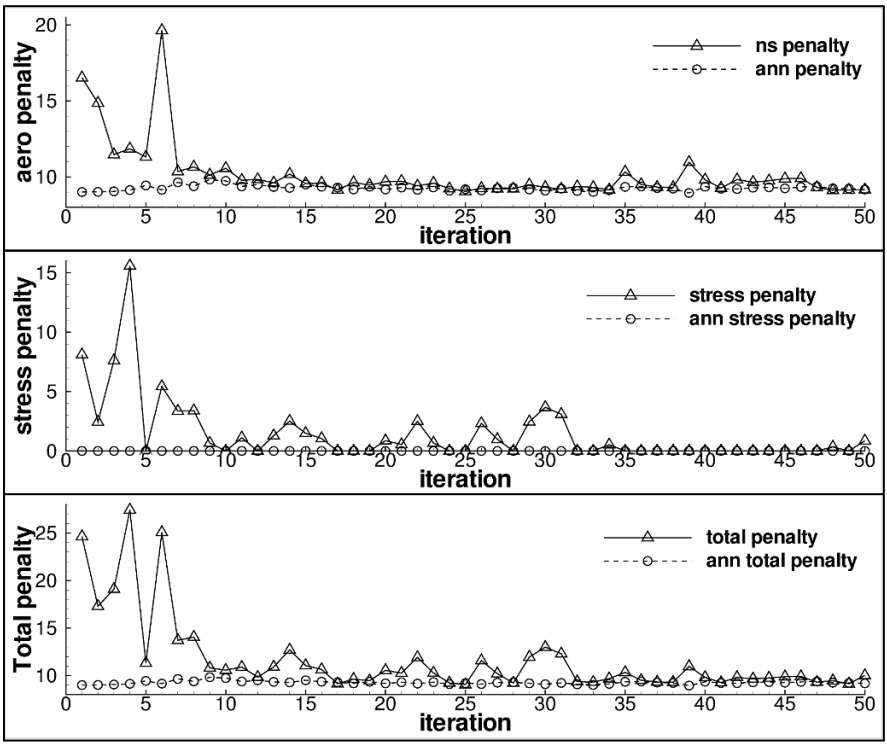

Fig. 6.27 Convergence history of the optimization

split into dedicated ANNs enhances the accuracy by which important values such as efficiency and maximum stress can be predicted.

A 2^{n-p} factorial design is used where n is 23 while p is fixed at 17. This results in a total of $2^6 = 64$ samples in the initial database. An initial database contains only 53 geometries since 13 geometries defined by the DOE technique could not be analyzed due to geometrical constraints (intersection between the main blade and the splitter blade). Two additional geometries have been added, namely the baseline geometry and the central case. The latter is a geometry with all parameters at 50% of their range.

Figure 6.27 shows the convergence history of the optimization. The "aero penalty" (based only on NS predictions of the efficiency, the Mach number distribution and mass flow), the "stress penalty" (based on the result of the FEA analyses) and the "total penalty" are all compared to the ones predicted by the ANNs. One observes a decrease in the discrepancy between both prediction methods with iteration number. This is the consequence of an increasing number of samples in the database, resulting in more accurate ANNs.

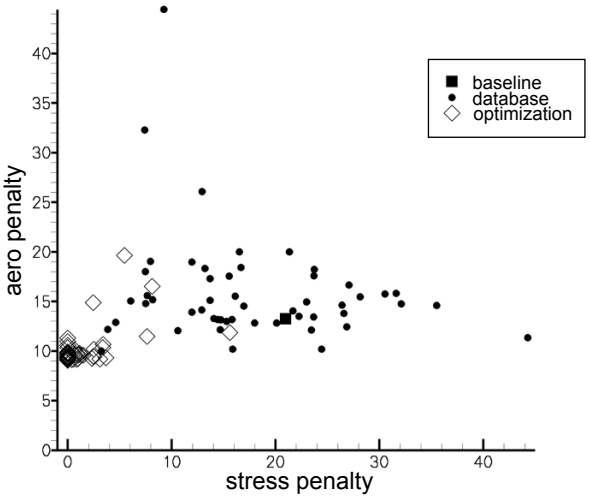

Fig. 6.28 Aero penalty versus stress penalty for baseline, database and optimized geometries

Only 10 iterations are needed to obtain a very good agreement for the aero penalty. The ANN predicted stress penalty is zero for every geometry proposed by the GA. However it takes more than 15 iterations before the FEA confirms that the proposed geometries do not violate the mechanical constraints.

