International Journal of Computational Intelligence Research.
ISSN 0973-1873 Vol.2, No.3 (2006), pp. 255–272
© Research India Publications http://www.ijcir.info

Handling Uncertainty in Indicator-Based Multiobjective Optimization

Matthieu Basseur[1] **and Eckart Zitzler**[2]

[1]LIFL/CNRS/INRIA, University of Lille,
59655 Villeneuve d'ascq, France
email: *basseur@lifl.fr*

[2]IT & EE Department, ETH Zurich,
8092 Zurich, Switzerland
email: *zitzler@tik.ee.ethz.ch*

Abstract: **Real-world optimization problems are often subject to uncertainties caused by, e.g., missing information in the problem domain or stochastic models. These uncertainties can take different forms in terms of distribution, bounds, and central tendency. In the multiobjective context, some approaches have been proposed to take uncertainties into account within the optimization process. Most of them are based on a stochastic extension of Pareto dominance that is combined with standard, non-stochastic diversity preservation mechanisms. Furthermore, it is often assumed that the shape of the underlying probability distribution is known and that for each solution there is a 'true' objective value per dimension which is disturbed by noise.**

In this paper, we consider a slightly different scenario where the optimization goal is specified in terms of a quality indicator—a real-valued function that induces a total preorder on the set of Pareto set approximations. We propose a general indicator-model that can handle any type of distribution representing the uncertainty, allows different distributions for different solutions, and does not assume a 'true' objective vector per solution, but in general regards a solution to be inherently associated with an unknown probability distribution in the objective space. To this end, several variants of an evolutionary algorithm for a specific quality indicator, namely the ϵ-indicator, are suggested and empirically investigated. The comparison to existing techniques such as averaging or probabilistic dominance ranking indicates that the proposed approach is especially useful for high-dimensional objective spaces. Moreover, we introduce a general methodology to visualize and analyze Pareto set approximations in the presence of uncertainty which extends the concept of attainment functions.

Keywords: uncertainty, multiobjective optimization, evolutionary algorithms, quality indicators

I. Motivation

Knowledge about the set of Pareto-optimal solutions is useful in many applications involving multiple objectives. Therefore, considerable research, particularly in the context of evolutionary computation, has been devoted to generating methods, i.e., techniques that try to generate the entire Pareto set or approximations of it. One recent approach is based on quality indicators where a function I assigns each Pareto set approximation a real value reflecting its quality, cf. [26]; the goal is to identify a Pareto set approximation that minimizes (or maximizes) I. As such, I induces a total order of the set of *approximation sets* in the objective space, in contrast to the classical aggregation functions like weighted sum that operate on single solutions only and gives rise to a total order of the corresponding *objective vectors*. In [16] and [24], different indicator-based multiobjective optimizers have been proposed. The advantage of the indicator concept is that no additional diversity preservation mechanisms are required, and 24) have demonstrated that indicator-specific search can yield superior results in comparison with popular algorithms such as SPEA2 [25] and NSGA-II [6] and with respect to the indicator under consideration.

Since evolutionary multiobjective optimization using quality indicators is a relatively new concept, it is an open question how to deal with uncertainties in this framework. Many real-world optimization problems are subject to uncertainties and therefore this aspect needs to be accounted for. Among the different types of uncertainty one can distinguish, cf. [15], we here consider the case that the calculation of the objective function values is randomized, i.e., every time a solution is evaluated, a different objective vector may be returned. Such a scenario emerges, e.g., if the underlying computational model involves stochastic components such as Monte Carlo simulation [21].

While uncertainty in the objective functions gained some attention in the single-objective context [1; 15], only few studies address this problem within a multiple criteria setting[1]. [13] were among the first to discuss uncertainty in the light of generating methods, although they did not propose a particular multiobjective optimizer for this purpose. Several years later, 14) and 22) independently proposed stochastic extensions of Pareto dominance and suggested similar ways to integrate probabilistic dominance in the fitness assignment procedure; both studies consider special types of probability distributions. In [2], another ranking method is proposed which is based on the average value per objective and the variance of the set of evaluations. Similarly, 7) suggested to consider for each dimension the mean over a given sample of objective vectors and to apply standard multiobjective optimizers for deterministic objective functions.

Most of the aforementioned approaches cannot directly be used in indicator-based search, only averaging the objective values as, e.g., in [7] is a generally applicable strategy. Furthermore, existing studies assume certain characteristics (symmetry, shape, etc.) of the probability distribution that determines to which objective vectors a solution may be mapped to. In other words, the corresponding methods rely and exploit problem knowledge, which may not be available, particularly with real-world applications. Therefore, we propose a general indicator-based model for uncertainty where every solution is associated with an *arbitrary* probability distribution over the objective space. Extending [24], different algorithms for this model and a particular quality indicator, namely the ϵ-indicator, are presented and investigated in comparison with the methods in [14] and [7]. Moreover, this paper also considers the issue of analyzing and visualizing Pareto set approximations in the presence of uncertainty, which to our best knowledge has not been considered so far in the multiobjective literature. To this end, an extension of the empirical attainment function [11] and corresponding algorithms are suggested. In the last section, we will summarize the different issues of this paper and discuss about some open questions.

II. An Indicator-Based Model for Uncertain Environments

A. Multiobjective Optimization Using Quality Indicators

Let X denote the search space of the optimization problem under consideration and Z the corresponding objective space. Without loss of generality, we assume that $Z = \mathbb{R}^n$ and that all n objectives are to be minimized. In the absence of uncertainty, each $\mathbf{x} \in X$ is assigned exactly one objective vector $\mathbf{z} \in Z$ on the basis of a vector function $f : X \to Z$ with $\mathbf{z} = f(\mathbf{x})$. The mapping f defines the evaluation of a solution $\mathbf{x} \in X$, and often one is interested in those solutions that are Pareto optimal with respect to f.[2] However, generating the entire set of Pareto-optimal solutions is usually infeasible, e.g., due to the complexity of the underlying problem or the large number of optima. Therefore in many applications, the overall goal is to identify a good approximation of the Pareto-optimal set.

Different notions of what a good Pareto set approximation is are possible, and the definition of approximation quality strongly depends on the decision maker and the optimization scenario. We here assume that the optimization goal is given in terms of a binary quality indicator I, as proposed in [24]. A binary quality indicator, cf. [26], is a function $I : \mathcal{M}(Z) \times \mathcal{M}(Z) \to \mathbb{R}$, where $\mathcal{M}(Y)$ stands for the set of all possible multisets over Y, that can be regarded as a continuous extension of the concept of Pareto dominance to multisets of objective vectors. The value $I(A, B)$ quantifies the difference in quality between $A, B \in \mathcal{M}(Z)$. Now, if R denotes the set of Pareto-optimal solutions (or any other reference set), then the overall optimization goal can be formulated as

$$\text{argmin}_{S \in \mathcal{M}(X)} I(f(S), f(R)) \tag{1}$$

with $f(Y) := \{f(\mathbf{x}) \,|\, \mathbf{x} \in Y\}$. Since R is fixed, I actually represents a unary function that assigns each Pareto set approximation a real number; the smaller the number, the more preferable is the approximation.

B. Handling Uncertainty

In the following, the above optimization model will be extended to take uncertainty into account; to this end, we will assume that X is finite which simplifies the following presentation. Later, we will discuss how to estimate and compute expected indicator values for uncertain environments.

As to uncertainty, the basic difference to the classical settings is that the vector function f does not represent a deterministic mapping from X to Z, but can be regarded as a randomized procedure: every time a solution $\mathbf{x} \in X$ is evaluated using f, it may be mapped to a different objective vector $\mathbf{z} \in Z$. The higher the degree of uncertainty, the larger the variance among the objective vectors resulting from multiple, independent evaluations of $\mathbf{x}$. Thus, with each solution $\mathbf{x}$ a random variable $\mathcal{F}(\mathbf{x})$ is associated the range of which is Z; the underlying probability distribution is usually unknown and may be different for other solutions.

Now, consider an arbitrary solution multiset $S = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\} \in \mathcal{M}(X)$. Based on the random variables

[1] Note that the problem considered here is different from the issue of robustness, where the goal is to find solutions that are robust regarding parameter perturbations.

[2] A solution $\mathbf{x} \in X$ is Pareto optimal if and only if there exists no $\mathbf{x}' \in X$ such that (i) $f(\mathbf{x}')$ is component-wise smaller than or equal to $f(\mathbf{x})$ and (ii) $f(\mathbf{x}') \neq f(\mathbf{x})$.

$\mathcal{F}(\mathbf{x}_i)$ associated with the elements $\mathbf{x}_i$ of S, a corresponding random variable $\mathcal{F}(S)$ is defined for S which takes values in $\mathcal{M}(Z)$; $P(\mathcal{F}(S) = A)$ denotes the probability that (i) all members of S are mapped to elements of $A \in \mathcal{M}(Z)$ and (ii) there is at least one $\mathbf{x} \in S$ per $\mathbf{z} \in A$ for which $\mathbf{z} = f(\mathbf{x})$. Using this notation, we can now reformulate the optimization goal for uncertain environments as

$$\text{argmin}_{S \in \mathcal{M}(X)} \; E(I(\mathcal{F}(S), \mathcal{F}(R))) \tag{2}$$

where R is an arbitrary reference set from $\mathcal{M}(X)$ and $E(\cdot)$ stands for the expected value.

Note that there is a fundamental difference to other approaches, cf. [15]: we do not assume that there is a 'true' objective vector per solution which is blurred by noise; instead, we consider the scenario that each solution is inherently associated with a probability distribution over the objective space.

C. Estimating the Expected Indicator Value

If the probability distributions are known in advance and identical for all solutions $\mathbf{x} \in X$, then the expected value for any indicator can be computed according to

$$\begin{aligned} &E(I(\mathcal{F}(S), \mathcal{F}(R))) \\ &\quad = \sum_{A,B \in \mathcal{M}(X)} P(\mathcal{F}(S) = A, \mathcal{F}(R) = B) \cdot I(A,B) \\ &\quad = \sum_{A,B \in \mathcal{M}(X)} P(\mathcal{F}(S) = A) \cdot P(\mathcal{F}(R) = B) \cdot I(A,B) \end{aligned} \tag{3}$$

since $\mathcal{F}(S)$ and $\mathcal{F}(R)$ are independent from each other.

