Author: Kruisselbrink, Johannes Willem
Title: Evolution strategies for robust optimization
Date: 2012-05-10

Chapter 2

Optimization

This chapter lays out the background of this work by providing a summary of black-box optimization. It introduces the "classical" view on optimization together with the concepts and terminology that are commonly used within this context. This classical view on optimization will be extended in Chapter 3 to form a definition of robust optimization.

Section 2.1 starts with providing a global description and a formal definition of a black-box optimization problem as it will be used in this thesis. Section 2.2 discusses how the practice of optimization is generally perceived in practical applications. Section 2.3 zooms in on the concept of objective function landscape, which is a frequently used metaphor for perceiving optimization problems. Section 2.4 focuses on real-parameter optimization problems, being the main type of optimization problem discussed in this thesis. Section 2.5 provides an overview of black-box optimization algorithms and the general goal of automated optimization. Section 2.6 closes with a summary and discussion.

2.1 Optimization Problems

The model of Figure 2.1 describes an optimization problem. It considers a system that produces output y as a function of input x. Keeping it as general as possible, we note that x and y can be of any form and assume that there is no knowledge about the internal mechanisms of the system, i.e., it is considered to be a black-box. Given such a system and a large number of possible input settings, the central problem of optimization can be loosely formulated as:

What setting(s) of the input x yield(s) the best possible (optimal) output y?

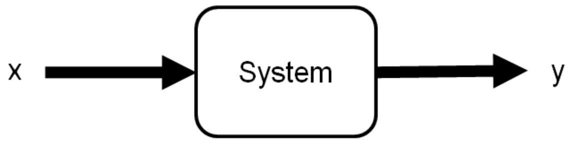

Figure 2.1: The general black-box model of a system.

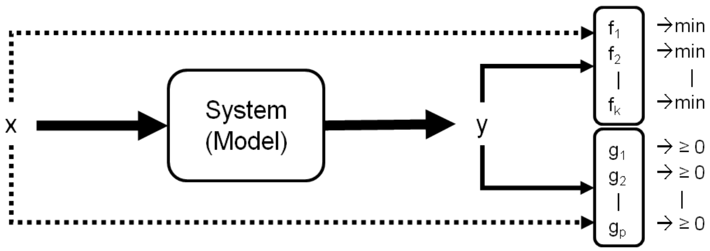

Figure 2.2: The general model of an optimization problem.

To deal with problems of this type, the model of Figure 2.1 is commonly transformed into a form as depicted in Figure 2.2. In many cases the system that is considered is replaced by an abstract model of the system (e.g., a mathematical model or a simulator). Furthermore, one or more objective functions $f_1, \dots, f_k$ and optionally also a number of constraint functions $g_1, \dots, g_p$ are introduced. The objective functions are of the form $f_i : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and assign score values to each input $x \in \mathcal{X}$ based on its respective output $y \in \mathcal{Y}$. Note that we could also neglect the intermediate mapping $\mathcal{X} \to \mathcal{Y}$, which yields the more common form $f_i : \mathcal{X} \to \mathbb{R}$. Without loss of generality, we assume that each score function is to be minimized. The constraint functions are of the form $g_i : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and rate the feasibility of each possible input x, again using y. Also here the intermediate mapping could be neglected to obtain the more common form $g_i : \mathcal{X} \to \mathbb{R}$. For a possible input x, it should hold that $g_i(x) \geq 0$ in order for x to be feasible[1].

The model of an optimization problem of Figure 2.2 can also be described mathematically:

Definition 2.1.1 (Optimization Problem)**:** An *optimization problem* is a triple $(\mathcal{X}, \mathcal{F}, \mathcal{G})$, where:

- $\mathcal{X}$ is the search space, which is the nonempty set of all possible solutions.
- $\mathcal{F} = \{f_1, \dots, f_k\}$, $k \in \mathbb{N}_1$, is a set of one or more objective functions that are to be minimized. Each objective function is a function of the form $f : \mathcal{X} \to \mathbb{R}$ that maps elements of the search space to a score value.
- $\mathcal{G} = \{g_1, \dots, g_p\}$, $p \in \mathbb{N}_0$, is a set of constraint functions that need to be satisfied. Each constraint function is of the form $g : \mathcal{X} \to \mathbb{R}$ mapping elements of the search space to a constraint value. For a certain input $x \in \mathcal{X}$ a constraint g is said to be satisfied if and only if $g(x) \geq 0$. Otherwise, if $g(x) < 0$, then solution x violates the constraint and is therefore infeasible.

