Chapter 1

CONSTRAINED EVOLUTIONARY OPTIMIZATION

— the penalty function approach

Thomas Philip Runarsson
Department of Applied Mathematics and Computer Science
Science Institute, University of Iceland
tpr@raunvis.hi.is

Xin Yao
School of Computer Science
The University of Birmingham
x.yao@cs.bham.ac.uk

Abstract The penalty function method has been used widely in constrained evolutionary optimization (CEO). This chapter provides an in-depth analysis of the penalty function method from the point of view of search landscape transformation. The analysis leads to the insight that applying different penalty function methods in evolutionary optimization is equivalent to using different selection schemes. Based on this insight, two constraint handling techniques, i.e., stochastic ranking and global competitive ranking, are proposed as selection schemes in CEO. Our experimental results have shown that both techniques performed very well on a set of benchmark functions. Further analysis of the two techniques explains why they are effective: they introduce few local optima except for those defined by the objective functions.

Keywords: Constrained evolutionary optimization (CEO), penalty function method, ranking.

1. Introduction

The general nonlinear programming problem can be formulated as

$$\text{minimize} \quad f(\mathbf{x}), \quad \mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{R}^n \tag{1.1}$$

where $f(\mathbf{x})$ is the objective function, $\mathbf{x} \in \mathcal{S} \cap \mathcal{F}$, $\mathcal{S} \subseteq \mathcal{R}^n$ defines the *search space* which is an n-dimensional space bounded by the *parametric constraints*

$$\underline{x}_j \leq x_j \leq \overline{x}_j, \quad j \in \{1, \ldots, n\}, \tag{1.2}$$

and the *feasible region* $\mathcal{F}$ is defined by

$$\mathcal{F} = \{\mathbf{x} \in \mathcal{R}^n \mid \mathrm{g}_k(\mathbf{x}) \leq 0 \ \forall \ k \in \{1, \ldots, m\}\}, \tag{1.3}$$

where $\mathrm{g}_k(\mathbf{x}), k \in \{1, \ldots, m\}$, are inequality *constraints*. Equality constraints $\mathrm{h}(\mathbf{x})$ can be approximated by inequality constraints using $|\mathrm{h}(\mathbf{x})| - \delta \leq 0$, where δ is a small positive number that indicates the degree of constraint violation. Only minimization problems are considered in this chapter without loss of generality since $\max\{f(\mathbf{x})\} = -\min\{-f(\mathbf{x})\}$.

The penalty function methods considered in this chapter belong to the exterior penalty approach. They are used widely in evolutionary constrained optimization (ECO), although some of the methods are equally applicable to non-evolutionary optimization algorithms. In contrast to numerous penalty function methods proposed for ECO (Michalewicz and Schoenauer, 1996), few theoretical analysis are available to explain how and why a penalty function method works. This chapter fills in this gap by providing an in-depth analysis of penalty function methods and their relationship to search landscape transformation. Such analysis has led to the development of new constraint handling techniques for CEO. In essence, a penalty function method transforms the search landscape by adding a penalty term to the objective function. Such transformation influences the relative fitness of individuals in a population. It also alters the characteristics of the search landscape, e.g., ruggedness. A previously fit individual according to the objective function might not be fit anymore on the transformed search landscape. Since the primary, if not the only, place in an evolutionary algorithm that fitness is used is selection, it is easy to see that an effective approach to "implementing" a penalty function method is to design a new selection scheme. Two rank-based selection schemes are described in this chapter to illustrate how penalty function methods can be "implemented" effectively by designing new ranking schemes in ECO.

The rest of this chapter is organized as follows. Section 2 analysis the penalty function method in CEO and discusses how different penalty function methods influence evolutionary search. In particular, the relationship between different penalty function methods and the ranking of individuals in a population is discussed in detail. Sections 3 and 4 present the ideas and algorithms of two constraint handling techniques based on ranking, i.e., stochastic ranking (Runarsson and Yao, 2000) and global competitive ranking. Section 5 provides further analysis of penalty function methods and shows how the penalty function method works through two detailed examples. Section 6 gives our experimental results on the two constraint handling techniques. Finally, Section 7 gives a brief summary of this chapter.

2. The Penalty Function Method

Constrained optimization problems have often been transformed into unconstrained ones by adding a measure of the constraint violation to the objective function (Fiacco and McCormick, 1968). This constrained handling technique is known as the penalty function method.

The introduction of the penalty term enables the transformation of a constrained optimization problem into a series of unconstrained ones, e.g.,

$$\psi(\mathbf{x}) = f(\mathbf{x}) + r^{(g)}\ \phi(\mathrm{g}_j(\mathbf{x}); j = 1, \ldots, m) \tag{1.4}$$

where $\phi \geq 0$ is a real valued function which imposes a penalty, $\phi(\mathrm{g}_j(\mathbf{x})$, controlled by a sequence of *penalty coefficients* $\{r^{(g)}\}_0^G$. G indicates the maximum number of generations used in CEO. The general form of function ϕ includes both the generation counter g (for dynamic penalty) and the population (for adaptive penalty). In our current notation, this is reflected in the penalty coefficient $r^{(g)}$. This transformation, i.e. equation (1.4), has been used widely in CEO (Kazarlis and Petridis, 1998; Siedlecki and Sklansky, 1989). In particular, the following quadratic loss function (Fiacco and McCormick, 1968), whose decrease in value represents an approach to the feasible region, has often been used as the *penalty function* (Michalewicz and Attia, 1994; Joines and Houck, 1994):

$$\phi(\mathrm{g}_j(\mathbf{x}); j = 1, \ldots, m) = \sum_{j=1}^{m} \max\{0, \mathrm{g}_j(\mathbf{x})\}^2. \tag{1.5}$$

However, any other penalty function is equally valid. Different penalty functions characterize different problems. It is unlikely that a generic penalty function exists which is optimal for all problems. The introduction of penalties may transform a smooth objective function into a rugged one. The search may then become more easily trapped in local minima. For this reason, it is necessary to develop a penalty function method which attempts to preserve the topology of the objective function and yet enables a CEO algorithm to locate the optimal feasible solution.

The penalty function method may work quite well for some problems. However, deciding an optimal (or near-optimal) value for $r^{(g)}$ turns out to be a difficult optimization problem itself! If $r^{(g)}$ is too small, an infeasible solution may not be penalized enough. Hence an infeasible solution may be evolved by an evolutionary algorithm. If $r^{(g)}$ is too large, then a feasible solution is very likely to be found but could be of very poor quality. A large $r^{(g)}$ discourages the exploration of infeasible regions even in the early stages of evolution. This is particularly ineffective for problems where feasible regions in the whole search space are disjoint. In this case, it may be difficult for an evolutionary algorithm to move from one feasible region to another unless they are very close to each other. Reasonable exploration of infeasible regions may act as bridges connecting two or more different feasible regions. The critical issue here is how much exploration of infeasible regions (i.e., how large $r^{(g)}$ is) should be considered as reasonable. The answer to this question is problem dependent. Even for the same problem, different stages of evolutionary search may require different $r^{(g)}$ values.

There has been some work on the dynamic setting of $r^{(g)}$ values in CEO (Joines and Houck, 1994; Kazarlis and Petridis, 1998; Michalewicz and Attia, 1994). Such work usually relies on a predefined monotonically nondecreasing sequence of $r^{(g)}$ values. This approach worked well for some simple problems but failed for more difficult ones because the optimal setting of $r^{(g)}$ values is problem dependent (Reeves, 1997). A fixed and predefined sequence cannot solve a variety of different problems satisfactorily. A trial-and-error process has to be used in this situation in order to find a proper function for $r^{(g)}$ for each problem, as is done in (Joines and Houck, 1994; Kazarlis and Petridis, 1998).

An adaptive approach, where $r^{(g)}$ values are adjusted dynamically and automatically by an evolutionary algorithm itself, appears to be most promising in tackling different constrained optimization problems. For example, population information can be used to adjust $r^{(g)}$ values adaptively (Smith and Coit, 1997). Different problems lead to different populations in evolutionary search and thus lead to different $r^{(g)}$ values. The advantage of such an adaptive approach is that it can be applied to problems where little prior knowledge is available because there is no need to find a predefined $r^{(g)}$ value, or a sequence of $r^{(g)}$ values.