However, in practice the underlying probability distributions are in general unknown, may vary for different solutions, and therefore can only be estimated by drawing samples. Let us assume that $\mathcal{S}(\mathbf{x}) \in \mathcal{M}(Z)$ represents a finite sample, i.e., a multiset of objective vectors, for solution $\mathbf{x}$. Now, the expected indicator value $E(I(\mathcal{F}(\{\mathbf{x}\}), \{\mathbf{z}_1^*, \ldots, \mathbf{z}_q^*\})$ of $\mathcal{F}(\mathbf{x})$ with respect to a given set of objective vectors $\{\mathbf{z}_1^*, \ldots, \mathbf{z}_q^*\}$ can be estimated as follows

$$\begin{aligned} &\hat{E}(I(\mathcal{F}(\{\mathbf{x}\}), \{\mathbf{z}_1^*, \ldots, \mathbf{z}_q^*\})) \\ &\quad = \sum_{\mathbf{z} \in \mathcal{S}(\mathbf{x})} I(\{\mathbf{z}\}, \{\mathbf{z}_1^*, \ldots, \mathbf{z}_q^*\}) / |\mathcal{S}(\mathbf{x})| \end{aligned} \tag{4}$$

where $\hat{E}$ stands for the estimated expected value and $|\cdot|$ for the cardinality of a set. For a multiset S of solutions with $S = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$, the formula is

$$\begin{aligned} &\hat{E}(I(\mathcal{F}(S), \{\mathbf{z}_1^*, \ldots, \mathbf{z}_q^*\}) \\ &\quad = \sum_{\mathbf{z}_1 \in \mathcal{S}(\mathbf{x}_1)} \cdots \sum_{\mathbf{z}_m \in \mathcal{S}(\mathbf{x}_m)} \frac{I(\{\mathbf{z}_1, \ldots, \mathbf{z}_m\}, \{\mathbf{z}_1^*, \ldots, \mathbf{z}_q^*\})}{\prod_{1 \le i \le m} |\mathcal{S}(\mathbf{x}_i)|} \end{aligned} \tag{5}$$

and if one considers a reference set R of solutions with $R = \{\mathbf{x}_1^*, \ldots, \mathbf{x}_r^*\}$, then the estimate amounts to

$$\begin{aligned} &\hat{E}(I(\mathcal{F}(S), \mathcal{F}(R)) \\ &\quad = \sum_{\mathbf{z}_1^* \in \mathcal{S}(\mathbf{x}_1^*)} \cdots \sum_{\mathbf{z}_r^* \in \mathcal{S}(\mathbf{x}_r^*)} \frac{\hat{E}(I(\mathcal{F}(S), \{\mathbf{z}_1^*, \ldots, \mathbf{z}_r^*\}))}{\prod_{1 \le i \le r} |\mathcal{S}(\mathbf{x}_i^*)|} \end{aligned} \tag{6}$$

With this approach, the probability of a solution $\mathbf{x} \in X$ to be mapped to any objective vector $\mathbf{z}$ is estimated using the relative frequency of $\mathbf{z}$ with respect to $\mathcal{S}(\mathbf{x})$, i.e., $\hat{P}(\mathcal{F}(\{\mathbf{x}\}) = \{\mathbf{z}\}) = \sum_{\mathbf{z}' \in \mathcal{S}(\mathbf{x}),\, \mathbf{z}' = \mathbf{z}} 1/|\mathcal{S}(\mathbf{x})|$.

D. Computing Estimates for Expected Indicator Values

Computing the estimated expected indicator value for two multisets of solutions in the aforementioned manner is usually infeasible due to combinatorial explosion. Suppose each multiset contains 100 solutions with a sample size of 10 each, then equation 6 contains $100^{10} \cdot 100^{10} = 10^{40}$ summands. However, if particular properties of the indicator used can be exploited, then the exact calculation for $\hat{E}(\ldots)$ may become feasible. We here propose an algorithm for the (additive) ϵ-indicator [26] to compute an estimate for the expected quality difference between a multiset $S \in \mathcal{M}(X)$ and a reference set R with one element only - for reference sets of arbitrary size the computation is still too expensive to be useful in practice. Later in Section III it will be discussed how this procedure can be integrated into an evolutionary algorithm.

For a minimization problem, the additive ϵ-indicator $I_{\epsilon+}$ is defined as follows:

$$\begin{aligned} I_{\epsilon+}(A,B) = \inf\{\epsilon \in \mathbb{R} \mid \forall \mathbf{z}_2 = (z_{2_1}, \ldots, z_{2_n}) \in B :& \\ \exists \mathbf{z}_1 = (z_{1_1}, \ldots, z_{1_n}) \in A : \forall 1 \le i \le n : z_{1_i} \le \epsilon + z_{2_i}\}& \end{aligned} \tag{7}$$

It gives the minimum ϵ-value by which B can be moved in the objective space such that A is at least as good as B; a negative value implies that A is better than B in the Pareto sense. If B consists of a single objective vector $\mathbf{z}^*$, then the formula reduces to

$$\begin{aligned} I_{\epsilon+}(A, \{\mathbf{z}^*\}) = \inf\{\epsilon \in \mathbb{R} \mid \exists \mathbf{z}_1 = (z_{1_1}, \ldots, z_{1_n}) \in A :& \\ \forall 1 \le i \le n : z_{1_i} \le \epsilon + z_i^*\}& \end{aligned} \tag{8}$$

Now, to compute $\hat{E}(I_{\epsilon+}(\mathcal{F}(S), \{\mathbf{z}^*\}))$ it is not necessary to consider all combinations of objective vectors to which the elements $\mathbf{x} \in S$ could be mapped to. Instead, one can exploit the fact that always the minimum $I_{\epsilon+}(\{\mathbf{x}\}, \{\mathbf{z}^*\}))$-value determines the actual indicator value. By sorting the objective vectors beforehand, it suffices to consider the ϵ-values in increasing order.

In detail, this works as follows. We consider all pairs $(\mathbf{x}_j, \mathbf{z}_k)$, where $\mathbf{x}_j \in S$ and $\mathbf{z}_k \in \mathcal{S}(\mathbf{x}_j)$, and sort them in increasing order regarding the indicator value $I_{\epsilon+}(\{\mathbf{z}_k\}\{\mathbf{z}^*\})$. Suppose the resulting order is $(\mathbf{x}_{j_1}, \mathbf{z}_{k_1})$, $(\mathbf{x}_{j_2}, \mathbf{z}_{k_2}), \ldots,$

Algorithm 1 (Estimation of the Expected $I_{\epsilon+}$-Value)

```
Input:   $S \in \mathcal{M}(X)$          (multiset of decision vectors)
         $\mathbf{z}^* \in Z$            (reference objective vector)
Output:  $\hat{E}(I_{\epsilon+}(\mathcal{F}(S), \{\mathbf{z}^*\}))$ (estimate for the expect. value of $I_{\epsilon+}$)
Step 1:  Determine $\bar{\epsilon} = \min_{\mathbf{x} \in S} \max_{\mathbf{z} \in \mathcal{S}(\mathbf{x})} I_{\epsilon+}(\{\mathbf{z}\}, \{\mathbf{z}^*\})$
Step 2:  Set $L = \emptyset$. For each $\mathbf{x} \in S$ and $\mathbf{z} \in \mathcal{S}(x)$ do:
             1. $\epsilon = I_{\epsilon+}(\{\mathbf{z}\}, \{\mathbf{z}^*\})$.
             2. If $\epsilon \leq \bar{\epsilon}$ then append $(\epsilon, \mathbf{x})$ to list L.
Step 3:  Sort L in increasing order according to the $\epsilon$-values.
Step 4:  Set $\hat{E} = 0$. For each $\mathbf{x} \in S$ set $N[\mathbf{x}] = 0$.
Step 5:  While L is not empty do:
             1. $(\epsilon', \mathbf{x}')$ = first element of L.
             2. $p = 1/(|\mathcal{S}(\mathbf{x}')| - N[\mathbf{x}']) \cdot \prod_{\mathbf{x} \in S} 1 - N[\mathbf{x}]/|\mathcal{S}(\mathbf{x})|$.
             3. $\hat{E} = \hat{E} + p \cdot \epsilon'$.
             4. $N[\mathbf{x}'] = N[\mathbf{x}'] + 1$.
             5. Remove first element of L from L.
Step 6:  Return $\hat{E}$.
```

$(\mathbf{x}_{j_l}, \mathbf{z}_{k_l})$. Then, the estimate of the expected indicator value is

$$\begin{aligned}
&\hat{E}(I_{\epsilon+}(\mathcal{F}(S), \{\mathbf{z}^*\})) = \\
&\quad I_{\epsilon+}(\{\mathbf{z}_{k_1}\}, \{\mathbf{z}^*\}) \cdot \hat{P}(\mathcal{F}(\{\mathbf{x}_{j_1}\}) = \{\mathbf{z}_{k_1}\}) + \\
&\quad I_{\epsilon+}(\{\mathbf{z}_{k_2}\}, \{\mathbf{z}^*\}) \cdot \\
&\qquad \hat{P}(\mathcal{F}(\{\mathbf{x}_{j_2}\}) = \{\mathbf{z}_{k_2}\} \,|\, \mathcal{F}(\{\mathbf{x}_{j_1}\}) \neq \{\mathbf{z}_{k_1}\}) \} + \\
&\quad \ldots \\
&\quad I_{\epsilon+}(\{\mathbf{z}_{k_l}\}, \{\mathbf{z}^*\}) \cdot \\
&\qquad \hat{P}(\mathcal{F}(\{\mathbf{x}_{j_l}\}) = \{\mathbf{z}_{k_l}\} \,|\, \forall_{1 \leq i < l} \mathcal{F}(\{\mathbf{x}_{j_i}\}) \neq \{\mathbf{z}_{k_i}\})
\end{aligned} \tag{9}$$

Note that not necessarily all l summands need to be computed and thereby the computation time can be reduced. As soon as for one solution $\mathbf{x}$ all objective vectors $\mathbf{z} \in \mathcal{S}(\mathbf{x})$ have been considered, the remaining summands will yield 0 since $\hat{P}(\forall_{\mathbf{z} \in \mathcal{S}(\mathbf{x})} \mathcal{F}(\mathbf{x}) \neq \mathbf{z}) = 0$, i.e., at least one objective vector $\mathbf{z}$ in the sample $\mathcal{S}(\mathbf{x})$ must occur. This scheme is detailed in Alg. 1.

III. Algorithm Design

24) proposed a general scheme of an evolutionary algorithm to perform a multiobjective search with respect to an arbitrary quality indicator. Based on this work, we will present and discuss different implementations for integrating Alg. 1 into the optimization process in order to achieve the overall goal as defined in Equation 2.

Algorithm 2 (Baseline Algorithm: Steady-State-$IBEA$)

```
Input:   $N$    (population size)
         $G$    (maximum number of generations)
Output:  $S$    (approximation set)
Step 1:  Initialization: Generate an initial population $S$ of size $N$; set the
         generation counter $g$ to 0.
Step 2:  Fitness assignment: Calculate fitness values of individuals in $S$,
         i.e., $Fit(\mathbf{x}) = \sum_{\mathbf{z} \in \mathcal{S}(\mathbf{x})} \hat{E}(I(\mathcal{F}(S \setminus \{\mathbf{x}\}), \{\mathbf{z}\}))/|\mathcal{S}(\mathbf{x})|$ for all
         $\mathbf{x} \in S$.
Step 3:  Environmental selection: If $|S| > N$, then remove the individual
         $\mathbf{x}^* \in S$ with the smallest fitness value, i.e., $Fit(\mathbf{x}^*) \leq Fit(\mathbf{x})$ for
         all $\mathbf{x} \in S$, from $S$.
Step 4:  Termination: If $g \geq G$ then return $S$.
Step 5:  Mating selection: Perform binary tournament selection on $S$ to
         select two parents.
Step 6:  Variation: Apply recombination and mutation operators to the se-
         lected parents and insert the generated individual into the popula-
         tion $S$. Increment the generation counter ($g = g + 1$) and go to
         the Step 2.
```

The baseline algorithm that will be considered in the following is outlined in Alg. 2. It represents a steady-state evolutionary algorithm where at each generation a single offspring is produced and added to the population; the environmental selection step removes the worst individual from the population. The fitness of an individual $\mathbf{x}$ is defined as the estimated expected loss in quality, if $\mathbf{x}$ would be removed from the population S, i.e.,

$$\begin{aligned}
Fit(\mathbf{x}) &= \hat{E}(I(\mathcal{F}(S \setminus \{\mathbf{x}\}), \mathcal{F}(S)) \\
&= \hat{E}(I(\mathcal{F}(S \setminus \{\mathbf{x}\}), \mathcal{F}(\{\mathbf{x}\})) \\
&= \frac{1}{|\mathcal{S}(\mathbf{x})|} \sum_{\mathbf{z} \in \mathcal{S}(\mathbf{x})} \hat{E}(I(\mathcal{F}(S \setminus \{\mathbf{x}\}), \{\mathbf{z}\}))
\end{aligned} \tag{10}$$

Regarding Equation 2, we here assume that no external reference set R is given; instead, R is the current population plus the newly generated child.