The good agreement in both stress and aero penalties over the last 18 iterations indicates that the ANN predictions are reliable. It means that the same optimum geometry would have been obtained if the GA optimization had been based on the more sophisticated NS and FEA analyses. Hence no further improvement can be expected and the optimization procedure can be stopped after 35 iterations.

The aero penalty is plotted versus the stress penalty in Fig. 6.28. The database geometries show a good spread. Geometries created during the optimization process are all in the region of low penalties. Most of them outperform the geometries of the database. Only a few geometries have penalties of the same order as the database samples. Those geometries are the ones created during the first 10 iterations when the ANN is still inaccurate.

Figure 6.29 is a zoom on the low penalty region of Fig. 6.28. A large number of geometries have zero stress penalties but with a different aero penalty. The geometries corresponding to iteration 17, 49 and 25 have the lowest aero penalty and satisfy the stress constraints. Details are listed in Table 6.5. Iteration 2 has the highest efficiency (60.4%) but has a high aero penalty due to negative loading and loading unbalance. This geometry is also added to Table 6.5.

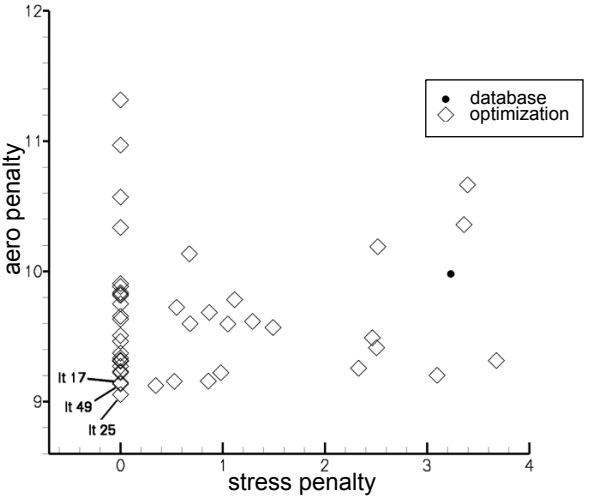

Fig. 6.29 Zoom on the low penalty region of Fig. 6.28

Table 6.5 Comparison between baseline and optimized impellers

	Baseline	Iter. 2	Iter. 17	Iter. 25	Iter. 49
$\eta_{TS}(\%)$	62.34	60.42	60.31	60.06	59.68
P_{aero}	13.24	14.86	9.14	9.05	9.14
$P_{\text{loadunbalance}}$	0.06	3.84	0.26	0.04	0.00
σ_{vonMises} (MPa)	749.	440	389	367	396
Blade lean (°)	-7.8	-11.8	-8.6	-7.3	-15.0
$\dot{m}$ (g/s)	25.9	19.6	20.2	20.2	20.1
Power (kW)	3.19	2.52	2.61	2.62	2.62
Spec.Pow. (W.s/kg)	1123.2	128.6	129.2	129.7	130.3

From all the geometries created during the optimization, iteration 25 performs best. It has an efficiency a little lower than iteration 17 but less loading unbalance and the stresses are 33 MPa below the limit. In spite of its high efficiency, the baseline impeller shows a high aero penalty because of a very high mass flow.

The influence of the stress penalty on the optimization is clear by comparing the values of the baseline impeller with the ones of iteration 25. The reduction of the maximum stress level with 370 MPa is at the cost of a 2.3% decrease of efficiency.

Figures 6.30 and 6.31 show the von Mises stresses in the baseline geometry and the one of iteration 25, respectively. The drastic reduction in stress of the optimized impeller is due to:

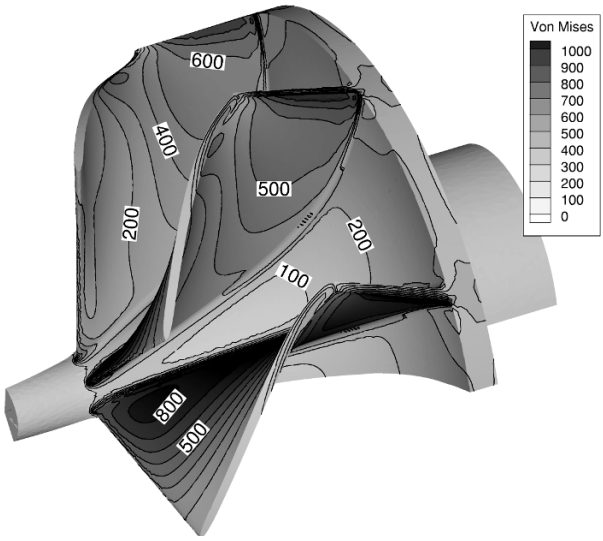

Fig. 6.30 von Mises stresses due to centrifugal loading in the baseline

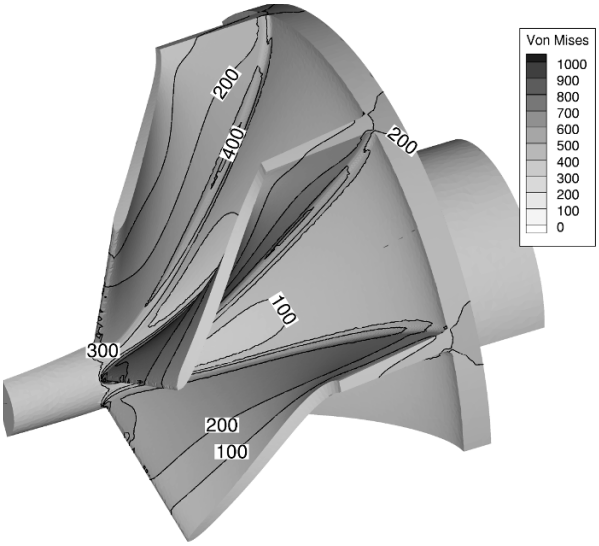

Fig. 6.31 von Mises stresses due to centrifugal loading in iteration 25

- the reduced blade height at the leading edge, resulting in lower centrifugal forces at the leading edge hub;
- the increase of blade thickness at the hub;
- the modified blade curvature.

The latter two result in a decrease of the bending stresses.

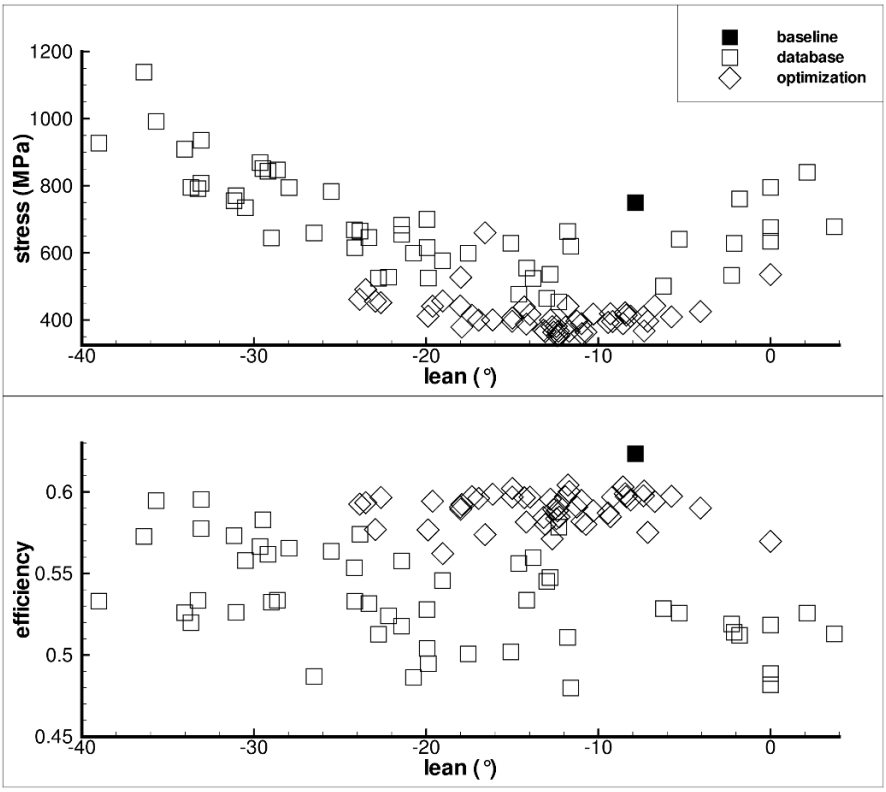

Fig. 6.32 Blade lean versus stress and efficiency for database and optimization geometries

The blade lean is positive in the direction of rotation (Fig. 6.7). It results from the integration of the β distribution at hub and shroud when the trailing edge rake is limited to 45.0°. It is often limited to a small arbitrary value for stress reasons. Iteration 49 on Table 6.5 shows that it can be as low as −15.0° without exceeding the maximum stress limit. Figure 6.32 confirms that minimum stresses are observed around −15.0°. Several geometries with good efficiency are found for lean angles between −40.0° to −5.0°. The decreasing efficiency for lean angles larger than −5.0° suggests that a limited negative lean may have a favorable effect on performance.

