# SYSTEMATIC ANALYSES OF MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS APPLIED TO REAL-WORLD PROBLEMS USING STATISTICAL DESIGN OF EXPERIMENTS

J. Mehnen[1], Th. Michelitsch[1], Th. Bartz-Beielstein[2], N. Henkenjohann[3]
[1]Department of Machining Technology, University of Dortmund, Germany
[2]Department of Computer Science, University of Dortmund, Germany
[3]Department of Statistics, University of Dortmund, Germany

**Abstract**
Solving multi-objective optimization problems is a challenging task that demands efficient software tools and systematic analytical approaches. In this paper two evolutionary multi-objective optimization algorithms – namely the evolution strategy (ES) and the NSGA II – are applied to two complex real-world problems. The parameter settings of the evolutionary algorithms have been chosen and optimized according to statistical design plans. A new ranking method for measuring the quality of pareto-fronts is introduced. The layout of mold temperature control systems and the scheduling of elevators show typical complexity aspects that are necessary to illustrate a systematic approach of solving real-world multi-objective optimization problems.

**Keywords:** Multi-objective evolutionary algorithms, Statistical design of experiments, Pareto-ranking method, Mold temperature control, Elevator supervisory group control.

## 1 INTRODUCTION

Evolutionary algorithms (EA) are general purpose methods for solving complex optimization problems. These meta-heuristics depend on optimal parameter settings that are usually not easy to find. Problems in industry often differ strongly from the well known 'toy'-functions that are typically used in the EA community. The complexity of time consuming real-world problems limits the number of experiments that can be performed to tune the parameter settings of an EA. Furthermore, typical real-world problems are multi-objective, i.e. they are combinations of several quality demands that may also contradict each other. In this article a systematic and constructive approach is introduced to handle these difficult problems by means of EA and statistical design of experiments (DoE). Classic DoE requires expressive scalar quality measures [1]. Providing these values for multi-objective problems is usually not straight forward [2]. Therefore, a new ranking method for measuring Pareto fronts is introduced.

The paper has the following structure: First, two selected real-world problems are described – the mold temperature control problem and the elevator supervisory group control problem. A description of two multi-objective evolutionary optimizers (an aggregating Evolution Strategy (ES) and the Nondominated Sorting Genetic Algorithm II (NSGA II)) is presented. A new multi-objective ranking method is introduced. Some fundamentals of the classical statistical design of experiments are given. Finally of the article the results of the experimental design are discussed to get an impression of the behavior of the algorithms.

## 2 MOLD TEMPERATURE CONTROL SYSTEMS (MTCS)

The optimal layout of mold temperature control systems is decisive for maximizing the product quality and for minimizing the production cycle times in injection molding and die casting [3]. Mold temperature control systems can be described as a network of deep hole bores that penetrate a casting tool.

The bores lead a water-oil emulsion near the die surface to heat up, cool down or stabilize the temperature of the tool. A well balanced die surface temperature can be established by the structure of the mold temperature control circuits. The layout of the mold temperature control system is usually designed by human expert knowledge. Objective and fast measures are necessary to estimate the temperature balance in a casting tool to support the complex decisions made by the experts. Here, an automatic optimization strategy is introduced that uses efficient temperature estimation models and intelligent geometrical encoding strategies [4, 5].

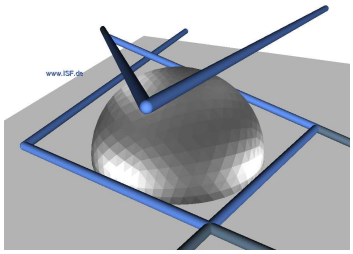

Figure 1 : Polyline description of a MTCS applied to a sphere.

A new polyline approach models efficiently the bore structures (see Figure 1). A sequence of 3D-vertices of the polyline defines uniquely a circuit of bores. A vector composed of the vertex co-ordinate values is use to describe a circuit (phenotype). The real-value vertex vector is the input (genotype) of the optimization strategies.

Several effects, restrictions and demands are essential to all MTCS: the temperature on the die surface depends on the geometric position and structure of the bores; each bore is interconnected to another bore to form a well defined circuit that starts and ends in two points at the tool surface; all bores must not intersect the tool surface, the die surface nor each other; the length and number of bores influence the manufacturing costs and stability of the tool negatively; an intensive but also well distributed temperature distribution has a positive effect on the temperature balance in the tool.