The fitness assignment step in Alg. 2 can be realized on the basis of Alg. 1, and we will denote the resulting algorithm as EIV (exact computation of the estimated expected indicator value). In this case, the running time complexity of Step 2 is of order $O(n(Ns)^2 \log(Ns))$ which can become computationally expensive in practice. To yield a faster computation of the fitness values, we suggest two variants of Alg. 1, namely BCK and Exp, which approximate the ranking of the individuals induced by the exact $\hat{E}(\cdot)$ values.

The first variant (BCK) uses bucket sort [4] for the sorting step in Alg. 1. The range of indicator values $[I_{min}, I_{max}]$,which can be either determined in advance or adaptively during the run, is partitioned into c buckets of equal size where c is user-defined and controls the accuracy and the run-time of the sorting procedure. The elements of L (see Alg. 1) are placed in the corresponding buckets, and a sorting is generated by reading out the buckets from lower to greater values. Here, the running-time complexity of the fitness assignment is bounded by $O((n + c)(Ns)^2)$. If c is chosen large enough and no elements are placed in the same bucket, then the correct sorting is computed, but also the running-time can be high. If c is small, then the running-time is low as it is dominated by the term $(Ns)^2$, but potentially all elements are in the same bucket and the resulting sorting can be arbitrary.

The second variant (Exp) does not make use of Alg. 1, but directly computes the fitness values in the following manner:

$$\mathit{Fit}(\mathbf{x}) = \sum_{\mathbf{z}\in\mathcal{S}(\mathbf{x})} \sum_{\mathbf{x}'\in S\setminus\{\mathbf{x}\}} \sum_{\mathbf{z}'\in\mathcal{S}(\mathbf{x}')} -e^{-I_{\epsilon^+}(\{\mathbf{z}'\},\{\mathbf{z}\})/\kappa} \quad (11)$$

This scheme represents an extension of the one proposed in [24] and transforms the indicator values using an exponential function. With one evaluation per solution ($s = 1$) and κ sufficiently close to 0, the corresponding fitness ranking tends to be identical to the fitness ranking resulting from applying Alg. 1; however, numerical precision issues strongly affect the outcome. The running-time complexity of this fitness assignment method is of order $O(n(Ns)^2)$.

To evaluate the three different schemes proposed above, we also consider two alternative fitness assignment schemes:

- The first approach, the averaging method (Avg), follows the idea presented in [7] and applies it to Alg. 1. In principle, the average value is computed for each objective function and individual, and then Alg. 1 is executed where the sample for each individual consist of only one objective vector, namely the one representing the averages. However, the computation can simplified using the following formula:

$$\begin{array}{rcl} \mathit{Fit}(\mathbf{x}) & = & \hat{E}(I_{\epsilon^+}(\mathcal{F}(S\setminus\{\mathbf{x}\}),\{\mathbf{z}^*\}) \\ & = & \min_{\mathbf{x}'\in S} I_{\epsilon^+}(\{f(\mathbf{x}')\},\{\mathbf{z}^*\}) \end{array} \quad (12)$$

  The running-time complexity is $O(Ns+N^2)$ (averaging step + indicator values computation).

- The second approach (PDR) is the fitness assignment method proposed by 14) which is based on the probabilistic dominance relation between solutions:

$$\mathit{Fit}(\mathbf{x}) = \sum_{\mathbf{z}\in\mathcal{S}(\mathbf{x})} \sum_{i=1}^{n} \sum_{\mathbf{x}'\in S\setminus\{\mathbf{x}\}} \sum_{\mathbf{z}'\in\mathcal{S}(\mathbf{x}')} \frac{h(z'_i, z_i)}{|\mathcal{S}(\mathbf{x}')|\cdot|\mathcal{S}(\mathbf{x})|} \quad (13)$$

  with $h(z'_i, z_i)$ equal to 0 (resp. 0.5, 1) if the ith objective value of $\mathbf{z}'$ is smaller (resp. equal, greater) than the ith objective value of $\mathbf{z}$. The running-time complexity of the PDR fitness assignment algorithm is of order $O(n(Ns)^2)$.

IV. Simulation Results

In the following, we investigate two questions concerning performance of the five different algorithms. First, we evaluate the ability of these algorithms to select the best solutions with respect to a single environmental selection step. To achieve this, we compute the average loss in quality, in terms of the ϵ-indicator value, obtained by deleting one individual of randomly created populations of size $N+1$. Secondly, we compare the outcomes of entire optimization runs on various multiobjective tests functions.

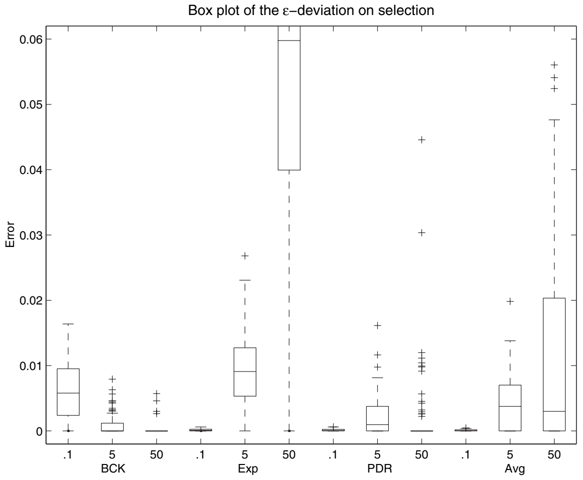

Figure. 1: Average selection error, with different levels of uncertainty.

A. Environmental Selection

The tests are performed with the different exact (EIV) and approximative (EIV, BCK, Exp, Avg, PDR) selection methods described above. We evaluate the selection process on randomly generated populations. The individuals generated are incomparable (using the dominance notion) in respect to their non-uncertain objective values. Each individual is created in the following way:

- Generate a (virtual) decision vector $\mathbf{x}$ with random value for each objective function.
- Scale $\mathbf{x}$ objective values to obtain $\mathbf{z}$ with $\sum_{i=1}^{n} f_i(\mathbf{x}) = n/2$. In the biobjective case, the resulting vector $\mathbf{z}$ is on the diagonal $[(1,0),(0,1)]$.
- Generate s different evaluations $[\mathbf{z}_1 \ldots \mathbf{z}_s]$ by adding uncertainty, i.e., a random value in the interval $[-\sigma,\sigma]$), for each objective function.

For each test, we consider 100 random populations of 100 individuals, with 10 biobjective evaluations per solution. Uniform noise defined on the interval $[-0.05, 0.05]$ is applied within each evaluation, and the number of objective functions, the sample size, and the level of uncertainty are systematically varied. The bucket sort approach was tested with $c = 50$.

We first evaluate the selection process for the exact I_{ϵ^+} value computation (EIV algorithm), which exactly determines the worst solution $\mathbf{x}_w$. Then, for each approximative fitness assignment algorithm i, we compute the worst solution $\mathbf{x}_{w_i}$. To evaluate the effectiveness of the approximation, we compute the difference, in terms of quality indicator I_{ϵ^+}, between the exact and the approximative approach: $I_{\epsilon^+}(S\setminus\{\mathbf{x}_{w_i}\}, S) - I_{\epsilon^+}(S\setminus\{\mathbf{x}_w\}, S)$ (smaller values are better).

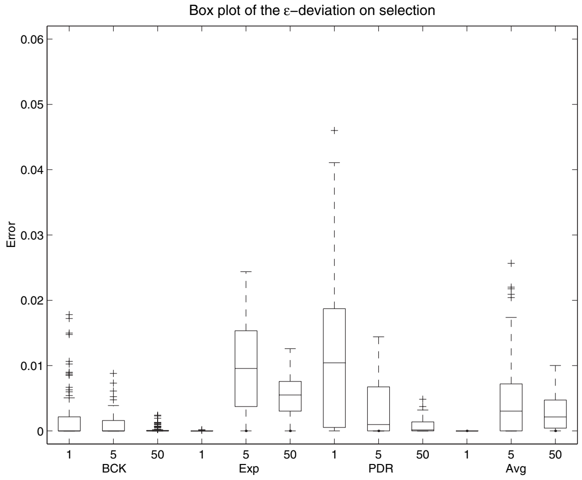

Figure. 2: Average selection error, with different number of evaluations.

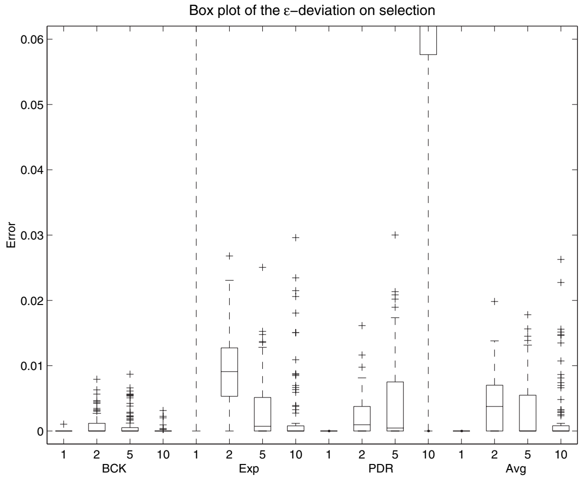

Figure. 3: Average selection error, with different number of objective functions.

The Figs. 1, 2 and 3 give the results obtained for different number of objective functions (1, 2, 5 and 10), sample size (1, 5 and 50), and levels of uncertainty (.1%, 5% and 50%). The smaller values are achieved by BCK in many cases, especially with an important level of uncertainty, many objectives, or a lot of evaluations. The other methods obtain small values only for small sample size or small level of uncertainty. When $s = 1$, the exponential approaches almost obtain the exact approach results, but only in this case. If the uncertainty considered is small, all approaches obtain the close-to-optimal results, except of BCK. In the non-uncertain case, we suggest to use the exponential function approach, which is not expensive to compute and almost gives the optimal results. With uncertainty, BCK seems to be more effective. The computational complexity of BCK mainly depends on its main parameter, i.e., the number of cells, which was set to

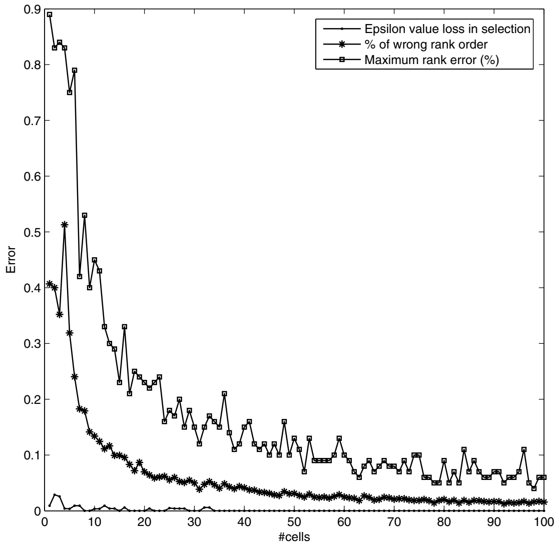

Figure. 4: Selection error in function of C value: comparison against EIV fitness assignment; (1) quality loss during ES(N+1) selection process; (2) rate of misorder between EIV and BCK ranking; (3) maximum ranking difference between EIV and BCK.

