On Uncertainty and Robustness in Evolutionary Optimization-Based MCDM

Daniel E. Salazar Aponte[1], Claudio M. Rocco S.[2], and Blas Galván[1]

[1] University of Las Palmas de Gran Canaria, Institute SIANI,
Edif. Central Pqe. Científico y Tecnológico, Las Palmas 35017, Spain
danielsalazaraponte@gmail.com, bgalvan@step.es

[2] Central University of Venezuela, Faculty of Engineering
Apartado Postal 47937, Los Chaguaramos. Caracas, Venezuela
crocco@reacciun.ve

Abstract. In this article we present a methodological framework entitled 'Analysis of Uncertainty and Robustness in Evolutionary Optimization' or AUREO for short. This methodology was developed as a diagnosis tool to analyze the characteristics of the decision-making problems to be solved with Multi-Objective Evolutionary Algorithms (MOEA) in order to: 1) determine the mathematical program that represents best the current problem in terms of the available information, and 2) to help the design or adaptation of the MOEA meant to solve the mathematical program. Regarding the first point, the different versions of decision-making problems in the presence of uncertainty are reduced to a few classes, while for the second point possible configurations of MOEA are suggested in terms of the type of uncertainty and the theory used to represent it. Finally, the AUREO has been introduced and tested successfully in different applications in [1].

1 Introduction

In this article we are concerned about the use Multiple Objective Evolutionary Algorithms (MOEA)[1] in Multiple-Criterion Decision-Making (MCDM) under uncertainty.

By MCDM we mean the process of selecting a final alternative from a group of more than one solving actions to a problematic (e.g. choice, sorting, ranking) within a common quality framework made up of various figures of merit called criteria, established by an entity called decision maker (DM). No matter what the problematic is, a rational DM is expected to maximize its level of satisfaction by choosing the alternative that scores best in terms of the criteria. Mathematically, we model it as a program of the form[2]:

$$\begin{aligned} &\mathbf{x}^* = \arg\max_{\mathbf{x}} \left(F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x}))^t \right) \\ \text{s.t.:} \quad &\mathbf{x} \in \Omega \\ &\Omega = \{\mathbf{x} \in \mathbf{X} : G(\mathbf{x}) \leq 0 \ , \ H(\mathbf{x}) = 0\} \end{aligned} \tag{1}$$

[1] Acronyms for singular and plural forms are spelled the same hereafter.
[2] With no loss of generality, optimality is expressed in terms of maximization hereafter.

M. Ehrgott et al. (Eds.): EMO 2009, LNCS 5467, pp. 51–65, 2009.

where $F : \mathbf{X} \rightarrow \mathbf{Y}$ is a vector of criteria $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ that map a vector of n decision variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)^t$ (called also *decision vector* or simply *alternative*) from the *decision space* $\mathbf{X}$, into a k-dimensional *objective vector* $\mathbf{y} = (y_1, y_2, \ldots, y_k)^t$ in the *objective space* $\mathbf{Y} \subseteq \mathbb{R}^k, k \in \mathbb{N}$. Additionally, the *feasible space* Ω is defined by two vectors $G(\mathbf{x})$ and $H(\mathbf{x})$ of inequality and equality constraints respectively.

Such a kind of problems are characterized by some conflict amongst the criteria, so that the set of alternatives cannot be arranged as a total order regarding their quality. Consequently, eq. 1 is not satisfied by a unique alternative but a subset of them called *efficient* or *non-dominated*. Typically the relation used for classifying the alternatives is the Pareto dominance, viz.:

Definition 1 (Pareto Dominance). $\mathbf{x}_1$ *dominates* $\mathbf{x}_2$*, denoted* $\mathbf{x}_1 \succ \mathbf{x}_2$ *, iff* $f_i(\mathbf{x}_1) \geq f_i(\mathbf{x}_2) \wedge F(\mathbf{x}_1) \neq F(\mathbf{x}_2);\ i \in \{1, 2, \ldots, k\}$*. If there is no solution dominating* $\mathbf{x}_1$*, then* $\mathbf{x}_1$ *is called non-dominated.*

In order to solve MCDM problems, MOEA are often used to approximate the set of non-dominated solutions. As a subclass of Evolutionary Algorithms, MOEA are searching methods based upon a population sequential sampling process ruled by heuristics. Such heuristics can be implemented in any fashion but in general they find inspiration in some natural processes (like mating and survival -Genetic Algorithms-, foraging -Ant colonies-, flocking -PSO) as well as mathematic (Differential Evolution) and thermodynamic (Simulated Annealing) principles.

Regardless of the final form given to their instances, all of the MOEA share a common principle of evolving towards a higher level of global fitness as iterations go on. In general MOEA associate fitness with Pareto optimality and approximation sets with spatial even distributions. In practice it is possible by defining a ranking procedure concerned by optimality and density built upon some fitness expression which turns out to be function of the mathematical model of eq. 1. Needless to say that, if $F(x)$ cannot be properly assessed as it happens in the presence of uncertainty, the very foundation of the operation of MOEA could be seriously compromised.

