Self-Organizing Maps for Pareto Optimization of Airfoils

Dirk Büche[1], Gianfranco Guidati[2], Peter Stoll[2], and Petros Koumoutsakos[13]

[1] Institute of Computational Science, Swiss Federal Institute of Technology (ETH),
CH - 8092 Zürich, Switzerland
{bueche, petros}@inf.ethz.ch
http://www.icos.ethz.ch

[2] Alstom (Switzerland) AG, Segelhof, CH - 5405 Dättwil, Switzerland
{gianfranco.guidati, peter.stoll}@power.alstom.com
http://www.alstom.ch

[3] NASA Ames Research Center, Moffett Field, CA, USA
http://www.arc.nasa.gov

Abstract. This work introduces a new recombination and a new mutation operator for an accelerated evolutionary algorithm in the context of Pareto optimization. Both operators are based on a self-organizing map, which is actively learning from the evolution in order to adapt the mutation step size and improve convergence speed. Standard selection operators can be used in conjunction with these operators.

The new operators are applied to the Pareto optimization of an airfoil for minimizing the aerodynamic profile losses at the design operating point and maximizing the operating range. The profile performance is analyzed with a quasi 3D computational fluid dynamics (Q3D CFD) solver for the design condition and two off-design conditions (one positive and one negative incidence).

The new concept is to define a free scaling factor, which is multiplied to the off-design incidences. The scaling factor is considered as an additional design variable and at the same time as objective function for indexing the operating range, which has to be maximized. We show that 2 off-design incidences are sufficient for the Pareto optimization and that the computation of a complete loss polar is not necessary. In addition, this approach answers the question of how to set the incidence values by defining them as design variables of the optimization.

1 Introduction

Real-world application often include multiple and conflicting objectives. A solution to such a problem represents always a compromise between the different objectives. The set of the best compromise solutions is referred as the Pareto set, characterized that starting from a Pareto solution, one objective can only be improved at the expense of at least one other objective.

Evolutionary Algorithms (EAs) are a standard tool for Pareto optimization. EAs perform a population-based search, which allows approximating the Pareto front in a single optimization run by evolving the population in a cooperative search towards the Pareto front.

In Pareto optimization, work has been performed in the development of selection operators and especially fitness assignment techniques for Pareto optimization, while the recombination and mutation operators are often copied from simple single-objective algorithms and are non-adaptive over the optimization run.

In single objective optimization, however, the key component is the mutation operator. The operator exploits knowledge from the evolution to adapt a mutation distribution in order to focus on areas of promising solutions. The Covariance Matrix Adaptation (CMA) [5] embeds information about the path of successful mutations (evolution path) in a covariance matrix. This leads to a significant performance gain, compared to non-adaptive EAs.

The evolution path for multi-objective optimization is far more unclear than for single objective optimization, since the population converges in a cooperative search towards the Pareto front and not to a single optimum. Since different solutions in the population converge towards different locations on the Pareto front, the mutation distribution can not be described by one covariance matrix and has to differ between the solutions.

Thus, we introduce a new adaptation method for the mutation step size in Pareto optimization using the self-organizing maps (SOM) of Kohonen [6]. The method is inspired by the work of Milano *et al.* [8], who develop the first SOM for single objective optimization. The SOM is continuously trained on the current best solutions and thus is tracking the evolution path in a learning process. The SOM adapts the mutation step size such that it focuses on areas of promising solutions in order to generate an accelerated convergence.

The aerodynamic design of modern aircraft wings as well as rotor blades from various areas as turbo machinery, helicopters, and wind energy plants relies significantly on the design of 2D cuts (profiles), which are then stacked to a 3D wing or blade. This simplification omits 3D flow effects but is adequate for the design due to the large aspect ratios between span and chord. Real 3D calculations are often performed for the design assessment, after the actual design process. The profiles can be designed individually or can be taken from a profile family.

The aerodynamic performance of a profile is mainly characterized by the thermodynamic losses at the design operating condition and by the operating range. The operating range can be described by the possible variation of the inlet flow angle from the design condition (incidence variation) until separation or stall occurs. These two characteristics are conflicting, thus requiring a set of profile designs for different compromises manifested on a Pareto front.