50 in our experiments. In Fig. 4, the error obtained by BCK with different values for the number of cells defined is shown. With more than 20 cells, the quality of the results appears to be sufficient. The number of cells has to be chosen carefully for a good trade-off between computational complexity and quality of approximation.

B. Entire Optimization Runs

For the entire optimization runs, we consider 6 multi-objective test functions taken from the literature: ZDT1, ZDT6 [23], DTLZ2 [8], KUR [19], COMET [9] and QV [20]. The number of decision variables has been fixed to 50 for all the test problems. Tests are realized by considering two different types of uncertainty:

- on the objective vector $f(\mathbf{x})$: an uniform-distributed random uncertainty is directly applied on the evaluation functions, in a fixed interval $[-\sigma, \sigma]$. We set $\sigma = 0.1$ for all problems, excepted for COMET ($\sigma = 1$) and KUR ($\sigma = 0.5$) problems.
- on the decision variables: a uniformly distributed uncertainty distribution is applied concerning the decision variables. The result is a variable uncertainty, depending on the form of the objective space around the envisaged solution. We set $\sigma = 0.1$ for all decision variables for every problem.

The population size N is set to 50, with $s = 5$ evaluations for each solution. An uniformly distributed uncertainty is applied on the solutions (on decision variables $\mathbf{x}$ or on objective vectors $\mathbf{z}$). The maximum number of generations is set to 5000. We perform 30 runs for each problem. The different methods are tested with the same initial populations. The 50 decision variables are real numbers defined over the interval $[0, 1]$. The SBX-20 operator is used for recombination and a polynomial distribution for mutation [5]; the mutation and recombination probabilities are set to 1 and 0.01, respectively, according to [10]; the other parameters are those used in [24].

1) Performance assessment: evaluation with the 'true' vector function

To evaluate the effectiveness of each method, we generate the 'true' objective vector for each solution. Then, for each approximation A, we compute $\hat{E}(I_{\epsilon^+}(A, R))$ value, where R is the reference set, determined by merging all solutions found during the runs and keeping only the non-dominated solutions. The comparison of the whole set of runs is realized using the Mann-Whitney statistical test [3], applied on the sets of $\hat{E}(I_{\epsilon^+}(A, R))$ values computed for each method.

Table 1 represents the comparison of the different selection methods for the $\hat{E}(I_{\epsilon^+})$ value, with two different types of uncertainty: on objective vectors and on decision variables. In order to compare the outcomes of the runs, we use the Mann-Whitney statistical test, as described in [17]. The columns give the adjusted P value of the corresponding pairwise test that accounts for multiple testings; it equals to the lowest significance level for which the null-hypothesis (the medians are drawn from the same distribution) would still be rejected (with a significance level of 5%). A value under 5% shows that the method in the corresponding row is significantly better than the method in the corresponding column. If the significance level of *method A better than method B* is greater than 5%, then no conclusion can be drawn ($A \equiv B$). u_1 (resp. u_2) column corresponds to the results with uncertainty applied on the objective space (resp. decision variables).

In many cases, the results are not significant in the bi-objective case, since the different approaches achieve similar results. Further observations from Table 1 are:

- The exponential approximation approach Exp, give the worst results in many cases, except for KUR and $COMET$ instances.
- BCK and EIV obtain similar results, which shows the efficiency of BCK to approximate EIV fitness assignment method.
- Uncertainty on objective vectors: in many cases, ϵ-indicator-based approaches Avg, BCK and EIV perform significantly better than Hughes selection mechanism PDR, especially for $COMET$ and $ZDT1$ instances—with respect to the ϵ-indicator.
- Uncertainty on decision variables: in many cases, Avg results are significantly worse than EIV, BCK and PDR (problems $DTLZ2$, $ZDT6$ and KUR).

As a conclusion, the ϵ-indicator-based fitness assignment EIV performs well, in a general case, for the continuous problems which have uncertainty on objective functions or decision variables, and the approximative algorithm BCK obtain similar results. Table 2 shows the results on the same problems, with higher uncertainty level (5 times higher). The main observations are:

- EIV and BCK obtain similar results.
- Except for QV with uncertainty on the decision space, the average results of EIV and BCK outperform the other approaches.
- As for the previous tests, Avg obtains good approximations with uncertainty in the objective space, and PDR obtain good approximations with uncertainty in the decision space.
- Exp approach obtain acceptable results with uncertainty in the decision space.

In the table 3, we represent the results obtained for experiments performed on the $DTLZ2$ test function, with different number of objective functions. This table shows a superior performance of the EIV and BCK fitness assignment methods when the number of objective functions is growing. Exp is inferior to the other approaches in many cases, but gives satisfactory results with many objective functions to optimize.

We make the same test series on another multiobjective test function: $DTLZ6$. With uncertainty on the objective vectors, PDR is worse than the other methods in many cases, with a very small significance level. However, no global conclusion can made about the comparison of the other approaches. With uncertainty on decision variables, Exp, BCK and EIV approaches obtain results significantly better than PDR and Avg. Moreover, Exp is the best approach on this problem.

In our first experiments, we consider two type of uncertainty, u_1 (uncertainty on decision variables) and u_2 (uncertainty on objective functions), which are uniformly distributed and a central tendency equal to 0. Some experiments have been realized also with different types of uncertainty, on $DTLZ2$ test problem. The uncertainties considered are defined as follow:

- Normal distribution of the uncertainty on the objective vectors, with a central tendency equal to zero, and a scale value defined as σ in the first series of experiments. Let p_1 and p_2 be two random numbers (uniformly distributed); we approximate the normal distribution of the uncertainty u_1 as follow (applied on every objective value $\mathbf{z}_i$):

$$u_3(\mathbf{z}_i) = \sigma * \sqrt{-2 * log(p_1) * cos(2\pi p_2)}$$

Table 1: Comparison of the different selection methods for the I_{ϵ^+}-indicator using the Mann-Whitley statistical test: P-value, with uncertainty on objective vectors (u_1) and on decision vectors (u_2).

		EIV		**BCK**		**Exp**		**Avg**		**PDR**	
		u_1	u_2	u_1	u_2	u_1	u_2	u_1	u_2	u_1	u_2
DTLZ2	**EIV**			$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$6.35 * 10^{-11}$	$\equiv$	$5.17 * 10^{-6}$	$\equiv$	$\equiv$
	BCK	$4.38 * 10^{-4}$	$\equiv$			$5.74 * 10^{-11}$	$9.47 * 10^{-11}$	$6.84 * 10^{-3}$	$2.81 * 10^{-3}$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$3.38 * 10^{-10}$			$\equiv$	$\equiv$
	PDR	$7.31 * 10^{-9}$	$4.38 * 10^{-4}$	$5.03 * 10^{-5}$	$3.08 * 10^{-4}$	$5.74 * 10^{-11}$	$9.47 * 10^{-11}$	$3.86 * 10^{-8}$	$7.60 * 10^{-8}$		
ZDT1	**EIV**			$\equiv$	$\equiv$	$1.90 * 10^{-4}$	$3.32 * 10^{-7}$	$\equiv$	$\equiv$	$2.11 * 10^{-5}$	$\equiv$
	BCK	$\equiv$	$\equiv$			$1.13 * 10^{-5}$	$1.14 * 10^{-8}$	$\equiv$	$\equiv$	$9.98 * 10^{-7}$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$3.86 * 10^{-5}$	$1.06 * 10^{-7}$			$4.47 * 10^{-6}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$\equiv$	$\equiv$		
ZDT6	**EIV**			$\equiv$	$\equiv$	$5.75 * 10^{-11}$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$
	BCK	$\equiv$	$\equiv$			$5.75 * 10^{-11}$	$\equiv$	$\equiv$	$4.24 * 10^{-3}$	$\equiv$	$\equiv$
	Exp	$\equiv$	$8.08 * 10^{-5}$	$\equiv$	$1.15 * 10^{-3}$			$\equiv$	$1.15 * 10^{-7}$	$\equiv$	$1.50 * 10^{-2}$
	Avg	$1.49 * 10^{-2}$	$\equiv$	$4.64 * 10^{-4}$	$\equiv$	$5.75 * 10^{-11}$	$\equiv$			$\equiv$	$\equiv$
	PDR	$1.40 * 10^{-3}$	$\equiv$	$2.26 * 10^{-5}$	$\equiv$	$5.75 * 10^{-11}$	$\equiv$	$\equiv$	$1.65 * 10^{-3}$		
KUR	**EIV**			$\equiv$	$\equiv$	$\equiv$	$5.84 * 10^{-4}$	$\equiv$	$4.91 * 10^{-4}$	$\equiv$	$\equiv$
	BCK	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$4.92 * 10^{-2}$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$3.31 * 10^{-2}$	$\equiv$	$8.91 * 10^{-3}$		
COMET	**EIV**			$7.52 * 10^{-3}$	$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$\equiv$	$\equiv$	$9.17 * 10^{-4}$	$5.74 * 10^{-11}$
	BCK	$\equiv$	$\equiv$			$\equiv$	$8.57 * 10^{-11}$	$\equiv$	$\equiv$	$\equiv$	$5.74 * 10^{-11}$
	Exp	$\equiv$	$\equiv$	$3.60 * 10^{-2}$	$\equiv$			$\equiv$	$\equiv$	$1.76 * 10^{-3}$	$1.55 * 10^{-10}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$9.47 * 10^{-11}$			$2.60 * 10^{-2}$	$5.74 * 10^{-11}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
QV	**EIV**			$\equiv$	$\equiv$	$1.39 * 10^{-5}$	$\equiv$	$\equiv$	$\equiv$	$3.90 * 10^{-2}$	$\equiv$
	BCK	$\equiv$	$\equiv$			$6.11 * 10^{-5}$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$1.67 * 10^{-3}$	$\equiv$			$\equiv$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$3.06 * 10^{-2}$	$\equiv$	$\equiv$	$\equiv$		

Table 2: Comparison of the different selection methods for the I_{ϵ^+}-indicator using the Mann-Whitley statistical test: P-value, with uncertainty on objective vectors (u_1) and on decision vectors (u_2) - 2 objective problems - great uncertainty (5*$\{u_1$,$u_2\}$).

