

Comparison of Evolutionary Algorithms for Design Optimization

Wilfried Jakob, Martina Gorges-Schleuter, Ingo Sieber

Forschungszentrum Karlsruhe, Institut für Angewandte Informatik
Postfach 3640, D-76021 Karlsruhe, Germany
e-mail: jakob@iai.fzk.de

Abstract. The production of specimen for microsystems or microcomponents is both, time and material-consuming. In a traditional design process the number of possible variations which can be considered is very limited. Thus, in micro-system technology computer-based design techniques become more and more important - similar to the development of microelectronics.

In this paper we compare Evolutionary Algorithms based on Evolution Strategies and the extended Genetic Algorithm GLEAM for solving the design optimization problem. The reference problem is the design optimization of a 2-lens-system being part of a heterodyne receiver, a microoptical communication module. As this is a real world problem, the design must be as insensitive to fabrication tolerances as possible. The results obtained are compared to a more complex task: the robot path planning problem.

1 Introduction

To achieve both, the improvement of the quality of a design of microsystems or micro-components and the reduction of the time and material-consuming production of specimen, advanced simulation and optimization techniques are required. The computer aided development is based on simulation models which must be computable sufficiently fast and need to be parameterizable. In addition they need to be accurate enough, as the quality of an optimization depends highly on the quality of the simulation model; for details we refer to [1]. This paper focuses on the optimization process itself. As the time required for a single evaluation varies from a couple of seconds to some minutes depending on the problem on hand the reduction of the number of fitness calculations is essential for the applicability of evolutionary search techniques to the design optimization task. On the other hand the search technique must be general enough in order to produce reliable results for different applications. We compare the results of the Evolutionary Algorithm GLEAM [2,3,4] and the concept of foreruns with those obtained by standard Evolution Strategies [5] and an extension of ES to spatial structured populations and local selection [6] as well as with a traditional hill climbing method.

Our **SIM**ulation and **Optimization Tool Environment SIMOT** [7] will on one hand support the designer to develop and optimize simulation models and on the other hand to optimize complex (micro-)systems or components. It includes optimization tools and simulators. The optimization tools GAMA (Genetic Algorithm for Model Adaptation) and GADO (Genetic Algorithm for Design Optimization) are based on GLEAM and are developments of our institute [8, 9]. The simulators are commercial tools: an FEM sim-

ulator, an analog network simulator and Mathematica¹. The optimizer and the simulator are loosely coupled and may be chosen depending on the problem to be solved. For the optical system described further on we used Mathematica for the simulation and GADO as optimizer. The optimization of the design of a collimation system under realistic production conditions shows how SIMOT is successfully used on a multiple objectives problem with conflicting criteria. The search space of the application is of complex nature although there are only few variables to be considered.

To confirm the forerun approach we use a more complex but nevertheless fast to simulate task, the robot path planning problem.

2 Evolutionary Design Optimization

During the design process the engineer is faced with a large search space of possible design solutions and parameterizations. Building models is limited to a few only. The situation becomes better by creating a computer model which might be evaluated by a simulator. During an optimization process many simulations with various parameter settings have to be done. The complexity of the search space is in general high so that a manual exploration is limited and mainly influenced by personal knowledge, previous experiments, intuition of the engineer and good luck. An optimal system design might not be expected under these conditions.

Assuming that we are able to build a simulation model being accurate enough and parameterizable, then the engineer's optimization task can be supported by evolutionary search techniques exploring and exploiting the search space. The engineer's task is now the specification of the optimization parameters and restrictions and the formulation of the criteria of optimization. In case of multiple objectives being not mutually independent we cannot optimize for the quality goals separately. The formulation of grading functions and priorities as described below gives the engineer the possibility to provide the optimizer with a suitable way of making its decisions.

3 Optimization Algorithms

In this section we give a brief description of the optimization algorithms included in our comparison.

3.1 GLEAM

GLEAM uses a list-like hierarchical data structure. The elements of the data structure depend on the actual application. The hierarchy may be used to treat parts of the data structure as a unit, termed section, and thus prevent them from being separated by the crossover operators or to hide them completely thus prevent them from being modified by any of the genetic operators.