As we shall see later on, several algorithms have been proposed to operate under uncertainty. Regardless the computational efficiency of such existing approaches or any other one to come, the variety of sources, types and targets of uncertainty as well as the current theoretical frameworks to represent it, hinders the ability of MOEA designers to develop approaches valid for a wide range of situations. In response, we propose an analytical methodology called *Analysis of Uncertainty and Robustness in Evolutionary Optimization* or AUREO that allows one to study how to use MOEA in MCDM problems under uncertainty from a broad view. First the effort is oriented towards finding the mathematical formulation that suits best the decision-making problem regarding the characteristics of the uncertainty involved, while later on the analysis focuses on the structure of the MOEA propounded as solving technique, according to the characteristics of the problem formulated beforehand and on its efficiency. The benefit of doing so is double. On the one hand, having uncertainty in MCDM problems does not necessarily imply that the MOEA have to cope with uncertainty. We argue that the definitive element to decide whether the MOEA actually have to is the available information. On the other hand, in the case of dealing with uncertainty, to make a device of the structural requirements of the MOEA in terms of uncertainty handling helps one select amongst the existing instances or design new ones.

The remainder of the article is organized as follows: the next section introduces some problems raised by uncertainty and the possible reasonings available to deal with it. Section 3 brings the methodology proposed with some examples while section 4 gives some concluding remarks.

2 Accounting for Uncertainty in Decision-Making

In this section we give a glance at the notion of uncertainty, its relation with the decision-making and the existing views and reasonings about it.

The term uncertainty is understood in different ways, all of them related to defects of knowledge and information (for further insight see [2]). We adopt the full identification of uncertainty with imperfection of information, data or evidence herein. When the lack of information is originated by the inherent variability of physical systems and thus it cannot be reduced by further empirical efforts we say the uncertainty is *aleatory*. By contrast when the actual state of uncertainty is reducible by additional information of the system or its environment we call it *epistemic* uncertainty. A mixed aleatory-epistemic uncertainty is also possible.

To account for the effect of uncertainty in decision-making, consider the different scenarios depicted in fig. 1. The first target of our uncertainty analysis is the domain. This one can be subject to aleatory (case 1), epistemic (case 3) or mixed uncertainty (not depicted). In all of these cases the uncertainty associated to $\mathbf{x}$ should be propagated through $F(\mathbf{x})$ onto space $\mathbf{Y}$. If $F(\mathbf{x})$ is free of uncertainty, the propagation of epistemic uncertainty will yield epistemic objective vectors $\mathbf{y}$ (trajectory 1-3-5). In this case both the decision and objective vectors will be characterized by bounding sets (usually an interval) enclosing the true but unknown values. Likewise the propagation of aleatory decision vectors through a function free of uncertainty will yield objective vectors that actually are random variables (trajectory 3-6-9). On the other hand, the second target of the uncertainty analysis is the function. Indeed, $F(\mathbf{x})$ can be intrinsically uncertain (e.g. noisy or dynamic functions) or the way we assess it can be subject of aleatory (e.g. Monte Carlo simulation) or epistemic uncertainty (e.g. interval approximation), although the functional expression is deterministic. In such a case the result will be uncertain no matter if the input is (trajectories {1,3},5,{7,8,9}) or not (trajectories 2,5,{7,9}) uncertain. Notice that we can have objective vectors $\mathbf{y}$ subject to mixed uncertainty, i.e. the result is a set of possible sets of outcomes.

As we just have seen, whether it is epistemic, aleatory or mixed, the presence of uncertainty always entails comparing sets of objective instead of precise vectors. Consequently, one of the challenges risen to decision-making is how to compare and classify alternatives in terms of sets comparisons. We shall consider next the

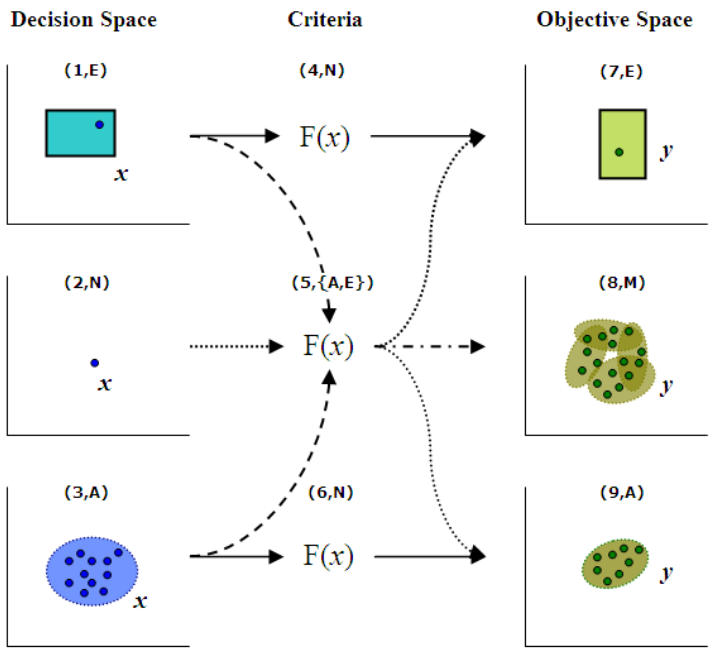

Fig. 1. Effect of Uncertainty in Decision-Making: labels (a,b) indicate the index a and the type of uncertainty b of each element. Uncertainty types are denoted by N (none), A (aleatory), E (epistemic) and M (mixed). Possible scenarios are denoted by different arrow types.

different theoretical frameworks for representing uncertainty and how they can influence decision-making and MOEA design.

2.1 Reasoning about Uncertainty

Theories about uncertainty provide us with logical frameworks to make statements about uncertain quantities. The basic principle that underpins the reasoning about uncertainty is that there is a set called *universe of discourse* denoted by $\mathcal{X}$ herein, that contains all the possible and pertinent states that an uncertain quantity can adopt. For instance, when we define $\mathbf{x}$ with an interval, we intrinsically state that, in principle, the evidence shows that all the values contained by the latter might be adopted by the former. Axioms and logic derivations formulated afterwards about the universe of discourse define the theoretical frameworks.

