Application of Free Form Deformation Techniques in Evolutionary Design Optimisation

Stefan Menzel, Markus Olhofer, Bernhard Sendhoff

2005

6th World Congress on Structural and Multidisciplinary Optimization
Rio de Janeiro, 30 May - 03 June 2005, Brazil

Application of Free Form Deformation Techniques in Evolutionary Design Optimisation

Stefan Menzel, Markus Olhofer and Bernhard Sendhoff

Honda Research Institute Europe GmbH
Carl-Legien-Str. 30
D-63073 Offenbach/Main, Germany
{stefan.menzel, markus.olhofer, bs}@honda-ri.de

1. Abstract
In the past decades evolutionary algorithms have been successfully applied to various design optimisation problems. It has been shown that the representation of the problem is crucial for a high performance of the method. One important aspect is the trade-off between the demands for a high geometric variability in shape generation and a minimum number of optimisation parameters. In this paper, free form deformation is coupled with evolutionary design optimisation in order to overcome this trade-off. Free form deformation methods have been initially developed in the field of solid and surface modelling allowing intuitive shape modifications by moving the control points of a lattice which encloses an arbitrary target geometry. Therefore it allows to decouple the complexity of the design to be optimised from the optimisation parameter. Evolutionary computation is a well established optimisation method for shape or design optimisation and belongs to the global stochastic optimisation methods. A population of possible designs is maintained and adapted from generation to generation via reproduction, variation operators like mutation and recombination and a selection method. The combination of free form deformation with evolutionary design optimisation of complex systems is illustrated by an optimisation of a three-dimensional high performance compressor airfoil. Hence the potential of coupling this kind of representation with a fluid mechanical design optimisation problem is emphasized. The optimisation parameters are defined by a lattice of control points which at the same time modify the geometry as well as the computational grid which is needed by the flow solver. Finally, the resulting shapes of the optimisation are presented and compared to the initial design to show the applicability of the proposed method.

2. Keywords: evolutionary design optimisation, free form deformation, representation, computational fluid dynamics

3. Introduction
The efficient development of optimal shape geometries plays a key role in every product design process. Computational simulation methods are more and more integrated as they promise to speed up the optimisation process and hence reduce costly product development time, as well as to support the designers and engineers in charge with any kind of information available. In the ideal case, a fully autonomous design optimisation environment evolves the final shape of the product without any human interaction. In contrast to classic deterministic approaches as gradient based methods evolutionary computation offers an interesting alternative for design optimisation. The possibility to develop new and unexpected designs which are not necessarily based on the heuristics of the engineers is provided by evolutionary algorithms which mimic the concepts of biological evolution.
From the point of view of evolutionary computation three important requirements can be mentioned which have to be fulfilled to realize such an autonomous environment. At first the core of the development process, namely the optimizer, has to be chosen adequately for the present problem. It should provide highly efficient algorithms and methods to push the performance of the design to its optimum in as few optimisation steps as possible even in case of complex quality functions. In the present paper an evolutionary strategy, more precisely the special variant using the covariance matrix adaptation (CMA-ES), is selected and will be the topic of Section 4. Due to the stochastic components and the use of a population of solutions evolutionary algorithms need more quality evaluations than other algorithms, but on the other hand they allow a global search and are able to overcome local optima. Therefore, the chosen algorithm shows a good trade-off between convergence speed and global optimisation. Furthermore, the considerable increase of available computing power especially on parallel architectures made the use of population based search methods, like the chosen evolutionary algorithm, more feasible and allows its application even in costly design optimisation.
Another requirement is a reliable, accurate and robust quality estimation of proposed solutions. In the present case, a CFD simulator is applied for modelling the behaviour of the evolved designs in real world situation. Its specifications are explained in Section 6 in more detail. Especially the feature of parallelizing the flow solver in a computer cluster environment allowed a time efficient optimisation.
The choice of an adequate shape representation as a third important component completes the optimal functionality of the autonomous optimisation environment. Using CFD for flow evaluation recommends a representation which is able to modify shape geometries and at the same time the computational grid for the flow solver. This allows to omit a costly re-meshing procedure. As a consequence, the required input data can be generated in a short period of time. Therefore, the methods of free form deformation (FFD) are applied to the present problem. Free form deformations promise a high geometric flexibility while keeping a low number of optimisation parameters as well as a comfortable way of mesh generation. The basics of this technique are the main focus of Section 5 and are finally illustrated by the example of a three-dimensional turbine blade optimisation in Section 7. It can easily be seen that the proposed methods can be extended to more complicated designs in a straightforward way. The term ‘complicated’ means in this context any kind of structure which also contains edges and ridges. If applied carefully free form deformations can even handle these kinds of geometric features and in such cases the possibility of omitting the re-meshing procedure is highly advantageous.

4. Evolutionary Design Optimisation

Evolutionary algorithms belong to the group of stochastic optimisation algorithms. They mimic the principles of Neo-Darwinian evolution, see e.g. [1], by applying operators for reproduction, mutation and/or recombination and selection. Prominent examples are Evolution Strategies (ES), Genetic Algorithms (GA) or Genetic Programming (GP), respectively. Among the advantages of evolutionary algorithms are robustness against noisy or discontinuous quality functions, the ability to escape from local optima and to enable global search. In the course of optimisation a population of possible solutions (e.g. a vector of continuous parameters, the objective variables) is adapted to solve a given problem over several generations. The adaptation occurs by variation of solutions contained in a population and by selection of the best solutions for the next generation. The variations can be classified as purely stochastic (usually called mutation) and combinatoric/stochastic (usually called recombination or in the context of genetic algorithms crossover). Schematically the evolution cycle is depicted in Figure 1, already with respect to the present turbine blade optimisation.

In this paper, a special variant of Evolution Strategies, the Covariance Matrix Adaptation (CMA), is applied which provides the advantage of a high convergence rate for real-valued problems compared to other evolutionary algorithms. This characteristic is of special importance in the case of very time consuming evaluations like CFD simulations. The successful application of this type of algorithm has been shown previously e.g. for a two-dimensional turbine blade optimisation [2]. The remainder of this section is organized as follows: An overview about the functionality of an Evolution Strategy is given in 4.1, followed by a brief discussion of the special type of the CMA-ES in 4.2.