		EIV		**BCK**		**Exp**		**Avg**		**PDR**	
		$5u_1$	$5u_2$	$5u_1$	$5u_2$	$5u_1$	$5u_2$	$5u_1$	$5u_2$	$5u_1$	$5u_2$
DTLZ2	**EIV**			$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$\equiv$	$\equiv$	$2.29 * 10^{-2}$	$\equiv$	$\equiv$
	BCK	$\equiv$	$\equiv$			$5.74 * 10^{-11}$	$\equiv$	$\equiv$	$1.09 * 10^{-2}$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$1.46 * 10^{-6}$	$\equiv$	$2.77 * 10^{-5}$	$\equiv$	$5.74 * 10^{-11}$	$\equiv$			$2.83 * 10^{-7}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$\equiv$	$\equiv$	$4.93 * 10^{-2}$		
ZDT1	**EIV**			$\equiv$	$\equiv$	$1.45 * 10^{-8}$	$2.51 * 10^{-8}$	$1.58 * 10^{-3}$	$\equiv$	$\equiv$	$8.67 * 10^{-4}$
	BCK	$\equiv$	$\equiv$			$4.26 * 10^{-9}$	$4.26 * 10^{-9}$	$\equiv$	$9.69 * 10^{-4}$	$\equiv$	$1.50 * 10^{-3}$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$1.84 * 10^{-6}$	$3.73 * 10^{-10}$			$\equiv$	$3.86 * 10^{-6}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$1.25 * 10^{-7}$	$7.76 * 10^{-4}$	$\equiv$	$\equiv$		
ZDT6	**EIV**			$\equiv$	$\equiv$	$1.72 * 10^{-5}$	$\equiv$	$\equiv$	$\equiv$	$9.94 * 10^{-3}$	$\equiv$
	BCK	$\equiv$	$\equiv$			$8.20 * 10^{-4}$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$9.16 * 10^{-6}$	$\equiv$			$7.17 * 10^{-3}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
KUR	**EIV**			$2.20 * 10^{-2}$	$\equiv$	$1.27 * 10^{-3}$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$
	BCK	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$1.49 * 10^{-2}$	$\equiv$	$\equiv$	$\equiv$		
COMET	**EIV**			$\equiv$	$\equiv$	$3.90 * 10^{-2}$	$1.06 * 10^{-7}$	$\equiv$	$2.60 * 10^{-5}$	$9.69 * 10^{-4}$	$\equiv$
	BCK	$\equiv$	$\equiv$			$1.21 * 10^{-3}$	$2.83 * 10^{-7}$	$\equiv$	$1.67 * 10^{-6}$	$4.00 * 10^{-3}$	$\equiv$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$3.81 * 10^{-3}$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$5.92 * 10^{-3}$	$\equiv$			$6.41 * 10^{-6}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
QV	**EIV**			$\equiv$	$\equiv$	$4.26 * 10^{-9}$	$\equiv$	$\equiv$	$\equiv$	$4.56 * 10^{-2}$	$\equiv$
	BCK	$\equiv$	$\equiv$			$1.77 * 10^{-8}$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Exp	$\equiv$	$5.21 * 10^{-4}$	$\equiv$	$3.90 * 10^{-2}$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$2.46 * 10^{-9}$	$\equiv$	$8.98 * 10^{-8}$	$9.37 * 10^{-8}$	$1.72 * 10^{-5}$			$\equiv$	$1.25 * 10^{-7}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$4.21 * 10^{-8}$	$\equiv$	$\equiv$	$\equiv$		

Table 3: Evaluation with several number of objectives to optimize: $DTLZ2$ and $DTLZ6$ test functions, using the Mann-Whitley statistical test: P-value, with uncertainty on objective vectors (u_1) and on decision vectors (u_2).

		EIV		**BCK**		**Exp**		**Avg**		**PDR**	
		u_1	u_2	u_1	u_2	u_1	u_2	u_1	u_2	u_1	u_2
n=3	**EIV**			$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$	$4.56 * 10^{-2}$	$9.06 * 10^{-3}$	$1.71 * 10^{-6}$	$\equiv$
DTLZ2	**BCK**	$\equiv$	$\equiv$			$5.74 * 10^{-11}$	$5.74 * 10^{-11}$	$\equiv$	$1.09 * 10^{-4}$	$1.49 * 10^{-5}$	$\equiv$
	Exp	$\equiv$		$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$			$1.02 * 10^{-3}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$1.15 * 10^{-10}$	$5.74 * 10^{-11}$	$\equiv$	$\equiv$		
n=3	**EIV**			$2.02 * 10^{-2}$	$\equiv$	$3.19 * 10^{-2}$	$\equiv$	$\equiv$	$\equiv$	$7.76 * 10^{-11}$	$5.74 * 10^{11}$
DTLZ6	**BCK**	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$3.99 * 10^{-2}$	$3.39 * 10^{-10}$	$7.76 * 10^{-11}$
	Exp	$\equiv$	$1.41 * 10^{-10}$	$\equiv$	$3.07 * 10^{-10}$			$\equiv$	$1.17 * 10^{-9}$	$7.02 * 10^{-11}$	$1.97 * 10^{-5}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$7.76 * 10^{-11}$	$9.47 * 10^{-11}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
n=4	**EIV**			$\equiv$	$\equiv$	$1.17 * 10^{-9}$	$6.02 * 10^{-10}$	$\equiv$	$2.20 * 10^{-2}$	$3.89 * 10^{-9}$	$6.52 * 10^{-5}$
DTLZ2	**BCK**	$\equiv$	$\equiv$			$6.69 * 10^{-9}$	$1.89 * 10^{-10}$	$2.50 * 10^{-2}$	$3.45 * 10^{-3}$	$1.62 * 10^{-8}$	$6.77 * 10^{-6}$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$1.62 * 10^{-8}$	$1.25 * 10^{-8}$			$1.25 * 10^{-7}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$8.55 * 10^{-7}$	$\equiv$	$\equiv$		
n=4	**EIV**			$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$3.26 * 10^{-4}$	$5.74 * 10^{-11}$	$\equiv$
DTLZ6	**BCK**	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$1.77 * 10^{-2}$	$5.74 * 10^{-11}$	$\equiv$
	Exp	$\equiv$	$1.89 * 10^{-10}$	$\equiv$	$1.47 * 10^{-10}$			$\equiv$	$7.02 * 10^{-11}$	$5.74 * 10^{-11}$	$2.81 * 10^{-3}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$5.74 * 10^{-11}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$4.64 * 10^{-3}$		
n=5	**EIV**			$\equiv$	$\equiv$	$4.58 * 10^{-8}$	$3.55 * 10^{-9}$	$6.18 * 10^{-2}$	$7.91 * 10^{-7}$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$
DTLZ2	**BCK**	$\equiv$	$\equiv$			$6.69 * 10^{-9}$	$1.49 * 10^{-9}$	$1.19 * 10^{-2}$	$3.90 * 10^{-7}$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$1.71 * 10^{-10}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$5.55 * 10^{-6}$	$1.14 * 10^{-2}$			$5.74 * 10^{-11}$	$7.73 * 10^{-11}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
n=5	**EIV**			$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$8.27 * 10^{-8}$	$1.05 * 10^{-10}$	$2.74 * 10^{-6}$
DTLZ6	**BCK**	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$8.55 * 10^{-7}$	$1.27 * 10^{-10}$	$3.39 * 10^{-5}$
	Exp	$\equiv$	$6.35 * 10^{-11}$	$\equiv$	$7.76 * 10^{-11}$			$\equiv$	$5.74 * 10^{-11}$	$6.35 * 10^{-11}$	$9.57 * 10^{-11}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$9.47 * 10^{-11}$	$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
n=6	**EIV**			$\equiv$	$\equiv$	$6.11 * 10^{-9}$	$2.24 * 10^{-9}$	$3.47 * 10^{-4}$	$7.60 * 10^{-8}$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$
DTLZ2	**BCK**	$\equiv$	$\equiv$			$1.60 * 10^{-7}$	$2.61 * 10^{-7}$	$9.49 * 10^{-3}$	$2.26 * 10^{-5}$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$
	Exp	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$5.74 * 10^{-11}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$4.13 * 10^{-4}$	$\equiv$			$5.74 * 10^{-11}$	$5.74 * 10^{-11}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		
n=6	**EIV**			$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$4.71 * 10^{-5}$	$7.02 * 10^{-11}$	$1.24 * 10^{-8}$
DTLZ6	**BCK**	$\equiv$	$\equiv$			$\equiv$	$\equiv$	$\equiv$	$2.02 * 10^{-4}$	$5.74 * 10^{-11}$	$1.49 * 10^{-8}$
	Exp	$\equiv$	$4.97 * 10^{-10}$	$\equiv$	$1.06 * 10^{-9}$			$\equiv$	$1.05 * 10^{-10}$	$6.35 * 10^{-11}$	$7.76 * 10^{-11}$
	Avg	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$			$5.74 * 10^{-11}$	$5.37 * 10^{-5}$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$	$\equiv$		

- Normal distribution of the uncertainty on the objective vectors, with a central tendency up to 0. The resulting uncertainty u_2 equals to (applied on every objective value $\mathbf{z}_i$):
$$u_4(\mathbf{z}_i) = abs(u_1)$$

- Noise on decision variables, with different scale for every decision variable. The uncertainty u_3 applied on the decision variables decrease with the rank of each decision variable $\mathbf{x}_i$:
$$u_5(\mathbf{x}_i) = \frac{\sigma p_1}{i}$$

The results are given in Table 4. With the two first defined uncertainties u_1 and u_2 (see Table 1), PDR is significantly better than the other approaches. If we consider the uncertainty u_3, PDR is significantly worse than EIV and Avg methods. For the uncertainty u_4, PDR is significantly worse than EIV, BCK and Avg methods. For the uncertainty u_5, there are no significant relation between the algorithms.

As mentioned at the beginning of the paper, we do not assume a true objective vector per solution, but a solution associated with an unknown probability distribution in the objective space. Then, each possible evaluation has to be considered as a possible event. In the next section, we describe a method to evaluate outputs without knowledge about the true objective vector function.

2) Probabilistic performance assessment

In the previous section, we presented experimental results obtained by comparing the set of individuals according to their 'true' evaluations, i. e. the result of the evaluation function without uncertainty. In the real case, we will consider that each 'uncertain' evaluation is a possible event. In this case, we have to take into account all the possible evaluations to evaluate a set of solutions, and evaluate the different possible sets according to the I_{ϵ^+}-indicator. In order to compute this performance, we suggest the following procedure:
(1) At the end of the optimization process, proceed to k uncertain evaluations for each solution of the approximation S proposed by the optimizer. In order to delete the influence of the optimization process, the k evaluations of every solution replace the evaluations realized during the optimization process.
(2) Define a reference set R, which corresponds to the set of non-dominated solutions extracted from all the evaluations done in the step 1.
(3) Compute the exact value for $\hat{E}(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$, represented by equation 5.

The exact computation of $\hat{E}(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$ can be time consuming. Therefore, we propose to compute $I(\mathcal{F}(S), \{r\})$ for all elements r in R, which is different than $\hat{E}(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$. In order to compute an approximation of this expected quality indicator, we propose to compute 3 different values:

- $I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R))$ value, in the best case:
$$\begin{aligned}&min(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R))) = \\ &max(min(min(I_{\epsilon^+}(\{\mathbf{x}_a\}, \{\mathbf{x}_r\}), \mathbf{x}_a \in \mathcal{S}(\mathbf{x})), \\ &\mathcal{S}(\mathbf{x}) \in S),\ \mathbf{x}_r \in R)\end{aligned} \tag{14}$$

- $I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R))$ value, in the worst case:
$$\begin{aligned}&max(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R))) = \\ &max(min(max(I_{\epsilon^+}(\{\mathbf{x}_a\}, \{\mathbf{x}_r\}), \mathbf{x}_a \in \mathcal{S}(\mathbf{x})), \\ &\mathcal{S}(\mathbf{x}) \in S),\ \mathbf{x}_r \in R)\end{aligned} \tag{15}$$

- $I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R))$ average value, for every $\mathbf{x}_r \in R$:
$$max(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R))) = \frac{1}{|R|} \sum_{\mathbf{x}_r \in R} \hat{E}(I_{\epsilon^+}(S, \{\mathbf{x}_R\})) \tag{16}$$

This type of evaluation could be useful, especially if the 'true' objective vector is either unknown or does not exist. In real-world application, there is usually not a true objective vector, and for such scenarios this type of evaluation can be used. In order to estimate the difference between these probabilistic quality indicators and evaluation made with knowledge of a 'true' objective vector function, we compute $min(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$, $max(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$ and $avg(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$ on several outputs. We apply these quality indicators to the results obtained on $DTLZ2$ with the 5 different type of uncertainty $u_1 \dots u_5$.

