Natural Computing (2006) 5: 363–385
DOI 10.1007/s11047-006-0001-x

On three new approaches to handle constraints within evolution strategies

O. KRAMER[1,*] and H.-P. SCHWEFEL[2]
[1]*Department of Computer Science, University of Paderborn, D-33098, Paderborn, Germany;* [2]*Department of Computer Science, University of Dortmund, D-44221, Dortmund, Germany (*Author for correspondence, e-mail: okramer@upb.de)*

Abstract. Evolutionary algorithms with a self-adaptive step control mechanism like evolution strategies (ES) often suffer from premature fitness stagnation on constrained numerical optimization problems. When the optimum lies on the constraint boundary or even in a vertex of the feasible search space, a disadvantageous success probability results in premature step size reduction. We introduce three new constraint-handling methods for ES on constrained continuous search spaces. The death penalty step control evolution strategy (DSES) is based on the controlled reduction of a minimum step size depending on the distance to the infeasible search space. The two sexes evolution strategy (TSES) is inspired by the biological concept of sexual selection and pairing. At last, the nested angle evolution strategy (NAES) is an approach in which the angles of the correlated mutation of the inner ES are adapted by the outer ES. All methods are experimentally evaluated on four selected test problems and compared with existing penalty-based constraint-handling methods.

Key words: evolution strategies, evolutionary algorithms, nonlinear optimization problem, constraint-handling methods, fitness stagnation, minimum step size, sexual selection, metaevolution

Abbreviations: EA – evolutionary algorithm; ES – evolution strategy; NLP – nonlinear programming; DSES – death penalty step control evolution strategy; TSES – two sexes evolution strategy; NAES – nested angle evolution strategy

1. Introduction

Evolutionary algorithms (EA) and in particular evolution strategies (ES) are used for constrained numerical parameter optimization. The optimum quite often lies on the constraint boundary or even in a vertex of the feasible search space. In such cases the EA frequently suffers from premature convergence because of a low success probability near the constraint boundaries. First of all, in this section the NLP-problem is defined. After a short survey of constraint-handling techniques, the premature fitness stagnation problem is discussed. In Section 2 we present a new constraint-handling method, the death penalty step control evolution strategy (DSES). The DSES uses an adaptive mechanism that controls the reduction of a minimum step size depending on the distance to the infeasible search space. In Section 3 we introduce a biologically inspired concept of sexual selection and pairing for handling constraints. In Section 4 we present the nested angle evolution strategy (NAES), a metaevolutionary approach in which the angles of the correlated mutation of the inner ES are adapted by the outer ES. At last, all techniques are experimentally evaluated on four selected test problems and compared with the results of an existing penalty-based constraint-handling method.

1.1. *The NLP problem*

In general, the constrained continuous nonlinear programming problem is defined as follows: In the n-dimensional search space $\mathbb{R}$ find a solution $x = (x_1, x_2, \ldots, x_n)^T$, which minimizes $f(\mathbf{x})$:

$$\begin{array}{llll} f(\mathbf{x}) \rightarrow \min., & \mathbf{x} \in \mathbb{R} & \text{with subject to} & \\ \text{inequalities} & g_i(\mathbf{x}) \leq 0, & i = 1, \ldots, n_1 & \\ \text{equalities} & h_i(\mathbf{x}) = 0, & j = 1, \ldots, n_2 & \end{array} \tag{1}$$

A feasible solution $\mathbf{x}$ satisfies all n_1 inequality and n_2 equality constraints. Many constraint-handling techniques like penalty functions make use of a constraints violation measurement G:

$$G(\mathbf{x}) = \sum_{i=1}^{n_1} \max[0, g_i(\mathbf{x})]^\beta + \sum_{j=1}^{n_2} |h_j(\mathbf{x})|^\gamma \tag{2}$$

The parameters β and γ are usually chosen as one or two. In the following, only inequality constraints are taken into account.

1.2. *A short survey of constraint-handling methods*

1.2.1. *Penalty functions*

There exists a variety of constraint-handling techniques for EA. Most of them are based on penalty functions. An early, rather general penalty approach is the sequential unconstrained minimization technique (SUMT) by Fiacco and McCormick (1964). The constrained problem is solved by a sequence of unconstrained optimizations in which the penalty factors are stepwise intensified. In other penalty approaches penalty factors can be defined statically (Homaifar et al., 1994) or depending on the number of satisfied constraints (Kuri-Morales and Quezada, 1998). They can dynamically depend on the number of generations of the EA as Joines and Houck propose (1994);

$$\tilde{f}(\mathbf{x}) = f(\mathbf{x}) + (C \cdot t)^{\alpha} \cdot G(\mathbf{x}) \tag{3}$$

The parameter t represents the actual generation, the parameters C and α must be defined by the user. Typical settings are $C=0.5, \alpha=1$ or 2. Penalties can be adapted according to an external cooling scheme (Joines and Houck, 1994) or by adaptive heuristics (Bean and Hadj-Alouane, 1992). In the death penalty (DP) approach infeasible solutions are rejected and new solutions are created until enough feasible ones exist.

1.2.2. *Penalty-related methods*

Many methods revert to the penalty principle. In the segregated genetic algorithm by Riche et al. (1995) two penalty functions, a weak and an intense one, are calculated in order to surround the optimum. In the coevolutionary penalty-function approach by Coello Coello (2000b) the penalty factors of the inner EA are adapted by an outer EA. Some methods are based on the assumption that any feasible solution is better than any infeasible (Powell and Skolnick, 1993; Deb, 2001). An example are the metric penalty functions by Hoffmeister and Sprave (1996). Feasible solutions are compared using the objective function while infeasible solutions are compared considering the satisfaction of constraints.

1.2.3. *Decoders and repair*

Decoders build up a relationship between the constrained search space and an artificial search space easier to handle (Koziel and Michalewicz, 1999; Michalewicz and Fogel, 2000). Repair algorithms either replace infeasible solutions or only use the repaired solutions for evaluation of their infeasible pendants (Belur, 1997; Coello Coello, 2002).