According to (1.4), different $r^{(g)}$ values lead to different fitness functions. A fit individual under one fitness function may not be fit under a different fitness function. When rank-based selection is used in CEO, finding a near optimal $r^{(g)}$, adaptively, is equivalent to ranking individuals in a population adaptively. Hence, the issue of setting $r^{(g)}$ adaptively becomes how to rank individuals according to their objective and penalty values.

To facilitate later discussion, some notations are first introduced here. The individuals being ranked will be *arbitrarily* assigned some numerical labels. Then for any ranking of individuals, the corresponding permutation $\pi \in \mathcal{P}^\lambda$ will be a function from $\{1, \ldots, \lambda\}$ onto itself, whose arguments are the individuals and whose values are the ranks. The following notation is used: $\pi(i)$ is the rank given to individual i and $\pi^{-1}(j)$ is the individual assigned the rank j. Since $\pi^{-1}(j)$ is the individual assigned the rank j, the bracket notation

$$\pi = \langle \pi^{-1}(1), \pi^{-1}(2), \ldots, \pi^{-1}(\lambda) \rangle$$

corresponds to listing all individuals in their ranked order.

For a given penalty coefficient $r^{(g)} > 0$ let the ranking of λ individuals be

$$\psi(\mathbf{x}_{\pi^{-1}(1)}) \leq \psi(\mathbf{x}_{\pi^{-1}(2)}) \leq \ldots \leq \psi(\mathbf{x}_{\pi^{-1}(\lambda)}) \tag{1.6}$$

where ψ is the transformation function given by equation (1.4). Now examine the adjacent pair $\pi^{-1}(i)$ and $\pi^{-1}(i+1)$ in the ranked order:

$$f_i + r^{(g)}\phi_i \leq f_{i+1} + r^{(g)}\phi_{i+1}, \qquad i \in \{1, \ldots, \lambda - 1\}, \tag{1.7}$$

where notations $f_i = f(\mathbf{x}_{\pi^{-1}(i)})$ and $\phi_i = \phi(\mathrm{g}_j(\mathbf{x}_{\pi^{-1}(i)}), j = 1, \ldots, m))$ are used for convenience.

Define a parameter, $\check{r}_i$, which will be referred to as the *critical penalty coefficient* for the adjacent pair i and $i+1$, as

$$\check{r}_i = (f_{i+1} - f_i)/(\phi_i - \phi_{i+1}), \qquad \text{for } \phi_i \neq \phi_{i+1}. \tag{1.8}$$

For a given choice of $r^{(g)} \geq 0$, there are three different cases which may give rise to Inequality (1.7):

1 $f_i \leq f_{i+1}$ and $\phi_i \geq \phi_{i+1}$: the comparison is said to be *dominated by the objective function* and $0 < r^{(g)} \leq \check{r}_i$ because the objective function f plays the dominant role in determining the inequality. When individuals are feasible, $\phi_i = \phi_{i+1} = 0$ and $\check{r}_i \to \infty$.

2 $f_i \geq f_{i+1}$ and $\phi_i < \phi_{i+1}$: the comparison is said to be *dominated by the penalty function* and $0 < \check{r}_i < r^{(g)}$ because the penalty function ϕ plays the dominant role in determining the inequality.

3 $f_i < f_{i+1}$ and $\phi_i < \phi_{i+1}$: the comparison is said to be *nondominated* and $\check{r}_i < 0$.

When comparing nondominated and feasible individuals, the value of $r^{(g)}$ has no impact on Inequality (1.7). In other words, it does not change the order of ranking of the two individuals. However, the value of $r^{(g)}$ is critical in the first two cases as $\check{r}_i$ is the flipping point that will determine whether the comparison is objective or penalty function dominated. For example, if $r^{(g)}$ is increased to a value greater than $\check{r}_i$ in the first case, individual $\pi^{-1}(i+1)$ would change from a fitter individual into a less fit one. For the entire population, the chosen value of $r^{(g)}$ used for comparisons will determine the fraction of individuals dominated by the objective and penalty functions.

Not all possible $r^{(g)}$ values can influence the ranking of individuals. They have to be within a certain range, i.e. $\underline{r}_g < r^{(g)} < \overline{r}_g$, to influence the ranking, where the lower bound $\underline{r}_g$ is the minimum critical penalty coefficient computed from adjacent individuals ranked only according to the objective function and the upper bound $\overline{r}_g$ is the maximum critical penalty coefficient computed from adjacent individuals ranked only according to the penalty function. In general, there are three different categories of $r^{(g)}$ values (Runarsson and Yao, 2000):

1 $r^{(g)} < \underline{r}^{(g)}$: *All* comparisons are based only on the objective function. $r^{(g)}$ is too small to influence the ranking of individuals. This is called *under-penalization*.

2 $r^{(g)} > \overline{r}^{(g)}$: *All* comparisons are based only on the penalty function. $r^{(g)}$ is so large that the impact of the objective function can be ignored. This is called *over-penalization*.

3 $\underline{r}^{(g)} < r^{(g)} < \overline{r}^{(g)}$: *All* comparisons are based on a combination of objective and penalty functions.

Penalty function methods can be classified into one of the above three categories. Some methods may fall into different categories during different stages in evolutionary search. It is important to understand the difference among these three categories because they indicate which function (or combination of functions) is driving the search process and how search progresses. For example, most dynamic penalty methods start with a low $r^{(g)}$ value (i.e., $r^{(g)} < \underline{r}^{(g)}$) in order to find a good region that may contain both feasible and infeasible individuals. Towards the end of search, a high $r^{(g)}$ value (i.e., $r^{(g)} > \overline{r}^{(g)}$) is often used in order to locate a good feasible individual. Such a dynamic penalty method would work well for problems for which the unconstrained global optimum is close to the constrained global optimum. It is unlikely to work well for problems for which the constrained global optimum is far away from the unconstrained one, because the initial low $r^{(g)}$ value would drive the search towards the unconstrained global optimum and thus further away from the constrained one.

The traditional constraint handling technique used in evolution strategies (ESs) falls roughly into the category of over-penalization since all infeasible individuals are regarded as worse than feasible ones (Schwefel, 1995; Hoffmeister and Sprave, 1996; Deb, 1999; Jiménez and Verdegay, 1999). In fact, canonical evolution strategies allow only feasible individuals in the initial population. To perform constrained optimization, an ES is first used to find a feasible initial population by minimizing the penalty function (Schwefel, 1995, p. 115). Once a feasible initial population is found, the ES algorithm will then minimize the objective function and reject all infeasible solutions generated.

It has been widely recognized that neither under- nor over-penalization is a good constraint handling technique and there should be a balance between preserving feasible individuals and rejecting infeasible ones (Gen and Cheng, 1997; Runarsson and Yao, 2000). Such a balance can be achieved by adjusting our measure of how fit an individual should be in comparison with others. The adjustment can be done explicitly through ranking individuals in evolutionary algorithms. In order to strike the right balance, ranking should be dominated by a mixture of objective and penalty functions. That is, the penalty coefficient $r^{(g)}$ should be within the bounds: $\underline{r}^{(g)} < r^{(g)} < \overline{r}^{(g)}$. It is worth noting that the two bounds are not fixed. They are problem dependent and may change from generation to generation as they are also influenced by the current population.

One way to measure the balance of dominance of objective and penalty functions is to count how many comparisons of adjacent individual pairs are dominated by the objective and penalty functions respectively. Such a number of comparisons can be computed for any given $r^{(g)}$ by counting the number of critical penalty coefficients given by (1.8) which are greater than $r^{(g)}$. If there is a predetermined preference for the number of adjacent comparisons that should be dominated by the penalty function then a corresponding penalty coefficient can be determined.

It is clear from the analysis in this section that all a penalty function method tries to do is to obtain the right balance between objective and penalty functions so that the search moves towards the optimal feasible solution rather than the optimum in the combined feasible and infeasible space. One way to achieve such balance effectively and efficiently is to adjust such balance directly and explicitly. Possible methods of achieving this will be presented in the following two sections.

3. Stochastic Ranking

The ranking procedure introduced in this section is *stochastic ranking* (Runarsson and Yao, 2000) where ranking is achieved by a bubble-sort-like procedure. In this approach a probability P_f of using only the objective function for comparing individuals in the infeasible region of the search space is introduced. That is, given any pair of two adjacent individuals, the probability of comparing them (in order to determine which one is fitter) according to the objective function is 1 if both individuals are feasible, otherwise it is P_f.