If $\mathcal{X}$ is continuous we can define it as an interval. Now if our uncertain decision vectors can be treated as intervals, we can use *Interval Arithmetic* to propagate $\mathbf{x}$ though $F(\mathbf{x})$ to assess $\mathbf{y}$. Regardless the nature of the uncertainty, the resulting interval is expected to bound the true value(s) of $\mathbf{x}$. On the other hand, if one has more information about the nature of the uncertainty at her disposal, one should use it.

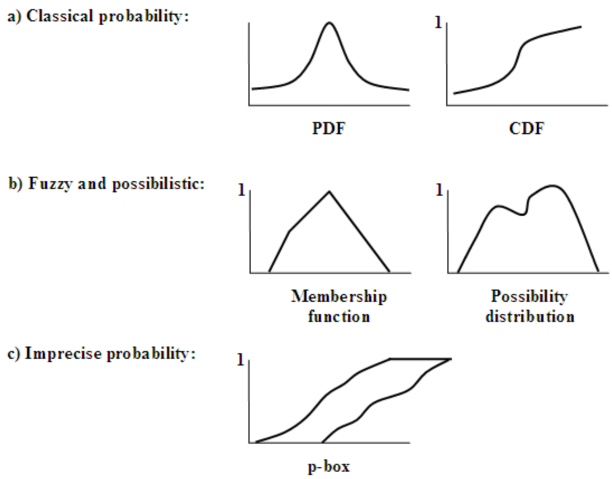

Fig. 2. Elements of the main theories for represetning uncertainty

Fig. 2 sketches some relevant elements of the main theories about uncertainty. In classical probability theory every element of the universe of discourse is assigned a probability. The relation between discrete domains and probabilities are captured by probability mass functions while the probability density function (PDF) are used with continuous domains. In both cases, it is possible characterize tendencies of variation within the universe of discourse by some symmetry axis of such variation (expected value) or its size (variance) amongst other things. It is also possible to assess the probability of the uncertain quantity adopting values within a set (like $P(X \leq x)$): this is expressed as a cumulative probability distribution (CFD).

Sometimes $\mathcal{X}$ is roughly or ill defined, as when one says '*it's cold*' and we know that 'cold' has different meanings according to the person who says it. This kind of uncertainty appears in natural language or when one handles blurred concepts (e.g. when defining the DM's preferences). In this case $\mathcal{X}$ is described by a membership functions that assigns numbers in [0,1] where 0 means the argument is not contained in the set and 1 the opposite. For insight into fuzzy logic see [3].

We can also extend the previous concept to talk about 'the possibility' of an event, using a bivalent logic (it is or not possible) or a graded logic captured by possibility distributions, which are in deep connection with the notion of probability, although saying that something is possible is different than saying it is probable. *Possibility Theory* provides therefore a non-probabilistic framework to represent epistemic uncertainty.

One of the shortcomings of classical probability is that it is not suitable for representing epistemic or mixed uncertainty. For instance, having limited evidence, an agent could make imprecise statements like '*the probability of x is in* $[0.3, 0.5]$' or '*vector* $\mathbf{x}$ *follows a normal PDF with mean in* $[3.26, 4.5]$ *and variance in* $[0.82, 0.97]$. Statements of the such can be captured by p-boxes e.g. saying that every CDF within the p-box is a possible representation for the actual aleatory uncertainty, whereas the epistemic uncertainty is captured by the fact that we don't actually know the true PDF. *Dempster-Shafer Theory* [4], *Walley's Theory* [5] or *p-boxes* [4] support theoretically this kind of approaches.

Table 1 summarizes the main elements of such approaches. From a practical viewpoint the relevant issue is that the theoretical frameworks mentioned previously are best used in certain situations as they cover distinct types and sources of uncertainty. Besides, all of them prescribe propagation methods, which means that in the plausible case of having uncertain domains represented by one of the theories mentioned so far, the outputs and therefore the ranking of alternatives within the MOEA will also be related to such theory.

Table 1. Some relevant elements of theories about uncertainty

Theory	Accounts for	Especially suitable for	Propagation method	Notable elements
Fuzzy logic	Graded membership of elements to sets	Linguistic epistemic uncertainty	Extension principle	Membership functions, core and support sets
Possibility	Binary or graded membership of elements to sets	Epistemic uncertainty	Choquet integrals and extension principle	Possibility distributions, possibility and necessity measures
Classical Probability	Likelihood of events	Aleatory uncertainty	Convolution and Monte Carlo simulation	PDF, percentiles, mean, variance and higher moments
Imprecise Probability	Imprecise probabilities and subjective judgements on sets	Epistemic and aleatory uncertainty	Convolution and Monte Carlo simulation	Belief and plausibility measures. Intervals for distributions and moments

3 Analysis of Uncertain and Robustness in Evolutionary Optimization (AUREO)

In this section we describe a two-stage methodology for the 'Analysis of Uncertain and Robustness in Evolutionary Optimization' (AUREO). The basic premise of this framework is that the analysis of the available information about a problem subject to uncertainty (fig. 3) determines the solving program (stage 1) and the MOEA structure (stage 2).