FEM models are exact and flexible but often very time consuming. Here, an innovative radiation model is used as a surrogate for generating fast evaluations of the temperature effects. Each bore is modeled as a "neon lamp" that illuminates the die surface. The light intensity is the analogon to the heat radiation of a bore. The die surface is described by triangulations. The radiation effect is calculated via the sum of influences that the bores between two consecutive points $(P_i, P_{i+1})$ have on each triangle $\tau_j$ with central point $M_j$. The influence on $\tau_i$ is calculated via a closed solution of the integral:

$$\tau_{i,j} = |P_i - P_{i+1}| \int_0^1 \frac{1}{(P_i + t \cdot (P_{i+1} - P_i) - M_j)^2} \, dt \quad (1)$$

The aggregated temperature effect $f_t$ is composed of three parts: the arithmetic average sum over all illumination influences of all bores on all triangles (describing the absolute cooling effect), the temperature mean distribution (modeling the uniformity of the illumination) and the normalized maximum effect value.

In the polyline approach the multi-objective fitness function has been modeled via a special multiplicative aggregation method or via a Pareto approach. The aggregated fitness $f$ is a function of the standardized restrictions $f_{pen}$, the manufacturing costs $f_l$ (i.e. the length of the bores), and the temperature effect $f_t$:

$$f = f_t \cdot (1.0 + f_{pen}) \cdot (1.0 + d \cdot f_l) \quad (2)$$

$$f_{pen} = a \cdot f_{bc} + b \cdot f_{sc} + c \cdot f_{ot}$$

The allover penalty $f_{pen}$ is a linear combination of the number of collisions between two bores $f_{bc}$, the factor $f_{sc}$ describes the collision between the bores and the tool surface and $f_{ot}$ is a penalty for the bores that reach out of the tool. $a$,$b$,$c$,$d$ are arbitrary weighting factors.

## 3 ELEVATOR SUPERVISORY GROUP CONTROL PROBLEM (ESGC)

High rise buildings require efficient elevator systems. The elevator group controller determines the floors where the elevators should go to and it assigns elevator cars to customers. This assignment is based on a heuristic or policy and should be optimal with respect to many (partially conflicting) goals, i.e. waiting times, energy consumption, maintenance costs, security, service intervals, or comfort.

Bartz-Beielstein and Markon proposed a simplified elevator group control simulation model: the sequential ring (S-ring). It abstracts from specific building configurations and enables the comparison of different heuristics. As it can be reproduced easily it is well suited as a benchmark problem.

The state of the system is mapped to a binary string and the system's dynamic is given by a state transition table. Only the number of floors $n$, the customer arrival rate $p$ and the number of elevator cars $m$ in the system have to be specified to perform a simulation. Despite the model's simplicity it is hard to find an optimal policy, since $N$, the number of different system states grows exponentially with the number of floors $n$ and $m$, the number of elevator cars: $N = \binom{n}{m} 2^m$.

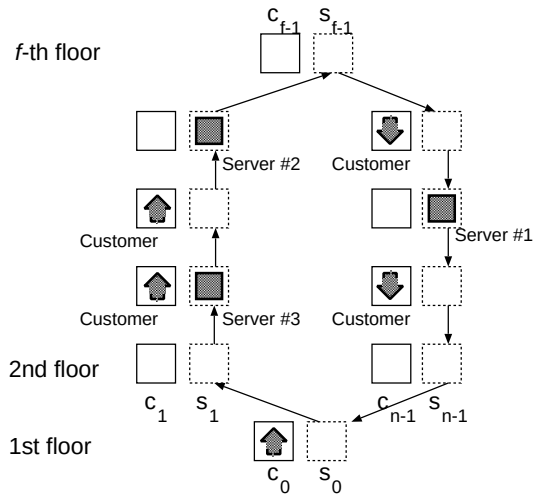


Figure 2 : S-ring elevator model.

The S-ring model determines approximatively the passenger waiting time. This is the time measured from the instance a passenger registers a call to the instance the passenger enters the elevator car. Different traffic patterns occur during the day, i.e. up-peak traffic in the morning, interfloor traffic during the day in office buildings. The S-ring model considered in the remainder of this article uses two different traffic patterns with corresponding fitness values $f_1$ and $f_2$. The minimization of the (expected) aggregated waiting times $f = f_1^2 + f_2^2$ is the subject of the optimization process. A "good" policy has to cope with these two conflicting objectives while minimizing the resulting waiting times. A $12$ dimension real-valued vector is used to represent the policy. In addition, stochastically disturbed fitness function values complicate the optimization task. Summarizing, the S-ring models a highly complex real-world optimization problem with conflicting objectives and noisy fitness function values.