[1]Constraints of the form $g(x) \geq 0$ are referred to as inequality constraints. In literature, also another type of constraint is used, namely the equality constraint, which is of the form $h(x) = 0$. In the definition given here, equality constraints are not included because they can easily be constructed using two inequality constraints (i.e., $g(x) \geq 0 \wedge -g(x) \geq 0 \Leftrightarrow g(x) = 0$).

Furthermore, we use the following definition of feasibility:

Definition 2.1.2 (Feasible Solution and Set of Feasible Solutions)**:** For an optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$, the *set of feasible solutions* $\mathcal{A}$ is the set

$$\mathcal{A} = \{x \in \mathcal{X} \mid g_i(x) \geq 0, i = 1, \ldots, p\}, \tag{2.1}$$

and each solution $x \in \mathcal{A}$ is called a *feasible solution.*

Given a problem that fits Definition 2.1.1, the goal of optimization can still vary and roughly be of one of the following forms:

1. Find *a* feasible solution that is optimal with respect to the objective function(s).
2. Find *all* feasible solutions that are optimal with respect to the objective function(s).
3. Find *a specific set of* feasible solutions that are optimal with respect to the objective function(s).

Here, “a specific set of solutions” loosely denotes the cases where, either implicitly or explicitly, also a secondary set selection criterion encompasses the optimization problem (e.g., searching for a diverse set of solutions requires a diversity notion based on sets of solutions). In addition to these three aims, a second class of optimization goals can also be identified in which the aim is to find solutions of which the objective function value(s) satisfies/satisfy (a) certain threshold value(s):

4. Find *a* feasible solution of which the objective function value(s) satisfies/satisfy (a) certain threshold value(s).
5. Find *all* feasible solutions of which the objective function value(s) satisfy (a) certain threshold value(s).
6. Find *a specific set of* feasible solutions of which the objective function value(s) satisfy (a) certain threshold value(s).

Note that the latter three aims could also be seen as constraint satisfaction problems (i.e., an objective function with a threshold is effectively a constraint function).

Given the definition of an optimization problem and the loosely defined possible goals of optimization, next we will give more formal definitions of optimality. However, in order to do this, we will make a distinction between *single objective optimization problems* and *multi-objective optimization problems* and provide separate definitions for both classes.

2.1.1 Single Objective Optimization Problems

A single objective optimization problem is a special instance of an optimization problem, defined as

Definition 2.1.3 (Single Objective Optimization Problem)**:** A *single objective optimization problem* is an optimization problem with precisely one objective function. For the triple $(\mathcal{X}, \mathcal{F}, \mathcal{G})$, the set $\mathcal{F}$ consists of exactly one objective function ($k = 1$ in Definition 2.1.1).

Given a single objective optimization problem, based on the definition of Törn and Žilinskas [TZ89], Bäck [Bäc96] defines the *global optimization problem* as the problem of determining a global minimizer:

Definition 2.1.4 (Global Minimum/Optimum and Global Minimizer/Optimizer)**:** For a single objective optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with $\mathcal{F} = \{f\}$ and a set of feasible solutions $\mathcal{A}$, the *global minimum* or *global optimum* f^* is the value

$$f^* = \min\{f(x) \,|\, x \in \mathcal{A}\}, \tag{2.2}$$

and every solution $x^* \in \mathcal{A}$ for which it holds that $f(x^*) = f^*$ is called a *global minimizer* or *global optimizer*.

This definition falls into the first item of the enumeration on page 13. Moreover, there are two other important remarks that have to be made. First, it is important to realize that there exist objective functions for which no global minimum exists. This can happen for objective functions of which the image $f[\mathcal{A}]$ of $f : \mathcal{A} \to \mathbb{R}$ is non-compact (e.g., the infimum of $f[\mathcal{A}]$ is not included in $f[\mathcal{A}]$). Secondly, one should note that when a global optimum does exist, there is only one global optimum, but there might be multiple global optimizers.

As it might happen that there are multiple solutions x for which holds that $f(x) = f^*$, an extended goal of global optimization is to find all global minimizers (see [TZ89]):

$$X^* = \{x \in \mathcal{A} \,|\, f(x) = f^*\}. \tag{2.3}$$

This extended goal falls into the second item of the enumeration above.