1. Mathematica is a registered trademark of Wolfram Research, Inc.

The mutation operator is inspired from its counterpart in evolution strategies in the sense that small variations of genetic values are more likely than larger ones. GLEAM allows the usage of any arbitrary alphabet for the internal representation being mostly naturally induced by the application considered. Assuming that the elements of the underlying alphabet (i.e. the values a certain gene can take) are sorted by some criteria, we create before applying the mutation operator a division of the range of values into classes. By mutation a change of the current gene value to a random value within the nearby classes is very likely and this probability shortens with the distance of a class as defined by a prespecified step function. There are various crossover operators implementing traditional n-point crossover and uniform crossover as used in genetic algorithms and crossover operators respecting the creation and existence of sections, which itself underlay the evolutionary process.

Each genetic operator may be independently activated on a percentage basis. Whenever an operator is chosen, a new offspring is generated. Thus if several genetic operators have a percentage of choice greater than zero, there is a chance that more than one offspring will be generated. The resulting set of full siblings will be evaluated and only the best will be considered to be included into the population as described by the survival rule. Thus there is a surplus of descendants and only the best may reproduce again.

3.2 Concept of Foreruns

Two different types of experiments were performed: the first type consists of a single more-or-less “large” population while the second one is split into a forerun and a main run. The forerun consists of small sized pre-populations performing only a small number of generations. The final best individuals obtained from the foreruns are used to initialize the main population. The idea of combining foreruns followed by a main run is inspired by the promising results of using previous knowledge for the initial population reported in [3] and shall hopefully reduce the number of required evaluations.

3.3 Spatially Structured Populations

The population used in GLEAM consists of uniformly distributed individuals over a geographic region. Interaction of the individuals, i.e. the selection process acting through both, mate selection for reproduction and the selection of descendants for survival, is limited to geographically nearby individuals, the locally overlapping neighborhoods.

In the following experiments with GADO a closed ring structure has been chosen and the size of the neighborhood of any individual is set to 9, thus including the centre individual and its immediate four neighbors to the right and left, respectively. Each centre individual and its partner being chosen within the neighborhood by linear ranking produce offsprings by means of mutation and / or crossover. The descendants are evaluated and the best of them is compared with the centre individual and replaces it immediately, but only if the offspring is better than the weakest in its neighborhood (survival rule better local least) and with the exception of those centre individuals being the best within their neighborhood, then the offspring must be better than the individual itself (local elitism) [4]. This process is continued until a termination criterion is reached, for example a maximum number of evaluations has been done or a prespecified quality is reached.

3.4 Standard and Diffusion Model Evolution Strategies

Evolution Strategies are especially powerful algorithms for numerical optimization. Their representation use n object variables x_i ($1 \leq i \leq n$) and a set of strategy parameters. We use a comma-ES with step size parameters only. The recombination of the object parameters is discrete and for the step size intermediate. There exist multi-modal problems being very difficult for the standard ES with global selection.

Now we use an ES with a diffusion model to clarify the influence of the population structure. As described above we have a spatially structured population with individuals limited to local interaction. Both, the selection of partners for recombination and the selection of descendants for survival are restricted to geographically nearby individuals.

The local parent selection method chooses both parents randomly with replacement from a neighborhood [10], the centric parent selection method takes the centre individual of a neighborhood as first parent and chooses only the second parent randomly from within the neighborhood [6]. In the runs reported we generated 6 descendants on each site at generation t . From this local pool of descendants only the best is chosen to survive and it replaces the centre individual in the next generation $t+1$. Thus, the local interaction ES, named LES, uses a generational approach whereas GLEAM uses a one-at-a-time transition. A notation, a growth curve analysis and an empirical comparison of the various classes of LES is presented in [6].

3.5 Local Hill Climbing Algorithm

Our simple derivation free hillclimber (Gauss-Seidel-Strategy with fixed step size for the line search and multiple restart) starts from a random initial setting of the parameters. One of them is chosen and optimized until no further improvement of this parameter is possible, then the next one is chosen and optimized and this is repeated until no further improvement is possible.