## 4 MULTI-OBJECTIVE EVOLUTIONARY OPTIMIZATION

Multi-objective optimization is the process of finding one or more vectors of decision variables that simultaneously satisfy all feasible constraints and optimize a vector of objective functions that map the decision variables to two or more performance criteria or objectives [2]. In real-world problems it is extremely rare that only one single objective has to be optimized. Usually, a solution for multiple optimization criteria that may also contradict each other has to be found. In a problem with multiple objectives, it is generally impossible to obtain a total-ordering of all of the alternative solutions, without invoking further rules or assumptions. To make choices between solutions one needs selection criteria. Typically, Pareto selection, scalarizing selection, and criterion selection are used.

In Pareto optimization the concept of dominance is used to introduce an order into a vector space. A dominance relation $\prec$ is defined that compares the elements of a vector component-wise [6]. The set of all Pareto optimal vectors in the decision variable space is called the Pareto set. The respective set in the objective space is called the Pareto front. Scalarizing methods have a great popularity in the domain of multi-criteria decision making (MCDM). These methods map the vector of objectives to a single objective. This allows to apply methods that are well known in single objective optimization. The disadvantage of scalarizing is that the scalars have to be chosen adequately, the complete

Pareto front cannot be approximated within one run, and non-convex Pareto fronts are difficult to handle [7].

Criterion selection considers just one objective in isolation, each time a solution is evaluated for selection. It is not taken into account in the context of this article.

| factor | symbol | parameter |
|---|---|---|
| $X1_N$ | $\mu$ | parent individuals |
| $X2_N$ | $p_c$ | crossover probability |
| $X3_N$ | $p_m$ | mutation probability |
| $X4_N$ | $d_c$ | distribution index for crossover |
| $X5_N$ | $d_m$ | distribution index for mutations |
| $X6_N$ | $g$ | number of generations |
| $X7_N$ | $n$ | chromosome length (dimension) |
| $X8_N$ | $s_t$ | selection type |
| $X9_N$ | $c_t$ | crossover type |
| $X10_N$ | $f_r$ | front ratio |

Table 1: NSGA II parameters description

One of the best known multi-objective evolutionary algorithm for Pareto optimization is the NSGA II [7]. This algorithms has been incorporated together with other multi-objective algorithms into a programming environment called KEA (Kit for Evolutionary Algorithms) [4]. The NSGA II algorithm is an elitist multi-objective algorithm. It utilizes a non-dominated sorting step that "peels" off Pareto front by Pareto front from the set of the combined offspring and parent population $P_t$ and clusters these groups into ranks. In a second step, a special crowding distance sorting method is used to order the new set according to the distances between the individuals to increase the diversification of the solutions. A special crowded tournament selection operator generates a new offspring generation using mutation and crossover. The operator selects mating partners according to their rank and crowding distance. A new iteration of the algorithm is performed using the new offspring population and the sorted parent population. The relevant parameters (in the following also called factors) of the NSGA II can be seen in Table 1.

| factor | symbol | parameter |
|---|---|---|
| $X1_{ES}$ | $\mu$ | parent individuals |
| $X2_{ES}$ | $\nu$ | offspring-parent ratio $\nu = \lambda/\mu$ |
| $X3_{ES}$ | $\sigma_0^0$ | initial standard deviation |
| $X4_{ES}$ | $\tau_0^m$ | variation multiplier for $\sigma_0$ |
| $X5_{ES}$ | $\kappa$ | maximum age |
| $X6_{ES}$ | $n_\sigma$ | number of standard deviation |
| $X7_{ES}$ | $\rho$ | mixing number |
| $X8_{ES}$ | $r_x$ | crossover for objective variables |
| $X9_{ES}$ | $r_\sigma$ | crossover for strategy variables |
| $X10_{ES}$ | $g$ | number of generations |
| $X11_{ES}$ | $seed$ | random seed |
| $X12_{ES}$ | $n$ | dimension |

Table 2: ES parameter description

A classic evolution strategy (ES [8]) has been used together with scalarizing selection. Additional operators, that may help to find solutions using additional problem specific knowledge, have been deactivated for the experiments. The relevant factors of the ES can be seen in table 2. $\lambda \in \mathbb{N}_+$ denotes the number of offspring. The concrete parameter settings used for the experiments can be seen in Tables 3 and 4.