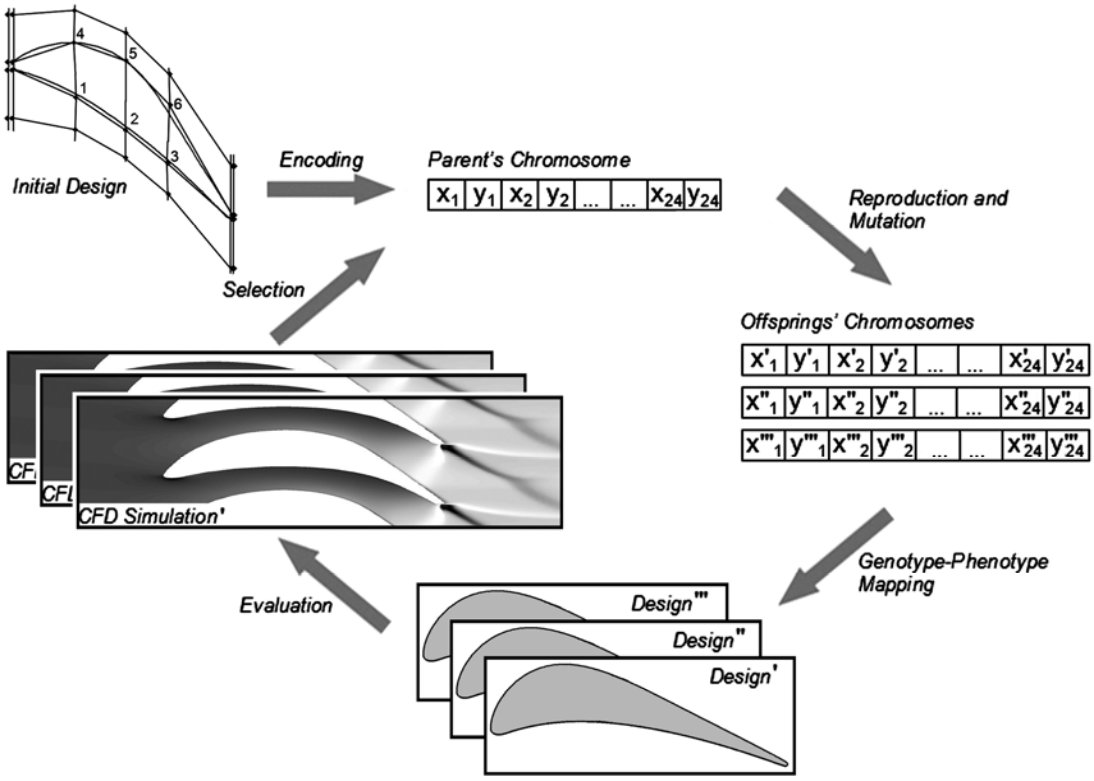

Figure 1. The generation cycle in evolutionary design optimisation

4.1. Concepts of Evolutionary Optimisation

A typical design optimisation procedure driven by an Evolution Strategy is shown in Figure 1 with respect to the turbine blade example. First, the initial blade design which is the starting point of the optimisation is parameterized and its significant parameters are determined. These parameters are encoded afterwards as the so-called genotype in the parent's chromosome. In the reproduction phase parents are chosen randomly and copied to form the offspring population. The generated offspring individuals which are exact copies of their parents are randomly mutated, without any recombination in our case. In Evolution Strategies this is done by adding a normally distributed random vector with zero mean as depicted in Figure 2. The shown curves determine the distribution of the offspring around the parent. The variances σ^2 of the normal distribution are called the strategy parameters of the search process and their values determine the width of the search and are comparable to a step size. The variances have to be adapted during the process to the local topology of the search space. This is realised by applying the same process of evolutionary optimisation as for the object parameters. This process which is usually referred to as self-adaptation is one of the key principles of an Evolution Strategy. It relies on a "second-order" or indirect selection of the strategy parameters which are part of each individual. In Figure 2 a bird's eye view of different normal distributions $N(\mathbf{a}_1,(\sigma_1^1, \sigma_1^1))$, $N(\mathbf{a}_2,(\sigma_2^1, \sigma_2^1))$ and $N(\mathbf{a}_3,(\sigma_3^1, \sigma_3^2))$ is shown for a two dimensional representation space. The "probability axis" would be perpendicular to the paper and the circles (ellipse) correspond to lines of equal probability. These lines are also called iso-density lines. Thus, if the parent vector is given by $\mathbf{a}_1$, approx. 46 % of all offspring will lie within the shown circle with radius σ_1^1.

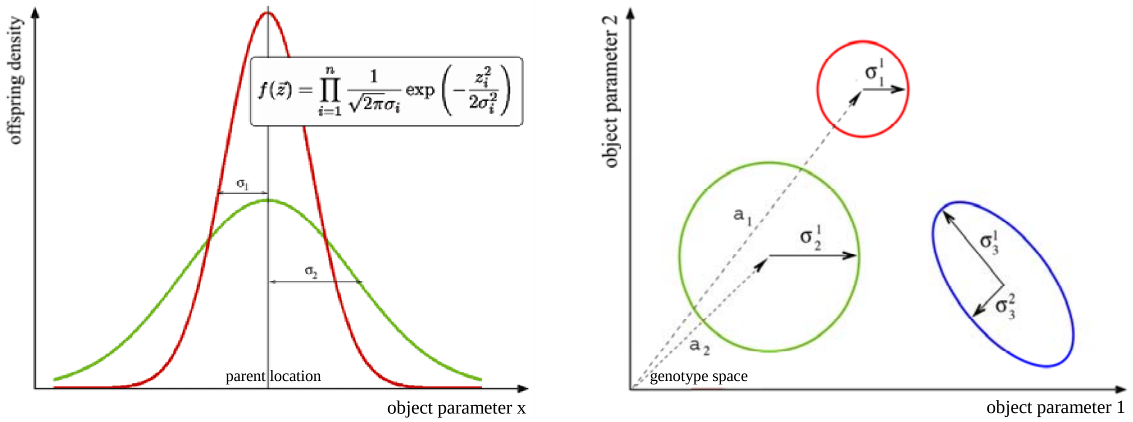

Figure 2. Normal distribution for various variances and Iso-density lines for three different normal distributions