The results for $avg(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$ quality indicator are shown in Table 5. Let us remark that there are no significant difference between the results obtained by $avg(I_{\epsilon^+}(\mathcal{F}(S), \mathcal{F}(R)))$ and the two other probabilistic quality indicators. For the two first uncertainties, which have an uniform distribution, the results obtained do not show a great difference of performance between evaluating with or without knowledge of the true evaluation function 1.

Considering the three last uncertainties, different results are obtained:

- u_3: Avg method is no longer better than the BCK and EIV methods. PDR is significantly worse than EIV (not BCK).

- u_4: Avg and BCK methods are significantly worse than EIV and BCK.

- u_5: Avg is significantly worse than BCK.

V. Probabilistic empirical attainment function for uncertain sets of solutions

Without uncertainty, the *attainment function* provides a description of the distribution of an outcome set $\mathbf{x} = \{x_j \in \mathbb{R}^n,\ j = 1, \dots, M\}$ in a simple and elegant way, using the notion of goal-attainment [11]. This function allows the user

Table 4: Evaluation with different type of uncertainty: $DTLZ2$ test function, using the Mann-Whitley statistical test: P-value.

		EIV	**BCK**	**Exp**	**Avg**	**PDR**
u_3	**EIV**		$\equiv$	$7.10 * 10^{-11}$	$\equiv$	$2.73 * 10^{-2}$
	BCK	$\equiv$		$1.59 * 10^{-10}$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$6.93 * 10^{-4}$	$2.41 * 10^{-5}$	$6.51 * 10^{-11}$		$3.35 * 10^{-7}$
	PDR	$\equiv$	$\equiv$	$2.59 * 10^{-10}$	$\equiv$	
u_4	**EIV**		$\equiv$	$5.74 * 10^{-11}$	$\equiv$	$1.98 * 10^{-6}$
	BCK	$\equiv$		$5.74 * 10^{-11}$	$\equiv$	$1.39 * 10^{-5}$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$5.74 * 10^{-11}$		$2.90 * 10^{-4}$
	PDR	$\equiv$	$\equiv$	$6.35 * 10^{-11}$	$\equiv$	
u_5	**EIV**		$\equiv$	$\equiv$	$\equiv$	$\equiv$
	BCK	$\equiv$		$\equiv$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$		$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	

Table 5: Evaluation with different type of uncertainty: $DTLZ2$ test function, using the Mann-Whitley statistical test: P-value - Probabilistic performance assessment ($avg(I_{\epsilon+}(A,R))$).

		EIV	**BCK**	**Exp**	**Avg**	**PDR**
u_1	**EIV**		$\equiv$	$5.74 * 10^{-11}$	$\equiv$	$\equiv$
	BCK	$1.08 * 10^{-3}$		$5.74 * 10^{-11}$	$5.92 * 10^{-3}$	$\equiv$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$5.74 * 10^{-11}$		$\equiv$
	PDR	$2.46 * 10^{-9}$	$3.86 * 10^{-5}$	$5.74 * 10^{-11}$	$3.25 * 10^{-8}$	
u_2	**EIV**		$\equiv$	$7.76 * 10^{-11}$	$2.48 * 10^{-6}$	$\equiv$
	BCK	$\equiv$		$8.57 * 10^{-11}$	$1.25 * 10^{-7}$	$\equiv$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$8.00 * 10^{-9}$		$\equiv$
	PDR	$2.90 * 10^{-4}$	$3.81 * 10^{-3}$	$7.76 * 10^{-11}$	$1.70 * 10^{-9}$	
u_3	**EIV**		$\equiv$	$1.40 * 10^{-10}$	$\equiv$	$1.76 * 10^{-2}$
	BCK	$\equiv$		$3.89 * 10^{-10}$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$		$\equiv 3.14 * 10^{-10}$		$\equiv$
	PDR	$\equiv$	$\equiv$	$6.76 * 10^{-10}$	$\equiv$	
u_4	**EIV**		$\equiv$	$5.74 * 10^{-11}$	$1.27 * 10^{-3}$	$2.11 * 10^{-2}$
	BCK	$\equiv$		$5.74 * 10^{-11}$	$9.17 * 10^{-4}$	$1.25 * 10^{-2}$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$5.74 * 10^{-11}$		$\equiv$
	PDR	$\equiv$	$\equiv$	$5.74 * 10^{-11}$	$\equiv$	
u_5	**EIV**		$\equiv$	$\equiv$	$4.05 * 10^{-2}$	$\equiv$
	BCK	$\equiv$		$\equiv$	$\equiv$	$\equiv$
	Exp	$\equiv$	$\equiv$		$\equiv$	$\equiv$
	Avg	$\equiv$	$\equiv$	$\equiv$		$\equiv$
	PDR	$\equiv$	$\equiv$	$\equiv$	$\equiv$	

to obtain a visualization of the outputs and some analysis results, such as the probability to attain a goal, i.e., objective vector $\mathbf{z}$. It is defined by the function $EAF_{\mathbf{x}}(\mathbf{z}) : \mathbb{R} \rightarrow [0,1]$ with

$$EAF_{\mathbf{x}}(\mathbf{z}) = P(\mathbf{x} \preceq \mathbf{z})$$

In this section, we discuss on how to adapt the *Empirical Attainment Function* (EAF) to an uncertain context. Consider r sets of objective vectors $S_1 \ldots S_r \in \mathcal{M}(Z)^r$, resulting from r different executions of the same couple $\{optimizer, problem\}$. The empirical attainment function $EAF_S(\mathbf{z}^*)$ is defined as the probability that the optimizer considered obtains at least one objective vector which dominates the goal $\mathbf{z}^*$, in a single run. In detail, the EAF gives for each objective vector in the objective space the relative frequency that it was attained, i.e., weakly dominated by the generated approximation set, with respect to r runs. In this section, we propose to adapt the EAF to the context of uncertain optimization. First, we consider the EAF for a single run (without uncertainty, the resulting EAF function returns 0 and 1 values), then we present an algorithm to compute the EAF in uncertain environments. In the last part, we propose a corresponding approximative algorithm.

A. Compute the exact probabilistic attainment surface for "uncertain" approximations

We introduce some notations before describing the global algorithm for EAF computation in the presence of uncertainty.

- $dom(\mathbf{z}_1, \mathbf{z}_2)$: compare the objective vector $\mathbf{z}_1 \in Z$ against the objective vector $\mathbf{z}_2 \in Z$. Return 1 if $\mathbf{z}_1 \preceq \mathbf{z}_2$, 0 otherwise.

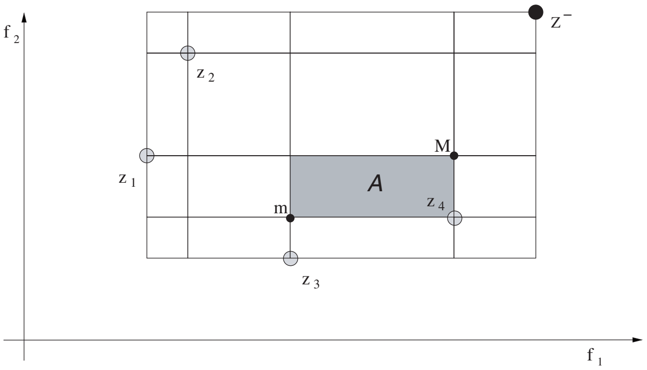

Figure. 5: There are no difference in terms of dominance relation between each point of the shaded area $\mathcal{A}$ and the set of circle solutions in the objective space (with $\mathcal{S}(\mathbf{x}) = \{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{z}_4\}$, $\forall \mathbf{z}^* \in \mathcal{A}, P(\mathcal{S}(\mathbf{x}) \preceq \mathbf{z}^*) = 0.25$ - only $\mathbf{z}_3$ dominates $\mathcal{A}$). In this example, $\mathcal{A}$ is defined by $m = (f_1(\mathbf{z}_3), f_2(\mathbf{z}_4))$ and $m = (f_1(\mathbf{z}_4), f_2(\mathbf{z}_1))$.

- $P(S \preceq \mathbf{z}^*)$: probability that a multiset $S \in \mathcal{M}(Z)$ dominates a reference point $\mathbf{z}^* \in Z$, computed as follow:

$$P(S \preceq \mathbf{z}^*) = \sum_{\mathbf{z} \in S} \frac{dom(\mathbf{z}, \mathbf{z}^*)}{|S|} \qquad (17)$$

- $P(S \preceq \mathbf{z}^*)$: probability that at least as one solution of the approximation set $\mathcal{F}(S) = \{\mathcal{S}(\mathbf{x}_1) \ldots \mathcal{S}(\mathbf{x}_t)\}$ of t multisets, dominates the reference point r, which corresponds to the formula:

$$P(\mathcal{F}(S) \preceq \mathbf{z}^*) = 1 - \prod_{S \in \mathcal{F}(S)} (1 - P(S \preceq \mathbf{z}^*)) \quad (18)$$

With the equation 18, we can compute EAF for one point of the objective space. In order to compute EAF for every point in the objective space, we need to sum the equation 18 result, for each point of the objective space. Considering a set of objective vector S and an hyper-volume $\mathcal{A}$ defined by the two points $m = [min_1 \ldots min_n]$ and $M = [max_1 \ldots max_n]$, with the relation: $\forall \mathbf{z} \in S, \nexists i \in [1 \ldots n]$, with $min_i < \mathbf{x}_i \leq max_i$, then the EAF function is constant for every point in $\mathcal{A}$ (see Fig. 5).

Then, we are able to compute the entire EAF function, with a finite number of dominance relation computation, as described in Alg. 3. The Fig. 6 shows an illustrated example with two objective functions, two solutions and 5 evaluations per solution.

$EAFs$ could be used for visualizing the outcomes of uncertain run of an optimizer. For instance, one may be interested in plotting all the goals that have been attained (independently) in 50% of the cases. This is defined in terms of a *probabilistic-k%-approximation set*:

> An approximation set S is called the *probabilistic-k%-approximation set* of an $EAF(S)$, iff it weakly dominates exactly those objective vectors that have been attained with a probability up to k per cent.

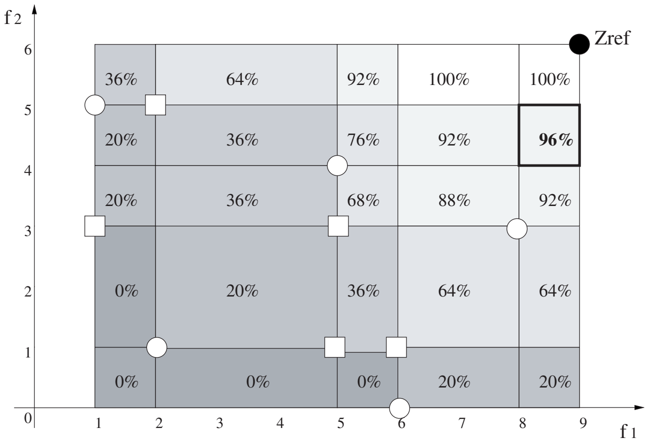

Figure. 6: Example with two objective functions and two solutions evaluated 5 times (circle and square points). There exist only one alternative over 25 for the point $(8, 4)$ to not be dominated by at least as one solution: if we take the circle and square sample on the top of the figure. Then $P(S \preceq (8, 4)) = 24/25 = 96\%$.

As described in [18] (k%-approximation set), the *probabilistic-k%-approximation set* can be derived in the same way to take into account multiple runs. It corresponds to the average *probabilistic-k%-approximation set* computed for each run done.

