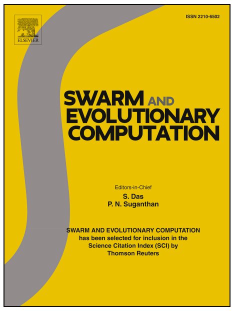
Accepted Manuscript



A generic fuzzy approach for multi-objective optimization under uncertainty

Oumayma Bahri, El-Ghazali Talbi, Nahla Ben Amor

PII: S2210-6502(17)30486-8

DOI: 10.1016/j.swevo.2018.02.002

Reference: SWEVO 355

To appear in: *Swarm and Evolutionary Computation BASE DATA*

Received Date: 16 June 2017

Revised Date: 28 January 2018

Accepted Date: 7 February 2018

Please cite this article as: O. Bahri, E.-G. Talbi, N.B. Amor, A generic fuzzy approach for multi-objective optimization under uncertainty, *Swarm and Evolutionary Computation BASE DATA* (2018), doi: 10.1016/j.swevo.2018.02.002.

A Generic Fuzzy Approach for Multi-objective Optimization under Uncertainty

Oumayma Bahri[a,b,*], El-Ghazali Talbi[a], Nahla Ben Amor[b]

[a] *INRIA Laboratory, CRISTAL-CNRS, 59650 Villeneuve d'Ascq, Lille, France*
[b] *LARODEC Laboratory, ISGT, 2000 Le Bardo, Tunis, Tunisia*

Abstract

Multi-objective optimization under uncertainty has gained considerable attention in recent years due to its practical applications in real-life. Many studies have been conducted on this topic, but almost all of them transformed the problem into a mono-objective one or just neglected the effects of uncertainty on the outcomes. This paper addresses specific uncertain multi-objective problems in which uncertainty is expressed by means of triangular fuzzy numbers. To handle these problems, we introduced a new approach able to solve them without any transformation by considering fuzziness propagation to the objective functions. The proposed approach is composed of two main contributions: First, a fuzzy Pareto dominance is defined for ranking the generated fuzzy solutions. Second, a generic fuzzy extension of well-known evolutionary algorithms is suggested as resolution methods. An experimental study on multi-objective Vehicle Routing Problems (VRP) with uncertain demands is finally carried to evaluate our approach.

Keywords: Multi-objective optimization, Fuzzy sets, Triangular fuzzy numbers, Pareto dominance, Evolutionary algorithms, Vehicle routing problem

1. Introduction

Multi-objective optimization is an important and complex field in decision making in which many scientific and industrials must cope. Indeed, in many

*Corresponding author
Email address: oumayma.b@gmail.com (Oumayma Bahri)
URL: oumaymabahri.com (Oumayma Bahri)

real-world applications, the decision maker is often confronted with several conflicting objectives that should be simultaneously optimized. Hence, the most common purpose is to choose the best trade-offs among all the predefined objectives. A wide variety of resolution methods and techniques for solving combinatorial multi-objective problems have been developed in the literature [1, 2, 3]. Despite the massive number of proposed methods, there are still many open issues in this topic. For example, there is no consideration of uncertainty aspect in the classical multi-objective concepts.

However, uncertainty characterizes almost all practical applications in which the big amount of data provides certainly some inevitable imperfections. These imperfections might result from using unreliable information sources caused by inputting data incorrectly, faulty reading instruments or bad analysis of some training data. It may also be the result of poor decision maker opinions due to any lack of its background knowledge or the difficulty of giving a perfect qualification for some costly situations. The classical way to deal with uncertainty is the probabilistic reasoning originated from the middle of the 17th century [4]. Nevertheless, probability theory was considered for a long time as a very good quantitative tool for uncertainty treatment, but as good as it is, this theory is only appropriate when probability distributions are available which is not always the case. Otherwise, there are some situations like the case of total ignorance, which are not well handled and so can make the probabilistic reasoning unsound. In this context, a panoply of non-classical tools for handling uncertainty appears such as fuzzy sets [5] to which we are interested in this work.

Moreover, the combination of uncertainty and multi-objectivity aspects has not been deeply studied so far. In particular, this combination leads to specific optimization problems characterized by the necessity of optimizing simultaneously many objectives while considering that some input data are not known beforehand. The challenge of solving these problems lies in their computational complexity. In fact, the effects of unavoidable uncertainties in preference parameters, decision variables, constraints, and/or objective functions could make such problems more complicated and difficult. Unfortunately, almost all existing approaches have been limited to transform the uncertain multi-objective problem into one or more mono-objective problems by using for example aggregation functions [6, 7]. Some other approaches have been focused on treating the problem in its multi-objective context while ignoring the propagation of uncertain inputs to the objectives and obviously to the resulting solutions [8]. Only few studies have been devoted to handle the problem as-is without erasing any of its characteristics by developing an interval-based multi-objective approach [9, 10].

The aim of the present study is to introduce a novel approach for dealing with fuzzy multi-objective problems, in which fuzziness is modeled via triangular fuzzy numbers. We first present details of our fuzzy Pareto dominance previously introduced in [11]. We next use it to extend two popular Pareto-based evolutionary algorithms to our fuzzy setting, namely SPEA2 and NSGAII algorithms, briefly studied in [12]. In addition, we propose a fuzzy extension of another competitive and third generation algorithm, namely IBEA [13].

The remainder of the paper is organized as follows. Section 2 recalls the basic definitions and concepts on which our approach is based. Section 3 describes in detail our proposal. Section 4 illustrates the application of the proposed approach on a multi-objective vehicle routing problem with fuzzy demands and finally section 5 summarizes the obtained results.

2. Background

This section first presents basic definitions related to deterministic multi-objective optimization [14] and then gives an overview of existing approaches for the uncertain case.

2.1. Deterministic multi-objective optimization

Deterministic multi-objective optimization is the process of optimizing systematically and simultaneously two or more conflicting objectives subject to some constraints. Contrary to the single-objective case, multi-objective optimization does not restrict to find a unique global solution but it aims to find a set efficient solutions. The following definitions gives more details about multi-objective concepts. Without a loss of generality, we only consider, throughout this paper, combinatorial *minimization problems.*

2.1.1. Definitions

Formally, a *multi-objective optimization problem (MOP)* consists of solving the following mathematical program:

$$\min\ F(x) = (f_1(x), f_2(x), \ldots, f_n(x))\ s.t.\ \ x \in S \tag{1}$$

where $F(x)$ is the vector of n ($n \geq 2$) objective functions to be minimized and $x = (x_1, \ldots, x_k)$ is the vector of decision variables from the set of feasible solutions S associated with equality and inequality constraints. In a combinatorial MOP, the feasible region S becomes a discrete set of solutions. Besides, $F(x)$ maps the decision variables x from the decision space to the objective space by assigning a cost function $y \in Y$ that evaluates the quality of each solution.

$$F : X \rightarrow Y \subseteq \mathbb{R}^n, \quad F(x) = y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \tag{2}$$

where $Y = F(S)$ represents the feasible points (solutions) in the objective space and $y_i = f_i(x)$ is a point of this space that represents the solution quality or fitness.

In order to identify efficient solutions of a given MOP, other concepts should be considered known as Pareto optimality.