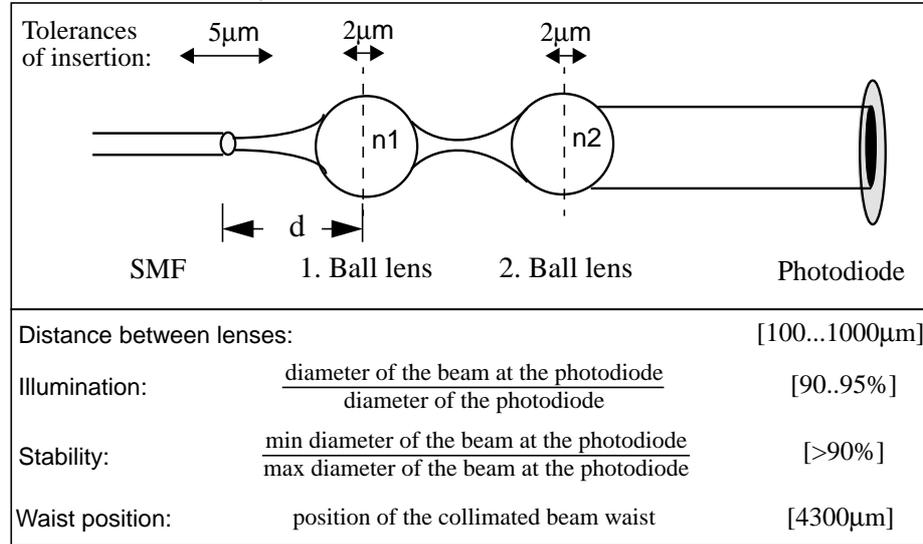
4 Optimization of a Microoptical Collimation System

The design of systems incorporating a laser beam, as microoptical applications do, mostly requires the modification of the “raw“ beam. The beam must be expanded, refocused and collimated. This modification can be performed by using lenses, mirrors or prisms [11]. For our application, the collimation system, we will use two microoptical ball lenses. The geometry of the 2-lens system is shown in Fig. 1.

The beam as it comes out of a single mode fiber is refocused by the first lens and then collimated by the second one in order to position the collimated beam waist at the location of the photodiode. In an ideal case of geometric optics it is possible under some restrictions to derive for each lens with refractive value n_1 a second lens with refractive value n_2 so that an arbitrary irradiation is yielded. In reality, we need to place the elements into prefabricated LIGA structures [12] and this can only be done with some tolerances. These tolerance values of insertion are given in the top row of Fig. 1.

The variations of the placement influence the position of the beam waist and the diameter of the beam at the photodiode. The optimization task is to determine a collimation system being as insensitive as possible with respect to the variances of insertion.

Fig. 1. Geometry of the collimation system. The bottom box shows the definition of the optimization criteria and the range of success values.



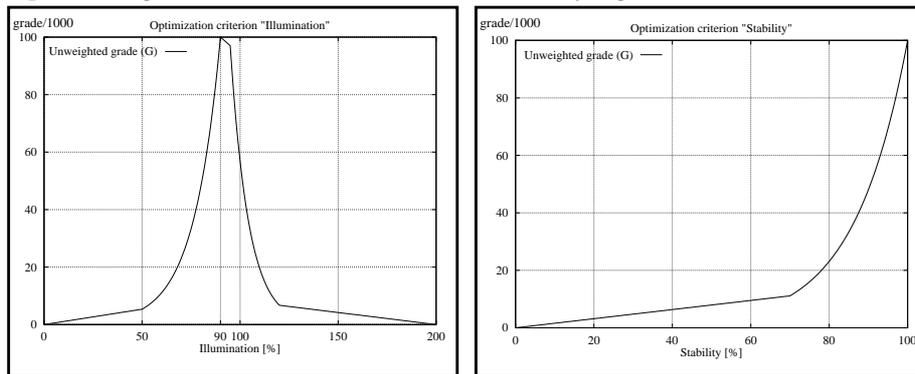
The optimization parameters are the refractive values $n1$ and $n2$ of the ball lenses in the range of 1.4 to 2.0 and a value z in the range of 1.0 to 2.0. Using z and the focus of the first ball lens we compute the distance of the fiber to the first lens as

$$d = \frac{z \cdot n1 \cdot R}{2 \times (n1 - 1)}$$

where $n1$ is the refractive value of the first lens and $R=450\mu\text{m}$ is the radius of this ball lens.