1.2.4. *Multiobjective optimization*

Multiobjective optimization techniques are based on the idea of handling each constraint as an objective. Under this assumption many multiobjective optimization methods can be applied. Such approaches were used by Parmee and Purchase (1994), Jiménez and Verdegay (1999), Coello Coello (2000a), and Surry et al. (1995). In the behavioral memory method by Schoenauer and Xanthakis (1993) the EA concentrates on minimizing the constraint violation of each constraint in a certain order and optimizing the objective function in the last step.

1.2.5. *Other approaches*

A further method is to avoid infeasible solutions by special constraints preventing representations and operators. An example is the GENOCOP-algorithm (Michalewicz and Fogel, 2000) that reduces the problem to convex search spaces and linear constraints. A predator–prey approach to handle constraints is proposed by Paredis (1994) using two separate populations. Schoenauer and Michalewicz (1996) propose special operators that are designed to search regions in the vicinity of active constraints. A comprehensive overview to constraint-handling techniques is given by Coello Coello (2002) and also by Michalewicz and Fogel (2000). Recently, Montes and Coello Coello (2005) introduced a technique based on a multimembered evolution strategy combining a feasibility comparison mechanism with several modifications of the standard ES.

1.3. *Evolution strategies*

For a comprehensive introduction to ES see Beyer and Schwefel (2002). Here, the most important features of the state of the art $(\mu/\rho \overset{+}{,} \lambda)$-ES for continuous search spaces are repeated. An ES uses a parent population with cardinality μ and an offspring population with cardinality λ. At first, the individuals, consisting of a vector of objective and strategy variables, are initialized. The objective variables represent a potential solution to the problem whereas the strategy variables, which are step sizes in the standard $(\mu/\rho \overset{+}{,} \lambda)$-ES, provide the variation operators with information how to produce new results. The initial values of our experiments can be found in Appendix B. During each generation λ individuals are produced. In the first step

ρ $(1 \leq \rho \leq \mu)$ parents are randomly selected for reproduction. After recombination of strategy and objective variables the ES applies log-normal mutation to the step sizes σ_i $(0 \leq i \leq n)$ and uncorrelated Gaussian mutation to the objective variables. After λ individuals are produced the best μ individuals are selected as parents for the next generation exclusively out of the offspring population in the case of comma selection or out of the offspring and the previous parental population in case of the plus selection scheme. As an extension of the comma selection scheme the parameter κ specifies the number of reproduction cycles individuals are allowed to survive in the parental population if they cannot be replaced by fitter offspring solutions.

1.4. *The problem of premature step size reduction*

ES on constrained optimization problems suffer from premature step size reduction in case of active inequality constraints. This results in a premature convergence. Figure 1 shows a typical situation. Consider the application of death penalty or other penalty based approaches as constraint-handling method and a small angle α between the contour lines of the fitness function and the constraint boundary. For the sake of better understanding we do as if all mutations fall within a σ-circle around individual P instead of taking a normal distribution with standard deviation σ into account. Only within the marked area the fitness of an offspring individual is better than the fitness of its parent

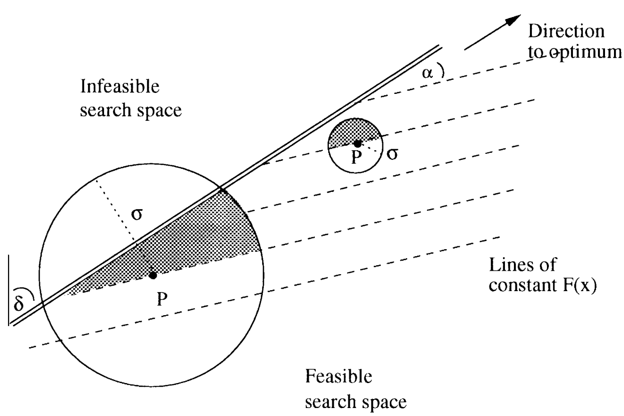

Figure 1. Premature step size reduction. The success probability p_s increases for decreasing step sizes.

P. The relation between this success area and the size of the whole circle resembles the success probability p_s, which is bigger for small step sizes (upper P) than for bigger step sizes (lower P). For a small angle $\alpha \ll \pi/2$ the infeasible search space cuts off a big area of the success region in the case of big mean step sizes. This means the success probability p_s increases for decreasing step sizes and therefore the self-adaptive selection process prefers solutions with smaller mean step sizes. Furthermore, up to one half of the mutations are produced in the opposite direction of the optimum. All circumstances lead to the mentioned premature step size reduction resulting in premature convergence. We explain the role of the angle δ in Section 4.

The situation becomes even worse for problems similar to the tangent problem, see Appendix A. Here, the angle α decreases when approximating the optimum and consequently the success probability p_s also decreases. The premature step size reduction can be shown experimentally on problem 2.40 (see Appendix A) for the DP method and the dynamical penalty function by Joines and Houck (1994), see Table 1. Problem 2.40 exhibits a linear objective function and an optimum with five active linear constraints. Each line of Table 1 shows the results of a (15,100)-ES after 50 runs. As termination condition fitness stagnation is chosen. If the difference between the fitness value of the best individual of a generation and the best of the following generation is smaller than $\theta = 10^{-12}$, then the ES terminates as the magnitude of the steps sizes is too small to effect further improvements. Both constraint-handling methods are not able to approximate the optimum of the problem satisfactorily. The standard deviation *Std. dev* show that the algorithms produce rather different results in the various runs.

Table 1. Experimental results of the DP method and the dynamic penalty function by Joines and Houck (Dyn) on problem 2.40

Algo	Best	Mean	Worst	Std. dev	FFC	CFC
DP	−4948.07919871	−4772.33867	−4609.98512	65.2821	50,624	96,817
Dyn	−4780.55456768	−4559.12982	−4358.44663	85.0955	31,878	31,878

The parameter *Best* shows the best fitness, *Mean* shows the average fitness with the standard deviation *Std. dev* whereas *Worst* shows the worst fitness achieved by the presented methods after all runs. The parameter *FFC* counts the fitness function calls and *CFC* the constraint function calls. Both constraint-handling techniques are not able to approximate the optimum of the problem satisfactorily. The relatively high standard deviations *Std. dev* show that the algorithms produce unsatisfactorily different results.