The procedure provides a convenient way of balancing the dominance in a ranked set. In the bubble-sort-like procedure, λ individuals are ranked by comparing adjacent individuals in at least λ sweeps[1]. The procedure is halted when no change in the rank ordering occurs within a complete sweep. Figure 1.1 shows the stochastic bubble sort procedure used to rank individuals in a population (Runarsson and Yao, 2000).

If at least one individual is infeasible in an adjacent pair, the probability of an individual winning a comparison, i.e., holding the higher

Stochastic bubble sort $(P_f,\ f,\ \phi)$:
  $\pi(j) = j\ \forall\ j \in \{1, \ldots, \lambda\}$;
   for $i = 1$ to N **do**
     for $j = 1$ to $\lambda - 1$ **do**
       sample $u \in U(0,1)$;
       if $(\phi(\mathbf{x}_{\pi^{-1}(j)}) = \phi(\mathbf{x}_{\pi^{-1}(j+1)}) = 0)$ **or** $(u < P_f)$ **then**
         if $(f(\mathbf{x}_{\pi^{-1}(j)}) > f(\mathbf{x}_{\pi^{-1}(j+1)}))$ **then**
           $swap(\pi^{-1}(j), \pi^{-1}(j+1))$;
         fi
       else
         if $(\phi(\mathbf{x}_{\pi^{-1}(j)}) > \phi(\mathbf{x}_{\pi^{-1}(j+1)}))$ **then**
           $swap(\pi^{-1}(j), \pi^{-1}(j+1))$;
         fi
       fi
     od
     if no *swap* done **break**; **fi**
   od
return (π)

Figure 1.1. Stochastic ranking procedure, where $U(0,1)$ is a uniform random number generator and N is the number of sweeps going through the whole population. When $P_f = 0$ the ranking is equivalent to over-penalization. When $P_f = 1$ the ranking is equivalent to under-penalization. The initial ranking is generated at random.

rank, in the ranking procedure is

$$P_w = P_{fw} P_f + P_{\phi w}(1 - P_f) \tag{1.9}$$

where P_{fw} is the probability of the individual winning according to the objective function and $P_{\phi w}$ is the probability of the individual winning according to the penalty function. In the case where adjacent individuals are both feasible $P_w = P_{fw}$, the probability of winning k more comparisons than losses can be computed. The total number of wins will be $k' = (N + k)/2$ where N is the total number of comparisons made. The probability of winning k' comparisons out of N is given by the binomial distribution

$$P_w(y = k') = \binom{N}{k'} P_w^{k'} (1 - P_w)^{N-k'}. \tag{1.10}$$

The probability of winning *at least* k' comparisons is

$$P'_w(y \geq k') = 1 - \sum_{j=0}^{k'-1} \binom{N}{j} P_w^j (1 - P_w)^{N-j}. \tag{1.11}$$

Equations (1.10) and (1.11) show that the greater the number of comparisons (N) the less influence the initial ranking will have. It is worth noting that the probability P_w usually varies for different individuals in different stages of ranking. Now consider the case where P_w is constant during the entire ranking procedure, which will be true if $f_i < f_j$, $\phi_i > \phi_j$; $j \neq i, j = 1, \ldots, \lambda$. Then $P_{fw} = 1$ and $P_{\phi w} = 0$. If $P_f = 0.5$ is chosen then $P_w = 0.5$. There will be an equal chance for a comparison to be made based on the objective or penalty function. Because we are interested in feasible solutions as the final solution, P_f should be less than 0.5 such that there is a pressure against infeasible solutions. The strength of the pressure can be adjusted easily by adjusting only P_f. When parameter N, the number of sweeps, approaches ∞, the ranking will be determined by P_f. That is, if $P_f > 0.5$, the ranking will be based on the objective function. If $P_f < 0.5$, the ranking is equivalent to over-penalization. Hence, an increase in the number of ranking sweeps is effectively equivalent to changing parameter P_f. Hence, $N = \lambda$ can be fixed and P_f adjusted to achieve the best performance.

The effectiveness and efficiency of stochastic ranking will be evaluated in Section 6 through experimental studies.

4. Global Competitive Ranking

A different method of ranking individuals in a population, in order to strike the right balance between objective and penalty functions, is the deterministic global competitive ranking scheme. In this scheme, an individual i is ranked by comparing it against all other members of the population. This is different from the stochastic ranking approach where only adjacent individuals compete for a given rank. In the global competitive ranking method, special consideration is given to *tied ranks*. In the case of tied ranks the same lower rank will be used. For example, for ranking $\pi = \langle 1,\, 3,\, (2,6),\, 7,(4,5)\rangle$, we should have $\pi(1) = 1$, $\pi(3) = 2$, $\pi(2) = \pi(6) = 3$, $\pi(7) = 5$ and $\pi(4) = \pi(5) = 6$.

Similar to the stochastic ranking approach, it is assumed that either the objective or the penalty function will be used in deciding an individual's rank. P_f indicates the probability that a comparison is done based on the objective function only. The probability that individual i holds its rank $\pi(i)$ when challenged by any other member of the population is,

$$P(\pi(i)) = P_f \frac{\lambda - \pi_f(i)}{\lambda - 1} + (1 - P_f)\frac{\lambda - \pi_\phi(i)}{\lambda - 1}, \tag{1.12}$$

where the permutations π_f and π_ϕ correspond to the ranking of individuals based on the objective and penalty functions, respectively. Equation (1.12) can be used to determine the final ranking. That is, the fitness function for the minimization problem becomes:

$$\psi(\mathbf{x}_i) = P_f \frac{\pi_f(i) - 1}{\lambda - 1} + (1 - P_f)\frac{\pi_\phi(i) - 1}{\lambda - 1}. \tag{1.13}$$

It is clear from the above that P_f can be used easily to bias ranking according to the objective or penalty function. In practice, the probability should take a value $0 < P_f < 0.5$ in order to guarantee that a feasible solution may be found. The close the probability is to 0.5, the greater the emphasis will be on minimizing the objective function. As the P_f approaches 0, not equal to zero, the ranking corresponds an over-penalization. The global competitive ranking scheme, unlike stochastic ranking, is deterministic. It can be summarized by Figure 1.2.

Global competitive ranking $(P_f,\ f,\ \phi)$:
Step 1: Determine the ranking, π_f, π_ϕ:
$f(\mathbf{x}_{\pi_f^{-1}(1)}) \le f(\mathbf{x}_{\pi_f^{-1}(2)}) \le \ldots \le f(\mathbf{x}_{\pi_f^{-1}(\lambda)})$
$\phi(\mathbf{x}_{\pi_\phi^{-1}(1)}) \le \phi(\mathbf{x}_{\pi_\phi^{-1}(2)}) \le \ldots \le \phi(\mathbf{x}_{\pi_\phi^{-1}(\lambda)})$
Step 2. Compute competitive fitness:
$\psi(\mathbf{x}_i) = P_f \frac{\pi_f(i)-1}{\lambda-1} + (1 - P_f)\frac{\pi_\phi(i)-1}{\lambda-1}$.
Step 3. Determine final ranking, π:
$\psi(\mathbf{x}_{\pi^{-1}(1)}) \le \psi(\mathbf{x}_{\pi^{-1}(2)}) \le \ldots \le \psi(\mathbf{x}_{\pi^{-1}(\lambda)})$

Figure 1.2. Global competitive ranking method for constraint handling.