The optimization criteria are stability, illumination, waist position and distance between the two lenses. The definition of these values as well as the range of valid values is given in Fig. 1. The optimum values are 100% for stability, 90% for illumination, 4300µm for the beam waist position and the distance between the lenses should be preferably be above 100µm and below 1000µm.

Fig. 2. Grading functions for illumination (left) and stability (right) .



The collimation system is simulated with Mathematica, where the extreme values of the displacement are used to determine the number of necessary Mathematica simulations for one design evaluation. Using the simulation outcome we compute the absolute value of the optimization criteria. The multiple objective optimization is done by using grading functions assigning to each absolute value a grade (N) between 0 and 100000. Fig. 2 shows these grading functions at hand of the illumination and stability criteria. For example, for the illumination criterion 90% is optimal and a value of up to 95% is regarded as a success; if the simulation detects a further underfill or overfill at the photodiode the outcome is degraded exponentially. A solution is regarded as a success, if the values of Fig.1 are fulfilled and with increasing stability values successful runs are ranked higher. All grades are then weighted, as specified by the weight functions given by the engineer, and summed up. In our setting a total maximum of 100000 might be reached in case of mutual independent criteria.

5 Results

5.1 Design Optimization Task

First we present the results of the design optimization of the heterodyne receiver. The hillclimber produced widely differing solutions. Especially, the demands on stability were hard to fulfil. This indicates the highly multimodal nature of the problem. The number of evaluations needed until this strategy converges differs in a wide range of 2000 to 42000 yielding in a range of quality grade values between 72340 and 79068. The best solution found has a stability of 90.3%, an illumination with small overfill of 90.7% and a waist position at 4294.3 μ m.

For reasons of comparability the runs using GLEAM and the ES were limited to an upper limit of about 36000 evaluations. For each setting (job) 40 runs were done and the quality threshold was set to a grade of 80500, which is not the best we could achieve (the best solution found has a quality of 81031), but a pretty good design quality. We recorded how many runs meet this requirements and how many evaluations were used by the “good” runs. The results are shown in Table 1.

As none of the HC runs meet the target grade the figures for the number of evaluations are calculated on the base of all runs and not of only the “good” ones as with the rest of the table. It is obviously that the HC approach is not sufficient to tackle the task.

As expected GLEAM delivers reliable good results with increasing population size. Thus we can take the GSP7 job as a reference for the GPP jobs. All GPP jobs were done using a size of 16 for the pre-populations. Due to the low number of successful runs GPP1 and GPP4 are not considered any further. GPP2 delivers reliable results with 21% less average evaluations than GSP7. Further reductions to 46% can only be achieved by a slightly loss of reliability as GPP3 shows. Fig. 3 summarizes these results. So for the problem on hand pre-populations can reduce the computational load significantly without a loss of reliability. The best solution of GLEAM has a stability of 91.22%, an illumination of 90.00%, a waist position at 4300.1 μ m and a grade of 81031. Compared with the HC results the obtained values for the refractive indices lead to completely different materials for the lenses and the resulting distances between the fiber and the first lens

Table 1. Results from hillclimber (HC), GLEAM with single (GSP) and pre-populations (GPP) and Evolution Strategy (ES) and local interaction ES (LES) for the 2 lens problem.