## 5 COMPARING THE PERFORMANCE OF DIFFERENT MOEA

Fonseca and Fleming introduced attainment surfaces that mark all solutions that are sure to be dominated by the set of already obtained non-dominated solutions [9]. Connecting the points of the approximated Pareto front found by the algorithm via horizontal and vertical line segments – as depicted in Figure 3 – divides the objective space into two regions. Repeated runs of the MOEA result in a set of attainment surfaces $\mathcal{A}_i = \{A_{ij}\}$ for the $i$th run configuration ($i = 1, 2, \ldots, m; j = 1, 2, \ldots, k_i$), if the $i$th run configuration is repeated $k_i$ times. Intersecting attainment surfaces with *cross-lines* enable us to define a metric for a comparison of several MOEA parameter design configurations. The set of cross-lines $L_k$ can be defined as

1. diagonal imaginary lines running in the direction of the improvement in all objectives [9],
2. lines intersecting the origin [10], or
3. lines that are parallel to the first bisector of the angle [11]. The following investigations are based on this approach.

For every cross-line and every attainment surface $A_{ij}$ the point of intersection $S_{ij,k}$ can be determined, see Figure 3. Hence, for every cross-line a distribution of points of intersection can be obtained. A simple geometric argument is used in the next step to determine the best run configuration $i^*$ for one crossing line ("the winner takes all" approach), see Figure 4. Finally, the percentage of cross-lines, where configuration $i$ performs best, is determined. Thus, we can conclude that configuration $i$ performs best compared to the other run configurations on $x_i$% of the cross-lines. This procedure is repeated $k$ times. The implementation of the comparison is straightforward and can be done efficiently.

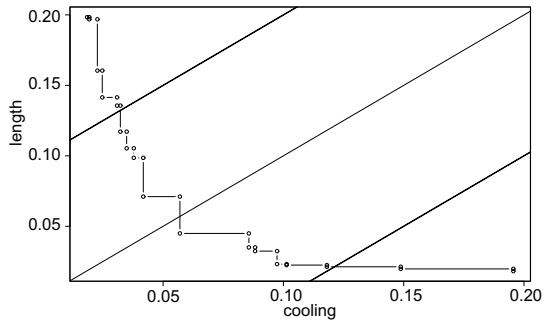


Figure 3: Attainment surface with cross-lines for one run of one simulation run-configuration.

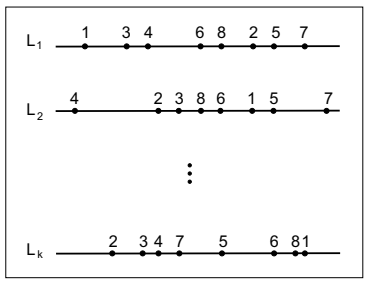


Figure 4: Points of intersection on cross-lines. The first run configuration performs best on cross-line $L_1$, whereas the fourth run configuration performs best on $L_2$.

Already a small number of cross-lines can give a good approximation of the quality of an parameter design configuration.

## 6 STATISTICS AND DESIGN OF EXPERIMENTS

Statistics are a powerful means to objectively analyze the results of computer experiments [1]. Designs of experiments (DoE) not only help to find optimal parameter settings very efficiently. They also support the comparison of parameter settings of algorithms when applied in different contexts. In this context linear regression models $\mathbf{y} = X\beta + \epsilon$ are used, where $\mathbf{y} \in \mathbb{R}^n$ is the known response, $X \in M(n \times q)$ is the design plan, i.e. a matrix of independent regression variables, $\beta$ is the vector of $q$ ($n \geq q$) unknown regression coefficients, and $\epsilon$ is the $n$-dimensional vector of errors, which are independently standard normal distributed. The least squares estimate of the regression coefficients is given by $\hat{\beta} = (X'X)^{-1}X'y$, where $X'$ denotes the transposed matrix of $X$. Generally, a three-stage approach for factor optimization is used: screening, modeling, and optimization. Screening is a technique to detect the significant main effects. From the coefficients of this model the "path of the steepest descent" towards locally better factor settings can be calculated. Classical types of screening designs are factorial and fractional factorial designs. Design generators with corresponding aliases and resolutions for fractional design plans can be found in [1].