5. How Penalty Methods Work

Convergence and convergence rate are two important issues in stochastic optimization and search algorithms, such as EAs. For a stochastic search procedure, average positive progress towards the global optimum, $\mathbf{x}^*$, is necessary in order to find the optimum efficiently. One approach of measuring progress is to compute the distance travelled between successive generations (Schwefel, 1995) towards $\mathbf{x}^*$. The distance from the best individual in generation (g) to the optimum $\mathbf{x}^*$ should be on average greater than that of the best individual at generation $(g+1)$. That is, the following $\varphi_{\mathbf{x}}$ should be greater than 0:

$$\varphi_{\mathbf{x}} = \mathrm{E}\Big[d\big(\mathbf{x}^{(g)}_{\pi^{-1}(1)}, \mathbf{x}^*\big) - d\big(\mathbf{x}^{(g+1)}_{\pi^{-1}(1)}, \mathbf{x}^*\big)\Big|\mathbf{x}^{(g)}_{\pi^{-1}(1)}, \ldots, \mathbf{x}^{(g)}_{\pi^{-1}(\mu)}\Big], \tag{1.14}$$

where the distance metric $d(\mathbf{x}, \mathbf{x}^*) = \parallel \mathbf{x} - \mathbf{x}^* \parallel$. A similar progress definition is given by (Rudolph, 1997, p. 207) in terms of fitness for the unconstrained problem:

$$\varphi_f = \mathrm{E}\Big[f\big(\mathbf{x}^{(g)}_{\pi^{-1}(1)}\big) - f\big(\mathbf{x}^{(g+1)}_{\pi^{-1}(1)}\big)\Big|\mathbf{x}^{(g)}_{\pi^{-1}(1)}, \ldots, \mathbf{x}^{(g)}_{\pi^{-1}(\mu)}\Big]. \tag{1.15}$$

However, the progress rate computed from fitness values, as the one given by (1.15), indicates the progress towards a local unconstrained minimum only. Progress towards the global minimum in a multimodal landscape can only be computed in terms of the distance and when the global minimum is known (Yao et al., 1999). Computing φ analytically is a difficult theoretical problem although there has been some published work on drift analysis (He and Yao, 2001).

If positive progress towards the global optimum is to be maintained, there must exist at least one parent $\mathbf{x}^{(g)}$ which produces at least one offspring that is closer than itself to the optimum $\mathbf{x}^*$ on average. Consider a simple $(1, \lambda)$ EA where there is only one parent $(\mu = 1)$ at each generation producing λ offspring. The offspring are produced using the following mutation operator:

$$\mathbf{x}^{(g+1)}_{\pi^{-1}(i)} = \mathbf{x}^{(g)}_{\pi^{-1}(1)} + N_i(0, \sigma^2) \quad i = 1, \ldots, \lambda, \tag{1.16}$$

where $N_i(0, \sigma^2)$ is a normally distributed random variable with zero mean and variance σ^2. We can now use two examples to illustrate how a penalty function method works by investigating the relationship between different penalty function methods and progress rates. In particular, we will examine how the progress in terms of fitness corresponds to that in terms of the distance to the global optimum.

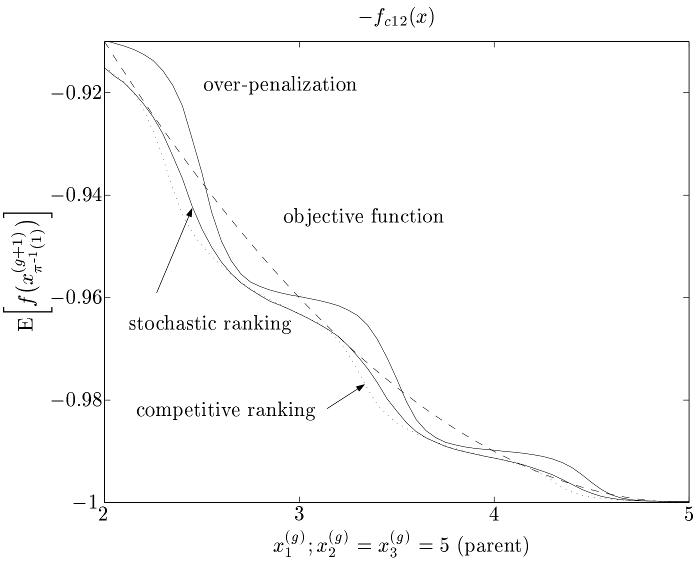

Figure 1.3. Expected fitness of the best offspring as a function of parent position for test function f_{12}. The curves lying below the dashed one (parent fitness) corresponds to positive progress towards the global optimum.

The first example is a the benchmark test function, f_{12} in (Koziel and Michalewicz, 1999):

$$\text{maximize: } f_{12}(\mathbf{x}) = \left(100 - (x_1 - 5)^2 - (x_2 - 5)^2 - (x_3 - 5)^2\right)/100$$

subject to:

$$\mathrm{g}(\mathbf{x}) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0.0625 \leq 0,$$

where $0 \leq x_i \leq 10$ $(i = 1, 2, 3)$ and $p, q, r = 1, 2, \ldots, 9$. The feasible region of the search space consists of 9^3 disjointed spheres. A point (x_1, x_2, x_3) is feasible if and only if there exist p, q, r such that the above inequality holds. Hence, the $\mathrm{g}(\mathbf{x})$ returned corresponds to its lowest value for given p, q, r values. The feasible global optimum is located at $\mathbf{x}^* = (5, 5, 5)$ where $f_{12}(\mathbf{x}^*) = 1$.

Figure 1.3 shows the results of 10,000 one-generational experiments for a number of different parent values. In Figure 1.3, variables x_2 and x_3 were fixed at 5 and only x_1 was adjusted between values 2 and 5. The mean search step size used was $\sigma = 0.2$ and the number of offspring $\lambda = 10$. This simulation was conducted using three different ranking strategies: *over-penalization*, *stochastic ranking*, and *global competitive ranking*. In both the stochastic and global competitive ranking, the value of P_f is 0.45. Over-penalization corresponds to a ranking with $P_f = 0$. The problem was treated as a minimization one.

In Figure 1.3, the expected objective function value of the highest ranked offspring is plotted versus the parent value of x_1. The dashed line corresponds to the objective function value of the parent. Hence, positive progress toward the global optimum will be achieved when the expected objective function value of the best offspring lies beneath the dashed line. The figure illustrates how the over-penalization approach has effectively transformed the original unimodal objective function to a multimodal fitness function. There existed large regions of negative progress when the over-penalization approach was used. The stochastic and global competitive ranking, however, maintained their positive progress towards the global feasible optimum even in infeasible regions, although the rate of progress is slower. This example shows that the penalty function method works by transforming the search landscape (Runarsson, 2000). Inappropriate penalty functions may make the optimization task more difficult than it should be.

The second example is also a well known benchmark test function in (Koziel and Michalewicz, 1999):

$$\text{minimize: } f_{11}(\mathbf{x}) = x_1^2 + (x_2 - 1)^2$$

subject to:

$$\mathrm{h}(\mathbf{x}) = x_2 - x_1^2 = 0,$$

where $-1 \leq x_1 \leq 1$ and $-1 \leq x_2 \leq 1$. The global feasible optimum is at $\mathbf{x}^* = (\pm 1/\sqrt{2}, 1/2)$ where $f_{11}(\mathbf{x}^*) = 0.75$. Figure 1.4 shows the objective function, $f_{11}(\mathbf{x})$, and the constraint curve $\mathrm{h}(\mathbf{x})$.

In this example both parent variables x_1 and x_2 were varied in our experimental study. *Stochastic ranking* ($P_f = 0.45$) was compared with *over-penalization* ($P_f = 0$). Since there exist two optima for this example, the progress was computed in terms of the maximum distance covered towards one of the optima:

$$\begin{aligned} \varphi_x = \mathrm{E}\Big[&\min\big\{d\big(\mathbf{x}^{(g)}_{\pi^{-1}(1)}, \mathbf{y}^*\big), d\big(\mathbf{x}^{(g)}_{\pi^{-1}(1)}, \mathbf{z}^*\big)\big\} \\ &- \min\big\{d\big(\mathbf{x}^{(g+1)}_{\pi^{-1}(1)}, \mathbf{y}^*\big), d\big(\mathbf{x}^{(g+1)}_{\pi^{-1}(1)}, \mathbf{z}^*\big)\big\}\Big] \end{aligned} \tag{1.17}$$

where $\mathbf{z}^*$ and $\mathbf{y}^*$ are the optima $(\pm 1/\sqrt{2}, 1/2)$.

