AIAA-2000-0838

MULTIDISCIPLINARY TURBOMACHINERY BLADE DESIGN OPTIMIZATION

Rolf Dornberger*, Peter Stoll†, Dirk Büche†
ABB ALSTOM POWER Technology, CH-5405 Baden-Dättwil, Switzerland.

Alexander Neu‡
ABB ALSTOM POWER, CH-5400 Baden, Switzerland.

ABSTRACT

The development of an optimization environment for multidisciplinary turbomachinery blade design is presented. The optimization environment simplifies the use of various optimization and evaluation methods. An advanced blade parameterization scheme is developed, in order to perform particular blade modifications with a limited number of optimization parameters. The generation of invalid designs is avoided a priori, which is particularly important in turbomachinery design with expensive 3D flow computations. The parameterization includes several components, such as profile, blade and grid generation tools. A CFD solver computes various turbomachinery stage designs in parallel. A preevaluation tool extracts necessary data and processes objectives and constraints. These objectives and constraints are assessed in an evaluator indicating the design quality. Interfaces between all tools of the optimization environment guarantee an automated, closed data flow. It is shown that this optimization environment enables easily sensitivity analyses as well as optimization runs. Investigations are made in order to derive reasonable 3D flow assessment criteria for the objectives and constraints. Results of performing multidisciplinary turbomachinery blade design optimizations on a turbine stage are presented.

INTRODUCTION

In turbomachinery design, the future trend in multidisciplinary optimization (MDO) is to cover not only aero- and thermodynamic performance of turbines and compressors, but also geometrical requirements, mechanical integrity, and manufacturing costs[1]. Life cycle costs, product cycle time, weight, emissions and heat transfer[2] are additional, possible criteria during the optimization process.

In the past, as CFD computations of entire 3D flow fields were computationally too expensive, design optimizations were limited to simpler models of one or two spatial dimensions only. Meanwhile, computers and CFD codes have become more powerful. Optimization methods have been improved in order to reduce the number of calls of computationally extensive programs, i.e. CFD codes. 3D optimization provides today the possibility of improved designs, where 1D, 2D and Q3D methods are already drained[3]. Thus, it is very promising to couple optimization tools with 3D CFD codes, additionally involving further solvers for performing entire 3D multidisciplinary design optimizations, where impressive results of entire airplane optimizations are available[4].

However, turbomachinery design business has its own specific design problems[5]. Large optimization environments can simply be created, which are theoretically very powerful, but practically useless. The designers' intuitions cannot be implemented as easily and completely as originally assumed. Optimization environments must involve special geometry modification tools for the handling of 3D designs, as well as mesh generation tools and CFD solvers. A sound interface programming is needed. Thus, optimization is often seen to be nothing else than a black box, respecting many constraints, producing many possible solutions, involving heaps of optimization parameters, and finding out of these quickly the optimal blade design.

Nevertheless, such batch-optimizations often do not converge satisfactorily, if at all. Designs are found, which would never be favored by the designers, up to totally absurd designs. Thus, more and more constraints are additionally applied in order to filter out unwanted and poor designs after they are computed. The effect is that the design space becomes increasingly constrained that finally no acceptable solutions can be found at all, although many time-consuming computations were

* Dr., Scientist, Thermo- and Aerodynamics Department, Member AIAA.
† Scientist, Thermo- and Aerodynamics Department.
‡ R&D Engineer, Steam Turbine Development.

made. Advantages of a sophisticated parameterization are obvious[6]. It reduces the number of design variables to a smaller set of optimization parameters. Thus, the optimizer has to generate only a few parameters, while the parameterization helps avoiding the generation of poor sets of design variables a priori. Therefore, many constraints can already be respected inside the parameterization before the design is computed, and not afterwards.

The entire design process is accelerated significantly, if all tools are running in an automated way. Several possibilities of saving computation time exist. We have laid emphasis on parallelization. As optimization methods always compute gradients, directions and/or regions of increasing design quality, the benefits of geometry modifications are normally found within fewer total optimization steps, if parallel execution of the solver computations can be applied. Although some optimization methods are more suited for parallelization than other ones, the principle of the optimization environment stays the same.