2. The death penalty step control approach

2.1. *Minimum step size reduction mechanism*

As mentioned in section 1.4 the DP method suffers from premature step size reduction because of insufficient birth surplus. The death penalty step control evolution strategy (DSES) is based on DP, i.e. rejection of infeasible solutions. For the initialization feasible starting points are required. The key principle of the approach is a minimum step size ε, a lower bound on the step sizes σ, that prevents the evolutionary process from premature step size reduction. But it also prevents the optimization process from unlimited approximation of the optimum when reaching the range of ϵ. Consequently, a control mechanism is introduced with the task of reducing ϵ when approximating the optimum. Intuitively, the reduction process depends on the number of infeasible mutations produced when reaching the area of the optimum at the boundary of the feasible search space. Consider the situation presented in Figure 2. Again, for the sake of better understanding we do as if all mutations fall within a σ-circle around the parental individual instead of using a normal distribution with standard deviation σ. On the left (a), the parent P has come quite close to the optimum at a vertex of the feasible search space. Further approximation (b) with the same minimum step size means an increase of infeasible mutations which are counted with the parameter z. The reduction process of ϵ depends on the number z of rejected

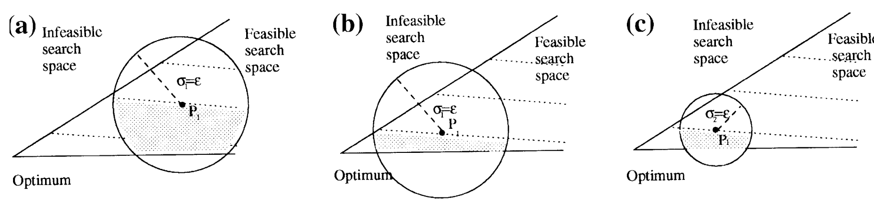

Figure 2. The optimum lies in the vertex of the feasible search space. For the sake of better understanding we do as if all mutations fall into a σ-circle around the parental individual instead of using a normal distribution with standard deviation σ. (a) The minimum step size ϵ enables the optimization process to approximate the optimum. (b) A further approximation is possible until the marked region of success becomes considerably small and many mutations fall into the infeasible region. (c) When the number of infeasible trials exceeds the parameter **mod** the minimum step size ϵ is reduced and a further approximation of the optimum becomes possible.

infeasible solutions: Every *mod* infeasible trials ϵ is reduced by a factor $0 < melt < 1$ according to the equation:

$$\epsilon' := \epsilon \cdot melt \tag{4}$$

The DSES is denoted by [*mod; melt*]-DSES.

2.2. *Experimental results*

The experimental results of the DSES with several settings on problem 2.40 are shown in Table 2. Again, the algorithm performs 50 runs with the same parameters as used in the previous Section 1.4. Table 2 shows that the DSES achieves satisfactory results on problem 2.40 as long as the minimum step size is not reduced too fast. The latter is the case for the [75; 0.5]- and the [75; 0.3]-ES. For the other test settings the fitness values and standard deviations show that the optimum of −5000.0 is reached in every run with high accuracy. A slow reduction of the minimum step size entails inefficiency. The [100; 0.7]-DSES requires about 15% more fitness and 35% more constraint function calls than the [75; 0.7]-DSES while achieving the same solution quality.

Figure 3 shows the development of the mean step size and the minimum step size ϵ during a typical run of the [75; 0.7]-ES on problem 2.40. The step sizes are presented on a logarithmic scale. As expected, the minimum step size ϵ is always located below the actual average step size. Further experiments in Section 5 confirm positive

Table 2. Results of the DSES with different settings on problem 2.40

DSES	Best	Mean	Worst	Std. dev	FFC	CFC
[100; 0.7]	−5000.0000	−5000.0000	−5000.0000	3.2×10^{-10}	93,944	1,168,387
[100; 0.5]	−5000.0000	−5000.0000	−5000.0000	1.1×10^{-9}	85,504	882,620
[100; 0.3]	−5000.0000	−5000.0000	−5000.0000	7.1×10^{-10}	88,472	747,183
[75; 0.7]	−5000.0000	−5000.0000	−5000.0000	2.2×10^{-10}	79,566	770,334
[75; 0.5]	−5000.0000	−4983.6316	−4823.2500	39.667	178,456	1,324,303
[75; 0.3]	−5000.0000	−4932.5880	−4811.6491	62.557	334,532	2,045,082

For four setting the DSES approximates the optimum in every run; Only in the cases of the [75; 0.5]- and the [75, 0.3]-DSES the minimum step size is reduced too fast to prevent premature step size reduction.

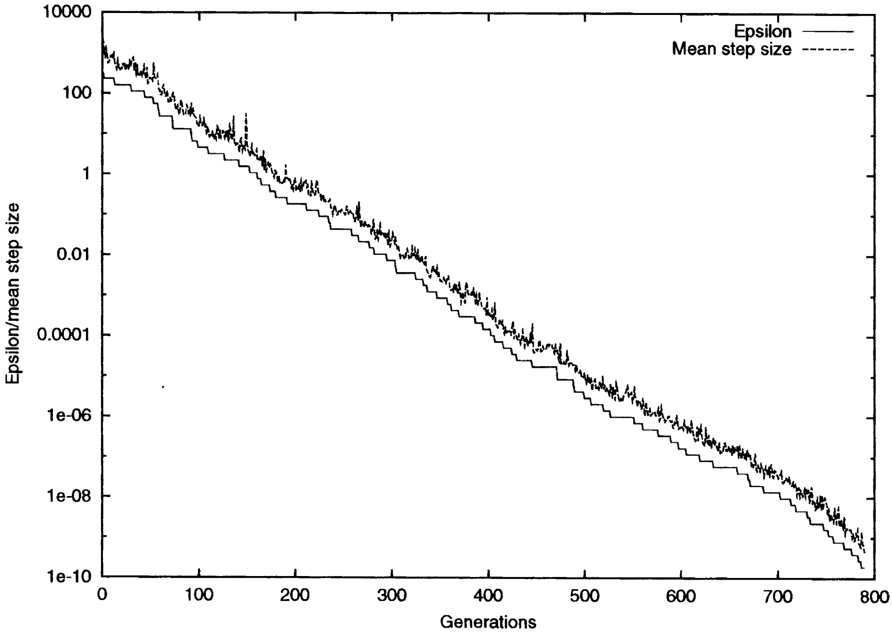

Figure 3. Development of the minimum and the mean step size during a typical run of the [75; 0.7]-DSES on problem 2.40 on a logarithmic scale. Both minimum and mean step sizes shrink exponentially.

results of the DSES on other test problems, but also show its limitations.