Two different mean step sizes were used in our experiments: $\sigma = 0.05$ and $\sigma = 0.1$. The number of offspring generated was again $\lambda = 10$. The progress rate given by Equation 1.17 is illustrated by contour plots

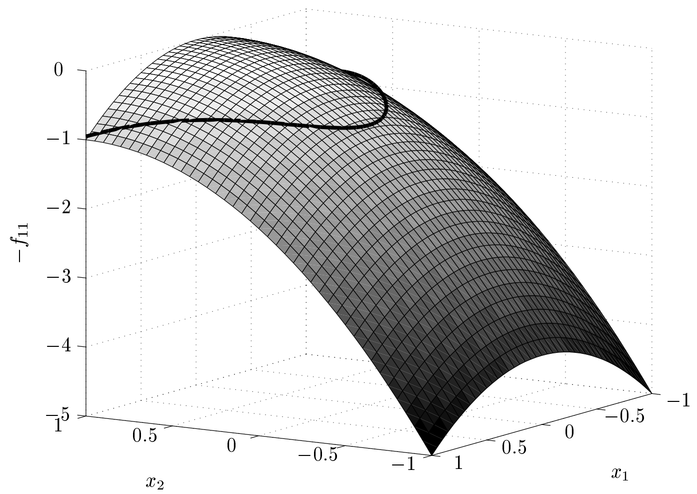

Figure 1.4. Fitness landscape for test function f_{11}. The curve represents the region of feasibility.

shown in Figure 1.5, where regions of negative progress are outlined with contour lines.

It is clear from Figure 1.5 that negative regions of progress were located around the global optima. This is not surprising since the mean search step size used was too large in these regions. A decreasing mean search step size should be used. For the over-penalization approach, however, there existed additional regions of negative progress which were not in the global optimum regions. These regions formed additional local attractors and would trap individuals as the mean search step size decreased. Stochastic ranking did not create any local attractors in this case. This is also true for global competitive ranking, as will be seen in the following section.

In summary, the introduction of constraints may produce additional local optima in the search landscape. A well designed constraint handling technique can minimize the number of such misleading local optima. This is the primary reason why our ranking methods worked so well on many test functions. Our ranking methods also make it easy to control constrained search by adjusting P_f for different problems.

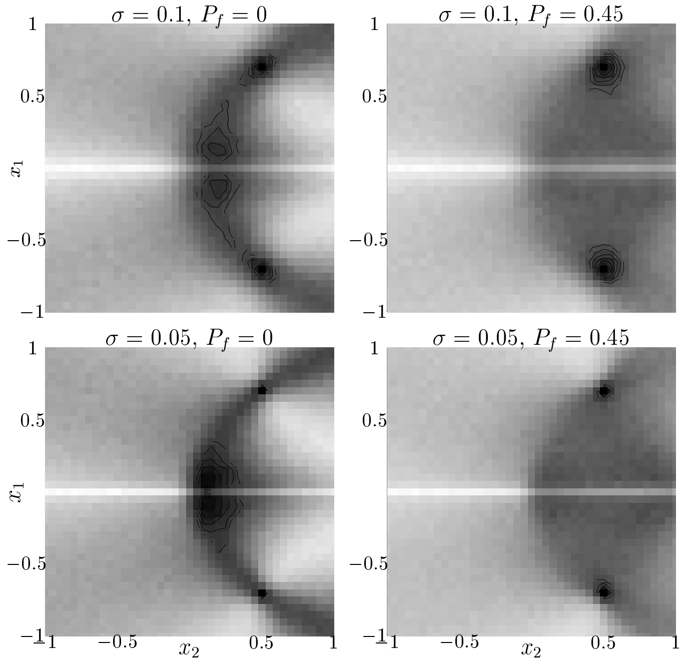

Figure 1.5. The figures show the progress rate in terms of the distance metric, i.e. φ_x where $\mu = 1$ and $\lambda = 10$, for test function f_{11}. The drawn contours mark regions of negative progress (darker regions). When $P_f = 0$ (over-penalization), there exists a region where no progress is maintained towards either global optima, and thus the search will get stuck in this region. This figure explains the poor performance observed in Table 1.1 for this function.

6. Experimental Study

6.1. Evolutionary Optimization Algorithm

The evolutionary optimization algorithm described in this section is based on the evolution strategy (ES) (Schwefel, 1995). One reason for choosing ES is that it does not introduce any specialized constraint-handling variation operators. It will be shown that specialized and complex variation operators for constrained optimization problems are unnecessary although they may be quite useful for particular types of problems (see for example (Michalewicz et al., 1996)). A simple extension to the ES, i.e., the use of the ranking schemes proposed in the previous sections, can achieve significantly better results than other more complicated techniques.

In a (μ, λ)-ES algorithm, an individual i is a pair of real-valued vectors, $(\mathbf{x}_i, \sigma_i)$, $\forall\ i \in \{1, \ldots, \lambda\}$. The initial population of $\mathbf{x}$ is generated according to a uniform n-dimensional probability distribution over the search space $\mathcal{S}$. Let δx be an approximate measure of the expected distance to the global optimum, then the initial setting for the 'mean step sizes' should be (Schwefel, 1995, p. 117):

$$\sigma_{i,j}^{(0)} = \delta x_j / \sqrt{n} \approx (\overline{x}_j - \underline{x}_j)/\sqrt{n}, \quad i \in \{1, \ldots, \lambda\}, j \in \{1, \ldots, n\}, \tag{1.18}$$

where $\sigma_{i,j}$ denotes the j-th component of the vector σ_i. These initial values will also be used as upper bounds on σ.

Following the ranking schemes presented, the evaluated objective $f(\mathbf{x})$ and penalty function $\phi(\mathrm{g}_k(\mathbf{x});\ k = 1, \ldots, m)$ for each individual $(\mathbf{x}_i, \sigma_i)$, $\forall\ i \in \{1, \ldots, \lambda\}$ is used to rank individuals in a population and the best (highest-ranked) μ individuals out of λ are selected for the next generation. The truncation level is set at $\mu/\lambda \approx 1/7$ (Bäck, 1996, p. 79).

Variation of strategy parameters is performed before the modification of objective variables. New λ strategy parameters are produced from the μ highest ranked individuals and then applied later for generating λ offspring. The 'mean step sizes' are updated according to the log-normal update rule (Schwefel, 1995): $i = 1, \ldots, \mu$, $h = 1, \ldots, \lambda$, and $j = 1, \ldots, n$,

$$\sigma_{h,j}^{(g+1)} = \hat{\sigma}_{h,j}^{(g)} \exp(\tau' N(0,1) + \tau N_j(0,1))), \tag{1.19}$$

where $N(0,1)$ is a normally distributed one-dimensional random variable with an expectation of zero and variance one. The subscript j in $N_j(0,1)$ indicates that the random number is generated anew for each value of j. The 'learning rates' τ and τ' are set equal to $\varphi^*/\sqrt{2\sqrt{n}}$ and $\varphi^*/\sqrt{2n}$ respectively where φ^* is the expected rate of convergence (Schwefel, 1995, p. 144) and is set to one (Bäck, 1996, p. 72). Recombination is performed on the self-adaptive parameters before applying the update rule given by (1.19). In particular, global intermediate recombination (the average) between two parents (Schwefel, 1995, p. 148) is implemented as

$$\hat{\sigma}_{h,j}^{(g)} = (\sigma_{i,j}^{(g)} + \sigma_{k_j,j}^{(g)})/2, \quad k_j \in \{1, \ldots, \mu\}, \tag{1.20}$$

where k_j is an index generated at random and anew for each j.

Having varied the strategy parameters, each individual $(\mathbf{x}_i, \sigma_i)$, $\forall\ i \in \{1, \ldots, \mu\}$, creates λ/μ offspring on average, so that a total of λ offspring are generated:

$$x_{h,j}^{(g+1)} = x_{i,j}^{(g)} + \sigma_{h,j}^{(g+1)} N_j(0,1) \tag{1.21}$$

Table 1.1. Over-penalization.

fcn	optimal	best	median	st. dev.	G_m
f_1	−15.000	−15.000	−15.000	0.0E+00	697
f_2	−0.803619	−0.803578	−0.785253	1.5E−02	1259
f_3	−1.000	−0.327	−0.090	7.2E−02	61
f_4	−30665.539	−30665.539	−30665.538	3.8E+00	632
f_5	5126.498	5126.945	5225.100	2.7E+02	213
f_6	−6961.814	−6961.814	−6961.814	1.9E−12	946
f_7	24.306	24.322	24.367	5.9E−02	546
f_8	−0.095825	−0.095825	−0.095825	2.7E−17	647
f_9	680.630	680.632	680.657	3.8E−02	414
f_{10}	7049.331	7117.416	7336.280	3.4E+02	530
f_{11}	0.750	0.750	0.953	5.4E−02	1750
f_{12}	−1.000000	−0.999972	−0.999758	1.4E−04	90
f_{13}	0.053950	0.919042	0.997912	1.5E−02	1750

Recombination is not used in the variation of objective variables. When an offspring is generated outside the parametric bounds defined by the problem, the mutation (variation) of the objective variable will be retried until the variable is within its bounds. In order to save computation time the mutation is retried only 10 times and then ignored, leaving the object variable in its original state within the parameter bounds.