OPTIMIZATION ENVIRONMENT

An optimization environment has been developed[7], which simplifies the implementation and testing of optimization and evaluation schemes (Fig. 1). In addition, the optimization environment provides a generic interface to external solvers, which can be exchanged easily for computing different cost functions.

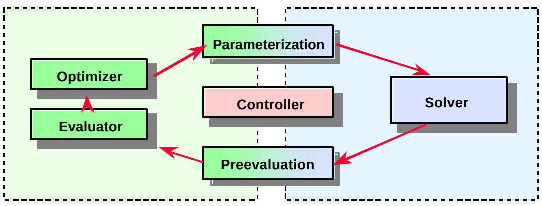

Figure 1: General optimization environment

In turbomachinery blade design optimization, the main solver is a CFD turbomachinery code. Applying multidisciplinary optimization, additional solvers process further disciplines, either parallel to the CFD computations, and/or afterwards using calculated flow quantities. The parameterization transforms the optimizer output into solver input. Thus, the parameterization surrounds the profile and blade generator as well as the grid generator, as shown in Fig. 2. The preevaluation tool extracts from the CFD results the values required by the evaluator, such as objectives and constraints. Therefore, it first uses CFD postprocessing tools in order to compute integral values of the 3D flow field, for example entropy production. If multiple disciplines are involved, further solver compute additional data. The preevaluation processes all these data in order to filter out additional objectives and constraints. Hence, the preevaluation tool includes postprocessing and data collection, as shown in Fig. 2. The evaluator finally evaluates the objectives and constraints by computing fitness values with respect to particular assessment schemes. A controller supervises all components of the optimization loop, in order to start the entire optimization process, to keep it running, and to terminate it, when appropriate stopping criteria are fulfilled. Thus, the controller sequentially executes succeeding tools, as soon as the results from the preceding ones are available. Due to distributed computations, it also has to re-compute and/or skip failing CFD computations in order to save the entire optimization process.

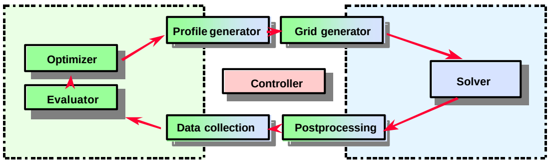

Figure 2: Optimization environment using profile and grid generator as well as postprocessing tools

Various optimization methods and search algorithms have been implemented using different evaluation schemes. All methods are used as minimization schemes with active range checking of the optimization parameters. Gradient methods and other analytic non-gradient optimization algorithms as well as discrete search algorithms and stochastic methods are available for optimizing 3D blade designs. However, depending on the optimization task, some methods are more suitable for identifying reasonable initial designs at all, other ones for further improving these designs.

Since the precise behavior of the objective functions in 3D design is not known a priori (e.g. the probability and appearance of local minima), robust stochastic optimization methods, such as evolution strategies, seem adequate for first optimization runs. The fundamental properties of evolution strategies lie in the parallel search, which leads to robust parallel optimization processes, reducing the risk of being stuck in local minima of the design space. If the design space is smooth and perhaps convex, the disadvantage of these methods compared to gradient methods is their lower convergence speed. We overcome this problem by applying massive parallelization of design evaluations.

In the evaluator, various evaluation schemes, such as weighted-sum using different norms, and Pareto

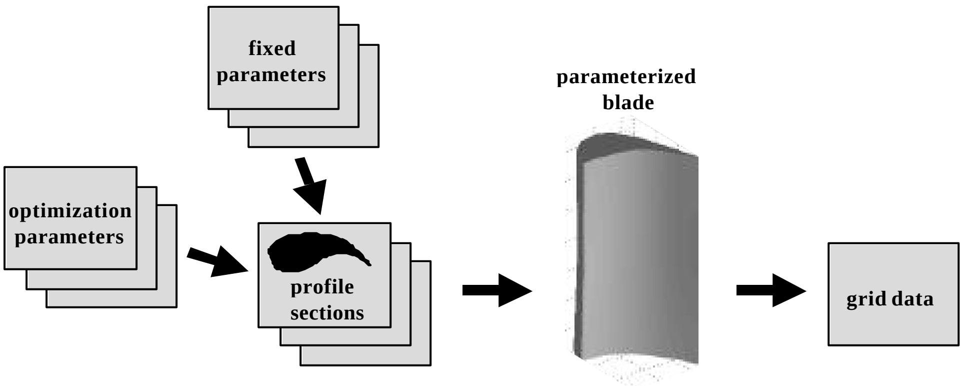

Figure 3: Generating profile sections, stacking them up to the parameterized blade, and creating its grid data

assessment, are implemented, where an additional scaling with reference values is provided. Depending on the optimization task, one overall fitness value or multiple fitness values are computed. Particularly in multidisciplinary optimization, it might be necessary to generate one overall fitness value for all disciplines, as well as individual fitness values for each discipline separately.

