Evolution strategies for robust optimization

Kruisselbrink, J.W.

Chapter 3

Robust Optimization

The traditional view on optimization as presented in Chapter 2 does not account for uncertainties and noise. However, this is not realistic for many real-world optimization problems. Consider, for instance, industrial engineering applications. A common optimization scenario is that a non-deterministic simulator replaces the real-world system and the aim is to find solutions such that the real-world realizations of these solutions are of a good quality, also when these are slightly perturbed due to manufacturing errors. In this scenario, uncertainties and noise arise in various ways, e.g., in the form of uncertainty because an approximate model is used instead of the real-world system, in the form of noise because the simulator is non-deterministic, and in the form of uncertainty introduced by the inability to generate exact realizations of the solutions. These observations give rise to two new questions: In what way can uncertainties and noise arise in the general model of an optimization problem as presented in Section 2.1? How do we account for this when optimizing on such systems?

The structure of this chapter is as follows: Section 3.1 starts with an overview/taxonomy of the various ways in which uncertainties and noise can emerge in optimization problems. In Section 3.2, the scope and goals of robust optimization are derived from this taxonomy. Section 3.3 presents three real-world optimization scenarios and discusses them in the context of robust optimization. Section 3.4 closes with a summary and discussion.

3.1 Uncertainties and Noise in Optimization Problems

Often, due to a variety of reasons, the theoretical model used for optimization differs from the real-world system for which optimal solutions are desired. Examples are:

1. The design variables cannot be controlled with unlimited precision. (**input**)

2. The operational (or environmental) conditions fluctuate or are known only to a certain extent. (**model**)

3. The output of the real-world system or of the (simulation) model is noisy. (**output**)

4. Approximation models replace the real-world system within the optimization loop. (**model/output**)

5. There is a degree of vagueness in the objectives and/or the constraint boundaries. (**objectives/constraints**)

These uncertainties can have a negative impact on the practical applicability of using idealized models or assumptions for solving real-world optimization problems. Not accounting for uncertainty and noise might lead to solutions that are found to be optimal with respect to the idealized model, but which are not useful or performing optimally in practice.

For optimization problems, the terms uncertainty and noise refer to behavior in any part of the optimization model that can not (fully) be predicted or controlled, or that is subject to vagueness. As illustrated, the causes of uncertainties and noise in real-world optimization problems can be manifold. In order to find good methods to deal with uncertainties and noise, a first step is to distinguish the different classes of uncertainty and noise that can arise in optimization problems. Among the different ways in which such a classification can be made (see, e.g., [BS07, JB05, ONL06]), the classification provided in this section will, to a great extent, follow the categorization of Beyer and Sendhoff [BS07].

3.1.1 Sources of Uncertainty and Noise

One way to distinguish different types of uncertainty and noise within optimization problems is to categorize them by looking at the parts of the optimization model in which they arise. When considering the model illustrated in Figure 2.2, one can identify five different locations where uncertainty and/or noise can enter the optimization model (see Figure 3.1):

A) Uncertainties and/or noise in the design variables

B) Uncertainties and/or noise in the environmental parameters

C) Uncertainties and/or noise in the output

D) Vagueness in the constraints

E) Preference uncertainty in the objectives

These sources of uncertainty/noise are integrated in different ways in the mathematical formulations of an optimization problem.

A) **Uncertainties and/or noise in the design variables**
With this, deficiencies or fluctuations in the real-world realizations of candidate solutions are addressed. These deficiencies/fluctuations can arise when in the real-world system, the design variables can only be realized or controlled with a limited precision. Manufacturing imprecisions in product engineering are exemplary for this type of uncertainty,