Figure 6.33 shows the impact of the leading edge blade height on the stress and efficiency. The radius at the LE shroud (see Fig. 6.22) can vary between 6.5 and 7.5 mm, resulting in a blade height of 4.25 and 5.25 mm, respectively. Values at 4.5 mm and 5.0 mm are database samples.

Shortening the blades lowers the stresses but the database samples suggest a small drop in efficiency. This explains the difficulty in maintaining a high efficiency when reducing the stress. However, the optimized geometries have shorter vanes and show a high efficiency. This indicates that the efficiency also

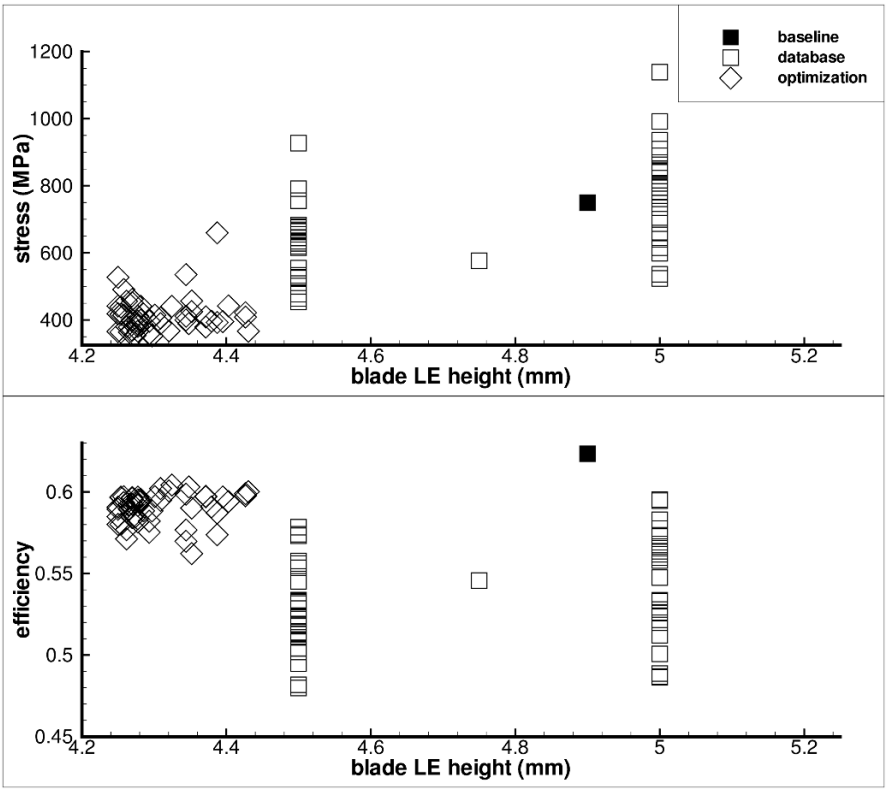

Fig. 6.33 Blade leading edge height versus stress and efficiency

depends on an optimum choice of other parameters. Although they have a less pronounced influence on stress and efficiency, a correct definition of their value is needed to reach the optimum. This illustrates the strongly coupled nature of the design problem and the need for an optimization tool.

6.7 Conclusions

It has been shown how a two-level optimization technique, an adequate parameter selection for the GA, the use of DOE for the definition of the database and an optimized learning technique for the ANN can considerably decrease the computational effort required by evolutionary theories. The proposed procedure is a self-learning system that makes full use of the expertise gained during previous designs.

The automated design method can be used with any flow solver and does not require the definition of a target pressure or Mach number distribution.

The Mach number-based criteria included in the Objective Function help enforce the convergence to the optimum and improve the off-design performance.

The reduction in computer effort makes the design of customized profiles and the multipoint optimizations affordable, as illustrated by the design of a transonic turbine blade and a Low Solidity Diffuser.

The use of a pseudo Objective Function to account for the mechanical and other constraints is presented, and the advantages and disadvantages are discussed. It is shown that the method is able to find the optimum combination of design parameters allowing a drastic reduction of the stresses with minimum penalty on efficiency.