3. Constraint-handling with two sexes

3.1. *Biologically inspired constraint-handling*

The idea of the concept called two sexes evolution strategy (TSES) is to handle the objective function and the constraint functions as separate objectives. Every individual of the TSES is assigned to a new feature called its sex. Similar to nature, individuals with different sexes are selected according to different objectives. Individuals with sex o are selected by the objective function. Individuals with sex c are selected by the fulfillment of constraints. The intermediate recombination operator plays a key role. Recombination is only allowed between parents of different sex. The treatment of objective function and constraints as separate objectives sounds similar to the multiobjective optimization approaches for constraint-handling. Instead of a multiobjective optimization method a biologically inspired concept of pairing two sexes is introduced. Consider the situation presented in Figure 4. Again, the optimum lies at the boundaries of the feasible search space. The optimum of the unconstrained objective function lies beyond the boundary in the infeasible search space. In the so-called $(\mu_o + \mu_c, \lambda_o + \lambda_c)$-TSES μ_o parents are selected out of λ_o individuals with sex o, whereas μ_c parents are selected out of λ_c offspring individuals of the previous generation with sex c. As the individuals with sex o are selected according to the objective function, they tend to lie finally in the infeasible search space (black squares) whereas the c-individuals are selected by the fulfillment of all constraints and mostly lie in the feasible search space (white circles). The measurement G for the fulfillment of constraints has already been defined in equation (2). By means of intermediate recombination, all individuals get closer to the optimum of the problem, but still are found on opposite sides of the boundaries between the feasible and the infeasible search space. For the initialization feasible starting points are not required.

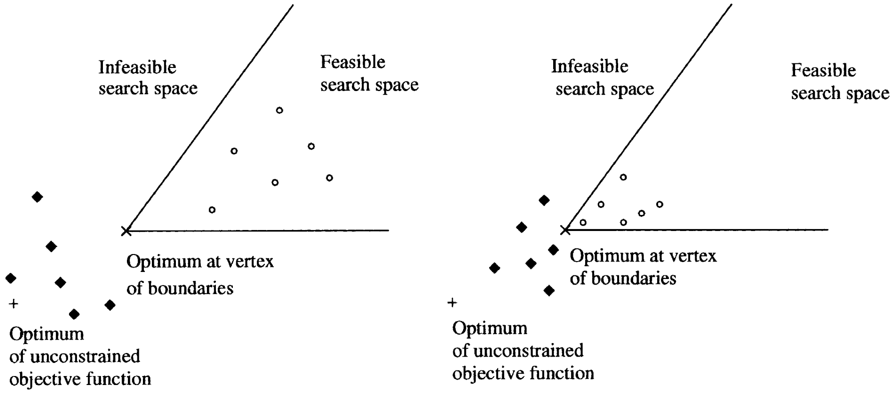

Figure 4. The effect of intermediary recombination within the TSES enabling individuals to reach a constrained optimum from both sides of the boundaries. Left: The individuals with sex o (black squares) enter the infeasible region. Right: After intermediary recombination all individuals get closer to the optimum on a vertex of the feasible region.

3.2. *Modifications of the basic TSES*

Several modifications of the basic concept of the TSES are necessary until the algorithm provides successful results. The usual self-adaptation process effects an explosion of the mean step sizes, because the invasion far into the feasible search space is rewarded with high fitness values for individuals with sex c as well as approaching the unconstrained optimum of the objective function is rewarded for the o-individuals. Modifications of the population ratios of the TSES aim at reducing the diversity in the population to avoid the overadaptation of the step sizes. Several experiments with different sex ratios and selection operators lead to the following heuristic modifications:

– sex ratio and birth surplus $(8+8,\ 13+87)$
– two-step selection operator for the sex c, according to the metric penalty function by Hoffmeister and Sprave (1996): First, selection by fulfillment of constraints, secondly, if enough feasible solutions exist: selection by objective function
– introduction of a finite life span $1<\kappa<\infty$ (see Section 1.3) for individuals with the sex c

These modifications lead to promising results on the test functions, see next Section 3.3. The sex ratio with a majority of 87 c-individuals and only 13 o-offspring show that the diversity within the o-individuals may not exceed a certain level. Otherwise, the population would be able to reach the region of the unconstrained optimum resulting, in an explosion of mean step sizes. The survival possibility for the most successful individuals over up to κ reproduction cycles emphasizes the role of the c-individuals.

3.3. *Experimental results*

In Table 3 the experimental results of the TSES are presented. Like above the TSES runs 50 times on problem 2.40. As termination condition fitness stagnation is chosen. The parameter κ is tested with two different settings. Table 3 shows that the $(8+8,13+87)$-TSES is able to approximate the optimum with the desired accuracy for both settings of κ. Both experiments differ in the average number of fitness and constraint functions calls as the TSES with $\kappa=200$ requires about 25% more FFC and CFC than the TSES with $\kappa=50$. Figure 5 shows

Table 3. The (8 + 8, 13 + 87)-TSES with two settings for parameter κ. Both TSES approximate the optimum of problem 2.40 in every run

κ	Best	Mean	Worst	Std. dev	FFC/CFC
50	−5000.00000	−5000.00000	−5000.00000	4.21×10^{-11}	529,468
200	−5000.00000	−5000.00000	−5000.00000	8.48×10^{-12}	709,536

The numbers of FFC and CFC are equal. As the constraint violation for the o-individuals does not have to be calculated the number of CFC may even be smaller.