The alternative goals of finding solutions with objective function values satisfying a certain threshold value (items 4–6 in the enumeration on page 13) are for single objective optimization problems known as *super/sub level set* optimization problems:

Definition 2.1.5 (Sublevel Set)**:** For a single objective optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with $\mathcal{F} = \{f\}$ and set of feasible solutions $\mathcal{A}$, the *sublevel set* below level d is the set

$$L_d = \{x \in \mathcal{A} \,|\, f(x) \leq d\}. \tag{2.4}$$

2.1.2 Multi-Objective Optimization Problems

A multi-objective optimization problem is a special instance of an optimization problem, defined as

Definition 2.1.6 (Multi-Objective Optimization Problem)**:** A *multi-objective optimization problem* is an optimization problem with more than one objective function. I.e., for the triple

$(\mathcal{X}, \mathcal{F}, \mathcal{G})$, the set $\mathcal{F}$ consists of at least two objective functions.

For multi-objective optimization problems, the definition of optimality is often based on the notion of Pareto dominance on the objective space. Pareto dominance introduces a partial order on the space of objective function values, being $\mathbb{R}^k$ for a problem with k objectives. In the context of minimization, this order is defined as:

Definition 2.1.7 (Pareto Dominance, Weak Pareto Dominance, Strict Pareto Dominance, and Incomparability)**:** For any two vectors $\mathbf{u}$ and $\mathbf{v}$:
$\mathbf{u}$ *dominates* $\mathbf{v}$ (notation $\mathbf{u} \prec_{\text{Pareto}} \mathbf{v}$ or just $\mathbf{u} \prec \mathbf{v}$) iff:

$$\forall i \in \{1, \ldots, k\} : u_i \leq v_i \tag{2.5}$$

$$\text{and} \quad \exists j \in \{1, \ldots, k\} : u_j < v_j, \tag{2.6}$$

$\mathbf{u}$ *weakly dominates* $\mathbf{v}$ (notation $\mathbf{u} \preceq \mathbf{v}$) iff:

$$\mathbf{u} \preceq \mathbf{v} \vee \mathbf{u} = \mathbf{v}, \tag{2.7}$$

$\mathbf{u}$ *strictly dominates* $\mathbf{v}$ iff:

$$\forall i \in \{1, \ldots, k\} : u_i < v_i, \tag{2.8}$$

$\mathbf{u}$ and $\mathbf{v}$ are *incomparable* (notation $\mathbf{u} \,||\, \mathbf{v}$) iff:

$$\mathbf{u} \npreceq \mathbf{v} \wedge \mathbf{v} \npreceq \mathbf{u}. \tag{2.9}$$

The partial order introduced by using the notion of Pareto dominance on the solution space can be used to define the goal of multi-objective optimization as to find Pareto optimizers:

Definition 2.1.8 (Pareto Optimizer and Pareto Optimum)**:** For a multi-objective optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with a set of feasible solutions $\mathcal{A}$, the *set of Pareto optimal solutions* X^* is the set of all solutions $x^* \in A$ with function values $\mathbf{f}(x^*) = [f_1(x^*), \ldots, f_k(x^*)]$ for which it holds that there does not exist another solution $x \in \mathcal{A}$ with function values $\mathbf{f}(x) = [f_1(x), \ldots, f_k(x)]$ such that $\mathbf{f}(x)$ dominates $\mathbf{f}(x^*)$. I.e.,

$$X^* = \{x^* \in \mathcal{A} \,|\, \nexists x \in \mathcal{A} : \mathbf{f}(x) \prec \mathbf{f}(x^*)\}. \tag{2.10}$$

An element of the set of Pareto optimal solutions $x^* \in X^*$ is called a *Pareto optimizer* and its objective function value vector is called a *Pareto optimum*.

Although for some multi-objective optimization problems it is sufficient to find a Pareto optimal solution, in general, when a problem is defined as a multi-objective optimization problem, it is intended also to get insight in the trade-offs between the various objectives. Therefore the more usual (customary) aim for multi-objective optimization is to find the set of all Pareto optimal solutions or at least a representative subset of it.

Definition 2.1.9 (Pareto Front and Efficient Set)**:** For a multi-objective optimization problem

$(\mathcal{X}, \mathcal{F}, \mathcal{G})$, the set of all Pareto optima is called the *Pareto Front* and the set of all Pareto optimizers is called the *Efficient Set*.