In the modeling stage interactions are taken into account. In the final optimization stage, central composite designs (CCD) with additional axial runs are used. The ability of these designs to approximate second-order surfaces is reflected in a relatively high number of experiments.

## 7 EXPERIMENTAL SETUP

To get a first impression on the relationship between the factors and the algorithms performance, screening designs have been performed. For the elevator group control problem the first $9$ factors shown in Table 4 have been considered. Since we wanted to focus on the analysis of the main factors, a fractional factorial design of resolution $III$ with $2^{9-4} = 32$ runs was sufficient. The MTCS problem was simplified by varying only the first 5 factors given in Table 3 and 4. For this situation an appropriate choice of a screening design was a resolution $V$ fractional factorial design with $2^{5-1} = 16$ runs.

The parameters of the applications have been kept constant for all experiments. They were chosen according to experience and reasons of clearness. In the mold temperature control problem the following weights were used in the penalty function $f_{pen}$: $a = 1.0$E+2 , $b = 1.0$E+4, $c = 1.0$E+6. In the term $f_t$, the value $e = 0.25$ and in the value $d = 1.0$E-6 in the effective fitness function $f$ have been chosen. The number of $6$ bores yields a reasonably complex problem. In each experiment a half-sphere (see fig. 1) had to be cooled. The number of generations of the ES applied to both problems depends on the number of fitness evaluations, which was kept constant. With $\lambda = \nu * \mu \in \{10, 35, 40, 140\}$ and $const = 1.0$E+4 evaluations, the generation numbers are $g = const/\lambda \in \{1000, 285, 250, 70\}$, respectively. The number of generations used for the NSGA II was $100$ and $200$ for $100$ and $50$ individuals, respectively.

It should be mentioned that an ES with only one step size $\sigma_0$ and $\rho = 1$ has been used. Experiments using more step sizes and changing values of $\rho$ have been tested only for the elevator group control problem.

Each complete design plan has been repeated $100$ times per experiment using the ES. This makes $16 \times 100$ runs for one complete plan. The best values and the arithmetic mean of the fitness values found by the population during each generations have been recorded. The same initial random seed was used for each factor setting. The random seed was increased linearly for each of the $100$ repetitions.

| symbol | range | design |
|---|---|---|
| $\mu$ | $\mathbb{N}_+$ | $X1_N \in \{50; 100\}$ |
| $p_c$ | $]0.0, 1.0]$ | $X2_N \in \{0.3; 0.7\}$ |
| $p_m$ | $]0.0, 1.0]$ | $X3_N \in \{0.3; 0.7\}$ |
| $d_c$ | $\mathbb{N}_+$ | $X4_N \in \{20; 70\}$ |
| $d_m$ | $\mathbb{N}_+$ | $X5_N \in \{20; 70\}$ |
| $g$ | $\mathbb{N}_+$ | $X6_N * X1_N =$ const. |
| $n$ | $\mathbb{N}_+$ | $X7_N =$ const. |
| $s_t$ | {roulette, tournament} | $X8_N =$ tournament |
| $c_t$ | {point, uniform, SBX} | $X9_N =$ SBX |
| $f_r$ | $\mathbb{R}_+$ | $X10_N = 1$ |

Table 3: Parameter settings for the NSGA II

| symbol | range | design |
|---|---|---|
| $\mu$ | $\mathbb{N}_+$ | $X1_{ES} \in \{5; 20\}$ |
| $\nu$ | $\mathbb{R}_+$ | $X2_{ES} \in \{2.0; 7.0\}$ |
| $\sigma_0^0$ | $\mathbb{R}_+$ | $X3_{ES} \in \{0.1; 0.3\}$ |
| $\tau_0^m$ | $\mathbb{R}_+$ | $X4_{ES} \in \{0.5; 2.0\}$ |
| $\kappa$ | $\mathbb{N}_+$ | $X5_{ES} \in \{1.0; \infty\}$ |
| $n_\sigma$ | $\{1; n\}$ | $X6_{ES} = 1$ |
| $\rho$ | $\{1; \mu\}$ | $X7_{ES} = \mu$ |
| $r_x$ | {intermediate, discrete} | $X8_{ES}$ = discrete |
| $r_\sigma$ | {intermediate, discrete} | $X9_{ES}$ = intermediate |
| $g$ | $\mathbb{N}_+$ | $X10_{ES} * \lambda =$ const. |
| $seed$ | $\mathbb{N}_+$ | $X11_{ES}$ = const. |
| $n$ | $\mathbb{N}_+$ | $X12_{ES}$ = const. |