In the standard evolutionary strategy only the diagonal elements of the covariance matrix are used, i.e. the mutation of the objective variables **x** is carried out by adding $N(0,\sigma_i^2)$ distributed random numbers z_i to each component x_i. The "step-sizes" σ_i are subject to mutations (log-normal distributed) both for each component separately (parameterised by τ) and overall (parameterised by τ'). Thus, the individual consists of both the objective and the step-size vector. Formally, the standard evolution strategy with self-adaptation can be expressed as follows: (see e.g. [3]).

$$\begin{aligned} \sigma_i(t) &= \sigma_i(t-1)\exp(\tau' z)\ \exp(\tau z_i) \\ \boldsymbol{x}(t) &= \boldsymbol{x}(t-1) + \tilde{\boldsymbol{z}} \end{aligned} \tag{1}$$

The strategy parameters determine the search strategy. A large step size indicates a more macroscopic search whereas a smaller one suggests a local search in a limited region of the search space. Finally, very small step-sizes also indicate the convergence of the optimisation. This behaviour will also be illustrated in the optimisation example in Section 7. For the adaptation of the strategy parameters several different methods have been proposed, the one which is chosen in the present optimisation will be mentioned below.

After the mutation of the offspring the genotype is mapped to the phenotype, i.e., the actual designs are created based on the encoded parameter set and prepared for the evaluation. In this phase a fitness value is assigned to each design which determines the performance with respect to realistic conditions. When optimizing a fluid dynamical system in a computational environment, typically a CFD calculation is performed and a selected result, e.g. a force, pressure or temperature, is taken as the controlling fitness value. Based on the calculated fitness the best offspring are picked according to the chosen selection method. For Evolution Strategies two different kinds of selection operators exist, one is called $(\mu+\lambda)$-selection, the other (μ,λ)-selection [4]. They differ in the way of choosing the offspring which form the parent generation of the next generation. In a $(\mu+\lambda)$-strategy the new parents are selected from the sum of the parents and the offspring of the current generation whereas in a (μ,λ)-strategy only the offspring are considered. In the first case a constant increase of the performance index is guaranteed because the best individual is always kept in every generation but at the same time there is the danger that the evolution gets stuck in a local optimum. After the selection of the best individual the process starts with a new reproduction cycle until the algorithm terminates when a certain stop criterion, e.g. a convergence threshold for the global step size, is fulfilled.

With respect to the turbine blade optimisation a (1,8)-strategy is applied, i.e. one parent and eight offspring individuals are considered and the selection takes place only on the offspring. In this context it should be noted that in evolutionary optimisation a standard population normally has a size of 50 individuals (or even more). But due to the expensive CFD-evaluation method in the present turbine blade optimisation the number of individuals had to be strongly limited. As an additional limitation, for this project totally 32 CPUs were available. Since each CFD simulation required 4 CPUs in parallel the size of the population was chosen to 8. To deal with these conditions, the CMA-ES had been chosen which had already proven its successful application for relative small population sizes.

4.2. Covariance Matrix-Adaptation (CMA-ES)

A crucial aspect for the performance of Evolution Strategies is the adaptation of the strategy parameters which highly influence the behaviour of the search and finally the convergence speed of the algorithm. As explained in the previous section the strategy parameters mainly define the variance(s) of the normally distributed random vector which is added to the object parameter during the mutation phase. In the simplest case, one single strategy parameter determines the variance of all elements of the random vector. The strategy parameter itself can be adapted by a self adaptation process. Therefore, the same process of evolution is applied to it as to the object parameter. Instead of one single strategy parameter the CMA algorithm adapts the whole covariance matrix of the distribution of the random vector. Furthermore, the path of the evolution is observed in order to predict favourable mutations based on the history of the optimisation. The theoretical concepts of the CMA-ES and its successful application in test scenarios as well as in a CFD optimisation environment have been discussed in various papers [2, 5, 6]. A detailed description of the algorithm can be found in [7] and an implementation in a software library in [8]. To summarize, there are mainly three features of the CMA-ES which are important. Firstly, the stochastic influence in the mutation step is reduced by introducing only one stochastic source which is used for modifying both, the object as well as the strategy parameters. Secondly, the so-called cumulative step-size adaptation is applied which extracts information from past generations to speed up and stabilize the adaptation of the strategy parameter. Thirdly, an adaptation of the full covariance matrix of the probability density vector takes place instead of independent variances for each single parameter. Therefore, correlated mutations can be realized which can significantly increase the convergence speed of the algorithm [4, 7].

5. Free Form Deformation as Representation

The choice of an adequate representation in evolutionary design optimisation depends on various requirements. It is important to keep the number of optimisation parameters as low as possible because a smaller parameter set implies a faster convergence of the optimisation process. The design subspace of interest should be defined by the smallest parametric description possible to guarantee a minimum of parameters. In most cases this conflicts with a highly flexible representation by which a wide variety of geometries can be realized and encoded. The fact that at the beginning of each optimisation the search direction and significant design modifications are unknown complicates a proper description even more. An adequate and efficient representation realizes this trade-off between minimality and completeness. A second aspect is the strong causality which is frequently quoted in this context [4]. The representation defines the properties of the map from the genotype space (the object parameter space) to the phenotype space (the space of possible designs). These properties are important because they influence the difficulty of the search process drastically. In this context, strong causality refers to the property that similar causes (in our case mutations on the genotype space) lead to similar effects (in our case the difference of the designs in phenotype space). Especially for evolutionary algorithms this property is meaningful since if similar mutations on the parameter set (the genotype space) imply similar changes on the geometric shape (the phenotype space) the self-adaptation of the step size is improved. Hence the probability for a fitness gain in the next generation is increased and the time for convergence reduces significantly.
There is one additional constraint on the representation which is specific to design optimisation problems where a CFD or FE calculation is performed. In this case in every generation for each offspring a computational grid has to be computed which is the basis for each CFD simulation. For the present problem free form deformation (FFD) has been chosen as representation as they promise to be a good trade-off for representing complex systems in this kind of design optimisation. They feature a good combination of minimality, completeness, causality and practicability. Since these methods are based on transfer functions instead of defining the geometry explicitly, the optimisation is decoupled from the geometric description of the design, e.g. via surfaces or splines, and hence the number of optimisation parameters is decreased. Additionally, they provide the possibility to modify the shape of one blade and the CFD mesh simultaneously reducing the time for each meshing process to a minimum. Even more, the adaptation possibilities of this kind of representation promise to allow a high degree of extensibility which is important with respect to future research work.