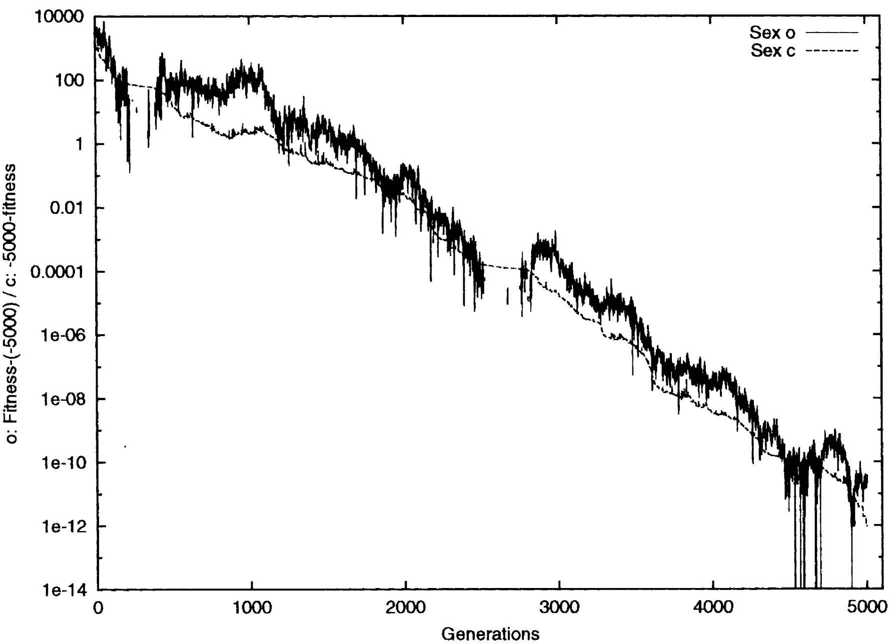

Figure 5. Development of the best fitness of both sexes of the (8 + 8, 13 + 87)-TSES on problem 2.40.

the fitness development of a typical run of the (8 + 8, 13 + 87)-TSES on problem 2.40. As expected in the description of the TSES-idea, the individuals of the two sexes lie on opposite sides of the constraint boundaries around the optimum. Individuals with sex *o* lie in the infeasible search space. Thus the graph shows the development of the difference between the optimum and the fitness values. Similarly, the fitness of the *o*-individuals develops this way oscillating around the fitness of sex *c* and being interrupted where the individuals lie in the feasible part of the search space. Concerning the performance, the TSES is less efficient than the DSES introduced in the previous Section 2. In Section 5 the qualities and efficiencies of all techniques are compared to each other.

4. The nested angle evolution strategy

4.1. *Metaevolution for mutation ellipsoid rotation*

The success probability situation at a boundary of the feasible search space can change considerably when the mutation ellipsoid is rotated by an angle in the range of δ, see Figure 6. Here, δ is the smaller angle between one parameter axis and the constraint. Using two different step sizes and one rotation angle the mutation ellipsoid can adapt to a situation where the infeasible search space does not cut off the success area and consequently prevents premature step size reduction. This consideration leads to the constraint-handling method proposed in this section. Figure 6 shows the situation when the mutation ellipsoid is rotated by angle $-\delta$. After the rotation the success probability p_s increases. Rotation of the mutation ellipsoid can be achieved by correlated mutations introduced by Schwefel (1974). But experiments with the correlated mutations show that the diversity in the population of a (15,100)-ES is not sufficient to achieve the adaptation of both mean step sizes and angles, see Table 4.

In our new nested angle evolution strategy (NAES) the outer ES adapts the angles for the rotation of the mutation ellipsoid of the

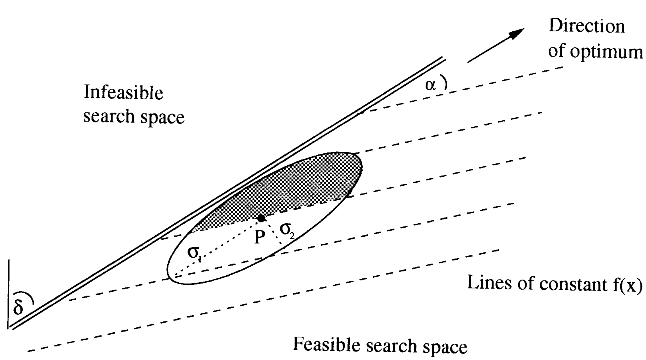

Figure 6. Situation at the boundary of the feasible search space after rotation of the mutation ellipsoid by the angle $-\delta$: The self-adaptation process enables the step sizes σ_1 and σ_2 to form an ellipsoid that is not cut off by the constraint.

Table 4. The (15,100)-ES with correlated mutations and standard parameter settings on problem 2.40

Pb	Best	Mean	Worst	Std. dev	FFC	CFC
2.40	−5000.00000000	−4942.9735056	−4704.2122103	88.653	78,292	180,450

The algorithm is far away from approximating the optimum in every run.

inner ES. Here we use the advanced notation for nested ES introduced by Rechenberg (1994). In the $[\mu'/\rho'+,\lambda'(\mu/\rho+,\lambda)^{\gamma}]$-ES λ' inner ES run for γ generations, also called isolation time. In our metaevolutionary approach the isolation time is conditioned by observing the fitness stagnation. This stops the inner ES after the step size has been reduced as a result of premature step size reduction. The parameters ρ and ρ' specify the number of parents used for recombination and are set to two during our experiments. They are omitted in the following notations.