With the definitions for a single- and multi-objective optimization problem, and the goal to find either one, multiple, or all optimizers, the basic problems of optimization have been introduced. How to solve optimization problems is another matter.

2.1.3 Discrete versus Real-Parameter Optimization Problems

Definition 2.1.1 only generally specifies the search space $\mathcal{X}$, the objective functions $\mathcal{F}$, and the constraint functions $\mathcal{G}$. Besides the separation of single- and multi-objective optimization problems, another distinction can be made by looking at the search space. Although it is not a comprehensive categorization, we distinguish two major classes of optimization problems: *discrete optimization problems* and *real-parameter optimization problems*.

Discrete optimization problems (which is a class that includes *combinatorial optimization problems*) are optimization problems where the search space is a discrete set of candidate solutions.

Definition 2.1.10 (Discrete Optimization Problem)**:** Any optimization problem of which the search space is a discrete set is called a *discrete optimization problem*.

This work focuses on the class of real-parameter optimization problems. Real-parameter optimization problems are optimization problems where the search space is a real-valued parameter space.

Definition 2.1.11 (Real-Parameter Optimization Problem)**:** A *real-parameter optimization problem* is an optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$, with $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{X}$ is of dimension n for some fixed $n \in \mathbb{N}_1$.

More specifically, for real-parameter optimization problems commonly a stricter class definition is taken, requiring the search space to be bounded by a box. We identify such types of problems as box-constrained real-parameter optimization problems.

Definition 2.1.12 (Box-Constrained Real-Parameter Optimization Problem)**:** A *box-constrained real-parameter optimization problem* is a real-parameter optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$, with

$$\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^n \mid (\mathbf{x}_\mathrm{l})_i \leq (\mathbf{x})_i \leq (\mathbf{x}_\mathrm{u})_i, i = 1, \ldots, n\}, \tag{2.11}$$

for some fixed $n \in \mathbb{N}_1$, and where $\mathbf{x}_\mathrm{l} \in \mathbb{R}^n$ is a vector of lower bounds and $\mathbf{x}_\mathrm{u} \in \mathbb{R}^n$ is a vector of upper bounds.

An example of a class of search spaces that does not fall into the categorization of discrete versus real-parameter is the class of mixed-integer optimization problems, consisting of a combination of real-parameter and discrete (integer and/or categorical) parameters.

2.2 The Practical Goal of Optimization

Given the definition of an optimization problem, the rough classification of optimization goals, the distinction between single- and multi-objective optimization, and the rough classification of discrete versus real-parameter optimization problems, the question is: How to solve such problems? The solvability of optimization problems much depends on the structure of the search space and the basic assumptions about the objective and constraint functions.

Discrete optimization problems with finite (enumerable) search spaces are solvable within a finite number of steps by means of *complete enumeration*. That is, by evaluating every solution in the search space it is possible to determine all optimizers. However, this has practical limits, for example when the search space is sufficiently large and/or evaluations are very time/cost expensive. Given that it is not possible to evaluate all candidate solutions we can follow the reasoning of Bäck [Bäc96] that when a strict subset of the search space is evaluated it is possible that the global optimum of a function f differs arbitrarily much from the optimum found so far. Hence, if we cannot afford to evaluate every solution in the search space, an optimization problem is generally unsolvable.

For real-parameter optimization problems an even more discouraging observation was made by Törn and Žilinskas [TZ89] who in the context of single objective optimization, according to Bäck [Bäc96], proved that

> *"The problem of determining a member of the level set $L_{f^*+\epsilon}$ of an arbitrary global optimization problem on a real-parameter objective function f on a compact feasible region M within a finite number of steps is unsolvable."* [Bäc96, p. 37]

Note that a similar message can be deduced for multi-objective optimization.

Given this observation, we can state that global optimization problems as generally formulated by Definition 2.1.1 on page 12 are practically unsolvable unless we a) restrict ourselves to a class of optimization problems for which the objective functions satisfy certain additional conditions, or b) relax the solvability requirement [TZ89]. Both are done in practice. Regarding the class of problems that is considered, it is commonly assumed that the optimization problem exhibits some kind of underlying structure. An implicit assumption that is often made is that similar solutions are believed to have similar performance. While agreeing that the notion of similarity is not well-defined, neither its intuitiveness nor its validity can be denied. The goal of optimization is relaxed by taking a more practical viewpoint. Based on the definition provided by Törn and Žilinskas [TZ89] we define the general goal of optimization as:

Definition 2.2.1 (Practical Goal of Optimization)**:** Given an optimization problem with an optimization goal and a limited number of resources (i.e., a number of trials or an evaluation budget), the *practical goal of optimization* is to use these resources in an optimal way to find (an) as good as possible solution(s).