Table 4: Parameter settings for the ES

The experiments of each complete design plan for the analyses of the NSGA II have been repeated only three times. This is due to the fact that the Pareto fronts generated by the NSGA II are more difficult and more time consuming to analyze than the single-criterion results generated by the ES. The cooling efficiency $f_t$ and the total length of the bores $f_l$ were recorded during each run yielding a two dimensional Pareto front.

All KEA and ES runs for the MTCS have been performed on a PC (Athlon XP+ 2600) with Windows 2000 and the ESGC optimization runs were performed on a heterogenous workstation cluster of 30 Linux PCs.

## 8 RESULTS

### 8.1 ES results for the MTCS

The best solutions found by the ES were disturbed by outliers with very high values. These outliers are due to the restrictions evaluated by the penalty functions. ES using small populations tend to get stuck in the restrictions, i.e. bores penetrate the surface of the tool and cannot be removed. Although, the number of generations performed by the ES was high enough to guarantee that the algorithm can stabilize in an equilibrium state, the bandwidth of the variations

of the ES-results was large. The multi-objective problem of the MTCS seems to have a lot of local optima. This problem increases with the number of bores. Therefore, it is likely for the ES to get stuck in one of the suboptima.

Due to the problems described above, the response value considered for the statistical analysis was chosen to be the global best value out of the $100$ runs for each factor setting. In the following these values will be called GBF values. Using the global best values is also practically reasonable, because the best of the hundred runs would also be selected for manufacturing.

The chosen fractional factorial design of resolution V allows to estimate all main effects and two-factor interactions uniquely. In order to avoid the problem of over-fitting, non-significant factors are deleted. Since the design is a saturated one, e.g. 16 parameters (including the intercept) are estimated based on 16 runs, no degrees of freedom are left to estimate the error variance. To assess the significance of the individual factors we generated a half-normal plot. Under the assumption that most of the investigated factors possess no significant influence, most of the estimated effects are normally distributed with mean zero and only some have a nonzero expectation value. This graphical tool plots the ordered absolute values of the standardized estimated effects against the corresponding quantiles of the standard normal distribution. The largest estimated effects that deviate clearly from the line formed by the other factors are declared to be significant. Figure 5 shows the half-normal plot for the ES problem.

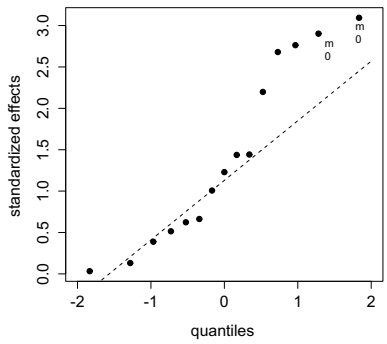


Figure 5: Half-normal plot.

Apparently, $\tau_0^m$, the multiplier of the learning rate, the population size $\mu$ and the selective pressure $\nu$ have a significant influence on the algorithm's performance. Additionally, the interactions between $\mu$ and $\tau_0^m$, and $\mu$ and $\nu$ play a significant role. A forward stepwise selection procedure verifies the results obtained by this graphical tool. The interaction between, for example, $\mu$ and $\tau_0^m$ can be interpreted as follows. If the multiplier of the learning rate $\tau_0^m$ is small, the population size plays no role on the performance of the algorithm. For large values of $\tau_0^m$ a larger population size $\mu$ results in a better performance of the algorithm. A value of the adjusted R-Square of $0.74$ indicates that the fit of this reduced model is sufficient.