An objective vector $y = (y_1, \ldots, y_n)$ *Pareto dominates* another objective vector $y' = (y'_1, \ldots, y'_n)$ (denoted by $y \prec_p y'$) if and only if no component of y' is smaller than the corresponding component of y and at least one component of y is strictly smaller:

$$\forall i \in 1, \ldots, n : y_i \leq y'_i \wedge \exists i \in 1, \ldots, n : y_i < y'_i. \tag{3}$$

A solution $x^* \in X$ is *Pareto optimal* (also known as efficient, non-dominated or non-inferior) if for every $x \in X$, $F(x)$ does not dominate $F(x^*)$, that is, $F(x) \not\prec_p F(x^*)$. A *Pareto optimal set* P^* is defined by:

$$P^* = \{x \in X / \exists x' \in X, F(x') \not\prec_p F(x)\}. \tag{4}$$

The image of this *Pareto optimal set* P^* in the objective space is called *Pareto front* PF^* defined by:

$$PF^* = \{F(x), x \in P^*\}. \tag{5}$$

Finding the *Pareto front* is known to be a difficult task [14]. In fact, identifying the Pareto optimal (non-dominated) solutions is generally an NP-hard problem. Thus, the main goal is to identify an approximation of the *Pareto optimal set*, from which the decision maker can choose a best solution based on the current situation. A good approximation of Pareto solutions should satisfy two properties: (i) convergence or closeness to the exact Pareto front and (ii) uniform diversity of the obtained solutions around the Pareto front.

2.1.2. Multi-objective evolutionary algorithms

Although a large variety of methods have been developed to solve MOPs, the family of multi-objective evolutionary algorithms (MOEAs) is the most widely used and successfully applied in diverse areas [1]. Otherwise, MOEAs have proved to be powerful and suitable for multi-objective optimization because of their population-based search and ability to find multiple optimal solutions with a good spread. These algorithms are mainly based on the following three components:

- **Fitness assignment:** allows to guide the search process toward Pareto optimal solutions for a better convergence. Examples of fitness assignment strategies are: Dominance-based approach and Indicator-based approach.

- **Diversity preservation:** allows to generate a diverse set of Pareto solutions by using density estimation techniques such as Nearest neighbor method, crowding strategy, etc.

- **Elitism:** allows to use and preserve the best solutions found during the search process. An elite population, called archive, is generally used to store the Pareto optimal solutions.

Almost all MOEAs follow the main steps described in the flowchart given in Figure 1. The process starts with an initial population generated randomly. This is followed by an evaluation of candidate solutions in the population using a fitness assignment strategy. Thereafter, a diversity preservation step is performed to select diverse efficient solutions. An external population or archive is then used to maintain the selected solutions. However, every solution is reproduced using variation operators (e.g., crossover and mutation) to generate new offsprings. At the last step, a replacement scheme is applied to determine which solutions in the population will survive from the offsprings and the parents. These steps are iterated until a stopping criteria hold.

2.1.3. Multi-objective quality indicators

Several quality indicators have been proposed in the literature in order to assess the performance of two sets or approximations of Pareto solutions. They can be classified depending on several features:

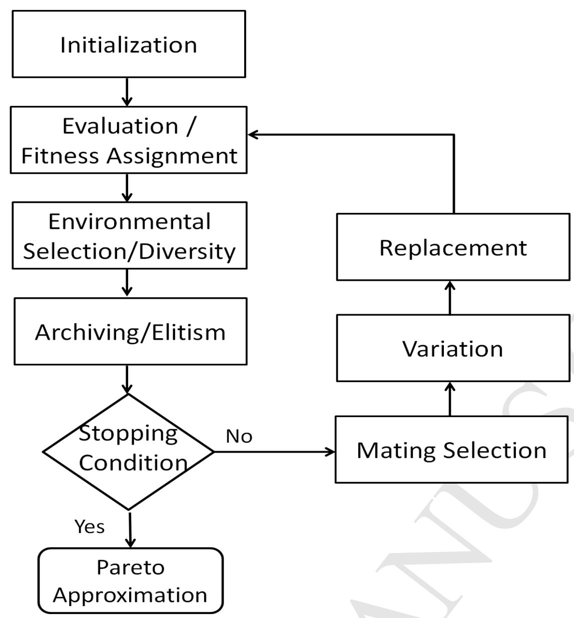


Figure 1: General scheme of MOEAs

- *Arity (Unary/Binary)*: A unary indicator represents the quality of each approximation by a unique scalar-value, whereas a binary indicator allows to compare the performance of two approximated Pareto fronts.

- *Performance goal*: It is mainly the convergence (or closeness) toward the optimal Pareto front and/or diversity of solutions along the front.

- *Required parameters*: They are usually defined by the user such as reference set, reference point, and/or ideal point.

In this paper, we concentrate our interest on two well-known indicators [15]:

(i) *Hypervolume indicator I_H:* considered as one of few indicators that measure the approximation quality in terms of convergence and diversity simultaneously. It belongs to the class of hybrid metrics and exists in both unary and binary forms . In its unary form, this indicator measures the volume of the objective space dominated by a given approximation. It requires the specification of a reference point Z_{ref} that denotes an upper bound over all the objectives (i.e. The higher this volume is, the better is the approximation). Its binary form, the so-called hypervolume difference I_H^- evaluates the quality of a given output set A in comparison to a reference set R. It computes the difference between these two sets by measuring the portion of objective space weakly dominated by R and not by A:

$$I_H^-(A) = I_H(R) - I_H(A) \tag{6}$$

where smaller values correspond to higher quality in contrast to the unary hypervolume I_H.

(ii) *$\epsilon-$indicator I_ϵ:* dedicated to the measure of approximations quality in term of convergence. The epsilon indicator family comprises a multiplicative and an additive version, both exist in unary and binary form. The unary additive ϵ-indicator ($I_{\epsilon+}$) gives the minimum factor ϵ by which an approximation set A has to be translated in the criterion space to weakly dominate a reference set R. It is formally expressed by:

$$I^1_{\epsilon+}(A) = I_{\epsilon+}(A, R) = min\{\forall x \in R, \exists x' \in A : x'_i - \epsilon \preceq x_i, \forall i \in 1 \dots n\} \tag{7}$$

Notice that, all the above concepts are only dedicated to deterministic MOPs where all inputs data are known in advance (i.e. with crisp values). Therefore, in presence of imperfections, all these concepts should be adapted and extended to uncertain setting. In the following, we propose a classification of existing approaches for handling uncertainty in MOPs.

2.2. Uncertain multi-objective optimization

Multi-objective optimization under uncertainty is nowadays one of the most important and difficult research fields in decision making. This field has attracted increasing attention since it appears in many real-life applications and poses several interesting challenges [16, 17, 18]. In general, a MOP under uncertainty is characterized by the necessity of optimizing simultaneously several conflicting objectives in presence of some uncertain input data.

However, the main issue is how to identify the type of inevitable uncertainties and their impacts on the results and optimal decision making. The purpose at this level is to analyse the manner in which such uncertainties are propagated through the optimization process. Usually, uncertainty propagation leads to an excessive increase in problem's complexity and difficulty for finding optimal solutions. In particular, the resolution stage will be much more complicated since propagating uncertainties may affect the key elements of multi-objective optimization process such as preference parameters, decision variables, constraints and/or objectives. Many studies addressing the effects of uncertainty propagation in multi-objective setting have focused on the case where uncertainty is assumed to occur in the objective functions. Yet, uncertainty in the objectives presents a critical and sensitive obstacle because it may influence the search process and consequently hamper the identification of efficient solutions. A MOP with uncertain objectives may be defined as:

$$\min\ F(x,\xi) = \min[f_1(x,\xi), f_2(x,\xi), \ldots, f_n(x,\xi)]\ s.t.\ \ x \in X, \xi \in U \qquad (8)$$

where F is the set of objective functions that may depend on uncertainty scenarios U, x is a decision variable vector from its admissible region $X \subseteq \mathbb{R}^n$ and $\xi = (\xi_1, \xi_2, \ldots, \xi_q)$ is a vector of independent uncertain variables. Clearly, the problem here is that each $f_i(x,\xi)$ is an uncertain quantity induced by ξ.