6.2. Experimental Results and Discussion

Thirteen benchmark functions are studied. The first 12 are taken from (Koziel and Michalewicz, 1999) and the 13th from (Michalewicz, 1995). The details, including the original sources, of these functions are listed in appendix 1.A. Functions f_2, f_3, f_8, and f_{12} are maximization problems. They are transformed to minimization problems using $-f(\mathbf{x})$. For each of the benchmark problems 30 independent runs are performed using a $(30, 200)$-ES and the ranking procedures described in the previous sections. All runs are terminated after $G = 1750$ generations except for f_{12}, which was run for 175 generations. The experimental results using the stochastic and global competitive ranking, with $P_f = 0.45$, are given in Tables 1.2 to 1.3. The results are compared against the over-penalization approach (Table 1.1) used in ES (Hoffmeister and Sprave, 1996). The over-penalization approach corresponds to the ranking schemes discussed for $P_f \to 0$. In the tables the best feasible objective value, median, standard deviation, and median number of generations (G_m) needed to find the best individual are given.

Table 1.2. Stochastic ranking ($P_f = 0.45$).

fcn	optimal	best	median	st. dev.	G_m
f_1	-15.000	-15.000	-15.000	0.0E+00	741
f_2	-0.803619	-0.803515	-0.785800	2.0E−02	1086
f_3	-1.000	-1.000	-1.000	1.9E−04	1146
f_4	-30665.539	-30665.539	-30665.539	2.0E−05	441
f_5	5126.498	5126.497	5127.372	3.5E+00	258
f_6	-6961.814	-6961.814	-6961.814	1.6E+02	590
f_7	24.306	24.307	24.357	6.6E−02	715
f_8	-0.095825	-0.095825	-0.095825	2.6E−17	381
f_9	680.630	680.630	680.641	3.4E−02	557
f_{10}	7049.331	7054.316	7372.613	5.3E+02	642
f_{11}	0.750	0.750	0.750	8.0E−05	57
f_{12}	-1.000000	-1.000000	-1.000000	0.0E+00	82
f_{13}	0.053950	0.053957	0.057006	3.1E−02	349

Table 1.3. Global competitive ranking ($P_f = 0.45$).

fcn	optimal	best	median	st. dev.	G_m
f_1	-15.000	-15.000	-15.000	0.0E+00	692
f_2	-0.803619	-0.803591	-0.792805	1.7E−02	1335
f_3	-1.000	-1.000	-1.000	2.6E−05	1725
f_4	-30665.539	-30665.539	-30665.538	5.4E−01	731
f_5 [2]	5126.498	5126.497	5126.721	1.1E+00	319
f_6	-6961.814	-6943.560	-6579.214	2.9E+02	13
f_7	24.306	24.308	24.361	1.1E−01	517
f_8	-0.095825	-0.095825	-0.095825	2.6E−17	398
f_9	680.630	680.631	680.657	5.8E−02	396
f_{10}	7049.331	–	–	–	–
f_{11}	0.750	0.750	0.750	7.2E−05	76
f_{12}	-1.000000	-1.000000	-1.000000	0.0E+00	63
f_{13}	0.053950	0.053943	0.053987	1.3E−04	247

As can be seen from Tables 1.1 to 1.3, both stochastic ranking and global competitive ranking performed very well for most test functions, especially for functions f_3, f_{11}, f_{12}, and f_{13}, for the reasons given in Section 5. They are also much faster than the over-penalization approach for most test functions. There are, however, two test functions that stand out: f_{10} and f_6. It is difficult to determine whether it is the constraint handling technique or the underlying search method which is contributing to the success or failure in locating the optimum. In (Runarsson and

Table 1.4. Over-penalization versus stochastic ranking for test function f_{10} and $\varphi = 1/4$.

P_f	optimal	best	median	st. dev.	G_m
0.45	7049.331	7049.852	7054.111	5.7E+00	1733
0.00	7049.331	7049.955	7062.673	3.1E+01	1745

Yao, 2000) the importance of the search method was illustrated on test function f_{10} by setting $\varphi = 1/4$. This results is given in table 1.4 and illustrates how significant the search method is.

Test function f_6 is the only test function solved more effectively using over-penalization. For this reason it is interesting to plot its progress rate landscape. The test function has two variables. The progress rate is simulated as before using 10.000 one generational experiments in the region where suboptimal solutions are found. The result is depicted in figure 1.6. Progress landscapes for the step sizes $\sigma = 0.05$ (dotted) and $\sigma = 0.01$ (dashed) are plotted as contours. Negative progress is maintained to the right of the last of the three contour lines plotted. The solid lines in the figure are the constraint curves and the circle marks the location of $\mathbf{x}^*$. The feasible region is the top narrow band formed by the two constraint curves. From the figure it becomes clear that in this case over-penalization guides the search directly to the optimal feasible solutions from the infeasible region. However, stochastic ranking approaches the optimal solution from the combined feasible and infeasible region. The progress contours are simply rotated. In this test case no additional

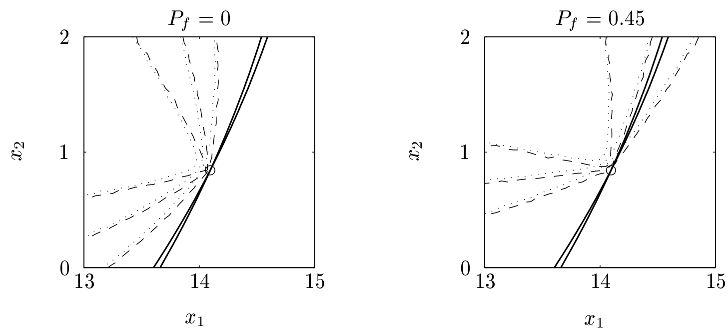

Figure 1.6. Progress landscape for test function f_6 for step sizes $\sigma = 0.05$ (dotted) and $\sigma = 0.01$ (dashed). Negative progress is to the right of the last of the three contour lines. The solid lines are the constraint curves and the circle the location of $\mathbf{x}^*$. The feasible region is the top narrow band formed by the two constraint curves.

attractors are created by the over-penalization method and therefore the two approaches should yield similar performance. This leads one to speculate whether the performance difference may be due to the lack of rotational invariance of the search method. To test this the coordinate system is rotated by $\pi/4$ and the experiment is re-run. The results are given in table 1.5. This simple experiment supports our prediction that the performance difference is due to the lack of rotational invariance of the search method.

Table 1.5. Over-penalization versus stochastic ranking for test function f_6 and coordinate system rotated by $\pi/4$.

P_f	best	median	mean	st. dev.	worst	G_m
0.45	−6954.352	−6913.419	−6909.142	2.7E+01	−6842.484	957
0.00	−6942.806	−6903.223	−6887.683	4.2E+01	−6782.945	864

7. Conclusion

The penalty function method is widely used in constrained optimization. It is emphasized in this chapter that the penalty function method transforms a constrained problem into an unconstrained one by modifying the search landscape. Different modifications lead to different search landscapes and thus different difficulties of optimization. We have given two concrete examples to illustrate how additional local optima could be introduced through inappropriate penalty methods and how such local optima could mislead search.

Selection in an EA depends primarily on fitness values of individuals. Modifications to a search (fitness) landscape can be achieved through modifications to the selection scheme, rather than to the fitness function. Ranking is a simple yet effective selection method that can be used to indicate which individuals are fitter than others and thus achieve the goal of modifying the fitness landscape. Two ranking schemes have been introduced in this paper to show how they can be used to handle constraints effectively and efficiently without adding a penalty term in the fitness function. Experimental results on a set of benchmark test functions are given in this chapter to support our analysis.

Notes

1. It would be exactly λ sweeps if the comparisons were not made stochastic.
2. Statistics based on 11 feasible solutions found.

References

Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, New York.