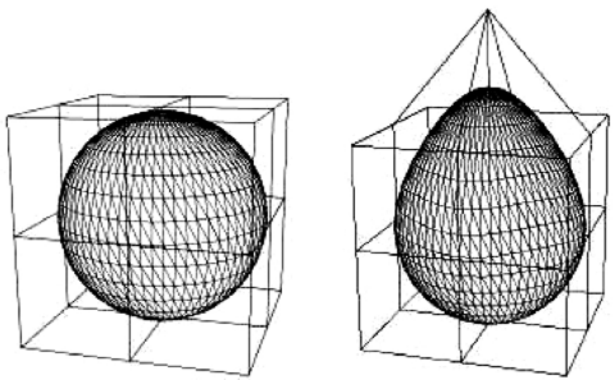

Figure 3. Free Form Deformation [9]

The concept of free form deformation has originally been introduced by Sederberg et al. [10] in the field of solid and surface modelling and has been extended and generalized by Coquillart [11]. Geometries are embedded within an arbitrary control volume which is defined by a set of control points and related knot vectors. By modifying these control points deformations can be applied to the geometry and the resulting shape is calculated based on a system of trivariate Bernstein polynomials or B-splines respectively as mathematical foundation. In the present optimisation, cubic B-splines have been used which allow a high degree of locality and flexibility. As already mentioned above, since each object within the control volume is described by transfer functions not only the surface of a body but also all grid knots within the control volume can be represented and shape designs as well as CFD grids can be evolved simultaneously so that a costly mesh generation process can be omitted [9]. This property is especially important for complex systems because the CFD mesh can be directly deformed while keeping its structural composition.
With respect to evolutionary design optimisation a free form deformation representation affects the development process in the following way. On the one hand, in the preparation phase of the optimisation the initial chromosome which is used for the first reproduction has to be prepared by encoding the initial geometry. On the other hand, directly before the evaluation phase the chromosomes have to be extracted and mapped to the geometric definition (genotype-phenotype mapping), i.e. the turbine blades as well as the computational grids needed for the CFD simulation have to be deformed based on the mutated control points.
Concerning the first aspect a lattice of control points has to be constructed which encloses either the whole object or the part of the object which will be modified during optimisation. In the second and most important step the geometry and grid coordinates have to be transferred into the parameter space of the lattice, a procedure which is also called "freezing". When freezing an object the u, v and w coordinates of the geometry in spline parameter space are calculated. This is usually done by Newton approximation which is regarded to be the fastest approach but it can also be done by similar gradient based methods [11, 12]. Finally, the positions of the

control points which are chosen for the deformation of the object are encoded in the parent's chromosome as start-up optimisation parameters. Practically, when speaking of a combination of this method with CFD, at first the initial design is meshed. In a second step the part of the design which is subject to optimisation and its surrounding part of the CFD mesh is embedded within an adequate control volume. Because of the concept of minimality the arrangement of the control points should be chosen carefully to keep the number of parameters as low as possible. Additionally, the objective control points should be fixed close to the surface of the design to maximize their influence. After the control points have been selected the coordinates in parameter space can be calculated by freezing the geometry and the CFD mesh. In Figure 4 a part of the control volume is depicted which is used in the blade optimisation which is explained in Section 7 in more details. For a better understanding it should be considered that the evaluation is done by CFD and hence the mesh which is situated between two neighbouring blades is of high importance. Because of this aspect the blade has been divided into two parts, the pressure and the suction side. Figure 4 shows the two dimensional layer at the hub section. The blue lines represent the pressure and suction side of one turbine blade whereas the grey area marks the CFD mesh region in this section. Both elements, i.e. blade surfaces and CFD grid, are embedded by the control volume and hence can be deformed in the pre-evaluation phase directly when the mutation of the chromosomes is finished.

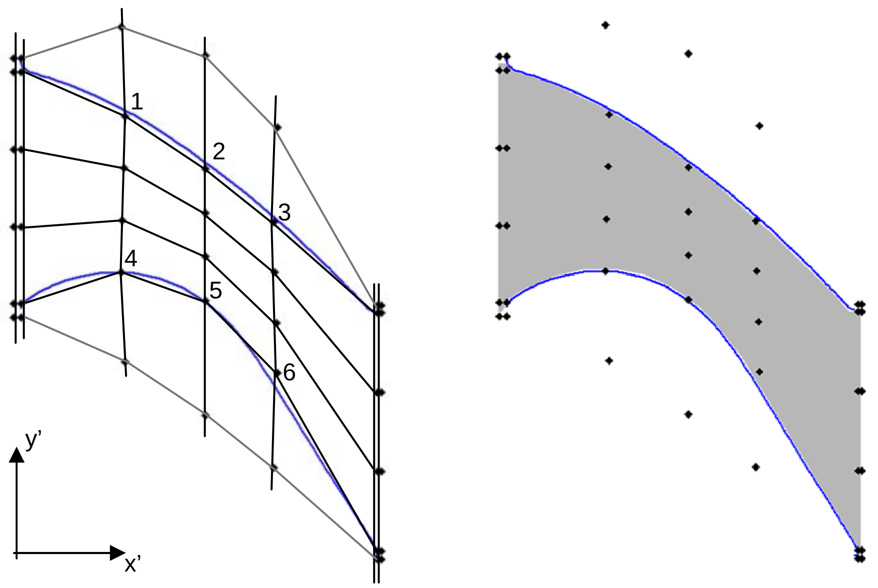

Figure 4. Set-up of the control volume (hub section)

After the mutation step the new coordinates of the control points are extracted from the chromosome and the deformations are transferred to the design geometry as well as to the CFD mesh. By solving the B-spline equations using the new spatial coordinates of the control points the x, y and z coordinates of the design surface and the grid knots are updated. In case of combining this method with CFD the new grid coordinates are calculated, too, and the new grid-file is generated by replacing the original grid coordinates with the updated ones. This guarantees that the mesh is modified while the structural composition is kept at the same time.