Once the uncertain scenarios and their effects are identified, the second relevant challenge consists to find a suitable way for handling uncertainty in multi-objective resolution methods. Nonetheless, very little research works have been proposed so far to deal with uncertain multi-objective optimization. Besides, although uncertainty in the objective functions has gained attention in recent years, the efforts devoted to this problem are still limited. In Figure 2, we propose to classify the existing approaches according to how the uncertain multiple objectives are managed.

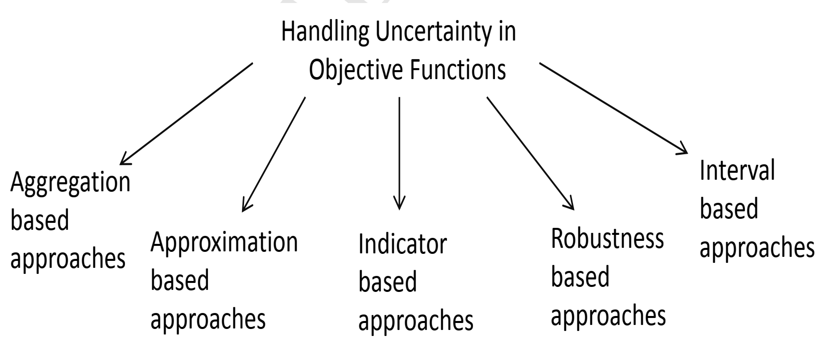


Figure 2: Taxonomy of approaches for MOPs with uncertain objectives

The first attempts to cope with uncertainty in objectives belong to the category of aggregation-based approaches [6, 7]. The basic idea of these traditional approaches is to combine the multiple objectives into a single uncertain one. In other words, they convert the MOP into a one or a set of single-objective problems. Furthermore, the different objectives can be rewritten into an aggregate objective f_A by applying a weighted sum function as follows:

$$f_A(x,\xi) = \sum[f_1(x,\xi), f_2(x,\xi), \ldots, f_n(x,\xi)] \tag{9}$$

These approaches have the advantage of simplicity because they do not require a particular development for uncertain multi-objective optimization. Indeed, the existing methods for handling uncertainty in single-objective optimization can simply be applied. Yet, these latter still not efficient since they limit the objective space, ignore the significant role of multi-objectivity and also relationship between the conflicting objectives. In consequence, the obtained results are very often useless and far from reality.

The second category encloses approximation-based approaches that use statistical functions to convert the uncertain objectives into their crisp equivalents [19, 8]. Otherwise, these approaches still abide to the certainty of objectives and usually allow to carry out an approximation of observed uncertainty. In this case, a statistical function may be applied to approximate each objective function as follows:

$$\Phi(f_1(x,\xi)), \Phi(f_2(x,\xi)), \ldots, \Phi(f_n(x,\xi)) \tag{10}$$

where $\Phi(.)$ denotes the statistical operator which can be the mean $M[.]$ or expected function $E[.]$ with respect to ξ. Clearly, this allows to transform the uncertain MOP into a crisp problem that can be resolved using standard deterministic multi-objective optimizers. A major limit of approximation-based approaches is that the propagation and effects of uncertainty on the objectives are neglected.

The third category includes different approaches [20, 21] that combine uncertainty of objectives and quality indicators (i.e., real-valued functions which allow assessment of Pareto approximations). This combination is done by estimating indicator evaluations for the uncertain objective vectors as:

$$I(f_1(x,\xi), X^*), I(f_2(x,\xi), X^*), \ldots, I(f_n(x,\xi), X^*) \tag{11}$$

where $X^* = \{x_1^*, \ldots, x_r^*\}$ is a variable reference set and $I(.)$ stands for the vector of indicator values that can be minimized or maximized depending on the quality goal. Thus, an indicator-based model is often applied to reflect the uncertainty of objectives.

Another category of approaches refers to the robustness aspect to handle uncertainty propagation [22, 23]. This aspect is connected to the idea that in presence of uncertain inputs, the outputs should be relatively insensitive (small uncertainty outputs). The robustness in objective functions can be modeled as:

$$(f_1(x,\xi), R_1), (f_2(x,\xi), R_2), \ldots, (f_n(x,\xi), R_n) \tag{12}$$

where R_i is the robustness criterion that should be maximized. It is defined in terms of the variation of $f_i(x)$ regarding the uncertainty associated with x. However, the main drawback of robustness-based and indicator-based approaches is that they rely on the assumption of a priori knowledge about decisive information such as the reference set of solutions or the robustness confidence level. Evidently, if such information is inappropriate or incorrect, the outputs of theses approaches can be misleading. In addition, these approaches only investigate the sensitivity and effects of uncertainty (i.e. by applying an indicator-based evaluation or robustness analysis), but they do not provide a specific representation of uncertain objectives. Otherwise, the uncertainty propagation from inputs to outputs is not considered.

Yet, this can lead to very poor decisions with often misleading simulation results. It is therefore necessary to account for the relationship between uncertain inputs and generated solutions, because if the input data or parameters are highly uncertain, how can the optimizer simply state that the outputs are robust ? It may be feasible only for simplicity or other practical reasons as long as the optimization performance will not be affected.

Further, very few studies assume to display uncertainty of objectives through intervals and thereby to perform the multi-objective optimization based on this uniform distribution. These studies fall under the category of interval-based approaches [9, 10], where the cost of evaluating $f(x,\xi)$, namely Y is represented by intervals as:

$$F(x,\xi) = Y = ([\underline{y_1}, \overline{y_1}], \ldots, [\underline{y_n}, \overline{y_n}]) \tag{13}$$

where $\underline{y_i}$ and $\overline{y_i}$ are respectively lower and upper bounds of the corresponding interval-valued function i. In this case, the generated solutions will be disrupted by the interval shape of objectives and so clearly the classic Pareto concepts cannot be used for comparing them. Indeed, extensions of the classic Pareto optimality have been discussed and addressed in recent literature [9, 24, 25, 26]. For instance, authors in [24, 25] involve the uncertainty in the objectives as fuzzy degrees. Then, an interval-based scheme is introduced to define the Pareto optimality of fuzzy solutions. In [9, 10], intervals of belief functions are used to represent the uncertain Pareto optimal solutions. However, all of these works consider intervals on the uncertain objectives and then apply a Pareto analysis between interval-valued outcomes. Consequently, the solutions are represented by finite bounding-boxes (rectangular form) in the objective space as shown in Figure 3, where each rectangular represents one solution.

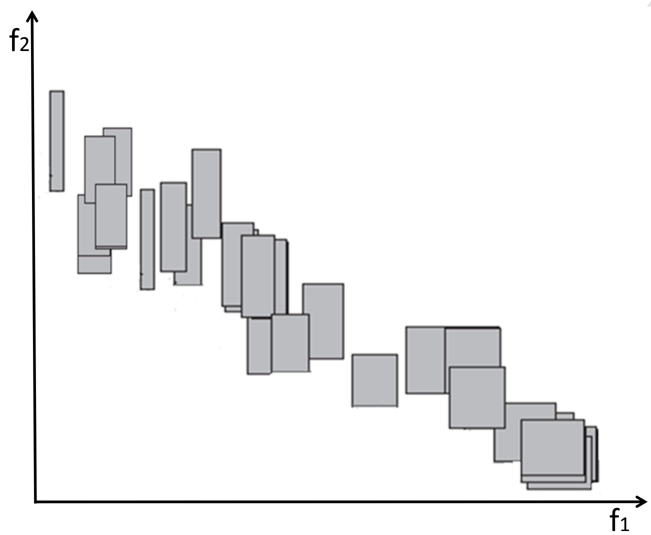


Figure 3: Examples of interval-based solutions

One of the most interesting questions is *how to analyse the Pareto optimality when uncertainty is modeled by non-crisp intervals ?* Unfortunately, it is not easy to compare among two or more non-crisp intervals (e.g. that can be possibility distributions, fuzzy numbers, etc). Our interest in this study focuses mainly on dealing with uncertain multi-objective optimization, when uncertainty in objectives is modeled by fuzziness. In particular, we aim to handle the specific case of MOPs with fuzzy-valued objective functions and to design a generic optimization approach for this purpose. To the best of our knowledge, there is no similar study of the propagation of fuzziness in multi-objective optimization process.