Deb, K. (1999). An efficient constrained handling method for genetic algorithms. In *Computer Methods in Applied Mechanics and Engineering*, page in press.

Fiacco, A. V. and McCormick, G. P. (1968). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques.* Wiley, New-York.

Floundas, C. and Pardalos, P. (1987). *A Collection of Test Problems for Constrained Global Optimization*, volume 455 of *Lecture Notes in Computar Science.* Springer-Verlag, Berlin, Germany.

Gen, M. and Cheng, R. (1997). *Genetic Algorithms and Engineering Design.* Wiley, New-York.

He, J. and Yao, X. (2001). Drift analysis and average time complexity of evolutionary algorithms. *Artificial Intelligence*, 127(1):57–85.

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

Hock, W. and Schittkowski, K. (1981). *Test Examples for Nonlinear Programming Codes.* Lecture Notes in Economics and Mathematical Systems. Springer-Verlag, Berlin, Germany.

Hoffmeister, F. and Sprave, J. (1996). Problem independent handling of constraints by use of metric penalty functions. In Fogel, L. J., Angeline, P. J., and Bäck, T., editors, *Proceedings of the Fifth Annual Conference on Evolutionary Programming*, pages 289–294, Cambridge MA. The MIT Press.

Jiménez, F. and Verdegay, J. L. (1999). Evolutionary techniques for constrained optimization problems. In *Proc. of the 7th European Congress on Intelligent Techniques and Soft Computing (EUFIT'99)*, Germany, Berlin. Springer-Verlag.

Joines, J. and Houck, C. (1994). On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GAs. In *Proc. IEEE International Conference on Evolutionary Computation*, pages 579–584. IEEE Press.

Kazarlis, S. and Petridis, V. (1998). Varying fitness functions in genetic algorithms: Studying the rate of increase in the dynamic penalty terms. In *Parallel Problem Solving from Nature*, volume 1498 of *Lecture Notes in Computer Science*, pages 211–220, Berlin, Germany. Springer.

Koziel, S. and Michalewicz, Z. (1999). Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. *Evolutionary Computation*, 7(1):19–44.

Michalewicz, Z. (1995). Genetic algorithms, numerical optimization and constraints. In Eshelman, L., editor, *Proceedings of the 6th International Conference on Genetic Algorithms*, pages 151–158, San Mateo, CA. Morgan Kaufman.

Michalewicz, Z. and Attia, N. (1994). Evolutionary optimization of constrained problems. In Fogel, L. J. and Sebald, A., editors, *Proc. of the 2nd Annual Conference on Evolutionary Programming*, pages 98–108, River Edge, NJ. World Scientific Publishing.

Michalewicz, Z., Nazhiyath, G., and Michalewicz, M. (1996). A note on usefulness of geometrical crossover for numerical optimization problems. In Fogel, L., Angeline, P., and Bäck, T., editors, *Proc. of the 5th Annual Conference on Evolutionary Programming*, pages 305–312. MIT Press, Cambridge, MA.

Michalewicz, Z. and Schoenauer, M. (1996). Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation*, 4(1):1–32.

Reeves, C. R. (1997). Genetic algorithms for the operations researcher. *INFORMS Journal on Computing*, 9(3):231–247.

Rudolph, G. (1997). *Convergence Properties of Evolutionary Algorithms.* Verlag Dr. Kovač, Hamburg.

Runarsson, T. P. (2000). *Evolutionary Problem Solving.* PhD thesis, University of Iceland, Reykjavik, Iceland.

Runarsson, T. P. and Yao, X. (2000). Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 4(3):284–294.

Schwefel, H.-P. (1995). *Evolution and Optimum Seeking.* Wiley, New-York.

Siedlecki, W. and Sklansky, J. (1989). Constrained genetic optimization via dynamic reward-penalty balancing and its use in pattern recognition. In *International Conference on Genetic Algorithms*, pages 141–149.

Smith, A. E. and Coit, D. W. (1997). Penalty functions. In Bäck, T., Fogel, D. B., and Michalewicz, Z., editors, *Handbook on Evolutionary Computation*, pages C5.2:1–6. Oxford University Press.

Yao, X., Liu, Y., and Lin, G. (1999). Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, 3(2):82–102.

Appendix: Test Function Suite

Minimize (Floundas and Pardalos, 1987):

$$f_1(\mathbf{x}) = 5\sum_{i=1}^{4} x_i - 5\sum_{i=1}^{4} x_i^2 - \sum_{i=5}^{13} x_i$$

subject to:

$$\begin{aligned}
g_1(\mathbf{x}) &= 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \le 0 \\
g_2(\mathbf{x}) &= 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \le 0 \\
g_3(\mathbf{x}) &= 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \le 0 \\
g_4(\mathbf{x}) &= -8x_1 + x_{10} \le 0 \\
g_5(\mathbf{x}) &= -8x_2 + x_{11} \le 0 \\
g_6(\mathbf{x}) &= -8x_3 + x_{12} \le 0 \\
g_7(\mathbf{x}) &= -2x_4 - x_5 + x_{10} \le 0 \\
g_8(\mathbf{x}) &= -2x_6 - x_7 + x_{11} \le 0 \\
g_9(\mathbf{x}) &= -2x_8 - x_9 + x_{12} \le 0
\end{aligned}$$

where the bounds are $0 \le x_i \le 1$ $(i = 1, \ldots, 9)$, $0 \le x_i \le 100$ $(i = 10, 11, 12)$ and $0 \le x_{13} \le 1$. The global minimum is at $\mathbf{x}^* = (1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$ where six constraints are active (g_1, g_2, g_3, g_7, g_8 and g_9) and $f_1(\mathbf{x}^*) = -15$.

Maximize (Koziel and Michalewicz, 1999):

$$f_2(\mathbf{x}) = \left| \frac{\sum_{i=1}^{n} \cos^4(x_i) - 2\prod_{i=1}^{n} \cos^2(x_i)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \right|$$

subject to:

$$\begin{aligned}
g_1(\mathbf{x}) &= 0.75 - \prod_{i=1}^{n} x_i \le 0 \\
g_2(\mathbf{x}) &= \sum_{i=1}^{n} x_i - 7.5n \le 0
\end{aligned}$$

where $n = 20$ and $0 \le x_i \le 10$ $(i = 1, \ldots, n)$. The global maximum is unknown, the best we found is $f_2(\mathbf{x}^*) = 0.803619$ (which, to the best of our knowledge, is better than any reported value), constraint g_1 is close to being active ($g_1 = -10^{-8}$).

Maximize (Michalewicz et al., 1996):

$$f_3(\mathbf{x}) = (\sqrt{n})^n \prod_{i=1}^{n} x_i$$

$$h_1(\mathbf{x}) = \sum_{i=1}^{n} x_i^2 - 1 = 0$$

where $n = 10$ and $0 \leq x_i \leq 1$ $(i = 1, \ldots, n)$. The global maximum is at $x_i^* = 1/\sqrt{n}$ $(i = 1, \ldots, n)$ where $f_3(\mathbf{x}^*) = 1$.

Minimize (Himmelblau, 1972):

$$f_4(\mathbf{x}) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141$$

subject to:

$$\begin{aligned}
\mathrm{g}_1(\mathbf{x}) &= 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leq 0\\
\mathrm{g}_2(\mathbf{x}) &= -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leq 0\\
\mathrm{g}_3(\mathbf{x}) &= 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \leq 0\\
\mathrm{g}_4(\mathbf{x}) &= -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leq 0\\
\mathrm{g}_5(\mathbf{x}) &= 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \leq 0\\
\mathrm{g}_6(\mathbf{x}) &= -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leq 0
\end{aligned}$$

where $78 \leq x_1 \leq 102$, $33 \leq x_2 \leq 45$ and $27 \leq x_i \leq 45$ $(i = 3, 4, 5)$. The optimum solution is $\mathbf{x}^*$ =(78, 33, 29.995256025682, 45, 36.775812905788) where $f_4(\mathbf{x}^*) = -30665.539$. Two constraints are active (g_1 and g_6).