4.2. *Experimental results*

Table 5 shows the experimental results of the [5, 50(5, 50)]-NAES on problem 2.40 after 15 runs. The variables of the outer ES which are the angles for the inner ES are initialized within the interval [0, $\pi/2$]. The corresponding initial step sizes are $\sigma_i = \pi/8$ for all i. The start position of all inner ES is $x^{(0)} = (250, 250, 250, 250, 250)$. Mutation and recombination parameters are chosen as usual. For both ES fitness stagnation is chosen as termination condition. For all inner ES the setting $\theta = 10^{-9}$ and for the outer $\theta = 10^{-7}$ is chosen. As the worst fitness values and the standard deviations show (see Table 5), the NAES is able to find the optimum with the accuracy the termination condition allows. The NAES is able to adapt the rotation angles and thus increases the success probability p_s. But obviously, the NAES is not very effcient. The reason for this lies in the nature of the metaevolutionary approach. Every evaluation of an individual of the outer ES causes a full run of λ' inner ES. As shown in the next Section 5

Table 5. The [5, 50(5, 50)]-NAES on problem 2.40

Pb	Best	Mean	Worst	Std. dev	FFC	CFC
2.40	−5000.000000	−5000.000000	−5000.000000	1.16×10^{-8}	3.6×10^{7}	1.5×10^{8}

In all runs the NAES is able to find the optimum with the desired accuracy.

with a [3, 15(3, 15)]-NAES we can reduce the number of fitness function calls by decreasing the population sizes. But smaller population sizes are not always sufficient to guarantee the diversity in the outer population that is necessary to find the correct rotation angles.

5. Comparison of experimental results

In this section, the proposed constraint-handling techniques are compared experimentally on four test problems with optima in vertices of the feasible search spaces. Table 6 summarizes the

Table 6. Survey of the experimental results of the DSES, TSES, and NAES on the problems TR2, 2.40, 2.41, and HB

	Best	Mean	Worst	Std. dev	FFC	CFC
TR2						
DP	2.000	2.000	2.001	3.8×1.0^{-4}	11,720	20,447
Dyn	2.000	2.001	2.007	0.0015	13,100	13,100
DSES	2.000	2.000	2.000	8.5×10^{-6}	796,2001	1,463,900
TSES	2.000	2.000	2.000	1.2×10^{-8}	1,100,872	1,100,872
NAES	2.000	2.000	2.000	3.1×10^{-16}	927,372	1,394,023
2.40						
DP	−4948.079	−4772.339	−4609.985	65.28	50,624	96,817
Dyn	−4780.555	−4559.130	−4358.447	85.10	31,878	31,878
DSES	−5000.000	−5000.000	−5000.000	2.2×10^{-10}	79,566	770,334
TSES	−5000.000	−5000.000	−5000.000	4.2×10^{-11}	529,468	529,468
NAES	−5000.000	−5000.000	−5000.000	1.2×10^{-8}	35,935,916	149,829,046
2.41						
DP	−17596.108	−17050.591	−16496.020	255.018	31,718	69,292
Dyn	−17754.605	−17187.705	−16172.738	342.9851	46,290	46,290
DSES	−17857.143	−17857.143	−17857.143	1.2×10^{-8}	51,660	469,866
TSES	−17857.143	−17857.143	−17857.143	8.6×10^{-10}	379,268	379,268
NAES	−17857.143	−17857.143	−17857.143	3.8×10^{-8}	12,821,710	41,720,881
HB						
DP	−30978.142	−30899.666	−30766.920	41.4454	37,540	77,298
Dyn	−30893.281	−30765.428	−30625.048	63.8514	24,186	24,186
DSES	−31025.560	−31025.560	−31025.560	1.3×10^{-9}	54,344	211,499
TSES	−31025.560	−31025.560	−31025.560	7.6×10^{-11}	241,290	241,290
NAES	−31025.560	−31025.560	−31025.560	2.1×10^{-8}	19,730,306	60,667,984

results of the three proposed constraint-handling methods in comparison to the methods DP and the dynamical penalty function (Dyn) by Joines and Houck 1994 on problems TR2 (two dimensional TR), 2.40, 2.41, and HB. For parameter settings and termination conditions we refer to Appendix B. Table 6 shows that the standard methods DP and Dyn are not able to approximate the optimum of problems TR2, 2.40, and 2.41 satisfactorily. The dynamic penalty function fails on problem HB, only the DP method performs well on problem HB to some degree, but cannot find the optimum in every run. As explained in Section 1.4 both techniques suffer from premature step size reduction.

In contrast to that, the small standard deviations *Std. dev* and the same fitness values for the best and the worst solutions show that the new proposed constraint-handling techniques do not suffer from premature fitness stagnation and therefore reach the optimum as far as the termination condition with the parameter θ allows on the problems 2.40, 2.41, and HB. The only exception is problem TR2 where only the NAES is able to approximate the optimum and find the correct rotation angle of $\pi/4$. In further experiments the NAES was able to approximate the optimum of the tangent problem of higher dimensions.

Obviously, the DSES cannot cope with the decreasing success probability of problem TR2 when approximating the optimum. The TSES achieves relatively satisfactory results on problem TR2. Concerning the performance, the DSES requires the least number of fitness function calls, but causes many constraint function calls. The reason for this is that the ϵ-reduction mechanism depends on the number of infeasible trials. The speed of conducting the ϵ-reduction depends on the parameters *mod* and *melt*, of course. The TSES is not as efficient as the DSES, but uses less constraint function calls.

The NAES causes the highest number of constraint and fitness function calls as we expect from the nature of a metaevolutionary approach. Every solution of the outer ES requires a full run of a couple of inner ES. But the NAES is the only constraint-handling method which is able to approximate the optimum of every type of constrained problem. In practice, the NAES might be too inefficient, but the performance can be improved by adequate parameter settings for the parameters μ, μ', λ, and λ'.

5.1. *Summary*

The premature step size reduction and the resulting premature fitness stagnation could be shown experimentally for two traditional constraint handling techniques, DP and a dynamic penalty function. Three new constraint-handling techniques were introduced with the aim of preventing an ES to suffer from premature fitness stagnation. The DSES achieves promising results on all test problems except on the tangent problem. The TSES shows similar results. The NAES is able to approximate the optimum of every kind of problem. All algorithms were compared by the numbers of necessary fitness and constraint function calls. We revealed the structural special feature of the tangent problem concerning the success probability at the optimum. The experimental results have to be expanded to other constrained test functions. In the future we will recommend further parameter settings for the proposed constraint-handling techniques in order to achieve fast and high quality results for constrained problems. Perhaps modifications of the proposed methods will help to improve the quality and efficiency of the results on the tangent problem and other structurally similar problems.