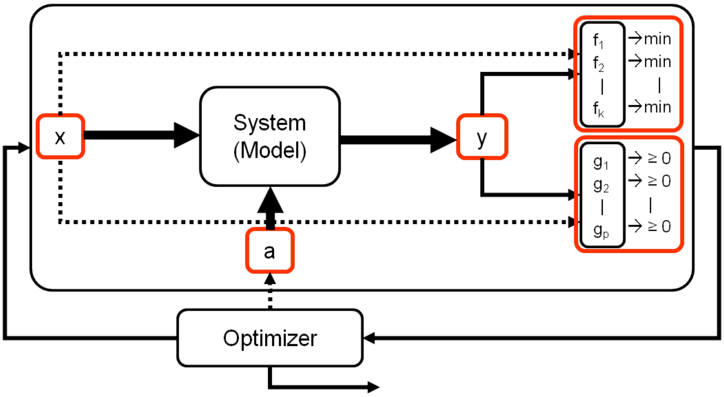

Figure 3.1: The general optimization model showing the different parts of the systems in which uncertainties can arise.

which is the main type of uncertainty considered by the robust design concept. This type of noise gives rise to two different scenarios:

Scenario 1: A simulation model replaces the real-world system within the optimization model. In this case the simulator accepts inputs as non-disturbed inputs, though in practice, solutions cannot be realized with unlimited precision. This can be compensated for by focusing the optimization on finding solutions that are also of high quality when slightly perturbed (i.e., finding robust optima).

Scenario 2: The real-world system is enclosed in the optimization loop (*experimental optimization*). In this case the disturbances in the input propagate through the output, hence to the objective and constraint functions. When the disturbances in the design variables can be measured a posteriori, the optimizer will receive deterministic measurements, but the sampling process cannot be controlled. In case that the disturbances cannot be measured a posteriori, these disturbances simply yield noise in the objective and constraint functions. The noise distributions can be of any kind depending on the distribution of the noise in the design variables, transformed through the output and the objective and constraint functions.

The effect of variations caused by uncertainties or noise in the design variables can be modeled within the formulation of an objective functions as:

$$\tilde{f}_i(x) = f_i(x + \delta_x), \quad i = 1, \dots, k. \tag{3.1}$$

And similarly for the constraint functions:

$$\tilde{g}_j(x) = g_j(x + \delta_x), \quad j = 1, \dots, p. \tag{3.2}$$

Here, $\delta_x \in \mathcal{X}$ is an uncontrollable random or uncertain variable representing the deviations/fluctuations in the input that are due to uncertainty and/or noise. An important observation is that this noise or uncertainty can depend on the values of the design variables themselves, i.e., it can vary for different x. Furthermore, due to this remodeling, the outputs of the objective and constraint functions become random variables (in case the uncertainty δ_x can be modeled as a random variable) or sets of possible outputs.

B) **Uncertainties and/or noise in the environmental parameters**
Uncontrollable (environmental) parameters of a system are not considered in the classical optimization model as presented in Section 2.1 (see Figure 2.2). These parameters are in the classical view considered to be constants of the system. However, in practice many of such constants are noisy or uncertain. Fluctuating or unknown operating conditions and deficiencies of internal parameters are possible ways in which uncertainties can manifest themselves within this part of the optimization model. Similar to uncertainties in the design variables, the setting in which this type of uncertainty can be actively compensated for is when a simulation model replaces the real-world system within the optimization model. Otherwise, the uncertainty propagates to the objective and constraint functions.

To incorporate such scenarios into the model presented in Section 2.1, we extend it as follows: Let the set $\mathcal{C}$ denote the (possible) settings of uncontrollable environmental parameters of the system. The models of the objective and constraint functions are extended by also being a function of the environmental parameters, i.e., $f : \mathcal{X} \times \mathcal{C} \to \mathbb{R}$ for all $f \in \mathcal{F}$ and $g : \mathcal{X} \times \mathcal{C} \to \mathbb{R}$ for all $g \in \mathcal{G}$.

The effect of variations caused by uncertainties or noise in the environmental parameters can be modeled in the objective functions as:

$$\tilde{f}_i(x, a) = f_i(x, \alpha), \quad i = 1, \ldots, k. \tag{3.3}$$

And similarly for the constraint functions:

$$\tilde{g}_j(x, a) = g_j(x, \alpha), \quad j = 1, \ldots, p. \tag{3.4}$$

Here $\alpha \in \mathcal{C}$ is a noisy or uncertain counterpart of the constant $a \in \mathcal{C}$. As can be seen, by modeling the uncertainty/noise in this way, the constant environmental parameters a are replaced by random or uncertain variables α.