Or, from a slightly different perspective, one could aim for finding solutions that are an improvement with respect to the currently best known solution (*melioration*).

In addition to the practical goal of optimization, it is generally noted that an optimization algorithm is a *global optimization algorithm* if, given an infinite evaluation budget, it will get arbitrarily close to the global optimum.

Definition 2.2.2 (Global Optimization Condition)**:** Let x_t denote the best solution found by the optimization algorithm at time t with function value f_t. We say that the optimization algorithm satisfies the *global optimization condition* iff for $t \to \infty$, $f_t - f^* < \epsilon$, for arbitrarily small positive values of ϵ.

Note that this goal is constructed in terms of finding one global optimizer (provided that it exists) and it should be extended when the goal is to find all (or a particular set of) global optimizers or level set solutions.

2.3 Objective Function Landscapes

A central dogma of geography formulated by Tobler is: "everything is related to everything else, but near things are more related than distant things" [Tob70]. This dogma represents an implicit assumption that is generally also used for optimization problems in practice, namely that there is a correlation between the (dis)similarity of two candidate solutions and the (dis)similarity of their objective function values. The conceptual step from a (dis)similarity measure to a distance measure $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is commonly a small one, leading to the assumption of the search space being a metric space $(\mathcal{X}, d)$ (e.g., Euclidean distance for real-parameter search spaces).

The assumption that the search space is a metric space introduces a view on objective functions as landscapes. The search space is the location space of this landscape and the objective function values denote the height or elevation at each location of the landscape. A landscape, consisting of peaks, valleys, ridges, etc., provides a way of visualizing optimization problems, but also allows us to talk about locality related properties of a given objective function. Especially for real-parameter search spaces, this point of view is very intuitive and often used implicitly, with Euclidean distance as distance measure. However, also in other types of search spaces it is possible to view a problem as an objective function landscape.

The simplest class of algorithmic methods that actively exploits correlation between solution similarity and objective function values is the class of the so-called *hill-climbing algorithms*. A hill-climbing algorithm is an iterative algorithm that starts with an arbitrary point in the search space and attempts to find a better solution by evaluating slight perturbations of that solution. If a perturbation produces a better solution, the algorithm proceeds with that new solution, repeating until no further improvements can be found. Hence, when viewing it from a maximization perspective, it takes steps uphill into the direction of the global optimum. A well-known example of a hill-climbing algorithm (for minimization) is the *Steepest Descent* algorithm (see, e.g., [NW06]) that follows the path of the gradient.

For optimization algorithms that use operators based on local perturbations (such as hill-climbing algorithms, but also Evolutionary Algorithms), the objective function landscape metaphor can be used to visualize different geographical scenarios that have a different impact on the performance of these algorithms. For instance, when an objective function landscape consists of multiple peaks of different heights, a hill-climber can get stuck in one of the lower height peaks, i.e., a locally optimal solution. Interestingly, one can herewith observe that a hill-climbing algorithm does not satisfy the global optimization condition of Definition 2.2.2 and is therefore qualified as a *local optimization algorithm*. This supports the viewpoint of objective function landscapes as a practical way to analyze optimization algorithms.

2.4 Single Objective Real-Parameter Landscapes

For single objective real-parameter optimization problems, the different geological concepts that can influence the behavior of perturbation-based optimization algorithms can formally be defined. Below, the most important concepts will be formalized exactly based on Euclidean distance as dissimilarity measure:

Definition 2.4.1 (Weak Local Minimizer/Optimizer)**:** For a single objective optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with $\mathcal{F} = \{f\}$ and the set of feasible solutions $\mathcal{A}$, a *weak local minimizer* or *weak local optimizer* is a solution $\mathbf{x}^* \in \mathcal{A}$ for which it holds that

$$\exists \delta \in \mathbb{R}_{>0} \left(\forall \mathbf{x} \in \mathcal{A} \left(||\mathbf{x} - \mathbf{x}^*|| < \delta \Rightarrow f(\mathbf{x}^*) \leq f(\mathbf{x}) \right) \right). \tag{2.12}$$