This paper is organized as follows: First, the principles of EAs for Pareto optimization are introduced, followed by a description of the SOM and the modifications for the learning in an EA. The new algorithm is illustrated on a test problem. Finally, an automated loop for the Pareto optimization of an aerodynamic profile concerning losses and operating range is described. The loop comprises the new optimization algorithm, a profile generation tool and a CFD analysis tool. The properties of the resulting profiles are discussed.

2 Multi-Objective Evolutionary Algorithms

The selection operator is often considered as a key operator for multi-objective evolutionary algorithms (MOEAs). It consists of the fitness assignment and the selection mechanism itself. The dominance criterion in combination with niching techniques is most popular for the fitness assignment [11], in order to select on average the less dominated solutions and preserve diversity in the population, respectively. Another key element is elitism [11], a technique of storing always the current best solutions in an archive. For a multi-objective problem, the elite solutions are the current nondominated solutions. The archive is then participating in the selection process.

The Nondominating Sorting Genetic Algorithm (NSGA-II) [2] and the Strength Pareto Evolutionary Algorithm (SPEA2) [12] are two common representatives of MOEAs and implement all of the previously stated techniques.

These algorithms, however, do not describe a mutation or recombination operator. To compare the performance on continuous problems, the authors of SPEA2 and NSGA-II use the polynomial distributed mutation and the simulated binary crossover of Deb *et al.* [1]. Both methods do not implement any learning process, so they do not exploit any knowledge from the evolution path.

2.1 Self-Organizing Maps

Self-organizing maps (SOM) [6] define a mapping of a highly dimensional input space $\mathbb{R}^n$ onto a regular lattice of s reference vectors (neurons). The lattice contains a fixed m-dimensional connectivity between the neurons, which is usually of lower dimension than the input space. Figure 1 illustrates a SOM with 25 neurons and a two-dimensional quadrilateral lattice, i.e. $n = 2$, $s = 25$, and $m = 2$. A reference vector $w_i \in \mathbb{R}^n$ is associated to each neuron i.

The response of the network to an input $x_j \in \mathbb{R}^n$ is defined as the best matching neuron c:

$$c(x_j) = \arg\min_i \{||x_j - w_i||\} \tag{1}$$

The SOM can be trained on a set of input data x_j. To each x_j the response c is computed and all SOM neurons are *updated* so as to become closer to the input x_j by the update rule:

$$w_i^{new} = w_i^{old} + h(c,i) \cdot (x_j - w_i), \quad i = 1...s, \tag{2}$$

where $h(c,i)$ is the so-called *neighborhood kernel*, defined so as $h(c,c) = 1$, $h(c,i) \geq 0 \; \forall \; w_i$. We use a kernel, which is known as *bubble kernel*, defined by:

$$h(c,i) = \begin{cases} \alpha \text{ , if } r(c,i) < r_0 \\ 0 \text{ , otherwise} \end{cases} , \tag{3}$$

where α is the learning rate, $r(c,i)$ is the Euclidean distance between node c and i in the lattice and r_0 defines the bubble size. The neighborhood function allows approximating a given distribution in an ordered fashion, by preserving neighborhood relationships. One update with all input dates is referred to as one training epoch.

2.2 Self-Organizing Maps for Multi-objective Evolutionary Algorithms

Here we use the SOM for the approximation of the Pareto front. To this aim we set connectivity m of the lattice to one dimension less than the objective space, that is the same dimension as the Pareto front; also, since the SOM is defined in design space, the dimension n of its reference vectors is equal to the number of design variables. The SOM is trained on the current parent population of the optimization process in order to approximate the parent population in an ordered fashion. Any *selection operator* like SPEA2 or NSGA-II can select the parent population.

We define a *recombination operator* by using the SOM. The SOM is trained on the design variables of the parent population. Thus, choosing a random point within the area that is covered by the SOM represents an intermediate recombination of the parent population. The recombination procedure chooses randomly a simplex of adjacent neurons in the lattice, and generates a recombined point u from a uniform probability distribution within the simplex (Figure 1).