6. Performance Evaluation via CFD

An important while computationally expensive task is the calculation of the performance of each evolved blade by CFD simulation. For each generated blade the mesh has to be created and the flow to be solved numerically. The overall grid size of one simulation was 175 x 52 x 64 = 582400 cells and the time for the calculation of one blade took about five to six hours on a PIII Xeon, 2.0 Ghz node, depending on the convergence behaviour. As flow solver the parallelized 3D Navier-Stokes flow solver HSTAR3D [13] has been used which is perfectly adjusted to the present problem. The solver is parallelized for four CPUs resulting in a total usage of 8 individuals x 4 CPUs = 32 CPUs at the same time. The node communication was realized via the Parallel Virtual Machine (PVM) framework in a master/client configuration [14, 15].

In each generation the parameters which are stored in the chromosome of each individual are decoded and used for a transformation of the initial CFD mesh. The grid coordinates of the initial mesh are replaced by the updated ones so that the structural composition of the cells is kept. After calculating all eight CFD grids in each generation the slave processes are called and each of these processes starts the numerical solution of one of these meshes. The return values are taken as inputs for the fitness equation which is given in Section 7. In the present problem, the pressure loss, the outflow angle, solidity, blade thickness and trailing edge thickness had to be determined based on the flow results and geometry dimensions respectively. These values have been weighed and summed up to a performance index which is further used during selection.

7. Example: 3D Turbine Blade Optimisation

In this section, a description of the problem set-up and the chosen constraints is given which illustrates the combination of free form deformation and evolutionary algorithms in an autonomous turbine blade optimisation environment.

7.1. Problem Description

Subject of the optimisation is a turbine stator blade which is part of a gas turbine used in jets. An illustration of the turbine is shown in Figure 5. Around the hub of the turbine eight blades are equally distributed. Because of this low number this type of stator is called ultra-low-aspect-ratio (ULAR) stator and has been already target of design optimisation. For more detailed information on the turbine functionality and on a design optimisation approach using spline representation please refer to [16] and [17].

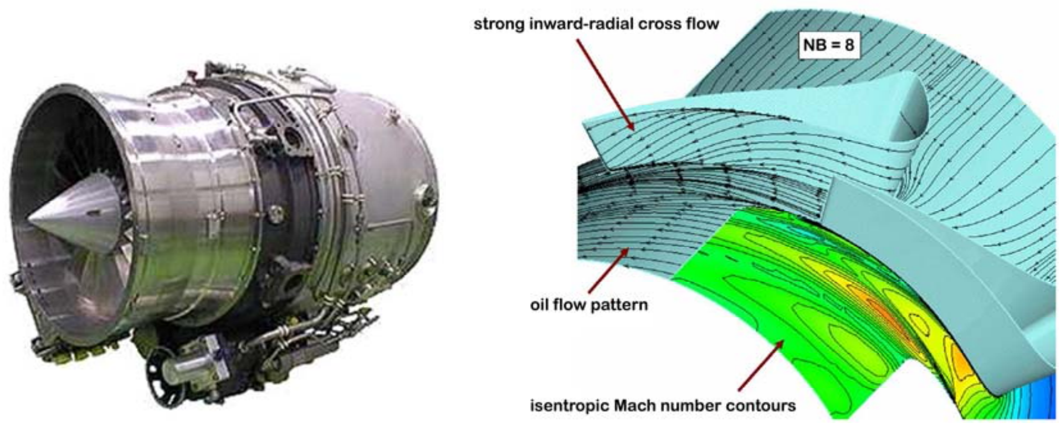

Figure 5. Gas Turbine and its fluid dynamics in one blade section [16]

7.2. Preparation of the evolutionary optimisation

The first step when it comes to numerical design optimisation is to extract the characteristic optimisation parameters from the representation of the given geometry. In terms of applying free form deformation, as mentioned above, a lattice of control points has to be constructed which encloses the target geometry. Fortunately, because of the rotational symmetry in the present problem only one of the eight turbine blade sections needs to be extracted for further evaluations. Additionally, it is important to mention that the control volume is set up between two blades, i.e. it spans from the suction side to the pressure side of two neighboring blades, because of the way the performance index is determined. In the present problem, the fitness value is calculated via CFD and accordingly the region between two blades has to be meshed for solving the numerical equations. This mesh including the blade's suction and pressure side as boundary layers was embedded by the lattice of control points to allow the simultaneous deformation of stator blade and computational grid. Finally, twelve control points have been taken from the lattice as optimisation parameters. To simplify the calculations and because of the bending of the turbine blade the global x, y and z coordinates of the design and of the knots of the CFD grid have been transferred to a local cylinder coordinate system x', y' and z'.

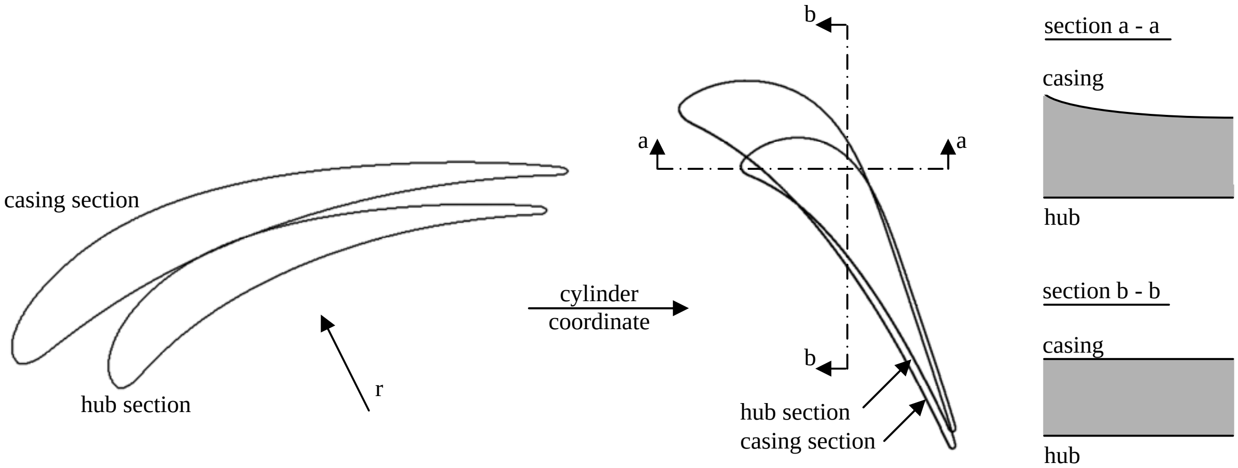

Figure 6. Transformation of global coordinate system to cylinder coordinates
(a-a and b-b denote the positions of the cross sections a - a and b - b respectively)