3. Proposed Approach

This section first defines the problem of our interest, especially the choice of fuzzy setting and its impact through the optimization process. Next, it describes our two main contributions for handling such a problem which are: (1) A new fuzzy Pareto dominance; (2) A generic fuzzy extension of well-studied MOEAs.

3.1. Fuzzy MOP formulation

We address here multi-objective optimization problems with fuzzy data in which fuzziness can be associated with the linguistic vagueness, polysemy or ambiguity of information due to limited knowledge [5]. In particular, we focus our attention on normalized fuzzy sets, so-called fuzzy numbers, which are frequently used to represent the approximate reasoning of linguistic values in many real-world applications [27, 28]. Yet, the building block of fuzzy numbers is a gradual membership function (MF) varying in the range of $[0, 1]$ and that can takes different shapes like triangular, trapezoidal, etc.

The choice of an appropriate MF depends entirely on the nature of imperfect data, the problem size and plays a substantial role in the overall optimization process. Accordingly, this specific issue has been widely studied in the literature [29, 30]. In fact, many researchers [31, 28] have discussed the performance of various fuzzy shapes by taking into account features such as simplicity, convenience, speed, and efficiency. Almost all of them convoyed that the triangular MF is found to be more intuitive, simple to implement and fast for computation compared to any other shape. Besides, it proved most popular and suitable shape for a human interpretation due to the simplicity, smoothness and concise notation of its linear piece-wise MF. For example, in order to model a product price estimation by fuzzy MFs, it is important to know how much the price is low, medium or high and so intuitively choosing the triangular shape with three degrees (i.e. lower price, most possible price and higher price).

Formally, a Triangular Fuzzy Number (TFN) is represented with a triplet of values $A = [\underline{a}, \widehat{a}, \overline{a}]$, where $[\underline{a}, \overline{a}]$ is the interval of possible values called its support and $\widehat{a}$ denotes its modal or kernel value (the most plausible). Another interesting advantage is that a TFN can be deduced from transformations of other different shapes by linguistic modifiers, compositions, projections and other operations [5]. For instance, a trapezoidal fuzzy number $B = [b_1, b_2, b_3, b_4]$ is transformed to triangular if $b_2 = b_3$.

All these advantages encourage us to consider the triangular fuzzy shape in the context of multi-objective optimization. Therefore, we suppose that fuzziness in inputs data will be modeled by means of TFNs for any MOP. Then as explained before, propagating fuzziness through the optimization process should not be ignored because this may distort the results. Otherwise, fuzziness in inputs data will clearly has great influence over the way a MOP is designed and optimized. Thus we should first consider this fuzziness when designing the multi-objective problem and then predict their unavoidable impacts on the search process. In that sense, we should analyse the effects of the triangular fuzzy shape on MOPs, especially on the problem outcomes and their optimality. Hence, the objective functions in such problems will be disrupted by the triangular fuzzy shape. Let us assume that a minimization MOP with triangular fuzzy-valued objectives is defined as follows:

Definition 1. *Triangular fuzzy MOP*

$$\min \ F(x^\tau) = (f_1(x^\tau), f_2(x^\tau), \ldots, f_n(x^\tau)) \ s.t. \ \ x \in X, \ \tau \in R \tag{14}$$

where $F(x^\tau)$ *is the vector objective functions, which are disrupted by the triangular form* τ *from the universal set* R *of fuzzy numbers. In the objective space, the vector* F *can be defined as a fuzzy cost function that represents the fitness of solutions by assigning a triangular-valued objective vector* Y^τ*:*

$$F : X \to Y \subseteq (\mathbb{R} \times \mathbb{R} \times \mathbb{R})^n,$$
$$F(x^\tau) = Y^\tau = \begin{pmatrix} y_1 = [\underline{y_1}, \widehat{y}_1, \overline{y_1}] \\ y_2 = [\underline{y_2}, \widehat{y}_2, \overline{y_2}] \\ \cdots \\ y_n = [\underline{y_n}, \widehat{y}_n, \overline{y_n}] \end{pmatrix} \tag{15}$$

It is clear that this formulation is a fuzzy counterpart of the classical MOP definition given in Section 2.1. In this case, the solutions are modeled by vectors of triangular fuzzy numbers. Figure 4 shows an example of the nature of solutions that may be obtained in our case of fuzzy bi-objective space.

Instead of the rectangular form in the case of crisp-intervals, the front here is composed by a set of triangles, where each triangle models one fuzzy solution. For instance, the triangular solution marked in bold in Figure 4 represents a vector of two triangular fuzzy numbers which are respectively: $[\underline{y_1}, \widehat{y}_1, \overline{y_1}]$ as solution values for the objective f_1 and $[\underline{y_2}, \widehat{y}_2, \overline{y_2}]$ as solution values for f_2. It should be noticed that the plot of these two fuzzy numbers in bi-objective space may lead to a rectangular form (like bounding boxes in Figure 3), considering the corner points of $[\underline{y_1}, \overline{y_1}]$ and $[\underline{y_2}, \overline{y_2}]$.

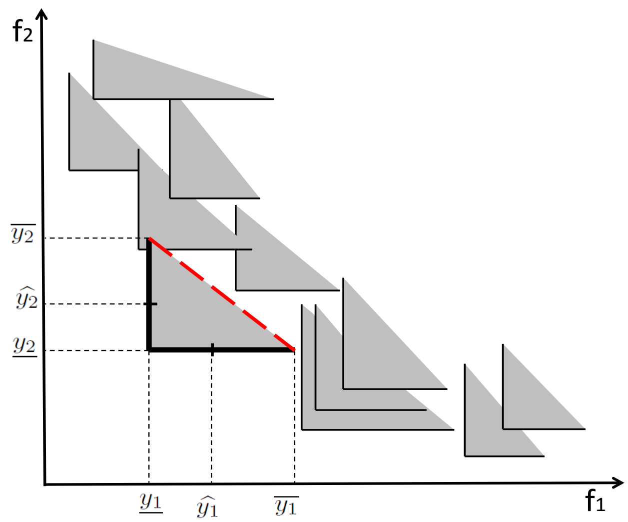


Figure 4: Examples of triangular solutions

For convenience, we propose to divide the linear box-structures into triangles by a splitting line (see red dotted line in Figure 4) between the upper values or extreme points $(\overline{y_1}, \overline{y_2})$. This condition allows us to delimit the most plausible area of solution values in sense of minimization and therefore leads to a better representation accuracy. In practice, this triangular shape reflects better the distribution of our data and thereby suits us in terms of ease of data manipulation and decision making.

Subsequently, once the form of problem solutions is predicted, the issue now is how to explore the optimality process between them. Yet, the classical Pareto concepts cannot be used in this case since they are only meant for deterministic case (i.e. when the solutions are exact values). To this end, a need for special optimality aspects capable to handle the generated solutions of triangular fuzzy values is evident.