6. Appendix

6.1. *A. Test problems*

For our experimental analysis we selected four constrained test problems with optima at the boundary of the feasible search space.

Problem 2.40. Schwefel's problem 2.40 (Schwefel, 1995)

Minimize:

$$F(x) = -\sum_{i=1}^{5} x_i$$

Constraints:

$$G_j(x) = \begin{cases} x_j \geq 0, & \text{for } j = 1, \ldots, 5 \\ -\sum_{i=1}^{5} (9+i)x_i + 50,000 \geq 0, & \text{for } j = 6 \end{cases}$$

Minimum:

$$x^* = (5000, 0, 0, 0, 0)^T, \quad F(x^*) = -5,000$$

G_2 to G_6 active.

Feasible starting point:

$$x^{(0)} = (250, 250, 250, 250, 250)^T, \quad F(x^{(0)}) = -1,250$$

The objective function as well as the constraints exclusively consist of linear equations. A similar problem was introduced by Schwefel (1995):

Problem 2.41. Schwefel's problem 2.41 (Schwefel, 1995)

Minimize:

$$F(\mathbf{x}) = -\sum_{i=1}^{5}(ix_i)$$

Constraints like problem 2.40.

Minimum:

$$x^* = (0, 0, 0, 0, \frac{50,000}{14})^T, \quad F(x^*) = -\frac{250,000}{14}$$

G_j active for $j = 1, 2, 3, 4, 6$.

Feasible starting point:

$$x^{(0)} = (250, 250, 250, 250, 250)^T, \quad F(x^{(0)}) = -3,750$$

Except of the scaling factor i this problem is structurally similar to problem 2.40. Both problems are flexible concerning their dimensionality, but here we limit to five dimensions. The third problem is the tangent problem, which is based on the sphere model subject to one constraint active at the optimum.

Problem TR (n-dimensional tangent problem)

Minimize:

$$F(\mathbf{x}) = \sum_{i=1}^{n} x_i^2 \quad (n\text{-dim. sphere model})$$

Constraint:

$$g(\mathbf{x}) = \sum_{i=1}^{n} x_i - t \geq 0, \quad t \in \mathbb{R} \text{ (tangent)}$$

For $n=k$ and $t=k$ the optimum is located at:

$$x^* = (1, \ldots, 1)^T, \text{ with } F(x^*) = k.$$

Feasible starting point:

$$x^{(0)} = (50, \ldots, 50)^T, \quad F(x^{(0)}) = n \cdot 50^2,$$

As mentioned in Section 1.4 the tangent problem differs from the other problems in the essential structural feature that the success probability for improvements decreases when approximating the optimum.

A further problem is Himmelblau's nonlinear optimization problem (Himmelblau, 1972), which is often used for the evaluation of constraint-handling methods.

Problem HB (Himmelblau's nonlinear optimization problem)

Minimize:

$$F(\mathbf{x}) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40,792.141$$

Constraints:

$$g_1(\mathbf{x}) = 85.334407 + 0.0056858x_2x_5 + 0.00026x_1x_4 - 0.0022053x_3x_5$$
$$g_2(\mathbf{x}) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2$$
$$g_3(\mathbf{x}) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4$$

$$\begin{gathered} 0 \leq g_1(\mathbf{x}) \leq 92 \\ 90 \leq g_2(\mathbf{x}) \leq 110 \\ 20 \leq g_3(\mathbf{x}) \leq 25 \\ 78 \leq x_1 \leq 102 \\ 33 \leq x_2 \leq 45 \\ 27 \leq x_i \leq 45 \quad (i = 3, 4, 5) \end{gathered}$$

Optimum:

$$x^* = (78.000, 33.000, 29.995, 45.000, 36.776)^T, \quad F(x^*) = -30,665.5$$

Feasible starting point:

$$x^{(0)} = (100, 40, 40, 40, 40)^T, F(x^{(0)}) = -25147.493180000005$$

6.2. *B. Parameter Settings*

In this section the parameter settings of the experimental results are presented. For all experiments the settings of the first part of the table are valid unless other settings are presented in the following parts. The initial mean step sizes are chosen according to the following equation:

$$\sigma_i = \frac{|x^{(0)} - x^*|}{n} \tag{B.1}$$

for $1 \leq i \leq n$ and a problem of dimension n.

Common setting	
Runs	50
Initial step sizes	TR2: $\sigma_i = 34.65$
	2.40: $\sigma_i = 955.25$
	2.41: $\sigma_i = 671.77$
	HB: $\sigma_i = 5.48$
Recombination variables, step sizes	$\rho = 2$, intermediate
Mutation	$n_\sigma = n$
	TR2: $\tau_0 = 0.45$, $\tau = 0.0$
	2.40,2.41, HB: $\tau_0 = \frac{1}{\sqrt{10}}, \tau = \frac{1}{\sqrt{2\sqrt{5}}}$
Termination condition	Fitness stagnation
	TR2: $\theta = 10^{-16}$
	2.40, 2.41: $\theta = 10^{-12}$
DP	
ES-type	(15,100)