C) **Uncertainties and/or noise in the output**
The third class is formed by output uncertainties or noise. Here we can distinguish two different types: 1) The system is non-deterministic and produces noisy outputs (e.g., the measurements have a stochastic nature and precise evaluation is impossible). 2) The system produces uncertain output (e.g., models where the quality of the output of the model cannot be guaranteed to be conform the actual output of the system).

Note that both types could be present simultaneously, e.g., when using non-deterministic simulation models.

Noise or uncertainty in the output can be modeled within formulations of the objective functions as:

$$\tilde{f}_i(x) = f_i(x) + z_{f_i}(x), \quad i = 1, \ldots, k, \tag{3.5}$$

and in the constraint function definitions as:

$$\tilde{g}_j(x) = g_j(x) + z_{g_j}(x), \quad j = 1, \ldots, p, \tag{3.6}$$

with $z_{f_i}(x)$ and $z_{g_j}(x)$ being random/uncertain variables (possibly indexed by space), denoting the propagation of the output uncertainty/noise to the objective and constraint functions.

Note that, strictly seen, the noise in the objective and constraint functions is a product of the noise in the output. Internally, $z_{f_i}(x)$ is a (possibly non-linear) function of the noise in the output, i.e., $y(x) + z_y(x)$. However, as we have chosen to model the objective functions and constraint functions in terms of x, this cannot be modeled explicitly.

Furthermore, note that uncertainties in the design variables and environmental parameters in principle propagate as a noisy or uncertain output, i.e., fluctuating/uncertain design variables and environmental parameters yield fluctuating/uncertain system output.

D) **Vagueness in the constraints**

Often it is hard to obtain strict and bounded definitions of constraints. When dealing with constraints like “the temperature should not be too high” or “the risk should not be too high”, it is not possible (or desirable) to draw a straight line between satisfied and not-satisfied. Such vagueness calls for methods that can account for this (e.g., by using fuzzy logic techniques [Zad65, BZ70]).

Fuzzy constraints can be described using the following notation:

$$g_j(x) \gtrsim 0. \tag{3.7}$$

Here, $\gtrsim$ is the fuzzified version of $\geq$ having the linguistic interpretation “$g_j(x)$ is essentially greater than 0”. However, this notation does not take into account the degree to which the constraint is fuzzy. For example, considering the “temperature should not be too high” example, the notation of Eq. 3.7 does not specify the margins for which the temperature could still be acceptable.

A typical way of modeling uncertainties of this kind is to transform the constraints by means of membership functions. A membership function is a monotonous (but not necessarily linear) function V_g which maps a constraint function $g(\mathbf{x})$ to an area bounded by 0 (strictly violated) and 1 (completely satisfied). Everything between 0 and 1 is in the

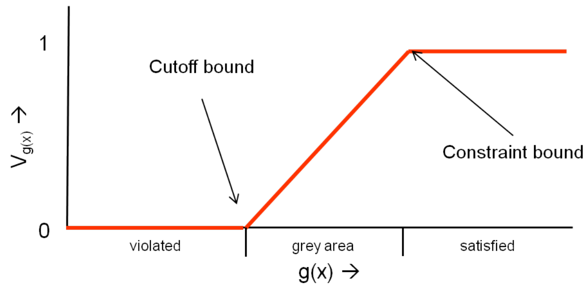

Figure 3.2: A simple linear membership function.

grey area (or transition area), and the higher the value of the membership function, the higher the degree of satisfaction. Figure 3.2 illustrates the basic idea of a membership function. Mathematically, this transformation has the following form:

$$g(x) \geq 0 \quad \text{becomes} \quad V_g(g(\mathbf{x})) \to \max . \tag{3.8}$$

A design issue that is introduced is that appropriate membership functions have to be constructed.

E) **Preference uncertainty in the objectives**
Preference uncertainty emerges when having multiple conflicting objectives, hence, when highly subjective trade-off choices have to be made regarding the importance of objectives. Preference might only be known afterwards, depending on the quality level that can be achieved for the objectives and the particularities of the trade-offs that exist. Approaches that use aggregate objective functions which combine multiple objectives functions into one objective function are prone to such uncertainties. Approaches that aim to approximate the complete Pareto frontier postpone such preference decisions.