The lattice is fully three dimensional and a sample cross-section is depicted in Figure 4 and in Figure 7. As explained above, the CFD mesh plays an important role in the blade optimisation and therefore all grid knots which can be found in the CFD mesh between two neighboring blade surfaces have to be fully embedded in the control volume. As a consequence the deformations are applied on the one hand to the turbine blade surfaces and simultaneously to the CFD mesh so that a re-meshing process can be omitted. In the present example the re-meshing costs would have not been crucial for the optimisation because of the 'simple' structured grid consisting of parallelepiped volumes but if the grid structure is more complex containing edges and ridges the omission of a costly re-meshing procedure is very advantageous.

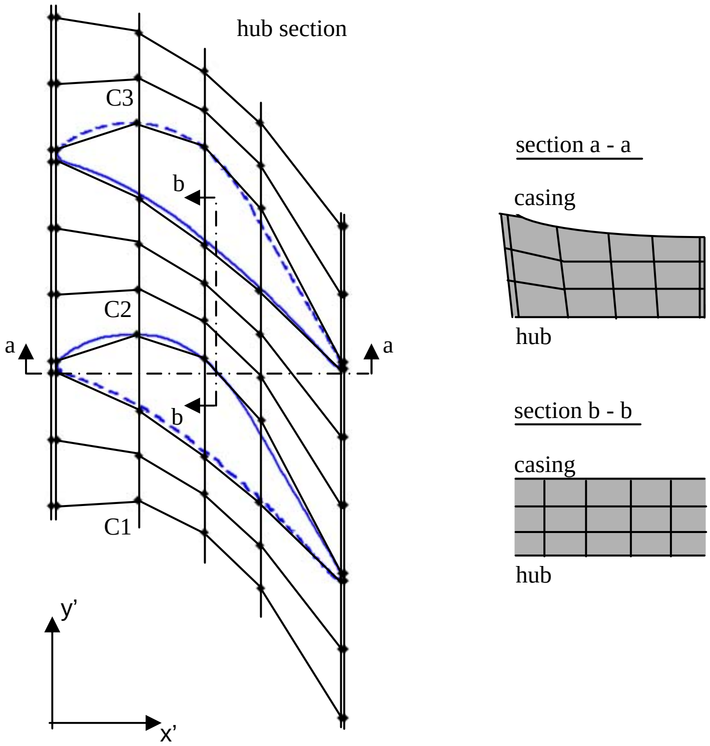

Figure 7. Embedding the turbine blade and the CFD mesh in an FFD lattice

The blue lines in Figure 4 and in Figure 7 illustrate the blade shape, the continuous upper line is the pressure side and the continuous lower one the suction side of two neighbouring turbine blades. The grey region marks the area of the knots and volume cells of the CFD grid. In Figure 7 two blade contours are depicted to show the position of the blades with respect to the control volume. As already mentioned, it should be kept in mind that not the shape of one whole blade is target to be embedded by the control volume but the passage between two blades where the CFD grid is found. In local x'-direction seven control points have been set. In local y'-direction the rotational symmetry influenced the number and positioning of the control points strongly. Each point repeats itself after four rows in y'-direction. Hence, the position of one specific control point is coupled to its 'friendly' point (in Figure 7, e.g. the points C1, C2 and C3) which can be found $\pm 2\pi r/8$ in y'-direction. Because of the mathematics of the cubic B-splines which form the basis in the free form deformation equations even a movement of the control point C3 influences the CFD grid in the passage where C2 is located. As a consequence, not only the four points of one passage but totally ten control points in y'-direction (the additional three points in positive and three points in negative y'-direction) had to be considered to define one complete blade section which is the target of the optimisation. In z'-direction four control points have been positioned which resulted in a total number of 7x10x4 = 280 control points.

Although all of these control points are important for freezing and deforming the geometry and the CFD mesh only 12 points had to be considered for optimisation. Six of these points are shown in Figure 4 for the hub section and another six points have been chosen analogously at the casing section. In total 24 parameters (x and y coordinates of the 12 points) have been considered in the evolutionary optimisation and were encoded in the parent's chromosome. To maximize the influence of these control points on the blade geometry they have been positioned as close as possible to the boundary layers of the blade so that the mutation of the control points provides a high impact on the design. In this first test scenario because of the small population size the number of optimisation parameters is kept as low as possible. As a consequence the movements of the 12 control points effect more or less global design changes of the blade which can also be seen in Figure 9. The present optimisation was motivated to analyze in how far these global design changes influence the performance. A higher degree of local changes can easily be realized by refining the control point lattice but with all its consequences on the optimizer, e.g. the increasing number of parameters.

Based on these control point settings the initial CFD grid and the blade geometry have been frozen, i.e. the coordinates of the grid knots have been calculated in spline parameter space. Afterwards the 24 parameters were encoded in the initial parent's chromosome and the optimisation was started. The first parameter sets were generated and extracted to calculate the new positions of the control points. To guarantee the rotational symmetry before evaluating the design in each generation the variations of the control points have not only been added to the twelve selected control points but also to the 'friendly' ones in each fifth row. The coordinates of the remaining control points have been linearly interpolated to have consistent deformations of the blade and of the CFD mesh. Based on these updated control point positions the free form deformation of the CFD grid and the blade geometry was performed and the CFD simulation started.