All constraints are treated as soft-bounded constraints[7] applying continuous penalty functions forming hills and valleys with more or less steep slopes. In order to allow such continuous penalty functions, the only limitation is that their soft and hard limits must not be identical. By sequentially applying the soft and hard limits, the resulting penalty functions describe either semi-infinite or closed intervals. By overlaying several penalty functions, continuous sequences of intervals can be applied at once.

Due to a wide range of complex models of different disciplines, the optimization of turbomachinery is an excellent example for developing and testing methods in MDO. In order to find good designs, sophisticated weighting between different objectives as well as sensible formulation of the constraints have to be found besides reasonable design parameterization and assessment criteria. Furthermore, the treatment of the constraints must not exclude good solutions a priori, only because some constraints lie on or just beyond the boundaries of the valid design space. Performing good designs with an optimization environment will always be an iterative process between the designer and system, understanding and improving systematically physics and design.

PARAMETERIZATION

Different possibilities of applying suitable blade parameterizations are possible. We investigated the method of generating engineering parameters by the optimizer within given ranges describing particular geometrical properties of the blade. In general, only some of all engineering parameters necessary to describe an entire blade are found reasonable for changing during optimization. Most geometrical parameters of the blades thus remain fixed, always depending on the design optimization task. For example, two totally different tasks are either changing the profile sections while keeping the stacking line of these sections constant, or changing the stacking line while keeping the profile sections constant.

In our representation, the entire blade consists of several profile sections. Free optimization parameters of each profile section are mixed with fixed parameters, as shown in Fig. 3. A profile generation tool creates profiles on different sections of the blade. The profiles are stacked up using particular stacking rules. As a simple radial stacking limits the possibility of locally influencing the flow significantly, the blades are modified with lean, sweep and/or twist. Lean is the bending of blades in circumferential direction, sweep the bending in axial direction, as shown in Fig. 4 for a stator and rotor of a turbine stage. Although twist as spiral deformation over the blade height is very important for adjusting the flow angles, this modification should rather be applied to each profile section individually than to the entire blade.

Lean and sweep are varied by adding first and higher order polynomials to the initial lean and sweep of an existing blade. These polynomials p can be expressed with the normalized radius r as

$$p(r) = C_1 \cdot r + C_2 \cdot 4 \cdot r(1-r) + C_3 \cdot \ldots$$

where C_i are the design parameters. If the function $p(r)$ consists only of two parameters, linear and bowed (quadratic) terms in radial direction are added to the blades. Corresponding polynomials are shown in Fig 5.

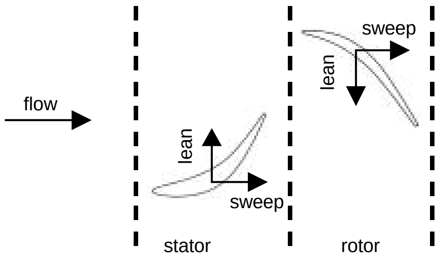

Figure 4: Lean and sweep for the stator and rotor

Positive lean and sweep are measured positively on the pressure side of the blades and in the direction of the axial flow respectively, as shown in Fig. 4, and vice versa.