Of these five sources of uncertainty, a distinction can be made between the first three (A, B, and C) and the last two (D and E). The former involve *uncertainty and/or noise in the system or (simulation) model*, whereas the latter involve *uncertainty in the optimization model* (i.e., the way in which a given problem is translated to an optimization problem).

3.1.2 Modeling Uncertainty and Noise

Besides the fact that uncertainties can arise in different parts of the optimization model, another way to distinguish different kinds of uncertainty and noise is to consider their nature. Up to now, a distinction has been made between uncertainty and noise, as being two distinct matters. The question is, however, whether these are indeed separate issues.

The distinction between uncertainty and noise is the same as the distinction between *aleatory uncertainty* and *epistemic uncertainty* which is a particularly popular view in the engineering field (e.g., [PC96, KD09]). Also, it can be related to the distinction between the two schools of statistical theory: frequentist and Bayesian statistics [O'H04, Cox05]. The term aleatory uncertainty is used to describe uncertainties within a system or model that have an intrinsic stochastic nature. These are the uncertainties associated with the pure (often said to be irreducible) randomness within a system. Epistemic uncertainty, on the other hand, is the uncertainty that is due to incomplete or inadequate information (i.e., due to a lack of knowledge). Epistemic uncertainty should be reducible when more knowledge becomes available. Regarding the two schools of statistics, the frequentists can be said to accept uncertainty as aleatory, whereas in the Bayesian statistics the focus is on the *degree of belief*, which can be related to epistemic uncertainty [O'H04].

Although the distinction between aleatory and epistemic uncertainty intuitively makes sense, it is often a source of confusion (see, e.g., [KD09] for a discussion). It is often hard to specify whether a specific uncertainty is purely due to inherent randomness or due to limited knowledge or modeling capabilities. A purely deterministic mind would attribute every uncertainty to limited knowledge, and indeed in practice many cases of uncertainty are due to abstractions of details of the system. Similarly, the distinction between frequentist and Bayesian statistics seems to touch upon the same subject. Here, the debate evolves around the difference between *uncertainty of knowledge* versus *variability in outcome* [Cox05].

To avoid resorting to a lengthy philosophical discussion on this topic, we will make a distinction based on the difference in the mathematical modeling of the uncertainties. This is in line with the view presented by Beyer and Sendhoff [BS07]. In fact, one could say that the decision on the type of uncertainty is left to the person providing the optimization model. Hence, from the perspective of optimization, we are not so much interested in the actual type of uncertainty, but rather in the way in which it is modeled within the optimization problem formulation. Looking at the different ways in which uncertainties and noise can mathematically be modeled within an optimization model, one can distinguish three classes:

1) **Fuzzy**
The uncertainty is formulated by fuzzy statements about the possibility or degree of membership by which a state of an uncertain variable is believed to be plausible (or desirable). Uncertainties of this type can be modeled with *fuzzy sets* [Zad65, BZ70]. Using fuzzy sets for this type of uncertainty requires modeling a particular uncertain variable from a given space of points A as a pair (A, m_A). Here, $m_A : A \to [0, 1]$ is a *membership function* that maps each $x \in A$ to a "grade of membership" in (A, m_A). The grade of membership is a value between 0 and 1. For a particular $x \in A$, the closer $m_A(x)$ will be to one, the higher its degree of membership of A or degree of plausibility.

2) **Deterministic**
The uncertainty is formulated by statements about the crisp possibility of whether a state of an uncertain variable is possible or not. Uncertainties of this type can be modeled

using *crisp sets*. A particular uncertain variable of this type is modeled as a pair (A, m_A), with A being the crisp set and m_A denoting the membership function. The membership function is of the form $m_A : A \rightarrow \{0, 1\}$. Hence, a particular $x \in A$ can be noted to be either a member of the set A, when $m_A(x) = 1$, or not to be a member of A, when $m_A(x) = 0$.