Job	Pre Populations		Main Population Size	# of Successful Runs	Speed-up wrt GSP7 [%]	Evaluations of Pre and Main Pops.		
	Number	Gen.				Median	Average	Variance
HC				1		6483	10986	15354
GSP1			60	28		2570	3844	3765
GSP2			90	28		4472	5845	6782
GSP3			120	36		4295	4802	3905
GSP4			150	39		4231	5252	4799
GSP5			180	39		5825	6276	3989
GSP6			210	39		5858	6644	4020
GSP7			240	40		7674	7743	4282
GPP1	10	10	60	33	41	4082	4601	3433
GPP2	20	10	60	40	21	5719	6113	4291
GPP3	10	20	60	39	46	3407	4147	3168
GPP4	10	10	90	37	33	5343	5195	3301
GPP5	10	20	90	39	30	3193	5448	4792
ES			60	18	53	3600	3600	1819
ES			120	28	7	6480	7200	3285
LES-1			60	33	56	3960	3442	1457
LES-1			90	35	33	4860	5164	2502
LES-c			60	37	45	4380	4287	1667
LES-c			90	40	30	4860	5439	2549

characterize two different optical systems ($n_{1\text{HC}} = 2.0$, $n_{1\text{G}} = 1.60$; $n_{2\text{HC}} = 1.58$, $n_{2\text{G}} = 1.55$; $d_{\text{HC}} = 495\mu\text{m}$, $d_{\text{G}} = 791.6\mu\text{m}$).

The rows labelled ES in Table 1 give the results for a (60,360)-ES and a (120,720)-ES, respectively. A population size of 60 (120) in the ES corresponds in terms of descendants generated in a single generation to a population size of 120 (240) in GLEAM. The standard ES with global selection has problems to find good solutions reliable for our multi-modal problem of fractal nature.

The rows labelled LES in Table 1 give the results for the local interaction ES with a ring population structure. Due to the results from [6] the neighborhood size is set to 3, the smallest possible. The LES uses only a single step size parameter for all object variables as experiments with a LES using multiple step sizes failed in all cases.

The LES with local parent selection (LES-1) is much more reliable than the global rule ES. The population size 60 found in 80% a solution better than the threshold 80500 and the larger population of size 90 in 88% of the runs.

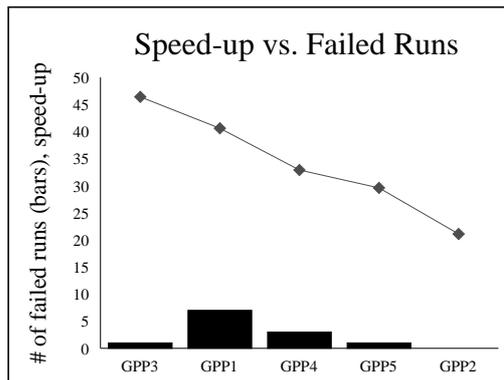


Fig. 3. Speed-up of the GPP runs from Table 1 with respect to GSP7 vs. number of failed runs

The best choice for a local interaction ES is one using the centric parent selection method (LES-c). Due to the soft selection pressure of this local strategy, balancing well exploitation and exploration, all runs for the larger population size converged to a quality above 80500. In addition the number of evaluations needed is comparable low to the GPP3 job of GLEAM.

5.2 Robot Path Planning Task

The question arises how general are these results or are they specific to the problem on hand? Unfortunately it is not possible to conduct such an amount of runs with our previous problem of optimizing a micropump [1] because of the long time of some minutes for each simulation. Thus we decided to fall back to a more complex but very fast to simulate problem, the collision free path planning of a robot movement. The difference to the numerical design optimization problem is that the nature of this task is closer to genetic programming as commands for the robot control are evolved.

The task is to move the end effector of an industrial robot from a given to a target position which is located behind an obstacle on a line as straight and smooth as possible. To do this the robot must turn around 135 degrees avoiding collision with two further obstacles. Beside collision avoidance the deviation from the target, the quality of the path, and the required time for the move are the most important optimization criteria. The robot is controlled on axis level and the sequence and parameters of control commands are subject to evolution. Details of this task and earlier investigations can be found in [3] and results of industrial applications of this approach in [13].