Minimize (Hock and Schittkowski, 1981):

$$f_5(\mathbf{x}) = 3x_1 + 0.000001x_1^3 + 2x_2 + (0.000002/3)x_2^3$$

subject to:

$$\begin{aligned}
\mathrm{g}_1(\mathbf{x}) &= -x_4 + x_3 - 0.55 \leq 0\\
\mathrm{g}_2(\mathbf{x}) &= -x_3 + x_4 - 0.55 \leq 0\\
\mathrm{h}_3(\mathbf{x}) &= 1000\sin(-x_3 - 0.25) + 1000\sin(-x_4 - 0.25) + 894.8 - x_1 = 0\\
\mathrm{h}_4(\mathbf{x}) &= 1000\sin(x_3 - 0.25) + 1000\sin(x_3 - x_4 - 0.25) + 894.8 - x_2 = 0\\
\mathrm{h}_5(\mathbf{x}) &= 1000\sin(x_4 - 0.25) + 1000\sin(x_4 - x_3 - 0.25) + 1294.8 = 0
\end{aligned}$$

where $0 \leq x_1 \leq 1200$, $0 \leq x_2 \leq 1200$, $-0.55 \leq x_3 \leq 0.55$ and $-0.55 \leq x_4 \leq 0.55$. The best known solution (Koziel and Michalewicz, 1999) $\mathbf{x}^* = (679.9453, 1026.067, 0.1188764, -0.3962336)$ where $f_5(\mathbf{x}^*) = 5126.4981$.

Minimize (Floundas and Pardalos, 1987):

$$f_6(\mathbf{x}) = (x_1 - 10)^3 + (x_2 - 20)^3$$

subject to:

$$\begin{aligned}
\mathrm{g}_1(\mathbf{x}) &= -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0\\
\mathrm{g}_2(\mathbf{x}) &= (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0
\end{aligned}$$

where $13 \leq x_1 \leq 100$ and $0 \leq x_2 \leq 100$. The optimum solution is $\mathbf{x}^* = (14.095, 0.84296)$ where $f_6(\mathbf{x}^*) = -6961.81388$. Both constraints are active.

Minimize (Hock and Schittkowski, 1981):

$$\begin{aligned} f_7(\mathbf{x}) = \; & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + \\ & 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45 \end{aligned}$$

subject to:

$$\begin{aligned} \mathrm{g}_1(\mathbf{x}) &= -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0 \\ \mathrm{g}_2(\mathbf{x}) &= 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0 \\ \mathrm{g}_3(\mathbf{x}) &= -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0 \\ \mathrm{g}_4(\mathbf{x}) &= 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0 \\ \mathrm{g}_5(\mathbf{x}) &= 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0 \\ \mathrm{g}_6(\mathbf{x}) &= x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0 \\ \mathrm{g}_7(\mathbf{x}) &= 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leq 0 \\ \mathrm{g}_8(\mathbf{x}) &= -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0 \end{aligned}$$

where $-10 \leq x_i \leq 10$ $(i = 1, \ldots, 10)$. The optimum solution is $\mathbf{x}^* = (2.171996$, 2.363683, 8.773926, 5.095984, 0.9906548, 1.430574, 1.321644, 9.828726, 8.280092, $8.375927)$ where $f_7(\mathbf{x}^*) = 24.3062091$. Six constraints are active (g_1, g_2, g_3, g_4, g_5 and g_6).

Maximize (Koziel and Michalewicz, 1999):

$$f_8(\mathbf{x}) = \frac{\sin^3(2\pi x_1)\sin(2\pi x_2)}{x_1^3(x_1 + x_2)}$$

subject to:

$$\begin{aligned} \mathrm{g}_1(\mathbf{x}) &= x_1^2 - x_2 + 1 \leq 0 \\ \mathrm{g}_2(\mathbf{x}) &= 1 - x_1 + (x_2 - 4)^2 \leq 0 \end{aligned}$$

where $0 \leq x_1 \leq 10$ and $0 \leq x_2 \leq 10$. The optimum is located at $\mathbf{x}^* = (1.2279713, 4.2453733)$ where $f_8(\mathbf{x}^*) = 0.095825$. The solution lies within the feasible region.

Minimize (Hock and Schittkowski, 1981):

$$\begin{aligned} f_9(\mathbf{x}) = \; & (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + \\ & 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \end{aligned}$$

subject to:

$$\begin{aligned} \mathrm{g}_1(\mathbf{x}) &= -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0 \\ \mathrm{g}_2(\mathbf{x}) &= -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0 \\ \mathrm{g}_3(\mathbf{x}) &= -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0 \\ \mathrm{g}_4(\mathbf{x}) &= 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0 \end{aligned}$$

where $-10 \leq x_i \leq 10$ for $(i = 1, \ldots, 7)$. The optimum solution is $\mathbf{x}^* = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227)$ where $f_9(\mathbf{x}^*) = 680.6300573$. Two constraints are active (g_1 and g_4).

Minimize (Hock and Schittkowski, 1981):

$$f_{10}(\mathbf{x}) = x_1 + x_2 + x_3$$

subject to:

$$\begin{aligned}
\mathrm{g}_1(\mathbf{x}) &= -1 + 0.0025(x_4 + x_6) \leq 0 \\
\mathrm{g}_2(\mathbf{x}) &= -1 + 0.0025(x_5 + x_7 - x_4) \leq 0 \\
\mathrm{g}_3(\mathbf{x}) &= -1 + 0.01(x_8 - x_5) \leq 0 \\
\mathrm{g}_4(\mathbf{x}) &= -x_1x_6 + 833.33252x_4 + 100x_1 - 83333.333 \leq 0 \\
\mathrm{g}_5(\mathbf{x}) &= -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leq 0 \\
\mathrm{g}_6(\mathbf{x}) &= -x_3x_8 + 1250000 + x_3x_5 - 2500x_5 \leq 0
\end{aligned}$$

where $100 \leq x_1 \leq 10000$, $1000 \leq x_i \leq 10000$ $(i = 2, 3)$ and $10 \leq x_i \leq 1000$ $(i = 4, \ldots, 8)$. The optimum solution is $\mathbf{x}^* = (579.3167, 1359.943, 5110.071, 182.0174, 295.5985, 217.9799, 286.4162, 395.5979)$ where $f_{10}(\mathbf{x}^*) = 7049.3307$. Three constraints are active (g_1, g_2 and g_3).

Minimize (Koziel and Michalewicz, 1999):

$$f_{11}(\mathbf{x}) = x_1^2 + (x_2 - 1)^2$$

subject to:

$$\mathrm{h}(\mathbf{x}) = x_2 - x_1^2 = 0$$

where $-1 \leq x_1 \leq 1$ and $-1 \leq x_2 \leq 1$. The optimum solution is $\mathbf{x}^* = (\pm 1/\sqrt{2}, 1/2)$ where $f_{11}(\mathbf{x}^*) = 0.75$.

Maximize (Koziel and Michalewicz, 1999):

$$f_{12}(\mathbf{x}) = (100 - (x_1 - 5)^2 - (x_2 - 5)^2 - (x_3 - 5)^2)/100$$

subject to:

$$\mathrm{g}(\mathbf{x}) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0.0625 \leq 0$$

where $0 \leq x_i \leq 10$ $(i = 1, 2, 3)$ and $p, q, r = 1, 2, \ldots, 9$. The feasible region of the search space consists of 9^3 disjointed spheres. A point (x_1, x_2, x_3) is feasible if and only if there exist p, q, r such that the above inequality holds. The optimum is located at $\mathbf{x}^* = (5, 5, 5)$ where $f_{12}(\mathbf{x}^*) = 1$. The solution lies within the feasible region.

Minimize (Hock and Schittkowski, 1981):

$$f_{13}(\mathbf{x}) = e^{x_1x_2x_3x_4x_5}$$

subject to:

$$
\begin{aligned}
\mathrm{h}_1(\mathbf{x}) &= x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10 = 0 \\
\mathrm{h}_2(\mathbf{x}) &= x_2 x_3 - 5 x_4 x_5 = 0 \\
\mathrm{h}_3(\mathbf{x}) &= x_1^3 + x_2^3 + 1 = 0
\end{aligned}
$$

where $-2.3 \leq x_i \leq 2.3$ $(i = 1, 2)$ and $-3.2 \leq x_i \leq 3.2$ $(i = 3, 4, 5)$. The optimum solution is $\mathbf{x}^* = (-1.717143,\ 1.595709,\ 1.827247,\ -0.7636413,\ -0.763645)$ where $f_{13}(\mathbf{x}^*) = 0.0539498$.