In addition, a *mutation operator* is defined in order to generate points outside the area covered by the SOM. Normally distributed random numbers are added to the new point u by:

$$u_k \leftarrow u_k + \frac{\sigma}{\sqrt{n}}\mathrm{N}(0,1), \quad k = 1...n, \tag{4}$$

where σ is the step size and is set equal to the Euclidean length of a randomly chosen edge of the simplex. This leads to an adaptive step size over the optimization process, since the SOM is adapting from a random initialization to an ordered approximation of the Pareto front. The step size differs for all possible simplexes of the SOM.

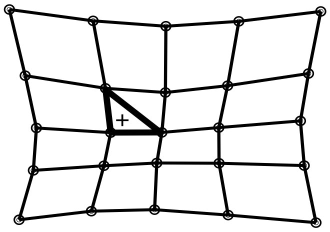

Fig. 1. SOM with 25 neurons [circles] and 2D quadrilateral lattice [thin lines]. A random simplex of adjacent neurons is created [bold line]. Within the simplex a uniformly distributed random point [plus symbol] is generated.

2.3 Experimental Results

The performance of the SOM-MOEA is analyzed for the two-objective test function of Fonseca and Fleming [4]:

$$f_{1/2} = 1 - \exp\left(-\sum_{i=1}^{n}\left(x_k \pm \sqrt{1/n}\right)^2\right) \tag{5}$$

with $x_{1\ldots n} \in [-1,1]$. The exact Pareto front is obtained for $x_{1\ldots n} = t, \quad -\sqrt{1/n} \leq t \leq \sqrt{1/n}$. The number of design variables is set to $n = 10$. An optimization run is started with the SOM-MOEA and a population size of 60 individuals. A simple selection operator is used, which selects only the current nondominated solutions in an elitistic fashion. In order to keep diversity within the selected set, the clustering algorithm of SPEA2 is used, allowing a maximum number of 30 nondominated solutions.

A one-dimensional SOM is initialized with $s = 20$ neurons, a learning rate $\alpha = 0.05$ and random values for the reference vectors w_i. After each generation, the SOM is trained with 30 training epochs on the selected set. The initial population is randomly generated. Figure 2 shows the initial population and SOM for two dimensions of the design space and for the objective space. Consider that a simplex for this SOM is a straight line.

The optimization run is started computing in total 3.000 solutions. The final population is shown in Figure 3. The figure shows that the SOM is aligned along the analytical Pareto front in design space, and the objective values of the final population are well distributed along the Pareto front. The step size σ of the mutation in Equ. 4 is related to the length of a simplex, i.e. for this one-dimensional network it is equal to the distance between two adjacent neurons. The ratio of the initial and final step size is equal to the distance between adjacent neurons of the SOM in Figure 2 and Figure 3.

3 Automated Design of Aerodynamic Profiles

We consider the automated profile design in the context of a constraint Pareto optimization. An optimization loop is implemented comprising the SOM-MOEA, a profile generation tool and a computational fluid dynamics (CFD) analysis tool. The profile generator describes the profile by a set of Bezier splines. The spline parameters are coded in a set of engineering parameters, specifying e.g. the profile length, the nose and trailing edge radius, the curvature distribution, etc. Subdividing the profile in several splines is common in profile design [7] [10]. The transformation of the spline parameters to engineering parameters simplifies the comparison and illustration of different profile parameter sets.

The flow analysis is performed with MISES [3], a quasi 3D computational fluid dynamics (Q3D CFD) solver, which solves the Euler equation with an integral, viscous boundary layer formulation. It takes into account the change in the streamline thickness along the profile (quasi 3D).