3.2. Fuzzy Pareto optimality

In the following, we present our new Pareto dominance for handling optimality in any MOP with fuzzy data, especially with triangular-valued objectives. Then as multi-objectivity usually involves problems with only two objectives, each solution here is a vector of two triangular fuzzy numbers (TFNs). Hence, our contribution is composed of two main phases which are: (i) the definition of mono-objective dominance relations between two TFNs;

(ii) the determination of multi-objective dominance based on the types of mono-objective dominance found for both objectives.

3.2.1. Mono-objective dominance

It should first be noted that our proposal is inspired by the paradigm of ranking fuzzy numbers, where a comparison procedure is often required to check the relationship between them. In this regard, we take into account that all possible topological relationships between two TFNs may be covered by only four topological situations [32] as shown in Figure 5. Based on these situations, we propose three mono-objective dominance relations which are: *Total dominance* ($\prec_t$), *Partial strong-dominance* ($\prec_s$) and *Partial weak-dominance* ($\prec_w$).

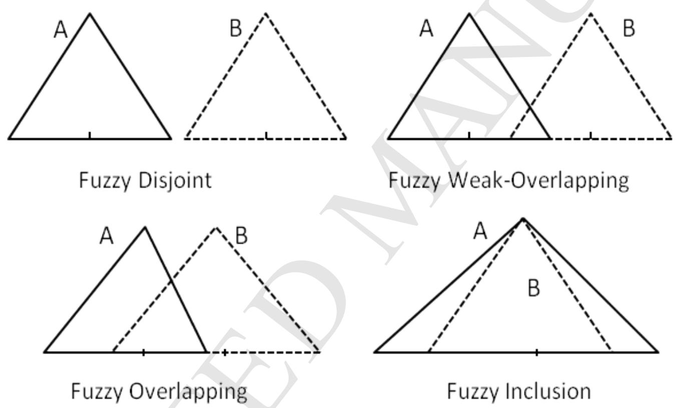


Figure 5: Possible topological situations for two TFNs

Definition 2. *Total dominance*
Let $y = [\underline{y}, \widehat{y}, \overline{y}] \subseteq \mathbb{R}$ and $y' = [\underline{y}', \widehat{y}', \overline{y}'] \subseteq \mathbb{R}$ be two triangular fuzzy numbers. y dominates y' totally or certainly, denoted by $y \prec_t y'$, iff:

$$\overline{y} < \underline{y}' \tag{16}$$

As shown in Figure 6, this dominance relation represents the fuzzy disjoint situation between two triangular fuzzy numbers and it imposes that the upper bound of y is strictly inferior than the lower bound of y'.

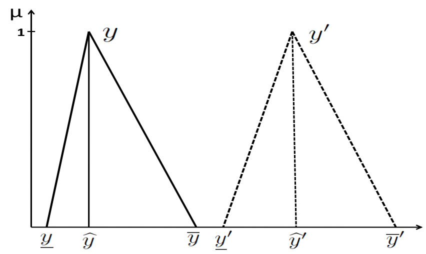


Figure 6: Total dominance

Definition 3. *Partial strong-dominance*
Let $y = [\underline{y}, \widehat{y}, \overline{y}] \subseteq \mathbb{R}$ *and* $y' = [\underline{y}', \widehat{y}', \overline{y}'] \subseteq \mathbb{R}$ *be two triangular fuzzy numbers. y strong dominates* y' *partially or uncertainly, denoted by* $y \prec_s y'$*, iff:*

$$(\overline{y} \geq \underline{y}') \wedge (\widehat{y} \leq \underline{y}') \wedge (\overline{y} \leq \widehat{y}') \tag{17}$$

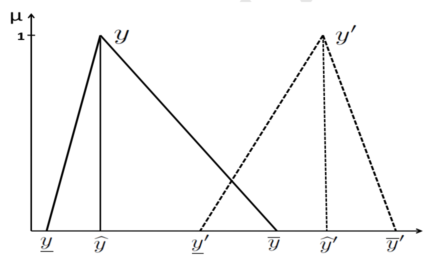


Figure 7: Partial strong-dominance

This dominance relation appears when there is a fuzzy weak-overlapping between both triangles. It imposes that firstly there is at most one intersection between them and secondly this intersection should not exceed the interval of their kernel values $[\widehat{y}, \widehat{y}']$ as shown in Figure 7.

Definition 4. *Partial weak-dominance*
Let $y = [\underline{y}, \widehat{y}, \overline{y}] \subseteq \mathbb{R}$ *and* $y' = [\underline{y}', \widehat{y}', \overline{y}'] \subseteq \mathbb{R}$ *be two triangular fuzzy numbers. y weak dominates* y' *partially or uncertainly, denoted by* $y \prec_w y'$*, iff we have:*
1. Fuzzy overlapping

$$\begin{gathered}[\underline{y} < \underline{y}' \wedge \overline{y} < \overline{y}'] \quad \wedge \\ [(\widehat{y} \leq \underline{y}' \wedge \overline{y} > \widehat{y}') \vee (\widehat{y} > \underline{y}' \wedge \overline{y} \leq \widehat{y}') \vee (\widehat{y} > \underline{y}' \wedge \overline{y} > \widehat{y}')]\end{gathered} \tag{18}$$

2. Fuzzy Inclusion

$$(\underline{y} < \underline{y}') \wedge (\overline{y} \geq \overline{y}') \tag{19}$$

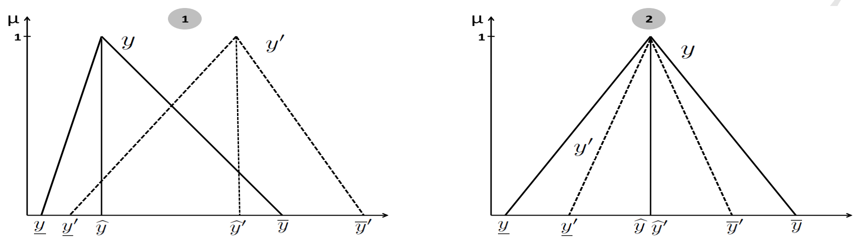


Figure 8: Partial weak-dominance

In this dominance relation, the two situations of fuzzy overlapping and inclusion may occur. In Figure 8-(1), where both numbers are overlapped, it is clear that y partially weak dominates y'. Yet, for case (2), where y' is included into y, a situation of incomparability is identified. Otherwise, the partial weak-dominance relation cannot discriminate all possible cases and leads often to some incomparable situations. In order to discriminate these cases, we suggest to use the middle value positions (the kernel or most plausible value) as an additional criterion of comparison. This may be formally defined by:

$$\widehat{y} - \widehat{y}' = \begin{cases} < 0, & y \prec_w y' \\ \geq 0, & y \text{ and } y' \text{ can be incomparable.} \end{cases}$$

Clearly, we remark that an incomparable situation is identified if we have $\widehat{y} - \widehat{y}' \geq 0$. At this level, the kernel criterion which consists in comparing the discard between both fuzzy triangles will be applied as follows:

$$y \prec_w y' \longleftrightarrow (\underline{y}' - \underline{y}) \leq (\overline{y}' - \overline{y})$$

It is easy to check that in the mono-objective case, we obtain a total pre-order between two triangular fuzzy numbers, contrarily to the multi-objective case, where the situation is more complex and it is common to have some cases of indifference.