Consider a refined mathematical program based on eq. 1. Let $F(\mathbf{x}, \mathbf{p})$ represents the DM's criteria in a free-of-uncertainty scenario in terms of the objective vector $\mathbf{x}$ and a vector of environmental parameters $\mathbf{p}$. As discussed in sec. 2 uncertainty may come up as lack of information about the variables $\mathbf{x}$ and $\mathbf{p}$. Besides the environmental parameters in $\mathbf{p}$ are often subject to change in real world. The assessment of $F(\cdot)$ might be a source of uncertainty as well. The first stage of AUREO, summarized in fig. 3, focuses therefore on the form of the MCDM problem considering the existing uncertainties.

If the model is accepted to be adequate the attention centres on the input vectors $\mathbf{x}$ and $\mathbf{p}$. If such vectors are free of uncertainty, no action is required, otherwise the analyst should ask about the type of such uncertainty and further investigate the best theory to represent it. Immediately the attention focuses on the outcomes of $F(\mathbf{x}, \mathbf{p})$ to find out if the $f_i(\mathbf{x}, \mathbf{p})$ are dynamic functions or if they will be assessed through surrogate or approximate models.

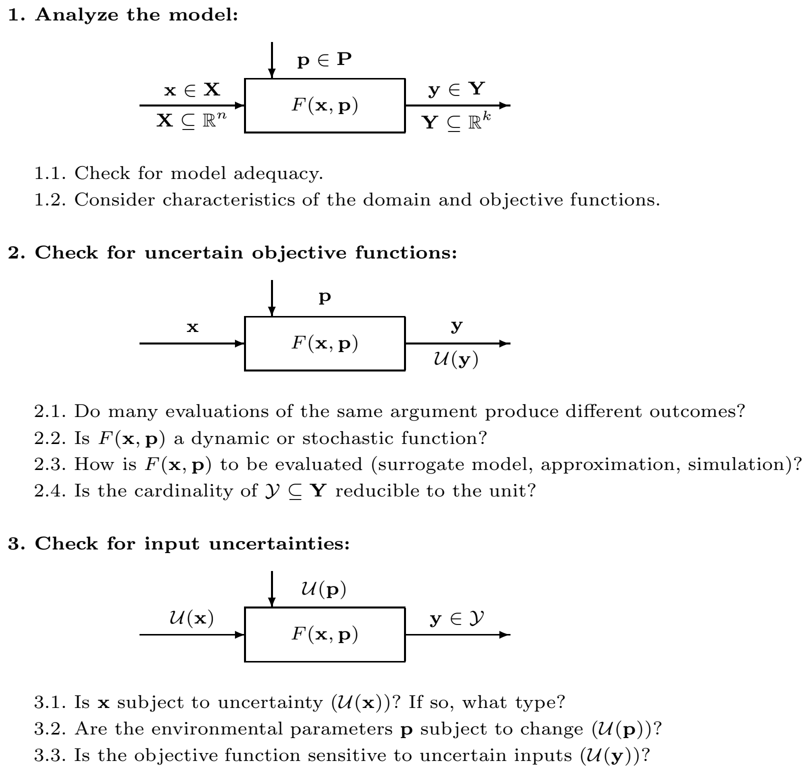

Fig. 3. AUREO Stage 1: Analysis of interactions between model and uncertainties

Once this analysis is ready, the original MCDM problem can be transformed into a new one based on new criteria defined in terms of the -possibly uncertain-input $(\mathbf{x}, \mathbf{p})$, the original criteria $F(\mathbf{x}, \mathbf{p})$, the uncertain outcome $\mathbf{y}$ and the theory employed to represent the uncertainty. For the sake of generality, let $\mathbf{x}$ be a nominal vector denoting a precise alternative and let $\mathcal{U}(\mathbf{x})$ denotes the uncertainty associated to $\mathbf{x}$, i.e. the universe of discourse and other particular elements related to the uncertainty representation (see sec. 2.1). The same notation stands for the uncertainty of $\mathbf{p}$ and $\mathbf{y}$.

Now, in the most general way, the new MCDM problem can be expressed through the following:

Definition 2 (Uncertainty-handling program). *Let $F(\cdot)$ be a measure of performance of a system determined by the decisional vector $\mathbf{x}$ and influenced by a vector of environmental parameters $\mathbf{p}$, each of which is subject to uncertainties $\mathcal{U}(\mathbf{x})$ and $\mathcal{U}(\mathbf{p})$ respectively. Let $C(\cdot)$ be a vector of constraints defined regarding the original constraints for the optimization of $F(\cdot)$. Finally let $I(\cdot)$ be a vector of requirements imposed upon the performance. The resulting uncertainty-handling formulation consists in solving the following program*

$$
\begin{array}{rc}
 & \max \; R\left(F(\mathbf{x},\mathbf{p}),\mathbf{x},\mathbf{p},\mathcal{U}(\mathbf{x}),\mathcal{U}(\mathbf{p})\right) \\
\textit{s.t.:} & \\
 & \mathbf{x}\in\mathbf{X}\;,\;\; \mathbf{p}\in\mathbf{P} \\
 & C(\mathbf{x},\mathbf{p},\mathcal{U}(\mathbf{x}),\mathcal{U}(\mathbf{p}))\le 0 \\
 & I(F(\mathbf{x},\mathbf{p}),\mathbf{x},\mathbf{p},\mathcal{U}(\mathbf{x}),\mathcal{U}(\mathbf{p}))\le 0
\end{array}
\tag{2}
$$

The new function denoted $R(\cdot)$ is typically an expression of risk, reliability or robustness. For example, in the case of pure uncertain functions, $R(\cdot)$ can be formulated as the original $F(\cdot)$ plus a measure of uncertainty to be minimized. $R(\cdot)$ can also account for reliability or robustness when the input is uncertain, resulting in a *robustness-seeking program*. Vector $I(\cdot)$ on the other hand, accounts for requirements formulated by the DM as additional performance constraints due to uncertainty (e.g. acceptance thresholds for variance or interquartile distances). From the previous program derive two classes of definitions of robustness with well defined solving procedures and a third mixed class that combines the reasoning of the preceding classes in order to solve problems with the least amount of information. Table 2 shows what class is applicable regarding the amount of information available.