3) **Probabilistic**
The uncertain variable is believed to be a stochastic random variable. A probability measure can be established measuring the probabilistic frequency of events that may occur. Uncertainties of the probabilistic type can be modeled using probability functions (or probability density functions in case of continuous domains). In this case, a function $p : A \rightarrow \mathbb{R}_{\geq 0}$ maps every event x in the space of all possible events A to a probability (density) value denoting the probability of that particular event. Note that the function p should conform to the classical Kolmogorov axiom system [Kol33].

From the perspective of mathematical modeling, this division can also be seen as a hierarchical structure of increased knowledge of the uncertainty. In the first type, there is uncertainty about the domain of the variation and the probabilities of the uncertainty events. In the deterministic type of uncertainty, the domain of the variation is known, but there is no knowledge about the probabilities of the events. In the probabilistic type of uncertainty, both the domain and the probabilities of the individual variation events are known. Although this distinction might be subtle (and the mathematical formulations might seem very similar), this difference is of great importance, as it requires different methods for treating these types of uncertainty.

Returning to the distinction between aleatory and epistemic uncertainty (and also the distinction between uncertainty and noise), we can see that aleatory uncertainty is the uncertainty of the probabilistic type and epistemic uncertainty is the uncertainty that is either of the deterministic or possibilistic type. A schematic view is given in Table 3.1. Moreover, when adopting this view, we can say that (agreeing with [KD09]) the aleatory/epistemic distinction is made by the person modeling the optimization problem and it is fair to make a distinction between these two types of uncertainty within the scope of the optimization model.

3.1.3 Stationary versus Non-Stationary Noise

One issue that has not yet been covered in the discussion about the modeling of noise is the distinction between *stationary noise* and *non-stationary noise*. Traditionally, these terms are used within the context of time-based systems in which the output is noisy. The term non-stationary noise indicates that the noise distribution changes over the course of time, whereas the term stationary noise indicates that the noise distribution is independent of time.

For optimization problems, this notion does not restrict to time, but can also be used in the context of space, i.e., non-stationary uncertainty/noise differs among the candidate solutions.

Conceptual classification	**Mathematical modeling**	**Mathematical properties**
Epistemic (uncertainty)	Possibilistic	Uncertainty domain unknown Probabilities unknown
	Deterministic	Uncertainty domain known Probabilities unknown
Aleatory (noise)	Probabilistic	Uncertainty domain known Probabilities known

Table 3.1: A classification of the types of uncertainty in terms of their mathematical properties based on the conceptual distinction between epistemic and aleatory uncertainty.

Looking back at the descriptions of the sources of uncertainty one notes that for uncertainties of type A and C, the noise/uncertainty might vary for different values of $x \in \mathcal{X}$. If the noise/uncertainty varies for different values of x, we call the noise/uncertainty *non-stationary*, if the noise/uncertainty is the same for all $x \in \mathcal{X}$ it is called *stationary*.

3.1.4 Cases of Uncertainty and Noise

Five sources combined with three types of uncertainties/noise yields theoretically 15 different concepts of uncertainty/noise that can be encountered within optimization problems. When including the distinction between stationary and non-stationary uncertainty/noise and the consideration that multiple types of uncertainties/noise can be present simultaneously within a real-world optimization problem, this picture becomes even more discouraging. Indeed, the variety of cases of uncertainty/noise that can arise from the combinations of the different cases complies with the complexity of real-world scenarios. In practice, however, some scenarios occur more often than others, and those are worthwhile studying in relative isolation (i.e., one type of noise within one part of the optimization model). Table 3.2 summarizes the combinations of class/type of uncertainty/noise as they can occur within optimization problems.

Uncertainty in the design variables (class A) is often uncertainty of type 3, i.e., probabilistic uncertainty. As this source of uncertainty is based on scenarios in which in the real-world the design variables cannot be set infinitely accurate, its relation to aleatory uncertainty with respect to the optimization model is undeniable. Hence, these are cases in which it is well doable and reasonable to describe the uncertainty using probability distributions. In cases where the uncertainty in the design variables is modeled as type 1 or type 2 uncertainty, these descriptions should be strict in order for optimization to make sense at all (i.e., having a broad uncertainty range for each candidate solution basically makes the solutions incomparable, hence not worth optimizing). Both stationary and non-stationary noise are possible for this uncertainty class. Hence, there are scenarios in which the fluctuations of the design variables depend on the settings themselves, but it may also be that the fluctuations have the same