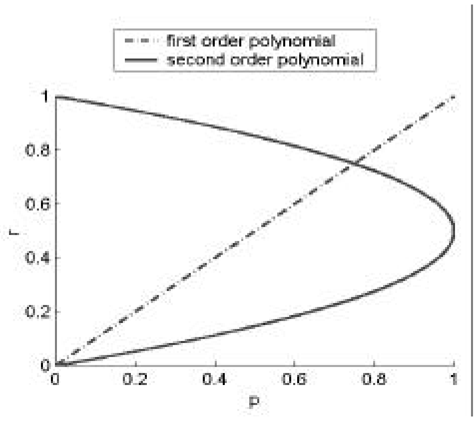

Figure 5: Polynomials describing linear and quadratic lean or sweep

After the generation of the blade geometry, the entire 3D geometry is stored for CAD visualization using spline control points, shown in Fig. 6, and then transformed into Cartesian coordinates lying on certain cutting planes. A grid generation tool uses these coordinates in order to create a mesh around the entire blades needed by the CFD code, as shown in Fig. 7. For

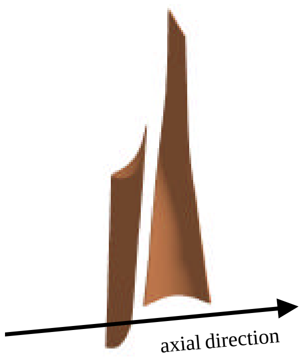

Figure 6: CAD geometry of the stator and rotor

the succeeding flow computations, the CFD code reads boundary and further flow data, i.e. thermodynamic and geometrical boundary conditions. It computes the flow field around the blades, assuming that one blade is a part of a turbomachinery row, or two blades part of a turbomachinery stage (Fig. 7).

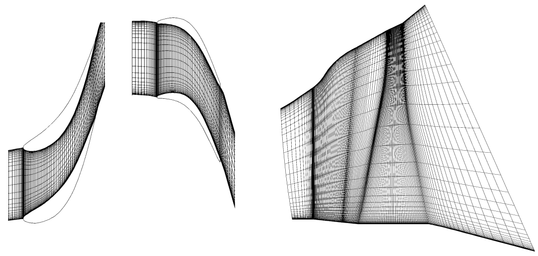

Figure 7: CFD mesh around the stator and rotor

DESIGN ASSESSMENT

Although 3D CFD computations are very common, they are less frequently being used in analyzing turbomachinery already during the design phase. The effects of lean and sweep are not known a priori in detail, but obviously, they have a large, potentially positive effect on the flow. The question is how to quantify and to measure this effect.

As the quality of automated design optimizations heavily depends on the definition of reasonable objectives and constraints, investigations are very sensitive in advance. One idea of getting an estimate of the importance of objectives and the influence of constraints is obtained by changing single design parameters between their geometrical limits, while keeping all other parameters constant. Different values are chosen in order to test their ability of particularly assessing the modified designs.

The results of the CFD computations are postprocessed in order to filter out characteristic values for describing the quality of the blades. These values are taken from the entire flow field or from the outflow section, a plain right behind the rotor blade with the turbine axis as perpendicular bisector. The values can be generated as 1D distributions by integrating the 3D flow field over the tangential direction of the plane, or even as scalar value by integrating over the whole domain.

However, the question of finding reasonable 3D assessment criteria remains a challenging task. Integral values are for example efficiency, overall losses, or entropy production rate. Although one common optimization goal is to maximize total efficiency of a turbomachinery (correspondingly to minimize its losses), the maximization of efficiency of single stages or rows might not be necessarily a suitable objective. The reason is that maximizing the efficiency of one stage easily leads to a significant reduction of efficiency in adjacent stages due to worsened flow conditions.

Furthermore, optimization becomes more complex, when MDO is applied. The quality of the blades is thus assessed not only considering aerodynamic flow field conditions, but also geometrical and mechanical properties, manufacturing costs, and so on. The results of different disciplines must concurrently be processed to objectives and constraints and sent together to the evaluator, which assesses all objectives with respect to all constraints. Nevertheless, the best turbomachinery in multidisciplinary manner is always a compromise of all these disciplines. As optimization is in that case very specific and controversial, we focus in the following examples on aerodynamic objectives, applying aerodynamic and geometrical constraints.

EXAMPLES

In order to test the components and to validate the methods implemented in the optimization environment, an example of a fictive turbine design optimization is presented. A common design task is to retrofit a new blading of turbine stages or rows into an existing casing. Hence, most of all geometrical constraints can already be covered within the parameterization. For example, the mean diameter of the turbine is more or less fixed, but the shape of the blades varies within particular ranges. A good parameterization supports the optimization process by producing only technically reasonable blades, which can be manufactured at all. This depends highly on the geometrical description of the problem, the chosen optimization parameters as well as on the applied parameterization of the blades.