Table 2. AUREO Stage 1: Classes of uncertainty-handling formulations according to the available information

Input: $\mathbf{x},\mathcal{U}(\mathbf{x}),\mathbf{p},\mathcal{U}(\mathbf{p})$		**Output:** $\mathbf{y},\mathcal{U}(\mathbf{y})$	
$\mathcal{U}(\mathbf{x})$	$\mathcal{U}(\mathbf{p})$	$I(\cdot)$ definable	$I(\cdot)$ undefinable
None	None	**Pure uncertain functions ≡ Class 1**	
None	Definable	**Class 1**	**Class 1**
None	Undefinable	**Class 2**	**Class 3**
Definable	None	**Class 1**	**Class 1**
Definable	Definable	**Class 1**	**Class 1**
Definable	Undefinable	**Class 2**	**Class 3**
Undefinable	None	**Class 2**	**Class 3**
Undefinable	Definable	**Class 2**	**Class 3**
Undefinable	Undefinable	**Class 2**	**Class 3**

Class 1: Uncertainty Propagating Programs. This class is characterized by a suitable description of $\mathcal{U}(\mathbf{x})$ and $\mathcal{U}(\mathbf{p})$ in such a way that the uncertainty can be propagated through $F(\cdot)$. If the uncertainty is aleatory, $\mathcal{U}(\mathbf{x})$ and $\mathcal{U}(\mathbf{p})$ have associated PDF. For instance, $\mathcal{U}(\mathbf{x})$ could be a normally distributed number $N(\mathbf{x},\sigma)$. On the contrary, if the uncertainty is epistemic, $\mathcal{U}(\mathbf{x})$ might be a crisp or a fuzzy interval, a p-box or something of the like.

Once this program has been identified, the second stage of AUREO consists in defining how the optimality and the density are to be assessed within the MOEA as well as in addressing efficiency issues. As shown in fig. 1 whenever the uncertainty is propagated the outcomes become sets. Thus, the first problem risen is how to decide about optimality using sets. Consider fig. 4: according

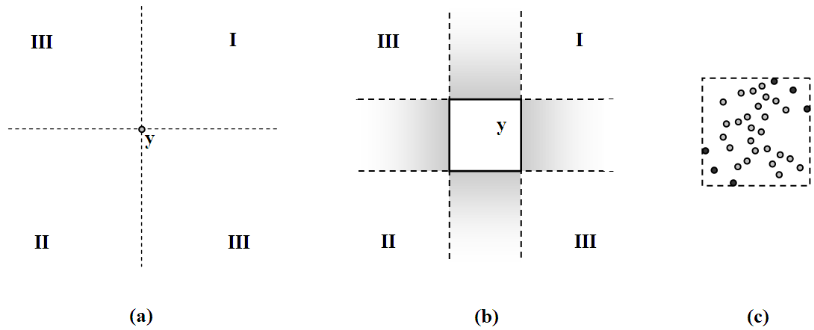

Fig. 4. Spatial dominance relationships regarding **y** in (panel a) the absence and (panel b) in the presence of uncertainty. The interval representing **y** in panel b can be constructed for discrete sets using the extreme (dark) points (panel c).

to the Pareto optimality (def. 1) in the absence of uncertainty every solution in region I (panel a) dominates **y**, in region II is dominated by **y** and in region III is non-dominated regarding **y**. In the presence of uncertainty (panel b) there is an additional (colored cross) region such that any set intersecting that area cannot be classified in terms of dominance without an additional criterion. In such a case the comparison of sets is often reduced to a comparison of representative points, but can also be settle using the whole sets. Density can also be assessed using the approaches just mentioned.

Working with representative points: If we can define a few crisp points representing the main features of the uncertain outcomes, we can solve the uncertainty-handling program with a regular application of existing MOEA. Defining such points, however, can be very tricky and computational cumbersome in practice. For representing a set, the extreme points, some symmetry axis and some size measures are commonly used. Let us consider some examples regarding the different theories to represent uncertainty.

Best and worst case are risk criteria that corresponds to extreme points of crisp and fuzzy intervals. Uniform distributions also exhibit finite extreme points, but in general such points are infinite in probability distributions. Nevertheless, extreme quantiles can be used to implement best and worst cases. With imprecise probabilities there are intervals of quantiles so one can use the best of the bests and the worst of the worst of the cases.

The mean value is commonly used as symmetry axis although the median can be used as well. A common approach is to optimize the mean value of the sets, minimizing sometimes its size simultaneously. This is the typical way to implement robustness (see *robust optimization* in tab. 3). In classical probabilistic contexts the size is measured as variance in $R(\cdot)$ although the interquartile range is another possibility. As one only have estimations of means and variances instead of the true statistics most of the time, it is more than desirable to have proper statistical tests supporting MOEA the ranking procedure. On the

Table 3. Some state of the art MOEA for optimization under uncertainty

Realm of study	Authors and works
Probabilistic Dominance:	
Optimization with interval fitness value	Teich [12]
Optimization with noisy fitness function	Hughes [13], Fieldsen et al. [14]
Quality Indicator-Based Procedures:	
Indicator-based optimization	Basseur and Zitzler [15]
Robust Optimization:	
Optimization with uncertainty propagation	Sörensen [16], Ray [17], Deb and Gupta [18], Barrico et al. [19]
Info-gap based robust design	Lim et al. [20]
Reliability-based optimization	Deb et al. [21]
Non-Probabilistic Procedures:	
Optimization with epistemic uncertainty	Limbourg [22] and Salazar A. [23]

other hand, in the case of handling alternative uncertainty theories, the interested reader might also define ranking procedure based on fuzzy and possibilistic means and variances [6,7,8,9] or mean values for imprecise probabilities [10,11].