The optimization algorithm provides the designer more insight into the multidisciplinary design problem. The main parameters that allow a reduction of the stresses are identified during the optimization process. This optimum combination may result in unexpected geometries that would not be accepted when using simplified stress criteria.

It has been shown how the use of computerized design techniques is a powerful tool to cope with the increasing complexity of advanced turbomachinery component design. However, the outcome of this valuable support still depends on the input of the designer in terms of a careful selection of design parameters, a clear definition of objectives and constraints as well as validation of results.

Acknowledgements The contributions by Dr. Stephane Pierret, Dr. Zuheyr Alsalihi and Dr. Tom Verstraete to the development of this method are gratefully acknowledged.

References

1. Aarts, E.H.L., Korst, J.H.M.: Simulated annealing in Boltzmann machines. Wiley Chichester (1987)
2. Alsalihi, Z., Van den Braembussche, R.A.: Evaluation of a design method for radial impellers based on artificial neural network and genetic algorithm. In: Proc. of ESDA 2002, 6th Biennial Conference on Engineering Systems Design and Analysis. Istanbul (2002)
3. Arnone, A.: Viscous analysis of three-dimensional rotor flow using a multigrid method. ASME Journal of Turbomachinery **116**, 435–445 (1994)
4. Bäck, T.: Evolutionary Algorithms in Theory and Practice. Oxford University Press, New York (1996)
5. Carroll, D.L.: FORTRAN genetic algorithm (GA) driver version 1.7.1a (2001). URL `URL:http:/cuaerospace.com/carroll/ga.html`
6. Cosentini, R., Alsalihi, Z., Van den Braembussche, R.A.: Expert system for radial compressor optimization. In: Proc. 4th European Conference on Turbomachinery. Firenze (2001)
7. Demeulenaere, A., Van den Braembussche, R.A.: Three-dimensional inverse design method for turbine and compressor blades. In: Design Principles and Methods for Aircraft Gas Turbine Engines, RTO MP-8 (1998)

8. Harinck, J., Alsalihi, Z., Van Buytenen, J.P., Van den Braembussche, R.A.: Optimization of a 3D radial turbine by means of an improved genetic algorithm. In: Proceedings of European Turbomachinery Conference. Lille (2005)
9. Kostrewa, K., Van den Braembussche, R.A., Alsalihi, Z.: Optimization of radial turbines by means of design of experiment. Tech. Rep. VKI-PR-2003-17, von Kármán Institute for Fluid Dynamics (2003)
10. Lichtfuss, H.J.: Customized profiles – the beginning of an area. In: ASME Turbo Expo 2004, Paper GT2004-53742 (2004)
11. Montgomery, D.C.: Design of Experiments. John Wiley & Sons, Inc. (1997)
12. Nursen, C., Van den Braembussche, R.A., Alsalihi, Z.: Analysis and multipoint optimization of low solidity vaned diffusers. Tech. Rep. VKI-SR-2002-31, von Kármán Institute for Fluid Dynamics (2002)
13. Pierret, S., Van den Braembussche, R.A.: Turbomachinery blading design using Navier Stokes solver and artificial neural network. ASME Journal of Turbomachinery **121**, 326–332 (1999)
14. SAMTECH group: SAMCEF FEA code. URL `www.samcef.com`
15. Siamion, J., Coton, T., Van den Braembussche, R.A.: Design and evaluation of a highly loaded LP turbine blade. In: Proc. 5th ISAIF Conference (International Symposium on Experimental and Computational Aerothermodynamics of Internal Flows). Gdansk (2001)
16. Thilmany, J.: Walkabout in another world. Mechanical Engineering **122**(11), 1–9 (2000)
17. Vanderplaats, G.N.: Numerical Optimization Techniques for Engineering Design. McGraw-Hill (1984)
18. Verstraete, T., Alsalihi, Z., Van den Braembussche, R.A.: Multidisciplinary optimization of a radial compressor for micro gas turbine applications. In: ASME Turbo Expo 2007, Paper GT2007-27484 (2007)
19. Verstraete, T., Alsalihi, Z., Van den Braembussche, R.A.: Numerical study of the heat transfer in micro gas turbines. Journal of Turbomachinery **129**(4), 835–841 (2007)
20. Volpe, G.: Geometric and surface pressure restrictions in airfoil design. In: Special Course on Inverses Methods for Airfoil Design for Aeronautical and Turbomachinery Applications, AGARD-R-780 (1990)