B. Compute an approached probabilistic empirical attainment function

The complexity of Alg. 3 is in $\theta(Ns^{n+1})$, where Ns is the total number of evaluations taken into account. Then, the exact value needs a lot of computation time. It will be interesting to approximate EAF, in order to apply it on large population size, or on problems with more than 2 objectives.

A simple way is to limit the number of areas defined. Let $c^n << Ns^n - 1$, be the number of hyper-squares envisaged. Then c^n hyper-square of size $1/(c-1)$, are defined by defining each set of objective values as $[1, 1 - 1/(c-1), 1 - 2/(c-1) \ldots, 0]$.

Then, we can use Alg. 3 as an approximation algorithm with a complexity of $Ns * c^n$: step 2 is replaced c^n cells, uniformly distributed. c could be set in order to favour computation time, or quality of the approximation.

We can also compute the probability that S dominates at least as one point of an hyper-surface $\mathcal{A}$. To do that, we only need to compare S against the nadir point of $\mathcal{A}$. Then we have a upper bound for the exact probabilistic EAF function (see Fig. 7).

Let us remark that computing the upper bound corresponds to compute the lower bound, with a translation of each solution of S by a vector $[-1/(c-1), \ldots, -1/(c-1)]$.

Algorithm 3 (Algorithm to compute EAF)

```
Input:   $n$              (Number of objectives)
         $S$              (The set to evaluate)
         $Z^-$            (nadir point)
Output:  $EAFT$           (EAF table)
Step 1:  $S = \{\mathcal{S}(\mathbf{x}^1) \ldots \mathcal{S}(\mathbf{x}^t)\}$
Step 2:  Scale the values of each objective vector $\mathbf{z} \in \mathcal{S}(\mathbf{x})$ ($\mathbf{x} \in S$), into
         the interval $[0 \ldots 1]^n$
Step 3:  for each objective $i \in [1 \ldots n]$, $T[i] \leftarrow$ Sort the different possible
         values of the objective i, in decreasing order ($T[i][j]$ = the $j^{th}$ worst
         solution of S, according to the objective function $f_i$)
Step 4:  $Ns \leftarrow$ Total number of evaluations
Step 5:  for i in $1 \ldots Ns^n$
           1. $V_1 \ldots V_n \leftarrow i$, written in base $Ns$
           2. $Ref \leftarrow [T[1][V_1], \ldots, T[n][V_n]]$ (temporary reference
              point)
           3. $p \leftarrow 0$
           4. for all $\mathbf{x} \in S$ do
              (a) $ps \leftarrow \sum_{\mathbf{z} \in \mathcal{S}(\mathbf{x})} \frac{dominates(\mathbf{z}, Ref)}{s_{\mathbf{x}}}$
              (b) $p \leftarrow 1 - ((1-p) * (1-ps))$
              (c) $EAFT[Ref] = p$
Step 6:  end for
Step 7:  return $EAFT$
```

An alternative algorithm: The complexity of the approximative EAF computation is in $Ns * c^n$. The next algorithm changes this complexity to $N(2c)^n$, which is interesting while $2^n < s$, i.e. for problems with only a few objectives to optimize. This algorithm proceeds as follows ($EAFT$ represents the set of areas where EAF value has to be computed):

- fill $EAFT$ values: initialize each cell to 0. Then, for each evaluation, define its cell by dividing each objective value by the range of one cell.
- then, for each solution $\mathbf{z} \in \mathcal{S}(\mathbf{x})$ and for each cell, compute how many solutions of the multiset $\mathcal{S}(\mathbf{x})$ dominates the solution $\mathbf{x}$. We can compute this in 2^n steps, using the previously computed values (Alg. 4). For example, in the bi-objective case, $EAFT[i][j] = EAFT[i][j-1] + EAFT[i-1][j] - EAFT[i-1][j-1]$, where i (resp. j) represents the i^{th} (resp. j^{th}) cell for the first (resp. second) objective function. The Alg. 4 computes the following recursive relation:

$$EATF[\mathbf{x}_1] \ldots [\mathbf{x}_n] = \sum_{i_1=0}^{1} \cdots \sum_{i_n=0}^{1} (-1)^{\sum_{j=1}^{n} i_j} EAFT[\mathbf{x}_1 - i_1] \ldots [\mathbf{x}_n - i_n] \tag{19}$$

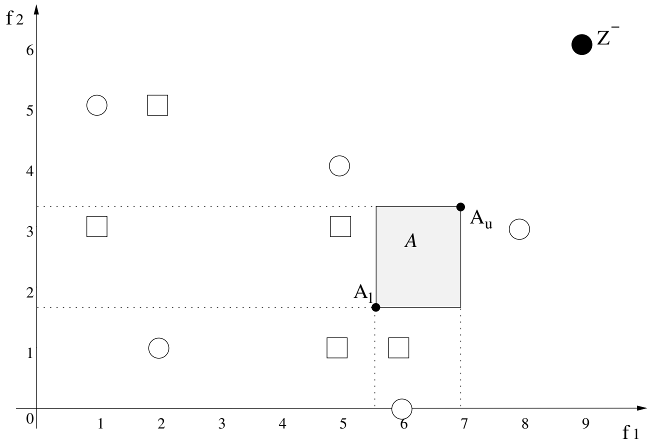

Figure. 7: Comparison of S against $\mathcal{A}_u$ give the upper bound of $P(S \preceq \mathbf{z}^*), \mathbf{z}^* \in \mathcal{A}$. If the comparison is realized against $\mathcal{A}_l$, we have a lower bound (S is represented by two multisets: circle and square solutions).

Algorithm 4 (*computeCell*: one *EAFT* value computation)

```
Input:   cell       (Current cell)
         dim        (The envisaged dimension)
Output:  EAFT[i]
Step 1:  $s \leftarrow 0$
Step 2:  for k in 0..$dim - 1$, $s \leftarrow s + computeCell(cell - c^k, k)$
Step 3:  return $EAFT[cell] - s$
```

C. Experimental results of EAF computation

We can apply the probabilistic EAF approximation to compute an approximated probabilistic Empirical Attainment Surface (EAS). Then, we can compute the maximum error of each bound: $\frac{(c+1)^n - c^n}{c^n}$. Then, by averaging the lower bound EAS_l and the upper bound EAS_u, the maximum error is reduced to $(EAS_l - EAS_u)/2$, which is less than $\frac{(c+1)^n - c^n}{2 * c^n}$.

We make several experimental EAF computation on uncertain random populations which are used in section IV-A. The Fig. 8 represents the computation time needed to compute EAF, with different number of objective functions and population size. This shows the interest of using the approximation algorithm, especially with many objective functions.

The quality of the approximation is represented in Figs. 9 and 10. The computation time corresponds to the ratio between the approximation and the exact EAF computation time. These figures show a good efficiency of EAF approximative algorithm, especially with 2 objective functions to optimize.

In Figs. 11 and 12, we represent the *probabilistic-k%-approximation set* for one uncertain run on KUR and $ZDT6$ test functions. This graphical representation could be applied for several runs, by averaging the probabilistic dominance of

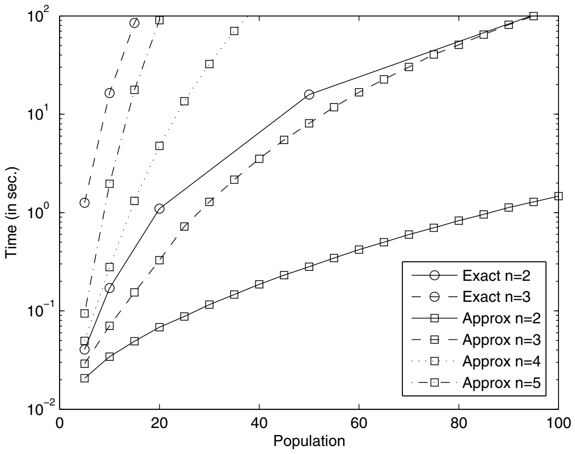

Figure. 8: Comparison exact/approximated approaches, with 10 evaluations per solution and a grid size of 10 for the approximation approach. The exact approach is time-expensive, and applicable only in bi-objective algorithm evaluation. Problems with 3 objective functions are practicable with the approximated measure.

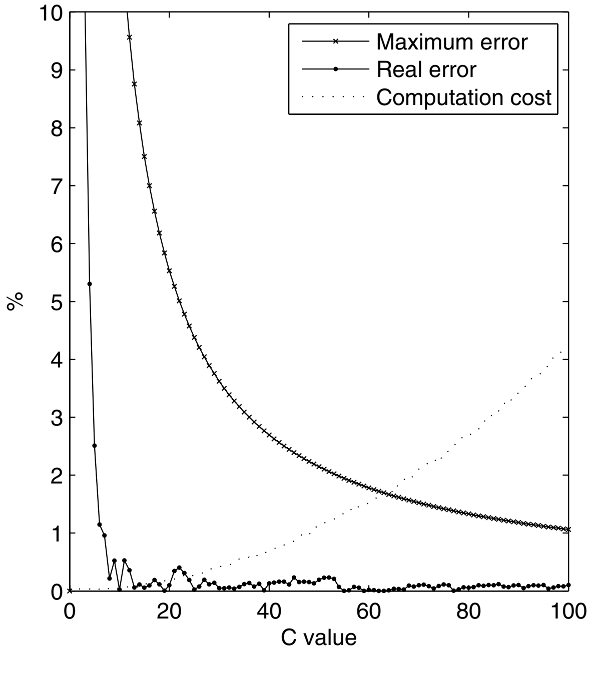

Figure. 9: Evaluation of the error ratio of the approximated approach, with different C values (example with 2 objectives, 50 individuals and 5 evaluation per individual).

each possible area of the objective space. For example, if we consider 50 runs, each run returning 50 individuals, which are evaluated 10 times, it corresponds, in the bi-objective case, to $(50{*}50{*}10)^2 = 6.25{\cdot}10^8$ potential different areas. Then, it is better to compute an *probabilistic-$k\%$-approximation set* by computing an approximated probabilistic EAF on a constant set of cells (only c^2 cells).

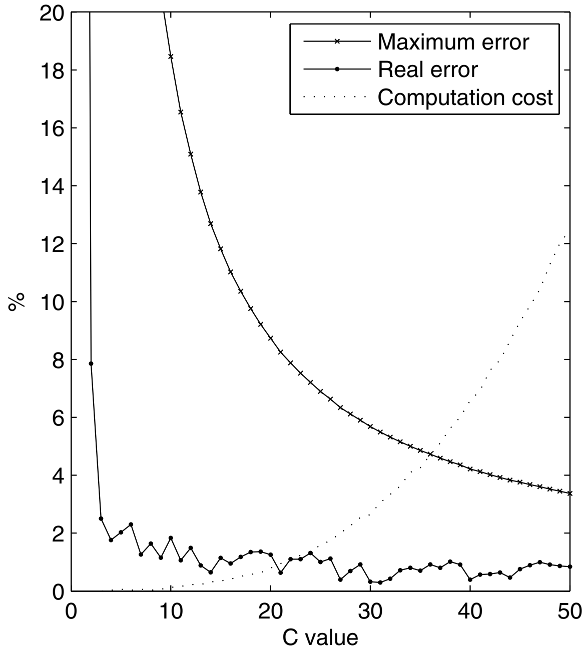

Figure. 10: Evaluation of the error ratio of the approximated approach, with different C values (example with 3 objectives, 20 individuals and 5 evaluation per individual).