Class	Type	Stationarity / non-stationary
A) Uncertainties and/or noise in the design variables	1) Possibilistic 2) Deterministic **3) Probabilistic**	Stationary or non-stationary
B) Uncertainties and/or noise in the environmental parameters	**1) Possibilistic** **2) Deterministic** **3) Probabilistic**	Stationary
C) Uncertainties and/or noise in the output	**1) Possibilistic** **2) Deterministic** **3) Probabilistic**	Stationary or non-stationary
D) Vagueness in the constraints	1) Possibilistic 2) Deterministic ~~3) Probabilistic~~	Stationary
E) Preference uncertainty in the objectives	1) Possibilistic 2) Deterministic ~~3) Probabilistic~~	Stationary

Table 3.2: Classification and categorization of different manifestations of uncertainty/noise as they can occur within optimization problems. Note that the bold types of uncertainty are considered to be “more common” and the crossed-out types are consdered to be nonexistent.

characteristics for all candidate solutions.

In class B, all three types can be encountered. Uncertainties of probabilistic nature occur when system parameters are due to fluctuations. Possibilistic and deterministic uncertainty occur when model parameters that represent real-world parameters are unknown. Class B uncertainties are always stationary with respect to the given design variables. That is, being fluctuations in the operation conditions of the system, they cannot depend on the particular settings of the design variables.

In class C, also all types of nature of uncertainty can be encountered. When dealing with noisy output, this uncertainty is of type 3. When the uncertainty arises due to the use of uncertain prediction models, the uncertainty is of type 1 or type 2. Moreover, when using stochastic simulation models, the uncertainty can be a composite of type 3 and type 1/2. Also this type of uncertainty can be both stationary and non-stationary with respect to the design variables.

In class D and class E, the nature of the uncertainties is always described as type 1 or 2 uncertainty. These are classes of uncertainty that are somewhat different than the other classes. These classes do not regard uncertainty within the model or the system, but rather the uncertainty of specifying what is desirable or acceptable. These types of uncertainty are only stationary with respect to the design variables.

3.2 Robust Optimization

Up to now, we have discussed the various ways in which uncertainty and noise can present themselves within optimization problems. Given this taxonomy, the next question is: what is robust optimization? In Section 1.1 it is noted that there exist different views on this matter. Some consider robust optimization to involve input uncertainties and/or noise in the design variables (e.g., [BBC11, BNT10]), whereas others consider a broader view, considering robustness with respect to uncertainties within the system or (simulation) model (e.g., [BS07]).

In this work, we consider robust optimization to be the practice of optimization given uncertainties and/or noise in the system or (simulation) model. Note that we exclude modeling uncertainty. For any kind of uncertainty and/or noise of class A, B, or C, robust optimization deals with the questions:

- In what way do uncertainties and/or noise within the system or (simulation) model affect optimization algorithms and the practical applicability of solutions found by optimization algorithms?
- How should optimization algorithms be adapted in order to account for uncertainties and/or noise within the system or (simulation) model?

With this we can extend the practical goal of optimization of Definition 2.2.1 and formulate the general goal of robust optimization as:

Definition 3.2.1 (Practical Goal of Robust Optimization)**:** Given an optimization problem with uncertainty and/or noise within the system or (simulation) model, and given an optimization goal and a limited number of resources. The *practical goal of robust optimization* is to use these resources to find (an) as good as possible solution(s) despite uncertainty and/or noise, that are also optimal and useful in the face of the uncertainties/noise.