For the investigations in this paper, we assume that the last stage of a low-pressure steam turbine should be optimized, consisting of a stator and rotor blade row, as shown in Fig. 6. This stage is of particular interest to fundamental 3D blade design investigations due to the large blade length compared to the hub diameter, and the opening angle of the casing in axial direction. In contrast to this low-pressure stage, high-pressure turbine blades are less changing in radial dimension due to the smaller flow channel width compared to the turbine diameter. The rotor cross-section of the low-pressure turbine is changing significantly in radial direction following the different flow conditions and circumferential speed. Therefore, the blade properties are highly dependent on its sophisticated 3D shape, achieved by the twisted, leaned, swept and bowed blade design (Fig. 6).

The turbine stage is discretized with a structured mesh of 280.000 elements, as shown in Fig. 7. In profound turbomachinery CFD analyses, this discretization might not be sufficient for the fully detailed resolution of all local flow effects. However, the discretization has been limited to this size for keeping the optimization process to adequate computation times, in order to test the principle of the optimization environment.

The optimization parameters of this example are the linear and quadratic lean and sweep of both rows, as given in Fig. 4. They are varied by adding first and second order polynomials to the existing blade design. Using these parameters for lean and sweep of the stator and rotor, a total number of eight optimization parameters results. These optimization parameters are dependent on other blade parameters like pitch or twist. Thus, modifying lean or sweep adjusts the blades in certain ways to the flow angles. This change could also be achieved by modifying locally the profile sections of the blade, but the number of optimization parameters would increase significantly in this case.

Lean and sweep are applied for improving design objectives like radial loading, reduction of secondary flows, and improvement of the vortex structure by adjusting the position of the blades in their rows. However, the effect of positive or negative lean and/or sweep can not be stated in general. This effect fully depends on type of the turbine, position of the row, pressure ratio, geometrical conditions, and so on.

The following paragraphs assess the basic effects of lean and sweep in this particular example. At the beginning, only one single design parameter is modified between its geometrical limits at a time, while all other parameters are kept constant. Due to the optimization environment, the necessary input and output interactions with the CFD solvers can be reduced to a minimal effort, allowing to run several computations in parallel. Different flow quantities are chosen for investigating them as possible assessment criteria.

Representatively for all eight optimization parameters, the influence of second order lean of the rotor blade is

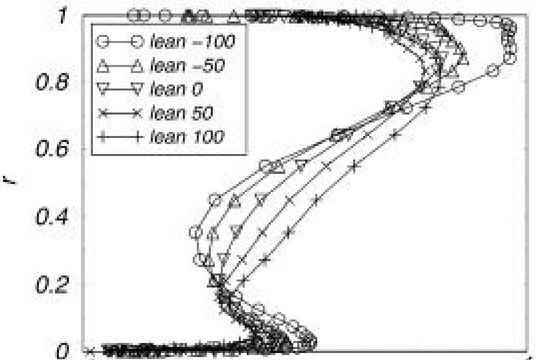

Figure 8: Stagnation pressure after the rotor dependent on second order rotor lean