Working with the whole sets: Treating the whole set instead of a few points is also possible. For probabilistic contexts the concept of probabilistic dominance provides a way of doing this. The basic idea is to assess the probability of one whole outcome dominating another one and to accept dominance if such probability surpass an acceptance threshold. Formulae exist for assessing dominance in classical [24] and imprecise probabilities [25]. The main drawback in practice is that there is no easy way to estimate the probability of dominance if the PDF of the concerned outcomes are not available.

There are also methods for assessing dominance in fuzzy and possibilistic contexts [26]. Such methods solve the problem of optimality but left the density control unattended. One possibility is to resort to representative points like the mean value to assess density, or to maximize the minimal distance between two neighbours.

Table 3 lists some of the existing MOEA that can be used in Class 1 problems. Chronologically speaking, the first attempts implemented probabilistic dominance. These approaches rely upon the assumption that the PDF of all the outcomes are known and share the same shape. In some real problems such an assumption stands but in general it constitutes a limitation. Current robust optimization approaches, on the other hand, do not make a sharp distinction between aleatory and epistemic uncertainty in their proposals. The user may be aware of this to avoid careless uncertainty handling. Other approaches make use of indicator quality measures to handle density assessment [23] or optimality and density as well [15]. Regarding the two referred here, the former were developed to work with intervals and seems suitable for epistemic uncertainty although the optimality is settle by rules that could not be universally accepted. By contrast, the latter approach relies on the assumption of probabilistic outcomes so it seems suitable for aleatory uncertainty albeit the algorithm makes no statistical treatment of the outcomes. A first conclusions is that making better and careful treatments of uncertainty taking into account the theoretical frameworks for reasoning about it is still a challenge for the design of MOEA.

Class 2: Robust Domain-Seeking Programs. Some situations may keep the uncertainty associated with the input variables from being propagated through $F(\cdot)$ and therefore handled as a Class 1 robustness problem. Whether the uncertainty is purely epistemic or mixed in nature, any additional assertion about $\mathcal{U}(\mathbf{x})$ or $\mathcal{U}(\mathbf{p})$ to transform the problem into one of Class 1, entails making some assumptions that could be wrong, leading to identify inadequate alternatives in $\mathbf{Y}$ as optimal solutions.

For example, if the DM knows that the real value of the nominal vector $\mathbf{x}$ is susceptible to vary but ignore the range of such variation, to assume the set of values that $\mathbf{x}$ can take or their likelihood may underestimate the uncertainty. The DM is therefore compelled to maximize the range of 'acceptable' realizations of $\mathbf{x}$ in order to hedge against regrettable consequences. In that sense, robustness is sought by widen the range of possible inputs, or in other words, by maximizing the cardinality of the $\mathcal{U}(\mathbf{x})$.

The Class 2 is therefore characterized by the existence of a constraints vector $I(F(\cdot)) \leq 0$ that constitute desired performance levels of attainment (quality requirements), and a robustness function $R(\cdot)$ that aims at maximizing the range of variation of the input variables that conform with $I(\cdot)$. The robust design problems is a good example of this class. For instance, [27] brings an application where the reliability R_s of a system is a function of the reliability R_i of its components. Since R_i may change, one is interested in knowing the effect of such variations over R_s although the $\mathcal{U}(R_i)$ are unknown. Instead of making assumptions about the $\mathcal{U}(R_i)$, the DM is asked about the desired performance requirements of R_i. This way the DM defines the restriction $I(R_s) = 0.90 \leq R_s \leq 0.99$ and the Class 2 program takes the form of *find* $\{[\underline{R_i}, \overline{R_i}]\} = \arg\max_i \prod_i (\overline{R_i} - \underline{R_i})$ *s.t.* $R_i \in [0.8, 1]$ *and* $I(R_s)$. The second objective is the minimization of the maximal cost that one can incur when selecting components within the range defined by the objective $R(\cdot)$. Notice that such a problem can be handled by a regular MOEA simply using interval arithmetic to check $I(R_s)$. For details about the formulation and the efficiency issues see [27].

Class 3: Mixed Robust-Seeking Procedure. To close this section, let us consider again the general robustness-seeking program formulated in def. 2. If the DM and the analyst are unable to characterize the input uncertainty nor the desired performance levels of attaintment, or alluding def. 2, if they cannot set $\mathcal{U}(\mathbf{x})$, $\mathcal{U}(\mathbf{p})$ nor $I(\cdot)$, the actual definition of the robustness function $R(\cdot)$ is not possible. It is therefore mandatory to generate information to help the DM to make their minds about $\mathcal{U}(\mathbf{x})$, $\mathcal{U}(\mathbf{p})$ or about $I(\cdot)$, in such a way that the problem collapses into a Class 1 or Class 2 program. The first case could be possible by means of further elicitation of uncertainty, while the second requires initial assumptions about $\mathcal{U}(\mathbf{x})$ and $\mathcal{U}(\mathbf{p})$ to roughly approximate the frontier, allowing to set $I(\cdot)$ and to solve the corresponding Class 2 program afterwards.