Definition 2.4.2 (Strict Local Minimizer/Optimizer)**:** For a single objective optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with $\mathcal{F} = \{f\}$ and the set of feasible solutions $\mathcal{A}$, a *strict local minimizer* or *strict local optimizer* is a solution $\mathbf{x}^* \in \mathcal{A}$ for which it holds that

$$\exists \delta \in \mathbb{R}_{>0} \left(\forall \mathbf{x} \in \mathcal{A} \left(||\mathbf{x} - \mathbf{x}^*|| < \delta \wedge \mathbf{x} \neq \mathbf{x}^* \Rightarrow f(\mathbf{x}^*) < f(\mathbf{x}) \right) \right). \tag{2.13}$$

Definition 2.4.3 (Weak/Strict Local Minimum/Optimum)**:** For a single objective optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with $\mathcal{F} = \{f\}$ and the set of feasible solutions $\mathcal{A}$, a *(weak/strict) local minimum* or *(weak/strict) local optimum* f^* is a value for which there exists a weak/strict local minimizer/optimizer $\mathbf{x}^*$ such that $f^* = f(\mathbf{x}^*)$.

The distinction between weak and strict local minima/optima is subtle, but important. In this work, we will consider a weak local minimum/optimum as the "default" type when referring to a local minimum/optimum. The existence and quantity of local optima is an indicator for the difficulty of a single objective optimization problem (or rather the likelihood of local hill-climbing algorithms to get stuck at local optima). Following Schwefel [Sch77]:

Definition 2.4.4 (Unimodal, Multimodal and Multiglobal)**:** An objective function is said to be *unimodal* if it has only one optimizer (i.e., only a global optimizer). Otherwise, it is said to be *multimodal*. A landscape is called *multiglobal* if there are several global optimizers.

Besides the existence of local optima and global optima, another phenomenon in objective function landscapes is the possible existence of plateaus. A plateau is defined as:

Definition 2.4.5 (Flat Region and Plateau)**:** For a single objective, real-parameter optimization problem $(\mathcal{X}, \mathcal{F}, \mathcal{G})$ with $\mathcal{F} = \{f\}$ and the set of feasible solutions $\mathcal{A}$, a *flat region* of $f|_{\mathcal{A}}$ is a connected set of points $P \subseteq \mathcal{A}$ such that

$$\forall \mathbf{x} \in P \left(f(\mathbf{x}) = c \right) \text{ and } \forall \mathbf{x} \in P \left(\exists \epsilon_{\mathbf{x}} > 0 \left(B_{\epsilon_{\mathbf{x}}}(\mathbf{x}) \cap \mathcal{A} \subseteq P \right) \right), \tag{2.14}$$

for some $c \in \mathbb{R}$, with $B_{\epsilon}(\mathbf{x}) = \{\mathbf{x}' \in \mathbb{R}^n \mid ||\mathbf{x} - \mathbf{x}'|| < \epsilon\}$, and with $\epsilon_{\mathbf{x}}$ defined separately for each $\mathbf{x} \in P$. A *plateau* P is a flat region with the additional property that there exists no flat region $P' \supset P$.

Note that when considering the possible existence of plateaus, there are solutions within a plateau that are both weak local minimizers and weak local maximizers, namely the solutions $\mathbf{x}^* \in \mathcal{A}$ for which it holds that

$$\exists \delta \in \mathbb{R}_{>0} \left(\forall \mathbf{x} \in \mathcal{A} \left(||\mathbf{x} - \mathbf{x}^*|| < \delta \Rightarrow f(\mathbf{x}^*) = f(\mathbf{x}) \right) \right). \tag{2.15}$$

This is a somewhat paradoxical property that emerges from using Definition 2.4.1. However, the alternative of restricting to Definition 2.4.2 and changing the $\leq$-sign by the $<$-sign leads to the problem that in a similar case, the global optimum is not a local optimum.

2.5 Black-Box Optimization Algorithms

An optimization algorithm is an algorithmic method that can be applied to solve (a specific class of) optimization problems. There is a wealth of optimization methods available and choosing one for solving a given optimization problem depends much on the characteristics of the optimization problem at hand. There are many aspects that vary from problem to problem. Many optimization methods are especially designed for specific types of search spaces, objective and constraint functions, or tailored for special objective function classes.