The influence of the population sizes on the problem quality is typical for difficult problems. The higher the sampling rate, the higher the possibility to find good alternatives among several sub-optima. The selection pressure $\nu$ influences the diversity in the population. Therefore, it is not surprising that the interaction of $\mu$ and $\nu$ has been detected. The dominant significance of factor $\tau_0^m$ is interesting. $\tau_0^m$ is an additional multiplier for the standard $\tau_0 = \sqrt{2n}^{-1}$ factor that is used in the self adaptation process of the step size in the ES. The smaller $\tau_0^m$ the smaller the standard deviation of the step sizes. Small values may be advantageous in complex problems where far "explorative" jumps in the fitness landscape yield too often disadvantageous results. Too small $\tau_0^m$ result in stagnation. The interaction of $\tau_0^m$ with the population size of the parents $\mu$ can be interpreted via formula of the convergence velocity which depends on both parameters [8].

Since the contribution of the main effects and the two factor interactions is almost identical to the corrected total sum of squares, it is not recommended to apply the method of steepest descent based on the main factor model. Instead, we determined the optimum predicted by the reduced regression model including the main effects and the interactions. Note that no extrapolation of the design space was permitted. Running the ES algorithm with this optimal parameter setting increased the performance. The best GBF value was $0.11221$. The optimal parameter setting is given in Table 5.

| symbol | parameter setting |
|---|---|
| $\mu$ | 5 |
| $\nu$ | 7 |
| $\sigma_0^0$ | 0.2 |
| $\tau_0^m$ | 0.5 |
| $\kappa$ | plus strategy |

Table 5: Optimal parameter setting

Compared to the mean of GBF values obtained from the fractional factorial design, a reduction of $18$ per cent was gained. A more sophisticated central composite design was run around this design point to investigate the vicinity of this local optimum. Although all of the observed responses were small, none resulted in a better algorithm performance than the optimal point given above.

### 8.2 ES results for the ESGC

In this section, we identify by means of regression trees [12] the main factors, which significantly improve the performance of the ES algorithm for the elevator group control problem. Each of the $32$ runs of the $2_{\text{III}}^{9-4}$ design was repeated three times.

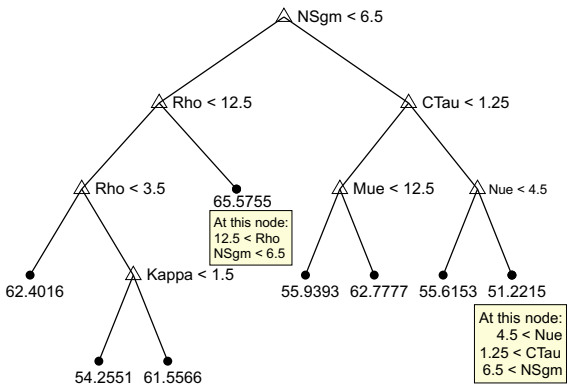


Figure 6: Regression tree.

The resulting regression tree (Figure 6) indicates that $n_\sigma$, the number of step-sizes has the largest influence on the algorithm's performance. Similar to a stepwise model selection from classical regression analysis, a cost complexity parameter can be determined for the tree (Figure 7). Superfluous nodes are snipped off to generate trees that can be interpreted intuitively.

The rightmost node has the lowest fitness value ($51.2$), the corresponding factor settings read: $\nu > 4.5$, $\tau_0^m > 1.25$, and $n_\sigma > 6.5$. Therefore, we can conclude that an increased selective pressure, combined with an increased learning rate and number of step-sizes, leads to an improved performance.

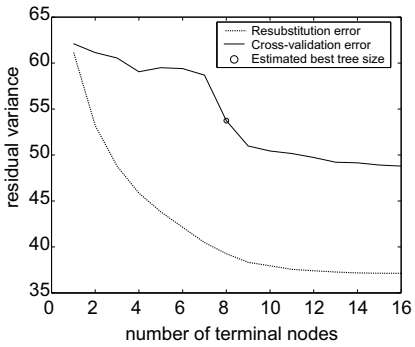


Figure 7: Tree pruning.

The regression tree analysis improved the algorithm's performance more than 15% and required only a few preliminary runs.

### 8.3 NSGA II results for the MTCS

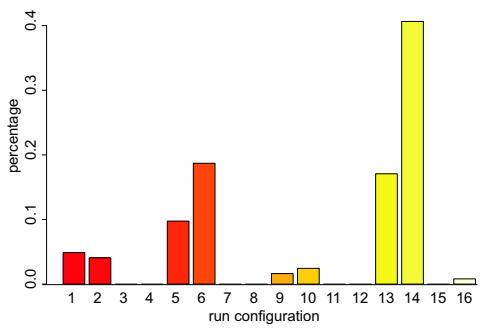


Figure 8: Barplot for the MTC problem. Levels as shown in Table 3, run configurations as shown in Table 4.