3.2.2. Pareto dominance

Our goal now is to determine an optimal ordering on the set of fuzzy solutions, where each solution is represented by a vector of triangular fuzzy numbers. Thus, we propose to use the mono-objective dominance relations, defined previously, in order to rank separately the triangular fuzzy values of each objective function. Then depending to the dominance types found for all objectives, we define the Pareto dominance between the triangular fuzzy solutions.

Definition 5. *Strong Pareto dominance*
Let Y and Y' be two triangular fuzzy solutions. Y strong Pareto dominates Y', denoted by $Y \prec_{SP} Y'$ iff:

$$\begin{aligned} \forall i \in 1,\dots,n : [y_i \prec_t y'_i \vee y_i \prec_s y'_i] \;\; \vee \\ \exists i \in 1,\dots,n : [y_i \prec_t y'_i \vee y_i \prec_s y'_i] \wedge \forall j \neq i : [y_j \prec_s y'_j \vee y_j \prec_w y'_j] \end{aligned} \tag{20}$$

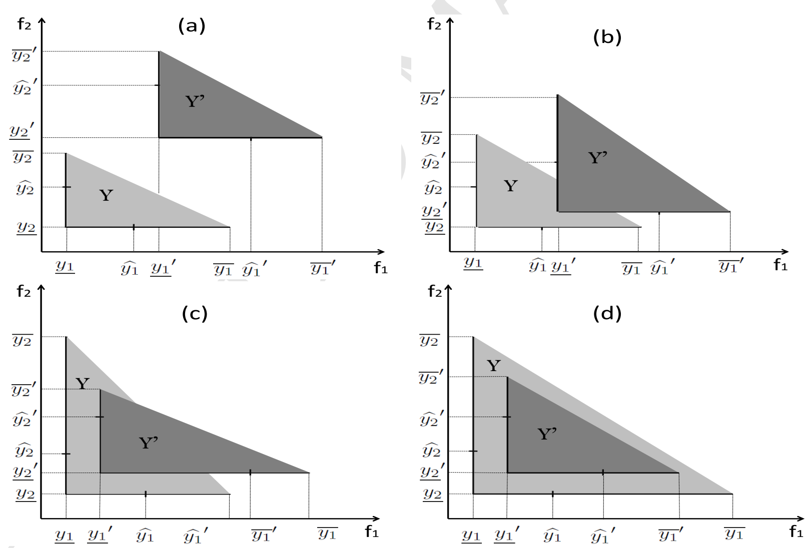


Figure 9: Pareto dominance relations

The strong Pareto dominance holds if either y_i total dominates or partial strong dominates y'_i in all the objectives (Figure 9-(a)), either y_i total or partial strong dominates y'_i in one objective and weak dominates it in another (Figure 9-(b)).

Definition 6. *Weak Pareto dominance*
Let $\overrightarrow{y}$ and $\overrightarrow{y}'$ be two triangular fuzzy solutions. $\overrightarrow{y}$ weak Pareto dominates $\overrightarrow{y}'$, denoted by $\overrightarrow{y} \prec_{WP} \overrightarrow{y}'$, iff:

$$\forall i \in 1, \ldots, n : y_i \prec_w y'_i \tag{21}$$

The weak Pareto dominance holds if y_i weak dominates y'_i in all the objectives (see Figure 9-(c)). Yet, a case of indifference (defined below) can occur if there is a weak dominance with inclusion type in all the objectives (see Figure 9-(d)).

Definition 7. *Case of indifference*
Two triangular fuzzy solutions are indifferent or incomparable, denoted by $\overrightarrow{y} \| \overrightarrow{y}'$, iff:

$$\forall i \in 1, \ldots, n : y_i \subseteq y'_i \tag{22}$$

3.2.3. Numerical Examples

We first present in Figure 10 three examples to give an idea of the usefulness and advantage of our mono-objective dominance by comparing it with other methods. Notice that, these examples can be considered as ones of well-defined cases of indiscrimination, that have failed to be resolved by some fuzzy ranking methods.

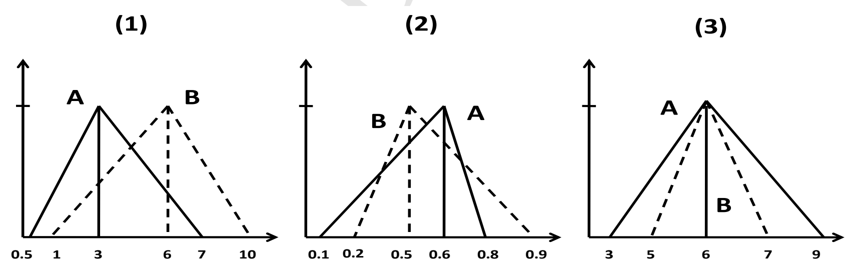


Figure 10: Three examples of triangular fuzzy numbers A and B

Example 1. *Consider the two triangular fuzzy numbers $A = [0.5, 3, 7]$ and $B = [1, 6, 10]$ in Figure 10-(1): The ranking order found by most of methods like Cheng's distance [33], Chu's index [34], Wang's centroid index [35] and kaufman's left and right scores [36], is $A \prec B$. By using our mono-objective dominance, it is easy to check that A partial weak dominates B: $A \prec_w B$ according to Equation 18 (Definition 4.1). Therefore, the ranking order in our case is the same as other tested methods ($A \prec B$).*

Example 2. *Consider the two triangular fuzzy numbers* $A = [0.1, 0.6, 0.8]$ *and* $B = [0.2, 0.5, 0.9]$ *in Figure 10-(2): By using some ranking methods such as Yao's signed distance [37], Chu's index [34] and Abbas's sign distance [38],* A *and* B *are indifferent i.e.* $A \approx B$*. This is the shortcoming of previous methods that rank two different fuzzy numbers equally. However, by applying our dominance method, we observe at the first step, that* A *and* B *are strongly overlapped. Thereby, the discrimination between these two numbers is not possible using Equation 18 (Definition 4-1), since the kernel-based criterion gives* $0.6 - 0.5 = 0.1 \geq 0$*. At this level, we apply the discard criterion* ($0.2 - 0.1 \leq 0.9 - 0.8$) *which leads to conclude that* A *partial weak dominates* B *and consequently* $A \prec B$*.*

Example 3. *Consider the two triangular fuzzy numbers* $A = [3, 6, 9]$ *and* $B = [5, 6, 7]$ *shown in Figure 10-(3): It is a common and very controversial problem, where an inclusion relation between the fuzzy numbers is defined. Yet, almost the majority of ranking methods such as [34, 35, 37, 38] failed to discriminate two fuzzy numbers having the same symmetrical spread, as for this example, it is deduced that* $A \approx B$*. Ezzati et al. [39] conclude that* B *dominates* A ($B \prec A$) *and consider this ranking order as a reasonable result since it agrees with human intuition. In fact, we can deduce intuitively that* B *has a greater tendency to be lower than* A*, according to its closeness to the most plausible value (the middle one). In other words, in order to estimate an unknown quantity in real-world situation, it is obvious that the interval estimation* $[5, 10, 15]$ *is better than* $[1, 10, 25]$ *(i.e. Saying that the quantity is around 10, between 5 and 15 units is more precise than saying that it is around 10, between 1 and 25). However, by applying our dominance method, we conclude that* B *partial weak dominates* A ($B \prec_w A$)*, since the discard criterion gives:* $5 - 3 > 7 - 9$*. Thus, we obtain the same rational result* $B \prec A$*.*

From the above examples, we may conclude that the proposed mono-objective dominance method can effectively rank two triangular fuzzy numbers and produces reasonable and intuitive results. The next example presents a comparison of our Pareto dominance relations with the interval-based dominance proposed by [9, 10].