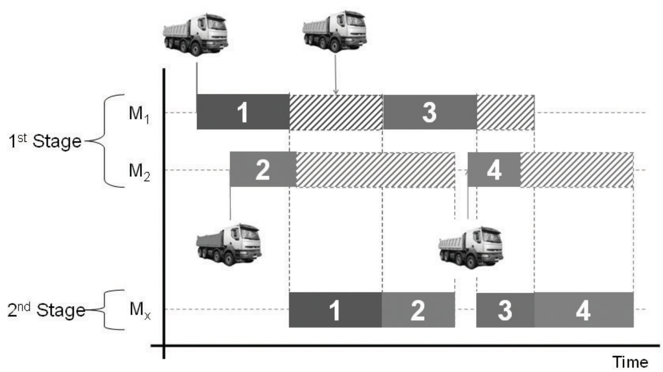

Fig. 5. Scheduling problem in a waste treatment plant: Machines M_m performs the unloading while the mixer M_x performs the waste processing [28]

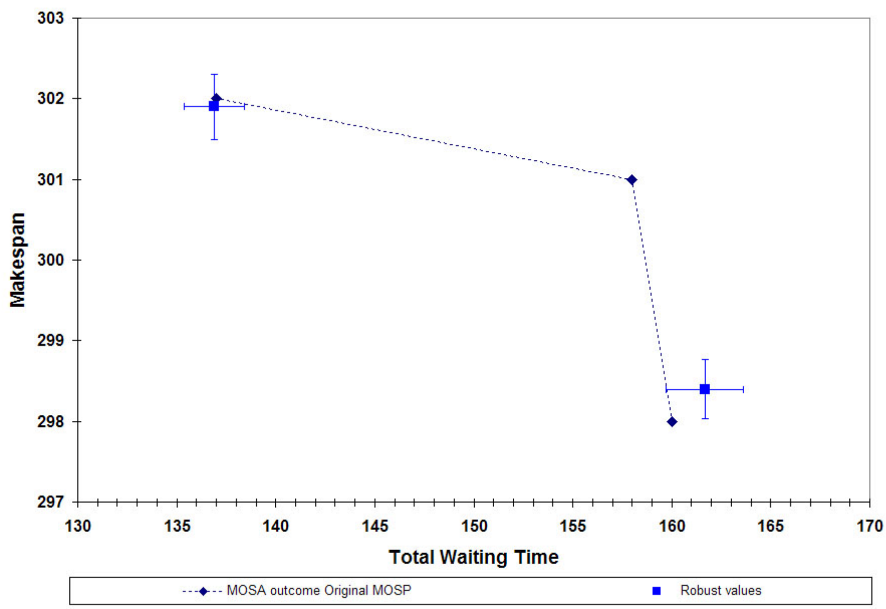

Fig. 6. Robust schedules that satisfy $I(\cdot)$ and maximizes the allowed variations in arrival times (objectives in minutes) [28]

This procedure is explored in [28]. The original problem consists in minimizing the makespan and the total waiting time of a waste treatment plant. The waste is carried by trucks that arrive at scheduled times and the operations are the unloading of the trucks into silos and the transference of the silos' content into a critical machine (see fig. 5). Assuming that the processing times do not change and the sequence of operations cannot be rescheduled on line, the new problem arises when considering uncertain arrival times for the trucks. The goal is to identify the more robust sequence of tasks given that no information about the possible variations of arrival times nor about $I(\cdot)$ were available a priori.

In order to solve this problem as one of the Class 2, a first assumption was made about the arrival time, accepting a uniform variation of $\pm 10'$ around the expected timing. Criteria were assessed using Monte Carlo simulation. The results obtained using MOSA [29] with this assumption showed that it is possible to absorb variations of 10' without deviations from the deterministic results greater than 1'. Constraint $I(\cdot)$ was set to allow at most deviations of 2' from the known deterministic optima. With this in mind the new goal is to find a sequence conforming $I(\cdot)$ that accepts the greater range of variations above 10' in the arrival time. The results are shown in fig. 6.

4 Final Remarks

In this article we offered an analysis of the interaction between MCDM and MOEA, emphasizing the importance of considering the different forms for representing uncertainty. The possible instances of MCDM problems under uncertainty were classified into three classes according to the elements concerned by uncertainty. In the Class 1 it is necessary to deal with uncertain outcomes so the MOEA designed to work with this group of problem have to implement mechanisms to propagate uncertainty and to rank the solutions in terms of optimality and density. Most of the existing approaches lie within this class, but there is still room for more research on the integration of MOEA and uncertainty theories to better treat aleatory and epistemic uncertainty and to cover scenarios not considered yet.

In Class 2 the problem does not require the MOEA to handle uncertain outcomes. Readers interested in this approach are referred to [1,27] for further details. Finally in Class 3 the decision-making problem requires additional efforts to generate information and to reduce it to a Class 1 or Class 2. We briefly exemplified the application of MOEA to solve a real Class 3 problem. For further details see [28].