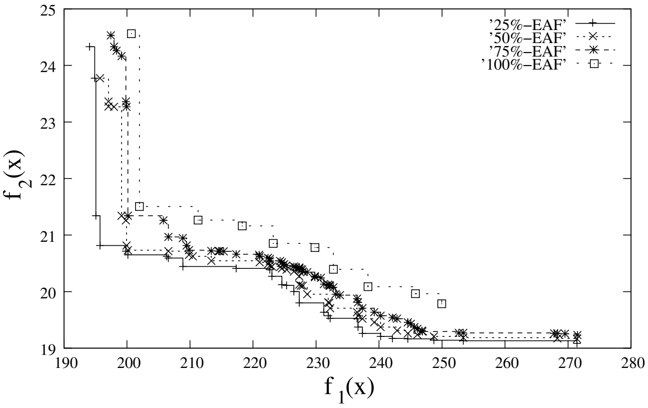

Figure. 11: EAF output, with a single run on KUR problem with EIV algorithm.

VI. Discussion

In this paper, we propose a method to handle uncertainty in indicator-based multiobjective optimization. Our approach tries to make no assumption about distribution, bounds and general tendency of the uncertainty. We propose the algorithm EIV, which computes the exact expected value of the I_ϵ-indicator. In order to apply this algorithm to environmental selection in EAs, we propose several algorithms which

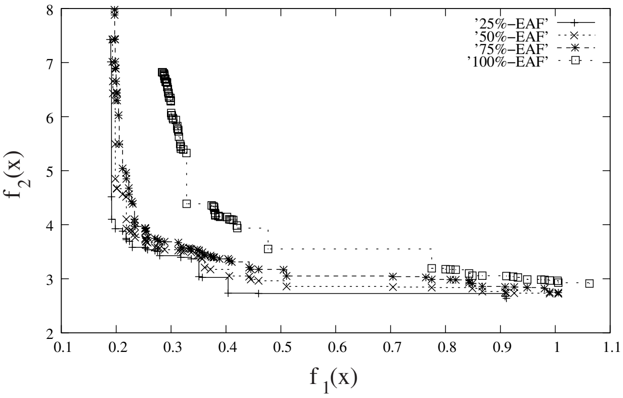

Figure. 12: EAF output, with a single run on $ZDT6$ problem with EIV algorithm.

approximate the results obtained by EIV, in order to select the best possible solutions during environmental selection, according to the I_ϵ-indicator quality indicator.

We made several experiments of the different methods proposed. First, we consider the goal to minimize the loss in quality during environmental selection: BCK give a good approximation of EIV selection, which is more time consuming. Then, we made some experiments on classical multiobjective tests functions. Regarding several other methods, Avg and PDR, EIV select the best solutions according to the I_ϵ-indicator, and BCK obtain the better approximation of the ideal selection, especially on functions with more than two objective to optimize. We use two different techniques to evaluate the outputs: by computing the 'true' objective vector value of every solution, then use the classical I_ϵ-indicator, and by computing a probabilistic I_ϵ-indicator value, without knowledge of the 'true' objective vector function.

In the section V, we propose an algorithm to compute an empirical attainment function, which evaluates the area of the objective space which is dominated by the output with different confidence levels. We propose one exact algorithm, and an approximative algorithm. We show some graphical results, concerning the computational time and efficiency of the algorithms. The exact algorithm is practicable only for a small number of solutions and only for biobjective functions. The approximation algorithms are practicable for 3 or 4 objective functions, but the approximation error increases with the number of objective functions.

There are some open questions resulting from this study:

- We propose indicator-based evolutionary approaches in order to handle uncertainty, but we only consider the $ES(N+1)$ evolution strategy. It will be interesting to evaluate the effectiveness of different evolution strategies, such as $ES(N+N)$, which allows a fast evolution in only a few generations. Only the $ES(N+1)$ evolution strategy has been investigated actually, in order to design an exact selection algorithm for this strategy. We could adapt our algorithm to the $ES(N+N)$ evolution strategy by using two different ways; (1) *Cut*: evaluate the fitness value of every solution, then select the N bests; (2) *Recursive*: evaluate the solutions, then delete the worst one, then re-evaluate the remaining solutions - stop when only N solutions remains. The first approach is no more expensive in time than our proposed approaches, the second approach seems more efficient and natural, but more expensive to compute. The algorithms presented in this article need to be adapted to the $ES(N+N)$ *recursive* selection strategy.
- In this article, we do not focus on the sample size setting, i.e. the number of evaluations to be done on the envisaged solutions. However, this is an important parameter of the optimization process. Indeed, a large sample size implies more computation time needed to compare the different solutions, but a good approximation of the real probability density function of the solutions (see equation 3). A good trade-off between computation time and quality of the approximation has to be found. This size could be defined dynamically during the optimization process, with a small initial value, which will increase during the run. Indeed, the difficulty to compare solution increases with the overall quality of the population. To achieve this, let set the initial sample size to 1, then increase its value every time a negative convergence rate is encountered after one new generation. Another idea is proposed in [12]: every generation, the ranking error of $s-1$ evaluations against s evaluations is computed. If the ranking error reach a threshold, the number of evaluations is increased by 1. Lastly, different number of evaluations could be defined for each solution of the population. Indeed, it would be interesting to refine the quality of the approximation for the oldest solutions of the populations, i.e. those which are selected every generation.

These different questions have to be answered, both in the bi-objective and multi-objective ($n > 2$) cases. Indeed, as shown in this study, the number of objectives has an impact on the results. Other parameters have to be checked, especially the uncertainty considered.

References

[1] D. V. Arnold. A comparison of evolution strategies with other direct search methods in the presence of noise, Computational Optimization and Applications, 24, pp. 135–159, 2003.

[2] M. Babbar, A. Lakshmikantha, and D. E. Goldberg. A modified NSGA-II to solve noisy multiobjective problems, In E. C. et al. (Ed.), Genetic and Evolutionary Computation Conference (GECCO'2003), late breaking papers, Volume 2723 of Lecture Notes in Computer

Science, Chicago, Illinois, USA, pp. 21–27, Springer, July 2003.

[3] W. J. Conover. Practical Nonparametric Statistics, New-York, NY: John Wiley and Sons, 1999.

[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein (Eds.). Introduction to Algorithms, MIT press and McGraw-Hill, Section 8.4: Bucket sort, pp. 174–177, 2001.

[5] K. Deb, and R. B. Agrawal. Simulated binary crossover for continuous search space. Complex Systems, 9, pp. 115–148, 1995.

[6] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In M. Schoenauer et al. (Eds.), Parallel Problem Solving from Nature (PPSN VI), Lecture Notes in Computer Science Vol. 1917, pp. 849–858, Springer, 2000.

[7] K. Deb, and H. Gupta. Searching for robust pareto-optimal solutions in multi-objective optimization, In C. A. Coello Coello, A. H. Aguirre, and E. Zitzler (Eds.), Conference on Evolutionary Multi-Criterion Optimization (EMO'05), Volume 3410 of Lecture Notes in Computer Science (LNCS), Guanajuato, Mexico, pp. 150–164, Springer-Verlag, March 2005.

[8] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. Technical report, TIK Report Nr. 112, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, July 2001.

[9] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham et al. (Eds.), Evolutionary Computation Based Multi-Criteria Optimization: Theoretical Advances and Applications, Springer, 2004a (To appear).

[10] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham et al. (Eds.), Evolutionary Computation Based Multi-Criteria Optimization: Theoretical Advances and Applications, Springer, 2004b (To appear).

[11] V. Grunert da Fonseca, C. M. Fonseca, and A. O. Hall. Inferential performance assessment of stochastic optimisers and the attainment function, In E. Zitzler, K. Deb, L. Thiele, C. A. C. Coello, and D. Corne (Eds.), Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO 2001), Volume 1993 of Lecture Notes in Computer Science, Berlin, pp. 213–225, Springer-Verlag, 2001.

[12] N. Hansen, A. Niederberger, L. Guzzella, and P. Koumoutsakos. Evolutionary optimization of feedback controllers for thermoacoustic instabilities, To be submitted to IEEE Transactions on Evolutionary Computation, 2006.

[13] J. Horn, and N. Nafpliotis. Multiobjective optimization using the niched pareto genetic algorithm, Technical report, University of Illinois, Urbana-Champaign, Urbana, Illinois, USA, 1993.

[14] E. Hughes. Evolutionary multi-objective ranking with uncertainty and noise. In EMO'01: Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization, London, UK, pp. 329–343, Springer-Verlag, 2001.

[15] Y. Jin and J. Branke. Evolutionary optimization in uncertain environments – a survey, IEEE Transactions on evolutionary computation, 9(3), pp. 303–317, June 2005.

[16] J. D. Knowles. Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization, Ph. D. thesis, University of Reading, 2002.

[17] J. D. Knowles, L. Thiele, and E. Zitzler. A tutorial on the performance assessment of stochastive multiobjective optimizers, Technical Report TIK-Report No. 214, Computer Engineering and Networks Laboratory, ETH Zurich, July 2005.

[18] S. Künzli, L. Thiele, and E. Zitzler. Multi-criteria decision making in embedded system design. In B. Al-Hashimi (Ed.), System On Chip: Next Generation Electronics, IEE Press, pp. 3–28, January 2006.

[19] F. Kursawe. A variant of evolution strategies for vector optimization. In H.-P. Schwefel and R. Männer (Eds.), Parallel Problem Solving from Nature, pp. 193–197, Springer, 1991.

[20] D. Quagliarella, and A. Vinici. In D. Quagliarella, J. Périaux, C. Poloni, and G. Winter (Eds.), Genetic Algorithms and Evolution Strategy in Engineering and Computer Science – Recent advances and industrial applications, Chapter Coupling genetic algorithms and gradient based optimization techniques, pp. 289–309, Wiley, Chichester, 1997.

[21] T. Siegfried and W. Kinzelbach. A multiobjective discrete stochastic optimization approach to shared aquifer management: Methodology and application, Water Resour. Res., 42, W02402, 2006. doi:10.1029/2005WR004321.

[22] J. Teich. Pareto-front exploration with uncertain objectives. In Conference on Evolutionary Multi-Criterion Optimization (EMO'01), Volume 1993 of Lecture Notes in Computer Science (LNCS), pp. 314–328, March 2001.

[23] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results, Evolutionary Computation, 8(2), pp. 173–195, April 2000.

[24] E. Zitzler and S. Künzli. Indicator-based selection in multiobjective search. In Proc. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII), Birmingham, UK, pp. 832–842, September 2004.

[25] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In K. Giannakoglou et al. (Eds.), Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001), International Center for Numerical Methods in Engineering (CIMNE), pp. 95–100, 2002.

[26] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review, IEEE Transactions on Evolutionary Computation, 7(2), pp. 117–132, 2003.

Author Biographies

Matthieu Basseur received the PhD of computer science from University of Lille in France, in June 2005. He is actually a temporary assistant professor in the Computer Science Laboratory at the University of Lille. His research focuses on different areas of multiobjective optimization: self-adaptive metaheuristics, application of metaheuristics to the multi-objective case, and cooperative methods (metaheuristics/exact approaches, parallel metaheuristics...)

Eckart Zitzler received degrees from University of Dortmund in Germany (diploma in computer science) and ETH Zurich in Switzerland (doctor of technical sciences). Since 2003, he has been Assistant Professor for Systems Optimization at the Computer Engineering and Networks Laboratory at the Department of Information Technology and Electrical Engineering of ETH Zurich, Switzerland. His research focuses on bio-inspired computation, multiobjective optimization, computational biology, and computer engineering applications. Dr. Zitzler was General Co-Chairman of the first three international conferences on evolutionary multi-criterion optimization (EMO 2001, EMO 2003, and EMO 2005).