The NSGA II parameterization from Table 1 was used for these experiments. The ranking method presented in section 5 reveals that the run configuration 14 apparently performs best (see Figure 8). This configuration uses the following parameters: $100$ parent individuals, a crossover probability $p_c = 0.7$, a mutation probability $p_c = 0.3$, a distribution index for crossover $d_c = 70$, and a distribution index for mutations $d_m = 20$.

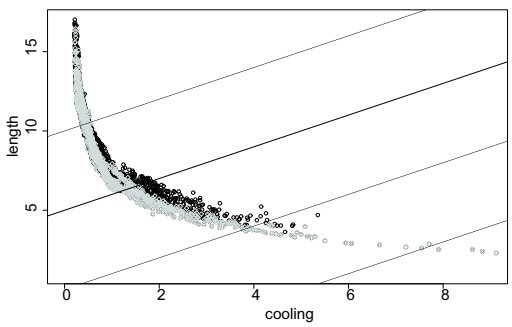


Figure 9: Pareto fronts for the MTC problem. Solutions of the improved setting are colored grey.

Combining these results with the results from a regression analysis that was performed additionally leads to the conclusion that $p_c = 0.7$ and $p_m = 0.3$ improve the algorithm's performance. A comparison of the related Pareto fronts support this assumption, see Figure 9.

## 9 CONCLUSIONS

Two very different real-world problems have been solved by evolutionary algorithms using an aggregation and a Pareto-front approach. The multimodal character of both complex problems resulted in high result variances. To circumvent this problem, attention was restricted the best value of the algorithms performance for repeated runs of one factor setting. A ranking method, based on attainment surfaces and cross-lines, provided an intuitively understandable and easily implementable way to compare the obtained solutions. By means of statistical methods, like Design of Experiments and regression trees, a better understanding of the influence of the parameters on the algorithm's performance was gained. As a result, for both complex problems a parameter setting of the algorithms could be identified that improved the performance of the algorithm significantly.

## 10 ACKNOWLEDGMENTS

These research projects (C12, B3) are supported by the German Research Foundation (DFG) as a project in the collaborative research center SFB 531 at the University of Dortmund, Germany.

## REFERENCES

[1] D.C. Montgomery. *Design and Analysis of Experiments*. Wiley, Hoboken, NJ, 5th edition, 2001.

[2] J.D. Knowles. *Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization*. PhD thesis, The University of Reading, Reading, UK, 2002.

[3] H. Gastrow. *Der Spritzgießwerkzeugbau in 130 Beispielen*. Hanser, München, 5 edition, 1998.

[4] J. Mehnen, T. Michelitsch, T. Bartz-Beielstein, and K. Schmitt. Evolutionary Optimization of Mold Temperature Control Strategies. *Journal of Engineering Manufacture*, 2004. in print.

[5] J. Mehnen, T. Michelitsch, and K. Weinert. Evolutionary Optimized Mold Temperature Control Strategies using a Multi-Polyline Approach. In *Proceedings of the EuroGP03*, pages 374–383, Colchester, UK, 2003. Springer.

[6] M. Ehrgott. *Multicriteria Optimization*, volume 491 of *Springer, Lecture Notes in Economics and Mathematical Systems*. Springer, Berlin, 2000.

[7] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley, New York, 2001.

[8] H.-G. Beyer. *The Theory of Evolutionary Strategies*. Springer, Berlin, 2001.

[9] C.M. Fonseca and P.J. Fleming. On the performance assement and comparison of stochastic multiobjective optimizers. In H.-M. Voigt, W.-E., I. Rechenberg, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature - PPSN IV, Berlin, Germany, Sept.22-26, 1996*. Springer, 1996.

[10] J. Knowles and D. Corne. Approximating the nondominated front using the pareto archived evolution strategy. *Evolutionary Computation*, 8(2):149–172, 2000.

[11] T. Bartz-Beielstein and B. Naujoks. Tuning multi criteria evolutionary algorithms for airfoil design optimization. Technical Report CI–159/03, University of Dortmund, 2003.

[12] T. Bartz-Beielstein. Experimental analysis of evolution strategies. Technical Report CI–157/03, University of Dortmund, 2003.