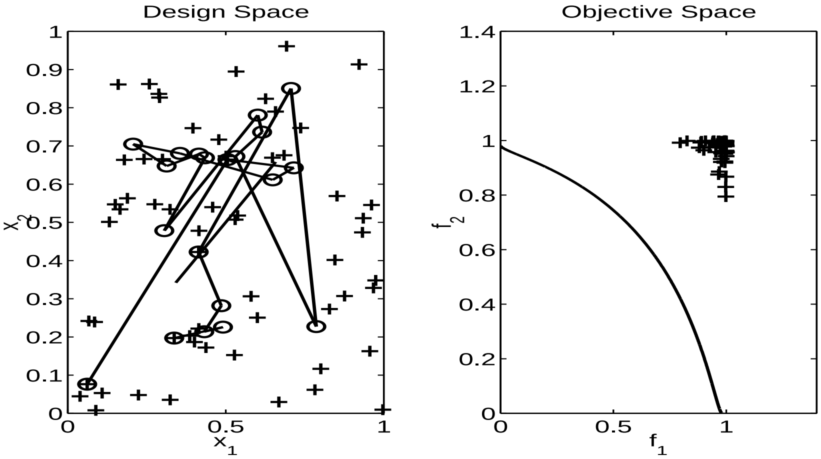

Fig. 2. Initialization of the SOM-MOEA for the two-objective problem of Fonseca and Fleming with 10 design variables: Random population [crosses], random 1-dim. SOM [connected circles] and analytical Pareto front [line] in a 2-dim subspace of the design space and in the objective space.

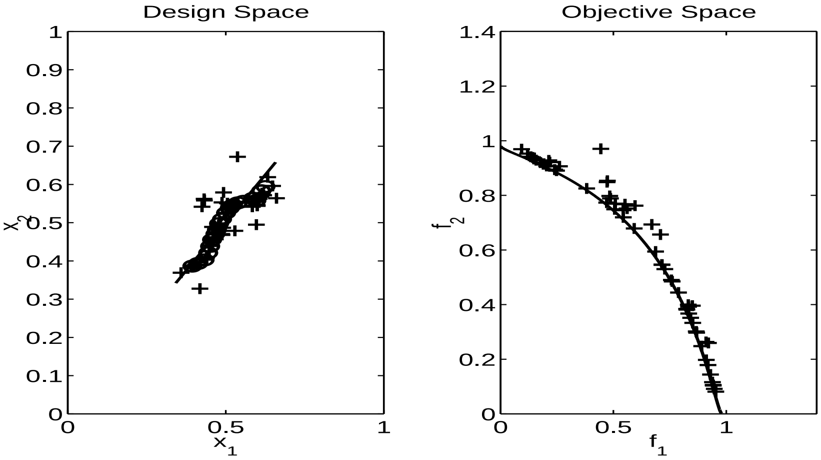

Fig. 3. SOM-MOEA after 50 generations (3000 evaluated solutions). The SOM alignes in the design space along the Pareto front.

3.1 Objective Functions

An aerodynamic profile should be optimal concerning its performance at design condition as well as for off-design conditions. This can be achieved by different approaches.

One approach is to follow a design philosophy, which can be specified by e.g. a Mach number or pressure distribution, maybe in conjunction with a non-dimensional shape factor of the boundary layer like H_{12}. Since the behavior of the design philosophy for certain profiles under off-design conditions is known, it is assumed to be sufficient to match the philosophy at design condition. This approach is used in the manual design process, the inverse design process (see e.g. [7], [10]) or in a direct optimization process (see e.g. [9]) and the quality of the result relies directly on the quality of the considered philosophy.
A second approach is the calculation of various incidences in order to approximate the loss polar of the profile as given in Fig. 4, which specifies the behavior of the profile over the complete operating range. A disadvantage is the large number of flow calculations, which are needed to specify the polar as in the optimization of [7]. Furthermore, there is the problem of how many incidences should be computed and for which values.
Our preference is on the second approach, since the first does not allow discovering new design philosophies and is difficult to apply to transonic or 3D flows. In addition, we present two modifications in order to improve the second approach. First, we formulate a Pareto optimization problem in order to obtain a family of profiles and not one single compromise solutions. The family represents all compromises between the conflicting objectives of minimizing the losses at the design condition and increasing the operating range. Second, we do not compute the complete loss polar and show that it is sufficient to compute 3 different incidences in order to assess a profile. The 3 calculations are performed for the design condition, i.e. 0° incidence and for one positive incidence I_1 and one negative I_2. The key concept is to define I_1 and I_2 variable by a free multiplier θ:

$$I_1 = 1.0 \cdot \theta \tag{6}$$

$$I_2 = -0.8 \cdot \theta \tag{7}$$

This definition takes into account that the positive incidence I_1 is more critical for stall than I_2. The incidence multiplier θ is an additional design variable.
The profile losses for the 3 incidences are summed to the first objective function f_1. For small values of θ, the losses are computed at small incidences. An optimization for small values of θ leads profiles which have minimal losses in the vicinity of the design condition, while for large values of θ, profiles are optimized for a large incidence range. Thus, θ is not only used as free design variable, but also as second objective function f_2. To both objectives penalties are added for violated constraints and the exact objective functions are given by:

$$\max(f_1), \quad f_1 = \theta - p_1 \tag{8}$$

$$\min(f_2), \quad f_2 = \sum_{i=1}^{3} (l_i) + p_1 + p_2 + p_3 + p_4, \tag{9}$$

where l_i is the profile loss for the incidence i and p_1 to p_4 are 4 penalties, which are non-zero, if the corresponding constraint is violated.
p_1 is a penalty for the convergence of the CFD solver and is equal to the number

of failed incidence calculations. Especially for large incidences, the convergence may fail due to flow separation. p_2 is a linear penalty on the deviation of the exit flow angle β to the design exit flow angle β_{design} at design condition, if the deviation exceeds an absolute value of $|\beta - \beta_{design}| > \delta\beta$. p_3 is a linear penalty on the profile area A, if A is smaller than the minimal area A_{min}. The minimal area is defined by the mechanical forces on the profile and the stress limit.
The free design variables are the parameters from the profile generator and the incidence multiplier θ. In total there are 15 design variables.

3.2 Optimization Results

An optimization run is performed for a profile design at an inlet Mach number of 0.67, a desired flow turning of 12° and $\delta\beta = 0.1°$. The SOM-MOEA of Sec. 2.3 is used, except the maximal size of the selected set is increased to 50. In total 10.000 solutions are evaluated. Among all evaluated solutions, 5.461 solutions do not violate any constraints and generate a Pareto front of 283 solutions (Fig. 4). Consider that the incidence multiplier is to be maximized and the losses are to be minimized. The Pareto front underlines the conflict in optimizing the two objectives. For small incidence multipliers, the losses are low, since all 3 incidences are computed almost at the design point. For large incidence multipliers, the loss increases for two reasons. First, the flow is computed at larger incidences leading to higher losses and second, the profile losses are higher at the design condition, since the design has to be more robust for converging at the high incidences. Two Pareto solutions are marked in the figure and their loss polar is given in Fig. 4. The minimal losses are at about 1.4%. The attainable operating range is considered to be bounded by the double of the minimal losses [7]. Solution A contains the smaller incidence multiplier and the loss polar shows lower losses close to the design incidence than solution B, but comprises a smaller operating range. For solutions A and B, the operating range is about 14.4° and 15.5°, respectively. Both polars are characterized by a smooth and continuous increase of losses over the absolute incidence. This indicates a soft stall behavior. Fig. 5 contains the profile shape. Solution A shows the smaller nose radius as well as the smaller maximal thickness.

4 Conclusions

A self-organizing map (SOM) is introduced as mutation and recombination operator for Pareto optimization. The network is actively learning from the current nondominated solutions. The SOM comprises the principle of cooperative search by interpolating the current nondominated front, thus it is sharing information about successful design variables values along the Pareto front. The mutation step size is related to the distance of neighboring neurons in the SOM. It varies within the network and is adaptive over the optimization run. The SOM represents a first step in the direction of developing mutation and recombination operators which are especially designed for Pareto optimization and which are