After the calculation has finished the result of the following fitness function is determined:

$$f = w_1 t_1 + \sum_{i=2}^{5} w_i t_i^2 \rightarrow \min \tag{2}$$

with:

- t_1 pressure loss
- t_2 difference to target outflow angle
- t_3 difference to target solidity
- t_4 difference to target minimum blade thickness
- t_5 difference to target minimum trailing edge thickness
- w_i weights for the different input data t_1, t_2, … t_5

As the main optimisation criterium the minimization of the pressure loss has been chosen. To keep the blade geometry within feasible constraints four additional values have been extracted from the CFD calculation and blade geometry respectively. Often an optimum of the fitness landscape is very close to the constraints and the boundary conditions have to be checked carefully. To avoid hard constraints which would directly exclude illegal designs weights have been introduced so that it was possible to determine a performance index for all evaluations. The parameters w_1t_2 to w_5t_5 are penalty terms and are calculated as follows. Before the optimisation a target range for the outflow angle, a target maximum solidity, a target minimum blade thickness and trailing edge thickness have been defined according to demands from other turbine parts and the used materials. After each evaluation these four geometric conditions have been calculated for the blade and if these values are not within the target range a penalty term is added to the pressure loss.

7.3. Results of the optimisation

The course of the fitness and the step size are depicted in Figure 8. A total number of 134 generations has been calculated resulting in an overall optimisation time of approx. six weeks. In the first ten generations a (1,6)-strategy has been used but was extended to a (1,8)-strategy starting with generation 11 because of the high variance of the fitness values. Generally, a population size beyond 12 to 14 is strongly recommended but could not be realized due to restrictions of the computational power. The fitness value of the initial blade is about 10.69 and is marked by the dashed line in the fitness graphs.

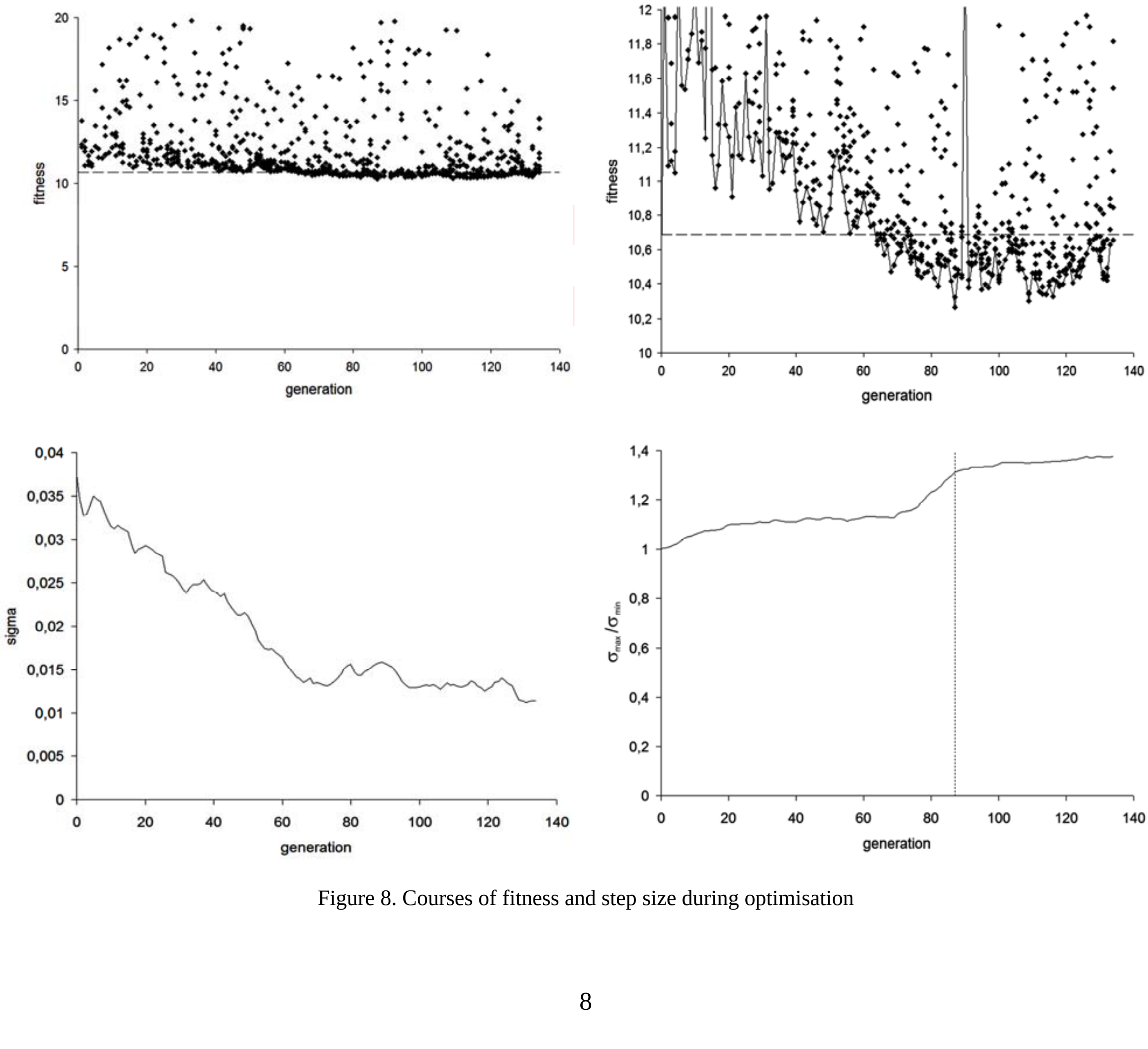

Figure 8. Courses of fitness and step size during optimisation

It can be seen that in the beginning the fitness value of the best design in each generation increases and reaches again the initial level at generation 60. After this point the best individual performs better than the initial one and stays within a range of 10,27 to 10,60. The best value of 10,27 is found in generation 87, which corresponds to a performance gain of 4 %. The course of the optimisation can also be analyzed by observing the step size. Right from the beginning the step size decreases and reaches a plateau at approx. generation 60. At the beginning large mutations were generated leading to an increase of the fitness value. This posed a serious problem due to the small population size of only 6 offspring. Therefore the population size was increased to 8 individuals starting with generation 11. Analyzing the results afterwards, it can be assumed that the step size has not been initialized properly and a lower value should have been chosen. But since the proper initial step size is unknown in most design optimisation it is more important to evaluate the behaviour of the optimizer, the evolutionary algorithm. Since the CMA-ES adjusts the step size by self-adaptation and decreases it in any generation the good behaviour and high robustness of this kind of Evolutionary Strategy is illustrated.
The ratio $\sigma_{max}/\ \sigma_{min}$ describes how correlated mutations in different dimensions are. As already described in Section 4.1 a value of 1 corresponds to a circle which does not prefer any kind of search direction (compare also Figure 2). This ratio increases and especially close to the minimum fitness a high gradient is identified. After reaching this optimum a plateau is formed and all evolved designs showed a lower performance because of a higher pressure loss and/or an additional penalty.
The initial blade and the shape of the best design of generation 87 are depicted in Figure 9 to visualize the changes which occurred.