Note here that optimality with respect to the uncertainties and/or noise should be defined within the scope of the uncertainty and/or noise at hand. Besides that, from this formal definition two intrinsically different targets of robust optimization can be distinguished, related to uncertainty and noise within optimization problems:

1. Target to find optimal solutions in noisy/uncertain environments.
2. Target to find robust solutions.

The first aim is to deal with the fact that the system on which the optimization is performed is not guaranteed to be noise/uncertainty-free and the problem is to design an optimization method that is able to deal with this in order to find solutions which are of good quality (also in practice). The other aim represents the desire to find solutions that are also of good quality when variations in the design variables or environmental parameters occur. This aim

Robust Optimization Target	**Uncertainty class**
Finding robust solutions	A) Uncertainties and/or noise in the design variables B) Uncertainties and/or noise in the environmental parameters
Optimization in noisy and uncertain environments	C) Uncertainties and/or noise in the output

Table 3.3: The two targets of optimization can roughly be related to the place in which the uncertainty/noise emerges.

emerges when instead of the real-world system, models are used for optimization and candidate solutions or internal model parameters cannot be guaranteed to be set infinitely precise in the real-world system. Hence, it focuses on the robustness of the solutions themselves. Table 3.3 summarizes this global categorization of robust optimization problems.

In literature often isolated scenarios for robust optimization are considered. That is, isolated in the sense that only one particular type of noise/uncertainty, present in one particular part of the system or (simulation) model is considered. The most prominent scenarios are:

- Optimization of noisy objective functions (e.g., [FG88, Bey00])
- Optimization of uncertain objective functions (e.g., [KAE+09])
- Optimization on systems in which the design variables are affected by uncontrollable and unmeasurable perturbations (e.g., [BOS03])
- Finding robust optima in anticipation of perturbations of the design variables (e.g., [TG97, GA05, ONL06, PBJ06])
- Finding robust optima in anticipation of fluctuations of the environmental parameters (e.g., [Hop09])
- Finding robust optima in anticipation of different operation conditions of the environmental parameters (e.g., [RKD+11])

In the remainder of this work, we will also consider such isolated scenarios. In particular, we consider two scenarios as exemplary for robust optimization, optimization of noisy objective functions and finding robust optima in anticipation of perturbations of the design variables, and study the way in which to solve such robust optimization problems. However, the reader should be aware of the fact that the practice of robust optimization comprises a broad variety of scenarios of uncertainties and/or noise.

3.3 Real-World Robust Optimization Scenarios

This section is devoted to present three real-world optimization scenarios in which uncertainty and noise are inherent parts of the system of interest. These scenarios are related to the view on robust optimization as presented in this chapter.

3.3.1 Deep Drawing Optimization

The optimization of the design of a deep drawing process for sheet metal forming in engineering is a typical example of a robust optimization problem (see, e.g., [SH04, PLBG07]). The design of a deep drawing process involves finding proper settings for the process (e.g., drawing forces), such that the end product of the deep drawing process is of the desired geometrical shape that complies to, or is as good as possible with respect to properties relating to plasticity, thickness, and probability of forming failure. In practice, finite element software is often used for virtual testing of candidate designs. Hence, the design of a deep drawing process is an optimization problem for which automated optimization techniques are in principle well-suited. Uncertainty and noise arise in several ways in such optimization problems:

A) **Uncertainties and/or noise in the design variables:**
In the real-world manufacturing process, the drawing forces, such as the drawbead forces, the binder force, and the punch force cannot be set infinitely accurately (type 3). Designs should be robust against fluctuations in these variables.

B) **Uncertainties and/or noise in the environmental parameters:**
Parameters within the simulation model are uncertain and/or noisy, e.g., the friction coefficient (type 1/2) or the blank thickness (type 3). Designs should be robust against these uncertainties and fluctuations.

C) **Uncertainties and/or noise in the output:**
Due to the limited accuracy of a (finite element) simulator, the simulation output is not guaranteed to match real-world behavior exactly (type 1/2). The optimization algorithm should be robust against this type of uncertainty and find good solutions despite the difficulty in the quality assessment of candidate designs.

The robust optimization goals for such problems are to find optimal designs (or solutions) that are robust against fluctuations and uncertain conditions in real-world practice, and furthermore to assure robustness of the optimization process itself, that has to deal with the uncertainty in the assessment of the quality of candidate solutions.