Table 2. Results from GLEAM with single (GSP) and pre-populations (GPP) for the path planning problem using small deme sizes of 5 for the pre-populations and 7 for the main populations.

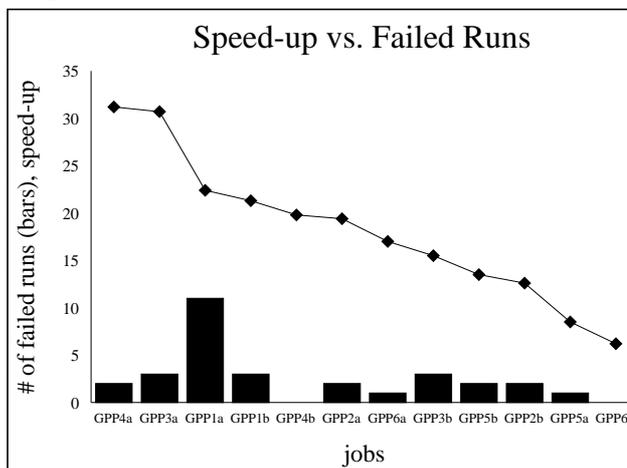
Job	Pre Populations		Main Population Size	# of Successful Runs	Speed-up wrt GSP4 [%]	Evaluations of Pre and Main Pops.		
	Number	Gen.				Median	Average	Variance
GSP1			60	181		40564	79418	110413
GSP2			90	197		42891	72161	107893
GSP3			120	200		48325	73853	109271
GSP4			180	200		65467	77624	59479
GSP5			240	200		77550	93107	84347
GPP1a	10	10	60	189	22	34199	60227	74277
GPP2a	20	10	60	198	19	45550	62543	50927
GPP3a	10	20	60	197	31	43382	53792	38634
GPP4a	10	10	90	198	31	43948	53416	33375
GPP5a	20	10	90	199	9	55375	70995	53907
GPP6a	10	20	90	199	17	48880	64499	49883
GPP1b	10	10	60	197	21	40318	61069	70152
GPP2b	20	10	60	198	13	55687	67849	49867
GPP3b	10	20	60	197	16	53823	65578	46968
GPP4b	10	10	90	200	20	49972	62233	48588
GPP5b	20	10	90	198	14	60893	67160	22477
GPP6b	10	20	90	200	6	61417	72851	50514

Due to the short simulation times of about 1 msec we performed 200 runs for each setting. A run is regarded as successful if the resulting path is collision free, the target deviation is less than 4 mm and the path is “acceptable smooth and straight”. The two groups of GPP jobs in Table 2 differ in the population size of 16 (a) and 24 (b) for the pre-populations. As with the design optimization increasing population sizes stabilize the results of GLEAM as the GSP runs show. From a population size of 120 onwards all runs are successful. Despite of the success of GSP3 we choose GSP4 as the reference because the maximum total number of evaluations is with 716000 much lower than the 1.26 million offsprings of GPP3. Thus 716000 is used as upper limit for the number of calculated offsprings for the GPP jobs which must undergo the average value of 77624 evaluations to be better than the single population approach. The first sequence of jobs were made with a deme size of 9 for both, the pre- and the main populations. Most of the jobs delivered 196 or more successful runs but 3 were significantly worse: those with the settings of GPP1a, GPP1b and GPP4a of Table 2. This indicates that settings based on 10 populations with a maximum of 10 generations are fairly small for this problem using this comparable large deme size.

Inspired from the results described in [6] we reduced the deme size of the pre-population to 5 and for the main population to 7. Table 2 shows the results. Now only the GPP1a job must be rejected. All others deliver a greater or the same number of successful runs as with the larger deme sizes. Thus the results from [6] are confirmed by applying it to the GLEAM algorithm on a complete different problem. The jobs with a setting of 10 populations with a maximum of 10 generations each have the greatest potential for improvement due to the low number of evaluations needed in the pre-populations. Now, with the smaller deme sizes they are no longer too unstable and deliver the best results, when they are combined with a main population size of 90, as GPP4a and GPP4b show.