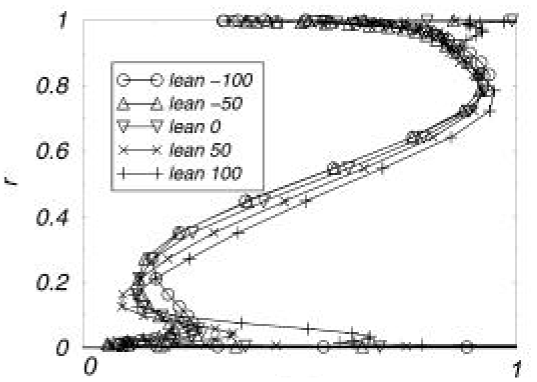

Figure 9: Flow angle after the rotor dependent on second order rotor lean

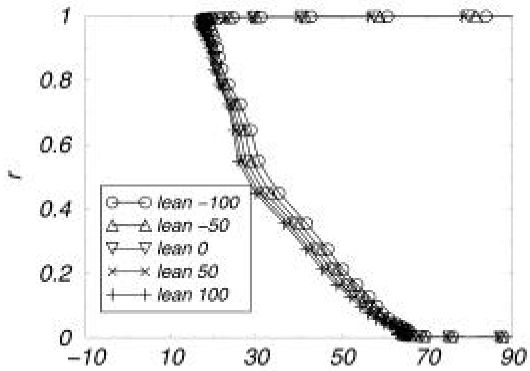

Figure 11: Entropy rise over the stator dependent on second order rotor lean

given in the following figures. Fig. 8 and 9 show the normalized radial distribution of stagnation pressure and flow angle after the rotor blade. In Fig. 10, the entropy rise (production) over the rotor blade is given. The figures emphasize the possibilities of lean and sweep in changing radial distributions as well as absolute values of flow quantities.

However, the blade modifications of one row definitively affect the flow in adjacent rows. For example, the entropy rise in the stator row is plotted in Fig. 11 for different rotor lean configurations of second order. As the entropy rise can be seen a measure for losses [5], the losses of this stator row correlate directly with the lean of the downstream rotor row. Nevertheless, other rows are not affected, since the chosen rotor row is in the last stage of the turbine.

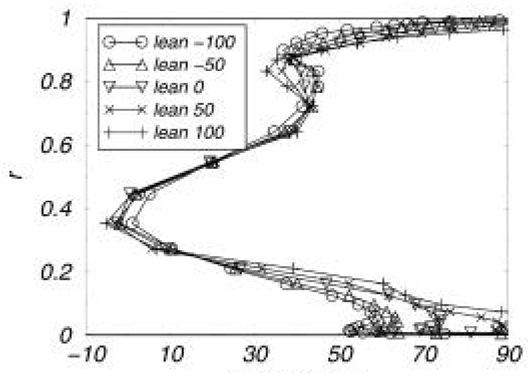

Figure 10: Entropy rise over the rotor dependent on second order rotor lean

As an overview over the sensitivity of the eight applied optimization parameters, Fig. 12 presents their influence to the entropy rise in the rotor row. Each parameter is changed separately. For all parameters, the entropy seems to change smoothly over the parameter space. The chosen parameterization is able to keep the blades inside reasonable geometrical ranges, and to change the flow conditions continuously. Since the entropy rise partly declines, partly increases over the whole parameter space, it can be assumed that at least one inner local extremum must exist. Nevertheless, in order to find the global minimum of this problem with high probability, a robust optimization method has been chosen for first optimization runs.

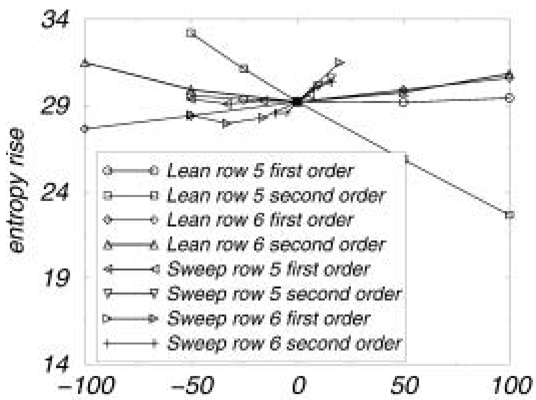

Figure 12: Entropy rise over the rotor dependent on lean and sweep of stator (row 5) and rotor (row 6)

In the following paragraphs, optimization of this turbine stage is presented. As objective function, the total entropy rise over the stator and rotor row is assumed an appropriate measure for the stage losses. The corresponding entropy values are combined by the weighted-sum method to one single resulting objective function. Further investigations have shown, that besides geometrical constraints following aerodynamic constraints must be set: mass flow, stagnation pressure after rotor row, and amount of work, which is done by the rotor row. While all geometrical constraints could already be treated inside the blade parameterization, in order to guarantee the compatibility of the rows with the turbine casing, the aerodynamic constraints are respected using the method of soft-bounded constraints[7]. As this method applies continuous penalty functions (smearing the hard jumps when constraints are violated), the direction, where constraints are not violated, is quickly found.

The optimization process is started with the initial solution that has been given in Fig. 6. As optimization method, a standard stochastic method following the strategies of evolutionary computing is applied. This method is appropriate for our optimization problem, because we are mainly interested in a robust optimization process, which converges independent to the properties of the objective function. Thus, this optimization method should rather be able to locate the position of the global extremum of the problem than to be able to converge fast with the danger of being stuck in local extrema. In order to accelerate the computation, this optimization method has been set up to run 20 CFD computations in parallel.