Example 4. *Figure 11 illustrates 4 possible cases between a pair of solutions* S_1 *and* S_2 *in a two-dimensional objective space. For each case, we intend to* *determine the type of Pareto dominance relation between both solutions by applying:*

- *Limbourg's interval-based Pareto dominance [9], denoted by* $<_{IP}$.
- *Our Fuzzy Pareto dominance: denoted by* $\prec_{SP}$ *for strong Pareto dominance,* $\prec_{WP}$ *for weak Pareto dominance and* $\|$ *for case of indifference.*

In that sense, every solution in our case is represented by a triangular fuzzy shape (colore in light and dark gray), that is a vector of two TFNs (respectively for objectives f_1 *and* f_2*). For example, in Figure 11-(c1),* $S_1 = ([2,4,9][1,3,6])$ *and* $S_2 = ([8,10,13][5,9,12])$*. On the other hand, by applying the interval-based approach, every solution will be represented by a rectangular shape (bounding boxes with dotted lines), that is a vector of intervals as for example in (c1):* $S_1 = ([2,9][1,6])$ *and* $S_2 = ([8,13][5,12])$.

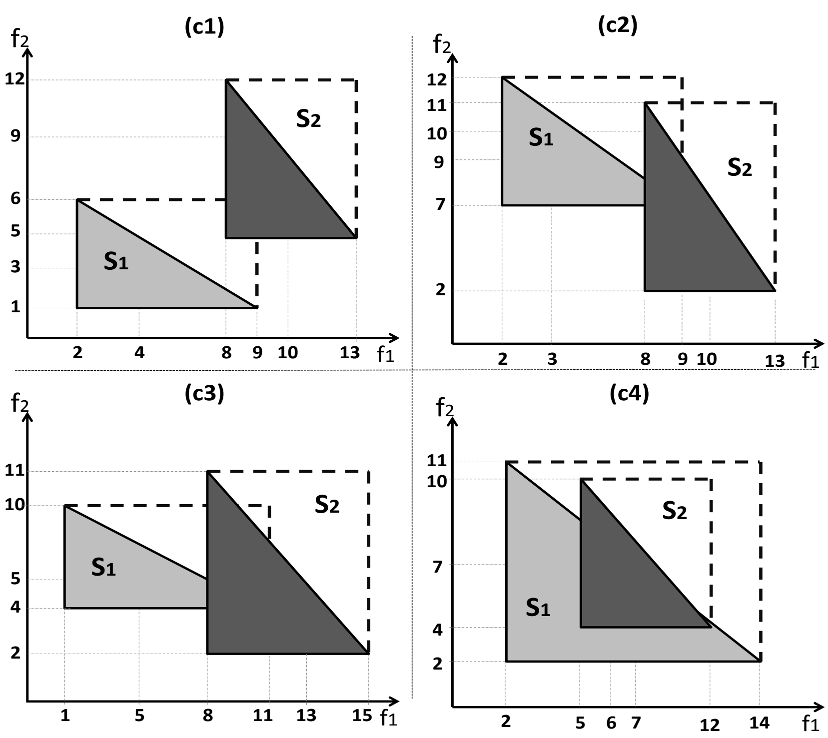


Figure 11: Pareto dominance examples

(c1) is the case where S_1 is lower than S_2 in both f_1 and f_2. By using the interval-based Pareto dominance, the lowest solution is preferred and thus $S_1 <_{IP} S_2$. In our triangular fuzzy setting, we can clearly deduce that $S_1 \prec_{SP}$, because S_1 strong dominates S_2 in all the objectives.

In cases (c2) and (c3), S_1 is better than S_2 in one objective but greater than it or incomparable in the other. In the case of interval-based approach, authors state that these cases depend heavily on the decision maker choice. Otherwise, this latter can interpret them as situations of indifference or use a preference-based decision. Yet, by using the fuzzy Pareto dominance, we may conclude that $S_1 \prec_{WP} S_2$ in both cases, because S_1 strong dominates S_2 in f_1 and weak dominates it in f_2. For instance in (c2), we have $S_{11} = [2, 3, 9] \leq_s S_{21} = [8, 10, 13]$ and $S_{12} = [7, 9, 12] \leq_w S_{22} = [2, 10, 11]$ with respect to their kernel values comparison, i.e., $(9 - 10) \leq 0$.

Likewise, (c4) represents always a case of incomparability because one solution encloses the other. Thus, we have $S_1 \| S_2$.

From these examples, we remark that our fuzzy Pareto dominance can successfully discriminate incomparable solutions in some critical decision cases. This is mainly due to the flexibility that our approach offers by giving us the choice between weak and strong dominance relation. This may also be explained by the use of the kernel values of our triangular fuzzy distributions as comparison criteria in the cases of indifference. In general, we may conclude that our fuzzy Pareto approach offers a better classification and more accurate knowledge comparing with the interval-based approach.

3.3. Extended MOEAs for fuzzy MOPs

The issue now is to design optimization algorithms for handling any MOP with fuzzy-valued objectives. The basic idea is to exploit the common steps from the general MOEA scheme (described in Section 2.1.2) and then adapt them to the fuzzy context. As mentioned before, the overall MOEA process is most strongly influenced by three main components: Fitness assignment, Diversity preservation and Elitism/Archiving. It is therefore necessary to adapt each of theses components in order to enable such algorithms working in a fuzzy context.

First, knowing that in a MOP with fuzzy-valued objectives, solutions are usually affected by fuzziness. In addition, we remember that in our case the encoding type of objective functions is a triangular fuzzy shape (see Equation 13 in Section 3.1). Consequently, every solution is composed of a vector of triangular fuzzy numbers (TFNs). This kind of fuzziness will be propagated step by step in the MOEA optimization proces. In the initial step, a population with a set of triangular fuzzy solutions is randomly sampled. These solutions should then be evaluated and ranked based on fitness assignment strategy such as dominance-based evaluation. However as the classical dominance concepts can only be applied between exact (or crisp) solutions, we propose to use our new fuzzy Pareto dominance relations introduced in the previous section. Similarly, if another fitness strategy is used, we should adapt it to the fuzzy setting.

Next, once all fuzzy solutions are ranked, the diversity preservation will be performed in the selection step. Usually, the local density is estimated by measuring distances between neighbor solutions. Yet, a standard distance measure is typically applied between exact values. Thus, in our case, we need clearly a specific measure for computing the distance between fuzzy values. Thereafter, the archiving tool must also be able to store the selected fuzzy non-dominated solutions. The remaining steps still unchanged because they are independent to the type of solutions.

Following these remarks, any MOEA can be extended to our fuzzy context by integrating these modifications into the search process. In this work, we illustrate the fuzzy extension on the most popular algorithms: NSGAII [40], SPEA2 [41] and IBEA [13]. These algorithms have proved to be very powerful tools for multi-objective optimization. Due to their population-based nature, they are able to generate multiple optimal solutions in a single run with respect to the good convergence and diversification of obtained solutions. Table 1 summarizes the differences between them at the main steps of unified scheme [42]. In the following, we describes in details the fuzzy extension of each algortithm.

Table 1: NSGAII, SPEA2 and IBEA components

	NSGAII	SPEA2	IBEA
Fitness	Pareto dominance	Pareto dominance	Quality indicator
Diversity	crowding-distance	nearest neighbor	-
Selection	binary tournament	binary tournament	binary tournament
Elitism	-	fixed size archive	-
Replacement	elitist	generational	elitist
Stopping	nb of generations	nb of generations	nb of generations

3.4. E-NSGAII

The *Non-dominated Sorting Genetic Algorithm II* (NSGA-II) [40] is one of the most popular PMOEAs for solving MOPs. Its popularity is often associated to its low computational complexity, explicit technique for diversity preserving and the no-archive strategy. Contrary to its original version, it does not use a fitness sharing mechanism for diversity preserving, but rely on a crowded comparison procedure.