Constraint-handling	Death penalty
Dyn	
ES-type	(15,100)
Constraint-handling	Dynamic penalty function
	(Joines and Houck, 1994)
Dynamic penalty parameters	$C=0.5$, $\alpha=1$, $\beta=1$
	HB: $C=10$, $\alpha=2$, $\beta=1$
DSES	
ES-type	(15,100)
Constraint-handling	DSES
TSES	
ES-type	(8 + 8, 13 + 87)
Constraint-handling	TSES
TSES	
	Two-step *c*-selection operator
	TR2: $\kappa=300$
NAES	2.40, 2.41, HB: $\kappa=50$
ES-type	TR2: [3,15(3,15)]
	2.40, 2.41, HB: [5,50(5,50)]
Constraint-handling	NAES
Runs	15
Initial interval, outer ES	$[0,\pi/2]$
Recombination variables, step sizes	$\rho=2$, $\rho'=2$, intermediate
Initial step sizes of outer ES	$\sigma_i=\pi/8$
Mutation, outer ES	$\tau_0=\frac{1}{\sqrt{2q}}, \tau=\frac{1}{\sqrt{2\sqrt{q}}}$ with $q=1/2\cdot n(n-1)$
Mutation, inner ES	$\tau_0=\frac{1}{\sqrt{2n}}, \tau=\frac{1}{\sqrt{2\sqrt{n}}}$
Termination condition, outer ES	TR2: $\theta=10^{-16}$
	2.40, 2.41: $\theta=10^{-7}$
Termination condition, inner ES	TR2: $\theta=10^{-16}$
	2.40, 2.41: $\theta=10^{-9}$
Conditions	Limitation of angles on interval $[0,\pi/2]$

References

Bean JC and Hadj-Alouane AB (1992) A dual genetic algorithm for bounded integer programs. Technical Report TR 92-53, Department of Industrial and Operations Engineering. The University of Michigan

Belur SV (1997) CORE: constrained optimization by random evolution. In: Koza JR (ed.) Late Breaking Papers at the Genetic Programming 1997 Conference, pp. 280–286. Stanford Bookstore, Stanford University California, California

Beyer H-G and Schwefel H-P (2002) Evolution strategies – A comprehensive introduction. Natural Computing 1: 3–52

Coello Coello CA (2000a) Treating constraints as objectives for single-objective evolutionary optimization Engineering Optimization 32(3): 275–308

Coello Coello CA (2000b) Use of a self-adaptive penalty approach for engineering optimization problems Computers in Industry 41(2): 113–127

Coello Coello CA (2002) Theoretical and numerical constraint handling techniques used with evolutionary algorithms: a survey of the state of the art Computer Methods in Applied Mechanics and Engineering 191(11–12): 1245–1287

Deb K (2001) An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanics and Engineering

Fiacco A and McCormick G (1964) The sequential unconstrained minimization technique for nonlinear programming – A primal-dual method. Management Science 10: 360–366

Himmelblau DM (1972) Applied Nonlinear Programming. McGraw-Hill, New York

Hoffmeister F and Sprave J (1996) Problem-independent handling of constraints by use of metric penalty functions. In: Fogel LJ, Angeline PJ and Bäck T (eds) Proceedings of the Fifth Annual Conference on Evolutionary Programming (EP'96), pp. 289–294. The MIT Press, San Diego, California

Homaifar A, Lai SHY and Qi X (1994) Constrained optimization via genetic algorithms. Simulation 62(4): 242–254

Jiménez F and Verdegay JL (1999) Evolutionary techniques for constrained optimization problems. In: Zimmermann H-J (ed.) 7th European Congress on Intelligent Techniques and Soft Computing (EUFIT'99), Verlag Mainz, Aachen

Joines J and Houck C (1994) On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GAs. In: Fogel D (ed.) Proceedings of the first IEEE Conference on Evolutionary Computation, pp. 579–584. IEEE Press, Orlando, Florida

Koziel S and Michalewicz Z (1999) Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. Evolutionary Computation 7: 19–44

Kuri-Morales A and Quezada CV (1998) A universal eclectic genetic algorithm for constrained optimization. In: Proceedings 6th European Congress on Intelligent Techniques & Soft Computing, EUFIT'98, pp. 518–522. Verlag Mainz, Aachen

Michalewicz and Fogel DB (2000) How to Solve It: Modern Heuristics. Springer

Montes EM and Coello Coello CA (2005) A simple multi-membered evolution strategy to solve constrained optimization problems. IEEE Transactions on Evolutionary Computation 9: 1–17

Paredis J (1994) Co-evolutionary constraint satisfaction. In: Proceedings of the 3rd Conference on Parallel Problem Solving from Nature, pp. 46–55. Springer, Berlin

Parmee IC and Purchase G (1994) The development of a directed genetic search technique for heavily constrained design spaces. In: Parmee IC. (ed.) Adaptive Computing in Engineering Design and Control, pp. 97–102. University of Plymouth, Plymouth, UK

Powell D and Skolnick MM (1993) Using genetic algorithms in engineering design optimization with non-linear constraints. In: Forrest S (ed.) Proceedings of the Fifth

International Conference on Genetic Algorithms (ICGA-93), pp. 424–431. Morgan Kaufmann, San Mateo, California

Rechenberg I (1994) Evolutionsstrategie '94. Frommann-Holzboog, Stuttgart

Riche RGL, Knopf-Lenoir C and Haftka RT (1995) A segregated genetic algorithm for constrained structural optimization. In: Eshelman LJ (ed.) Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA-95), pp. 558–565. Morgan Kaufmann, San Mateo, California

Schoenauer M and Michalewicz Z (1996) Evolutionary computation at the edge of feasibility. In: Voigt H-M, Ebeling W, Rechenberg I and Schwefel H-P (eds) Proceedings of the Fourth Conference on Parallel Problem Solving from Nature (PPSN IV), pp. 245–254. Springer, Berlin

Schoenauer M and Xanthakis S (1993) Constrained GA optimization. In: Forrest S (ed.), Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA-93), pp. 573–580, Morgan Kaufmann, San Mateo, California

Schwefel H-P (1974) Adaptive Mechanismen in der biologischen Evolution und ihr Einfluss auf die Evolutionsgeschwindigkeit. Technical Report of the Working Group of Bionics and Evolution Techniques at the Institute for Measurement and Control Technology Re 215/3, Technical University of Berlin

Schwefel H-P (1995) Evolution and Optimum Seeking, Sixth-Generation Computer Technology. Wiley Interscience, New York

Surry PD, Radcliffe NJ and Boyd ID (1995) A multi-objective approach to constrained optimisation of gas supply networks: The COMOGA Method. In: Fogarty TC (ed.) Evolutionary Computing. AISB Workshop. Selected Papers Lecture Notes in Computer Science, pp. 166–180. Springer, Sheffield, UK