The convergence of the optimization process is shown in Fig. 13. Always the best solution, which could be found, is plotted versus the number of computed solutions. The jumps in the convergence line is typical for evolution strategies, which generate always a certain number of solutions in parallel, and choose the best among these for the succeeding computations.

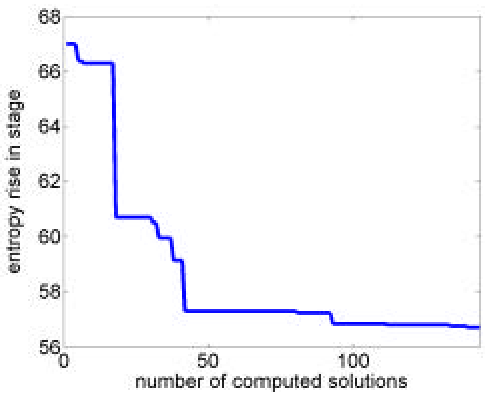

Figure 13: Convergence history of minimizing total entropy rise of the stage

After about 140 solver executions, the optimization process has converged to a solution (blade design) with a reduced total entropy production of 15% compared to the initial solution. In addition, the constraint functions for the mass flow and stagnation pressure could be kept almost constant within bounds of 0.5%. The benefit of the entropy reduction is found in the work of the turbine stage. Due to a reduced entropy production, the losses of the turbine are minor. Thus, its work output could be raised by 2.7% compared to the initial turbine.

Examining the change of the optimization parameters during the optimization, Fig. 14 shows the parameters of stator and rotor lean and sweep, and the resulting fitness values. Typical for the stochastic optimization method is the oscillating behavior of the fitness values. While the fitness is decreasing in average, the method tries to find other regions of feasible designs by randomly modifying the optimization parameters.

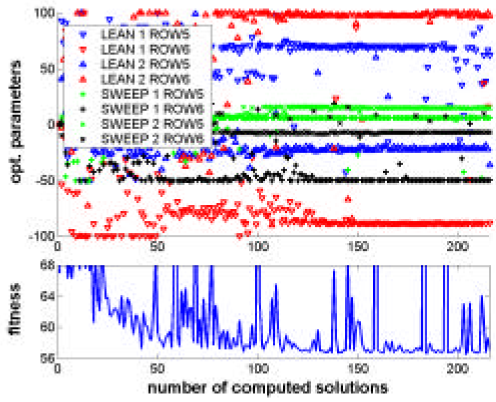

Figure 14: Convergence history of the optimization parameters and corresponding fitness values

As the fitness value is the sum of the total entropy production in both stator and rotor row, Fig. 15 and 16 show the single contributions of the radial distribution of entropy rise in the rotor and stator row. Typical for the solutions is the higher entropy production close to the hub and casing due to secondary losses. Lean and sweep do not seem to improve significantly the blade performance in these areas.

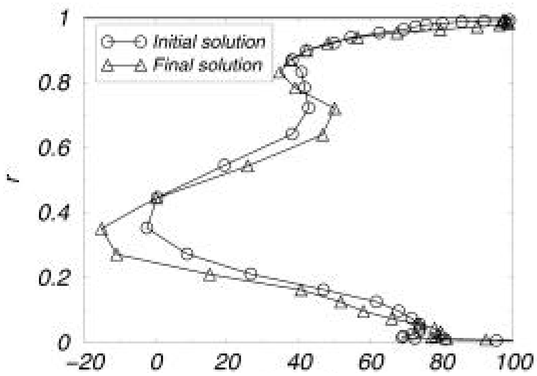

Figure 15: Entropy rise over the rotor in initial and final solution

The entropy production in the mid-span rotor area (Fig. 15) is only slightly reduced in average, mainly at the hub area part. However, the entropy production of the final stator design (Fig. 16) is significantly smaller over the whole blade height.

Thus, the optimization process leads to better turbine properties, resulting in higher efficiency, while all boundary conditions (constraints) of this particular retrofit case can be kept within reasonable limits.