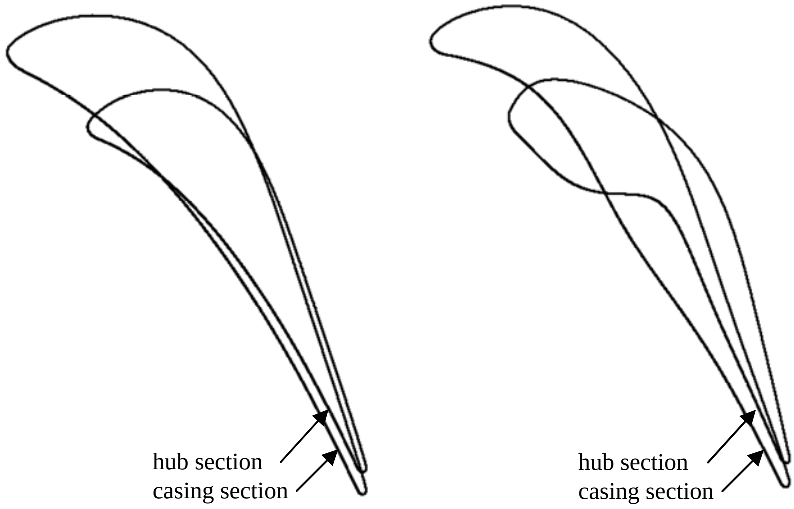

Figure 9. Initial and optimized shape of the turbine blade (hub and casing section)

8. Conclusions

In the present paper the techniques of free form deformation have been combined with evolutionary design optimisation. This combination is especially promising if coupled with CFD or FEM simulators for calculating the performance of the designs under realistic conditions. In contrast to a direct definition of the shape geometry, e.g. with the help of splines or spline surfaces, for free form deformation the geometry is embedded in a control volume and the shape modifications are calculated via transformation equations. Hence, for many complex systems the number of optimisation parameters can be reduced drastically. The additional possibility of deforming the shape geometry and the CFD/FEM mesh simultaneously is highly advantageous because the grid structure is kept and a re-meshing procedure is avoided. The good applicability of this representation and the comfortable integration in autonomous evolutionary design optimisation has been illustrated by the example of a turbine blade development.

9. Acknowledgements

The authors would like to thank M. Hasenjäger, T. Arima and T. Sonoda cordially for their support and advice on the turbine blade scenario.

10. References

1. Fogel D B. Evolutionary Computation: toward a new philosophy of machine learning New York, NY: IEEE Press, 1995
2. Sonoda T, Yamaguchi Y, Arima T, Olhofer M, Sendhoff B and Schreiber H-A. Advanced high turning compressor airfoils for low reynolds number condition, Part 1: Design and optimization. Proceedings of the ASME Turbo Expo, 2003
3. Schwefel H-P. Evolution and Optimum Seeking. John Wiley & sons. New York, 1995
4. Rechenberg I. Evolutionsstrategie ’94. Friedrich Frommann Verlag · Günther Holzboog, Stuttgart-Bad Cannstatt 1994
5. Jin Y, Olhofer M and Sendhoff B. A Framework for Evolutionary Optimisation With Approximate Fitness Functions. IEEE Transactions on Evolutionary Computation, Vol. 6, No. 5, pp. 481-494, October 2002
6. Jin Y, Olhofer M and Sendhoff B. Managing Approximate Models in Evolutionary Aerodynamic Design Optimization. Congress on Evolutionary Computation (CEC), volume 2, pages 592-599, Seoul, Korea, May 2001. IEEE Press
7. Hansen N and Ostermeier A. Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation, Vol. 9, No. 2, pp. 159-196, 2001

8. Shark – a modular C++ library for the design and optimization of adaptive systems. Open Source Software: http://sourceforge.net/projects/shark-project
9. Perry E C, Benzley S E, Landon M and Johnson R. Shape Optimization of Fluid Flow Systems. Proceedings of ASME FEDSM'00. 2000 ASME Fluids Engineering Summer Conference. Boston, Massachusetts, 2000
10. Sederberg T W and Parry S R. Free-Form Deformation of Solid Geometric Models. Computer Graphics, 20(4):151-160, 1986
11. Coquillart S. Extended Free-Form Deformation: A Sculpturing Tool for 3D Geometric Modeling. Computer Graphics, 24(4):187-196, August 1990
12. Piegl L and Tiller W. The NURBS Book. Springer-Verlag Berlin Heidelberg 1995 and 1997
13. Arima T, Sonoda T, Shirotori M, Tamura A and Kikuchi K. A numerical investigation of transonic axial compressor rotor flow using a low-Reynolds-number k-ε turbulence model. ASME Journal of Turbomachinery, 121(1), pp.44-58, 1999
14. Geist A et al. PVM: Parallel Virtual Machine – A Users' Guide and Tutorial for Networked Parallel Computing. Cambridge, Ma: The MIT Press, 1995
15. PVM: Parallel Virtual Machine. http://www.csm.ornl.gov/pvm/pvm_home.html
16. Hasenjäger M, Sendhoff B, Sonoda T and Arima T. Three dimensional Aerodynamic Optimization for an ultra-low aspect ratio transonic Turbine Stator Blade. ASME GT2005-68680
17. Hasenjäger M, Sendhoff B, Sonoda T and Arima T. Single and Multi-Objective Approaches to 3D Evolutionary Aerodynamic Design Optimisation. 6th World Congress on Structural and Multidisciplinary Optimization, Rio de Janeiro, Brazil, 2005