GPP1b and GPP3a deliver good improvements too, but as 3 runs failed and we have comparable improvements with lower or no failed runs we do not consider these settings any further. Fig. 4 summarises these results. On the other hand the improvement of 31% or 20% resp. is not as promising as with the 2-lens-problem. Thus we can state that pre-populations can lower the computational load for the robot task too but not in the same magnitude as with the design optimization problem.

Fig. 4. Speed-up with respect to GSP4 vs. number of failed runs.



So in general the concept of foreruns seem to be able to reduce the amount of evaluations while maintaining the quality of the search.

6 Conclusions

As a result of the investigation in this paper we can state that the Evolutionary Algorithm GLEAM as well as the extension of ES to spatially structured populations are able to solve the multi-modal problem of the 2-lens-system presented. The standard ES with global selection is not able to solve this problem with a sufficiently high convergence reliability. The theoretical investigation as well as the empirical findings using test-bed functions from literature presented in [6] are confirmed by the real-world problem of design optimization taking restrictions of the fabrication process into account.

The concept of foreruns for initialization of the main population of GLEAM as motivated in [3] proved its superiority to a single population concept. A halving of the computational load could be achieved with the design optimization problem at hand. The tendency of these results could be approved for a more complicated application, the collision free robot path planning problem.

Acknowledgements. We like to thank Prof. Christian Blume for the kind permission to use the robot simulator for the work outlined in section 5.2.

References

1. W. Süß, S. Meinzer, A. Quinte, W. Jakob, H. Eggert: *Simulation and Design Optimization of a Micropump*. Proceedings of MICROSIM'97, Lausanne, Sept. 1997, 127-135
2. C. Blume: *GLEAM - A System for "Intuitive Learning"*. Proc. of the 1st Int. Workshop on Parallel Problem Solving from Nature. LNCS 496, Springer-Verlag (1991)
3. W. Jakob, M. Gorges-Schleuter, C. Blume: *Application of Genetic Algorithms to Task Planning and Learning*. Proc. 2nd Int. Conf. PPSN, North-Holland (1992) 291-300
4. M. Gorges-Schleuter: *Parallel Evolutionary Algorithms and the Concept of Pop. Structures*. In: V. Plantamura et al. (Eds): *Frontier Decision Support Concepts*. Wiley (1994) 261-319
5. H.-P. Schwefel: *Evolution and Optimum Seeking*. Wiley & Sons, New York (1995)
6. M. Gorges-Schleuter: *A Comparative Study of Global and Local Selection in Evolution Strategies*. This Proc. on PPSN V.
7. W. Süß, W. Jakob, M. Gorges-Schleuter et al.: *Design Optimization of Microsystems based on Adaptive Search Techniques*. Computer Aided Optimum Design of Structures V, Computer Mechanics Publications, Southampton (1997) 121-130
8. W. Jakob, S. Meinzer, A. Quinte, M. Gorges-Schleuter, H. Eggert: *Partial Automated Design optimization Based on Adaptive Search Techniques*. Proceedings of ACEDC 96, PEDC, University of Plymouth (1996)
9. M. Gorges-Schleuter, W. Jakob, S. Meinzer et al.: *An Evolutionary Algorithm for Design Optimization of Microsystems*. Proc. PPSN IV, LNCS 1141, Springer (1996) 1022-1032
10. J. Sprave: *Linear Neighborhood Evolution Strategy*, Proc. 3rd Ann. Conf. on Evolutionary Programming, World Scientific, River Edge, NJ (1994) 42-51
11. D. C. O'Shea: *Elements of Modern Optical Design*. Wiley & Sons (1985)
12. P. Bley, J. Göttert, M. Harmening et al.: *The LIGA Process for the Fabrication of Micromechanical and Microoptical Components*. Micro System Technologies 91, VDE Verlag, Berlin (1991)
13. C. Blume: *Planning of Collision-free Movements for Industrial Robots Using Evolutionary Algorithms (in german)*. Automatisierungstechnische Praxis (atp) 12/97, Oldenburg Verlag, 58-67 (1997)