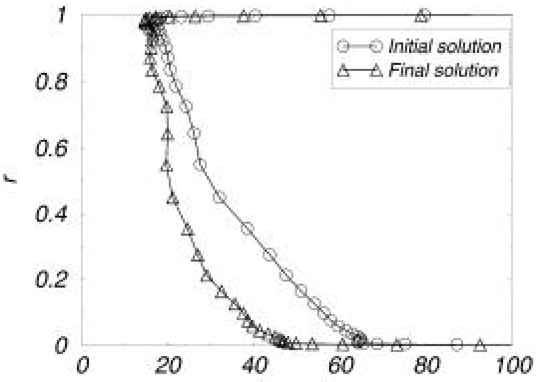

Figure 16: Entropy rise over the stator in initial and final solution

We emphasize that in this example the design optimization of the turbine stage was mainly based on aerodynamic considerations. Mechanical stresses pose additional constraints for MDO, particular important for large rotor blades. Additional solvers have to be integrated in optimization environment for computing further constraints. The methods and parameterization proposed might in general be used for the most axial turbines and compressors. The principle of the optimization environment stays the same as well as the blade parameterization. However, additional solvers have to be applied and reasonable objectives and constraints need to be chosen.

CONCLUSIONS

3D multidisciplinary blade design optimization has to focus first on the basic principle of the optimization environment and then on the diversity of additional features. Following this approach, we investigated in this work the possibilities of separately implementing and connecting tools forming the entire optimization environment. In this way, it is hoped to guarantee firstly an automated, closed data flow with parallel solver executions, and secondly the possibility of easily exchanging the components.

In a first step, standard optimization methods and evaluations schemes are sufficient for generating sets of optimization parameters, as well as for evaluating objectives and constraints. Stochastic optimization methods are suitable for this purpose, where the soft-bounded constraint treatment helps to guide the optimization process in the direction of feasible designs. In a second step, more expertise is required for a sophisticated parameterization, which allows the transformation of optimization parameters into design variables. For example, in 3D turbomachinery blade design it must be decided in advance which blade modifications are desired and useful, in order to keep the number of optimization parameters small. We demonstrated in this work that the polynomial lean and sweep blade parameterization needs only a few optimization parameters, but leads to important blade design modifications. Additionally, invalid blade designs were prohibited a priori by respecting all geometrical constraints inside the parameterization. Hence, this parameterization can produce automatically the blade geometry and create the CFD mesh for flow computations in a quick, reliable and robust way.

However, crucial points of the optimization are the assessment criteria. The definition of 3D aerodynamic assessment criteria as well as the choice of reasonable MDO objectives and constraints are heavily increasing the complexity. The identification of design rules, which lead to good designs, is not straightforward, in particular for 3D design. Nevertheless, with respect to 3D aerodynamic assessment criteria, integral values might easily be formulated, but their relation to improved designs is not always obvious. In absence of any information about realistic flow distributions, the design goal has been the minimization of the total entropy rise (production). As entropy correlates with the losses, its minimization may greatly improve the blade properties. Using this criterion, the optimization environment, the implemented methods, and the blade parameterization are successfully tested. As this optimization environment is thus able to accelerate the entire design process significantly, the work must be continued with the efficient optimization of real turbomachinery blading.

[1] S. Havakechian and R. Greim, Aerodynamic design of 50 per cent reaction steam turbines, Proc Instn Mech Engrs Vol. 213 Part C, IMech, 1999.

[2] S. Müller and P. Koumoutsakos, Film Cooling Optimization Using Evolution Strategies, submitted to AIAA J. 2000.

[3] S. Havakechian and R. Greim, Recent advances in aerodynamic design of steam turbine components, VGB conference, 1999.

[4] J.J. Reuther, A. Jameson et al., Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers, Part 1; Journal of Aircraft; Vol. 36; No. 1; 1999.

[5] J.D. Denton; Loss Mechanisms in Turbomachineries, Part I and II; in: Turbomachinery Blade Design Systems; Von Karman Institute; Feb. 1999; Brussels.

[6] J.M. Anders and J. Haarmeyer; A Parametric Blade Design System, Part I and II; in: Turbomachinery Blade Design Systems; Von Karman Institute; Feb. 1999; Brussels.

[7] R. Dornberger, Optimization Environment Using Soft-Bounded Constraints, 3rd WCSMO, Buffalo, 1999.