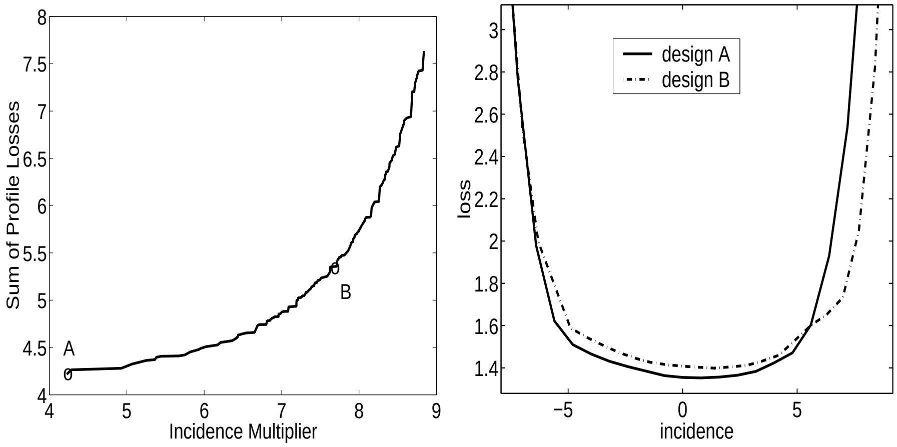

Fig. 4. Pareto front [left] for the profile optimization, and loss polar [right] for two selected Pareto solutions.

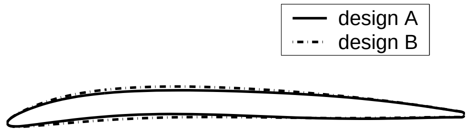

Fig. 5. Profile shape for the two selected Pareto solutions.

able to learn from the evolution in order accelerate the convergence.
The second part describes a formulation for the Pareto optimization of compressor profiles for minimal losses and maximal operating range, which operates with a minimal number of incidence calculations. The key feature is the definition of a multiplier for the incidences, which is used at the same time as design variable and objective function. This optimization introduces the concept of generating profile families in a single cooperative optimization run by using a Pareto optimization algorithm.

Acknowledgments

The first author would like to acknowledge support from the Commission for Technology and Innovation, Switzerland (Project 4817.1) and Alstom (Switzerland) AG.

References

1. Deb, K., Agrawal, R.B.: Simulated binary crossover for continuous search space. Complex Systems, No. 9 (1995) 115–148

2. Deb K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist nondominated sorting genetic algorithm for multi-objective optimization: NSGA-II. Parallel Problem Solving from Nature VI Conference (2000) 849-858.
3. Drela, M., Youngren, H.: A User's Guide to MISES 2.53. MIT (1998)
4. Fonseca, M.C., Fleming, P.J.: Multi-objective genetic algorithms made easy: Selection, sharing and mating restrictions. Proceedings of the 1st International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, London, UK, (1995) 45-52
5. Hansen, N., Ostermeier, A.: Completely Derandomized Self-Adaptation in Evolution Strategies. Evolutionary Computation, Vol. 9, No. 2 (2001) 159-195
6. Kohonen T.: Self-organizing maps. Springer series in information sciences, 3rd ed. (2001)
7. Köller U., Mönig, R., Küsters, B., Schreiber, H.-A.: Development of Advanced Compressor Airfoils for Heavy-Duty Gas Turbines, Part I: Design and Optimization. ASME Journal of Turbomachinery, Vol. 122, No. 3 (1999) 397-405
8. Milano, M., Schmidhuber, J., Koumotsakos P.: Active Learning with Adaptive Grids. International Conference on Artificial Neural Networks, Vienna, Austria (2001)
9. Naujoks, B., Willmes, L., Haase, W., Bäck, T., Schütz, M.: Multi-Point Airfoil Optimization Using Evolution Strategies, ECCOMAS 2000, Barcelona, Spain (2000)
10. Trigg, M.A., Tubby, G.R.,Sheard, A.G.: Automatic Genetic Optimization Approach to Two-Dimensional Blade Profile Design for Steam Turbines ASME Journal of Turbomachinery, Vol. 121, No. 1, (1999) 11-17
11. Van Veldhuizen, D.A., Lamont, G.B.: Multiobjective evolutionary algorithms: Analyzing state-of-the-art, Evolutionary Computing, Vol. 8, No. 2., MIT Press, Cambridge, MA (2000) 125-147
12. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. EUROGEN 2001, Athens, Greece (2001)