This work focuses on optimization methods that are not dependent on any knowledge about the system or model of the optimization problem. That is, the model or system that lies at the core of the optimization problem is considered to be a black-box. For such problems, optimization algorithms are challenged to find good solutions by sequential trial-and-error of candidate solutions. In the strictest sense, the term black-box optimization implies that there is no knowledge about the model or system whatsoever, however, often basic implicit assumptions or properties such as continuity, causality, or even the assumption that the problem belongs to a certain class of problems are used.

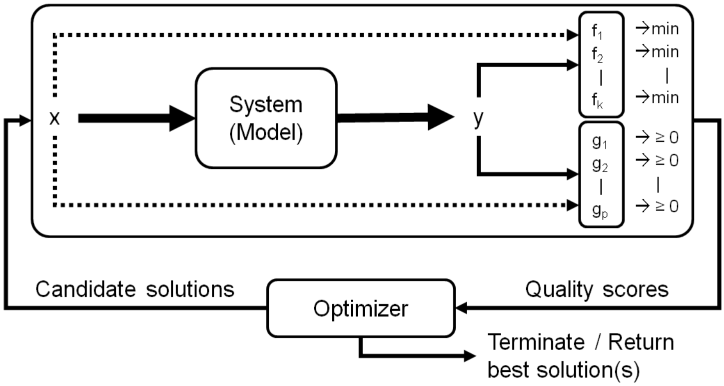

Figure 2.3: The general setup for black-box optimization.

Figure 2.3 visualizes the general black-box optimization loop. The principle is straightforward: An optimizer is coupled to the (model of the) system of interest (according to the model of Section 2.1). The optimizer generates one or more candidate solutions, feeds it/them to the system, and receives a quality score of these candidate solutions. Using this information, the optimizer generates a new set of candidate solutions, and again feeds those to the system to obtain their quality. This loop is repeated until either a satisfactory solution is found, a predefined evaluation budget has been reached, or any other termination criterion has been reached. Note that the term *optimizer* is used in two ways: for optimal solutions and for optimization algorithms. Evolutionary Algorithms form a sub-class of black-box optimization algorithms.

2.5.1 Quality Measures for Optimization Algorithms

A difficult issue in the field of black-box optimization is the assessment of the quality of optimization algorithms. The quality of an optimization algorithm depends much on the characteristics of the problem at hand and the class of optimization problems for which the algorithm is designed. Benchmark sets of test problems are often used for empirical comparison of multiple optimization algorithms, see, e.g., [SHL$^+$05, HFRA09b, HFRA09a, HFRA10]. For these benchmark sets, there are two types of indicators that can be used for determine the quality of the optimization algorithm: The quality of the (set of) solution(s)

1. versus the number of objective function evaluations needed to obtain that quality, or
2. versus the total computation time needed to obtain that quality.

When assuming that the evaluation time of the candidate solutions exceeds the computational overhead introduced by the operations of the optimization algorithm then the former is the most appropriate measure. Moreover, when the optimization algorithms contain stochastic elements, multiple runs should be used for obtaining averaged quality scores.

2.6 Summary and Discussion

In this chapter the background of this work is summarized, being the traditional view on black-box optimization. Given this view on optimization, the two main distinctions between single- and multi-objective optimization problems and between discrete- and real-parameter optimization problems are presented. For such problems, it is shown that there is a distinction between the theoretical goal of optimization and how this goal is used in practice.

Furthermore, the concept of objective function landscapes is summarized, which is based on the assumption that the search space is a metric space and that there is a correlation between (dis)similarity of two solutions and their objective function values. In particular, definitions of commonly used terms are given in the context of single objective real-parameter optimization problems. These types of problems are the main focus of this work.

Black-box optimization algorithms are algorithmic methods that aim to solve optimization problems. Such algorithms sequentially test candidate solutions in a trial-and-error fashion in order to find optimal solutions. Spatial correlation is commonly exploited by optimization algorithms, such as hill-climbing algorithms. Although the quality of an optimization algorithm much depends on the problem at hand, benchmark sets of test problems can be used for empirical comparison of optimization algorithms for certain problem classes.

An issue that is missing in the general model of an optimization problem as described in Section 2.1 is the possible existence of uncertainty and noise. However, these are frequently occurring phenomena when dealing with real-world optimization problems and they can have a large influence on optimization in practice. In Chapter 3, the general model of an optimization model as presented in this chapter will be extended to include uncertainties and/or noise in order to form a definition of a black-box robust optimization problem.