References

1. Salazar Aponte, D.E.: On Uncertainty and Robustness in Evolutionary Optimization-based Multi-Criterion Decision-Making, PhD Thesis, University of Las Palmas de Gran Canaria, Spain (2008)
2. Jousselme, A.L., Maupin, P., Bossé, E.: Uncertainty in a situation analysis perspective. In: Proceedings of the Sixth International Conference of Information Fusion 2003, pp. 1207–1214 (2003)
3. Klir, G.J., Folger, T.A.: Fuzzy Sets, Uncertainty, and Information. Prentice-Hall International, USA (1988)
4. Ferson, S., Kreinovich, V., Ginzburg, L., Myers, D.S., Sentz, K.: Constructing probability boxes and Dempster-Shafer structures. Technical report SAND2002-4015, Sandia National Laboratories (2002)

5. Walley, P.: Statistical Reasoning with Imprecise Probabilities. Chapman and Hall, London (1991)
6. Kröger, R.: On the variance of fuzzy random variables. Fuzzy Sets and Systems 92(1), 83–93 (1997)
7. Dubois, D., Prade, H.: The mean value of a fuzzy interval. Fuzzy Sets and Systems 24(3), 279–300 (1987)
8. Bronevich, A.G.: The Maximal Variance of Fuzzy Interval. In: Proceedings of the Third International Symposium on Imprecise Probabilities and Their Applications ISIPTA 2003. Electronic proceedings (2003), http://www.carleton-scientific.com/isipta/2003-toc.html
9. Carlsson, C., Fullér, R.: On possibilistic mean value and variance of fuzzy numbers. Fuzzy Sets and Systems 122, 315–326 (2001)
10. Kreinovich, V., Xiang, G., Ferson, S.: Computing mean and variance under Dempster-Shafer uncertainty: Towards faster algorithms. International Journal of Approximate Reasoning 42(3), 212–227 (2006)
11. Bruns, M., Paredis, C., Ferson, S.: Computational Methods for Decision Making Based on Imprecise Information. In: NSF Workshop on Reliable Engineering Computing, Savannah, GA (2006)
12. Teich, J.: Pareto-front exploration with uncertain objectives. In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D.W. (eds.) EMO 2001. LNCS, vol. 1993, pp. 314–328. Springer, Heidelberg (2001)
13. Hughes, E.J.: Constraint Handling with Uncertain and Noisy Multi-Objective Evolution. In: Proceedings of the IEEE Congress on Evolutionary Computation 2001 (CEC 2001), pp. 963–970. IEEE Service Center, Piscataway (2001)
14. Fieldsend, J.E., Everson, R.M.: Multi-objective Optimisation in the Presence of Uncertainty. In: Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC 2005), pp. 243–250. IEEE Service Center, Edinburgh (2005)
15. Basseur, M., Zitzler, E.: Handling Uncertainty in Indicator-Based Multiobjective Optimization. Int. J. Computational Intelligence Research 2(3), 255–272 (2006)
16. Sörensen, K.: A framework for robust and flexible optimisation using metaheuristics. OR4: A Quarterly Journal of Operations Research 1(4), 341–345 (2003)
17. Ray, T.: Constrained robust optimal design using a multiobjective evolutionary algorithm. In: IEEE Congress on Evolutionary Computation (CEC 2002), pp. 419–424. IEEE Service Center, Honolulu (2002)
18. Deb, K., Gupta, H.: Searching for robust pareto-optimal solutions in multi-objective optimization. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 150–164. Springer, Heidelberg (2005)
19. Barrico, C., Henggeler Antunes, C.: Robustness Analysis in Multi-Objective Optimization Using a Degree of Robustness Concept. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2006), pp. 1887–1892. IEEE Service Center, Vancouver (2006)
20. Lim, D., Ong, Y.S., Jin, Y., Sendhoff, B., Lee, B.S.: Inverse Multi-objective Robust Evolutionary Design. Genetic Programming and Evolvable Machines Journal 7(4), 383–404 (2006)
21. Deb, K., Padmanabhan, D., Gupta, S., Mall, A.K.: Reliability-based multi-objective optimization using evolutionary algorithms. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 66–80. Springer, Heidelberg (2007)
22. Limbourg, P.: Multi-objective optimization of problems with epistemic uncertainty. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 413–427. Springer, Heidelberg (2005)

23. Limbourg, P., Salazar Aponte, D.E.: An Optimization Algorithm for Imprecise Multi-Objective Problem Functions. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2005), pp. 459–466. IEEE Service Center, Edinburgh (2005)
24. Graves, S.: Probabilistic dominance criteria for comparing uncertain alternatives: A tutorial. Omega 37(2), 346–357 (2009)
25. Limbourg, P., Kochs, H.D.: Multi-objective optimization of generalized reliability design problems using feature models–A concept for early design stages. Reliab. Eng. Sys. Saf. 93(6), 815–828 (2008)
26. Hernandes, F., Lamata, M.T., Verdagay, J.L., Yamakami, A.: The shortest path problem on networks with fuzzy parameters. Fuzzy Sets and Systems 158(14), 1561–1570 (2007)
27. Salazar, A., D.E., Rocco, S., C.M.: Solving Advanced Multi-Objective Robust Designs By Means Of Multiple Objective Evolutionary Algorithms (MOEA): A Reliability Application. Reliab. Eng. Sys. Saf. 92(6), 697–706 (2007)
28. Salazar, D., Gandibleux, X., Jorge, J., Sevaux, M.: A Robust-Solution-Based Methodology to Solve Multiple-Objective Problems with Uncertainty. In: Barichard, V., Ehrgott, M., Gandibleux, X., T'Kindt, V. (eds.)) Multiobjective Programming and Goal Programming. LNEMS, vol. 618, pp. 197–207. Springer, Heidelberg (2009)
29. Ulungu, E.L., Teghem, J., Fortemps, P.H., Tuyttens, D.: MOSA Method: A Tool for Solving Multiobjective Combinatorial Optimization Problems. J. Multi-Criteria Decision Analysis 8, 221–236 (1999)