3.3.2 Building Performance Optimization

With increasing quality of Building Performance Simulation (BPS) tools, including optimization approaches for finding optimal building designs becomes more and more viable

[HHPW07, EHM$^+$08, Hop09, HTHB11]. Consider, for example, the problem of optimizing building designs for thermal comfort (maximization) and energy consumption (minimization) with respect to the following design variables: infiltration rate (air exchange rate per hour in the building), window fraction (the amount of glass percentage on one wall of the building), load equipment (power equipment per net floor surface), and load lighting (power lighting per net floor surface). By using BSP tools for the evaluation of candidate solutions (i.e., alternative designs), automated optimization can be used to optimize building designs. The ways in which uncertainty and noise arise in such optimization problems are:

B) **Uncertainties and/or noise in the environmental parameters:**
The operation conditions of a building in the real-world are uncertain and fluctuating. An obvious example is the outside temperature that changes from hour to hour and from day to day. For successful integration of optimization in this context, candidate designs should be evaluated with respect to these fluctuating environmental operating conditions.

C) **Uncertainties and/or noise in the output:**
BPS tools are not guaranteed to match the real-world behavior exactly (type 1/2). The optimization algorithm should be able to find high quality designs despite the difficulty in the quality assessment.

Moreover, although we do not consider them in the scope of robust optimization, also uncertainties of class E emerge in such problems. That is, thermal comfort is a typical objective in which preference uncertainty arises as this is an inherently subjective objective, these indicators are fuzzy measures.

3.3.3 Molecular Design Optimization

In *de novo* design of molecular structures, the challenge is to find molecular structures that could be active components of drugs. In order for a molecular structure (or ligand) to be suitable as a drug component, a number of criteria should be met. First, it should be active on the targeted receptor. Second, it should fulfill a number of criteria such that it is actually taken in by the body, it is not toxic or harmful in other ways, and it should be possible to actually create (synthesize) the structure (preferably as easily as possible).

In practice, automated methods can be used for *in silico* design of candidate molecular structures (see, e.g., [NAP09, KBIvdH08, KAE$^+$09]). A simple setup, for instance, is to use docking simulations to predict the binding affinity of a candidate ligand to a receptor, and simple descriptors (such as Lipinski's rule of five [LLDF01]) to determine the likelihood of a candidate ligand to be generally suitable as a drug. Given this, the design task (or optimization task) is to find molecular structures that have a high docking score, and also score well on the simple descriptors (hence, it is a multi-objective optimization problem). The ways in which uncertainty and noise arise in such optimization problems are:

C) **Uncertainties and/or noise in the output:**
The simulation output of the docking simulator is an approximation of the real-world behavior (type 1/2). Due to the use of a stochastic simulator, the outputs of the simulator are noisy (type 3). The optimization algorithm should be robust against these types of uncertainties and noise and find good solutions despite the difficulty in the quality assessment of candidate molecular structures.

Moreover, although we do not note them as being in the scope of robust optimization, also uncertainties of class D and class E emerge in such problems. Vagueness in the constraints can emerge when constraint bounds are used for the simple descriptors that determine the likelihood of a candidate ligand to be generally suitable as a drug (see, e.g., [KEB$^+$09b]). Preference uncertainty can emerge when the objectives are combined into one aggregate scoring function.

3.4 Summary and Discussion

In this chapter we have extended the model of optimization problems as presented in Chapter 2 to a model that includes uncertainties and noise as they can arise in real-world optimization problems. It has been shown that the various sources and types of uncertainty and noise yield a combinatorial explosion of different scenarios. Yet, a few isolated scenarios can be identified that emerge frequently, hence, are worthwhile subjects of study.

In this work, we consider robust optimization as the practice of optimization given uncertainties and/or noise in the system or (simulation) model. Hence, we exclude uncertainties that arise from the modeling of an optimization problem. Robust optimization deals with two goals: 1) the aim to find optimal solutions in noisy/uncertain environments, and 2) the aim to find robust solutions.

The concepts introduced in this chapter were exemplified by means of three real-world optimization scenarios; deep drawing optimization, building performance optimization, and molecular design optimization. For these three scenarios it was illustrated how the various forms of uncertainties/noise enter the optimization model, therewith stressing the importance of robust optimization.

In the second part of this work we will focus on two scenarios of robust optimization for real-parameter optimization problems: optimization of noisy objective functions and the problem of finding robust optima. For these two scenarios we will study how Evolutionary Algorithms, and in particular Evolution Strategies, should be adapted in order to deal with robust optimization problems.