To extend NSGAII, we propose at the first stage to replace the standard Pareto by our new fuzzy Pareto dominance defined in the previously. This modification allows to ensure the fitness assignment ranking in a fuzzy setting. At the second stage, we provide an adaptation of the diversity preserving technique where a crowded-comparison procedure is applied. This procedure is based on a *Crowding distance* CD that serves to get a discrimination of solutions having the same rank level. Formally, the CD of a solution is the sum of its individual objectives' distances, that in turn are the differences between the solution and its closest neighbors.

$$CD(i) = \sum_{i=1..n} (f_i(i+1) - f_i(i-1))/(f_i^{max} - f_i^{min}) \ \ s.t. \ \ i \in F \tag{23}$$

where $f_i(i+1)$ and $f_i(i-1)$ are the neighbor objective values of $i-$th objective, n is the number of objectives ,f_i^{max} and f_i^{min} are respectively the population maximum and minimum objective values and F is the $i-$th front to which solutions are associated. However, this distance cannot remain unchanged in our fuzzy context since it depends directly on the differences between objective values. Then as our objective functions are vectors of triangular fuzzy numbers (or TFNs), the distance measure must be adapted to fuzziness. Thus, we simply propose to approximate these objectives by computing their expected values before applying the *Crowding distance*. Indeed, the distance between each pair of objective values will be then substituted by the differences between the corresponding expectations. Formally, the *Expected value* E of a given triangular fuzzy number $y_i = [\underline{y_i}, \widehat{y}_i, \overline{y_i}]$ is calculated using the following formula [35]:

$$E(y_i) = (\underline{y_i} + 2 \times \widehat{y}_i + \overline{y_i})/4 \tag{24}$$

Notice that, we do not need a fuzzy extension of the elitism technique because there is no archive. Finally, the two simple refinements at fitness assignment and diversity preserving steps are incorporated into the search process of NSGAII. We describe E-NSGAII in Algorithm 1.

Algorithm 1: E-NSGAII

Input : Population size N
Maximum number of generations G
Output: Pareto set approximation
begin
1.*Initialization*. create a random population P of N fuzzy solutions;
repeat
2.*Fitness Assignment*. rank all solutions using fuzzy Pareto dominance;
3.*Environmental Selection*. select the non-dominated fuzzy solutions based on their expected crowding values and copy them in an external population P';
if *size of* P' *exceeds* N **then**
add the least crowded solutions to P';
else if *size of* P' *is less than* N **then**
set P' with dominated solutions;
else
the environmental selection is completed;
end
4.*Elitism*. update $P \cup P'$
until *5. Stopping condition. Number of generations* $> G$ *is satisfied*
6.*Mating Selection*. perform a binary crowded tournament selection to select parents from P';
7.*Variation*. apply crossover and mutation operators to the mating pool;
8.*Replacement*. replace old population by the resulting offspring population.
end

3.5. E-SPEA2

The *Strength Pareto Evolutionary Algorithm* (SPEA2) is an improved version of SPEA algorithm, where a mixed strategy of fitness ranking is adopted [41]. Otherwise, SPEA2 uses a Pareto-based fitness assignment which incorporates both rank and count dominance strategies.

Similarly to the previous algorithm, we first suggest to integrate our fuzzy Pareto dominance relations in the SPEA2 fitness assignment strategy. Then as detailed before, SPEA2 uses a nearest neighbor density estimation technique which allows a more precise guidance of the search process. This technique consists in calculating for each solution the Euclidean distance to its k^{th} nearest neighbor and then in adding the reciprocal value to the fitness vector. Formally, the *Euclidean distance EU* between two solutions (i.e.

deterministic objective vectors) y_1 and y_2 is given by:

$$EU(y_1, y_2) = \sqrt{\sum_{i=1}^{n} (y_{1i} - y_{2i})^2} \tag{25}$$

However, as *Euclidean distance* should be applied only between two exact vectors and in our case the solutions are vectors of TFNs, we should replace this latter by a specific fuzzy distance. Therefore, we choose to use the so-called *Bertoluzza metric* [43] in order to compute the distance between two fuzzy solutions based on α-cut principle. More precisely, given two vectors of TFNs $y = (y_1, ..., y_n)$ and $y' = (y'_1, ..., y'_n)$ such that $y_i = [\underline{y_i}, \widehat{y_i}, \overline{y_i}]$ and $y'_i = [\underline{y_i}', \widehat{y_i}', \overline{y_i}']$, the *Bertoluzza metric* is first applied to compute distances between every pair of fuzzy numbers y_i and $y - i'$. Then, we propose to compute the weighted mean of the overall distances $d(y_i, y'_i)$ in order to estimate the final distance between both vectors $D(y, y')$. Formally, it is given by:

$$d_\theta(y_i, y'_i) = \sqrt{\int_0^1 (mid(y_{i_\alpha}) - mid(y'_{i_\alpha}))^2 + \theta\ (spr(y_{i_\alpha}) - spr(y'_{i_\alpha}))^2 d_\alpha} \tag{26}$$

where y_{i_α} denotes the α-cut of y_i defined as as an α level set (or bijection) associating for any $\alpha \in [0, 1]$ a bounded interval $[\underline{y_{i_\alpha}}, \overline{y_{i_\alpha}}]$, $mid(y_{i_\alpha}) = \frac{1}{2}(\underline{y_{i_\alpha}} + \overline{y_{i_\alpha}})$ denotes the midpoint of y_{i_α}, $spr(y_{i_\alpha}) = \frac{1}{2}(\overline{y_{i_\alpha}} - \underline{y_{i_\alpha}})$ is the spread (or radius) of y_{i_α} and $\theta \in [0, 1]$ is a parameter that allows us to weight the effect of the deviation between spreads. For a sake of simplicity, we will consider in our case the 0-cut level ($\alpha = 0$) where $y_{i_0} = [\underline{y_{i_0}}, \overline{y_{i_0}}] = [\underline{y_i}, \overline{y_i}]$ is the topological support of y_i and we choose a parametrization with the value $\theta = \frac{1}{2}$.

By using this fuzzy distance, the distance to each k-nearest fuzzy solution can be estimated in order to select the well-distributed solutions. Thereafter, unlike the NSGAII algorithm, SPEA2 uses a regular population and also an external storage, the so-called archive. Thus, we need to extend this archive to the triangular fuzzy space in order to enable it keeping the best triangular solutions during the optimization process. In next steps, namely *mating selection*, *variation* and *replacement* remain unchanged in our fuzzy context. The above extensions are integrated into the search process of SPEA2 and leads to a new version, denoted E-SPEA2 and described in Algorithm 2.

Algorithm 2: E-SPEA2

Input : Population size N
Triangular archive A
Maximum number of generations G
Output: Pareto set approximation
begin
1.*Initialization*. create a random population P of N fuzzy solutions and create an empty triangular archive A of fixed size M;
repeat
2.*Fitness Assignment*. rank solutions using fuzzy Pareto dominance;
3.*Environmental Selection*. copy all non-dominated solutions from P to the triangular archive A;
if *size of* A *exceeds* M **then**
A is pruned by means of a clustering procedure;
else if *size of* A *is less than* M **then**
fill A with best dominated solutions;
else
the environmental selection is completed;
end
4.*Elitism*. update A;
until *5. Stopping condition. Number of generations* $> G$ *is satisfied;*
6.*Mating Selection*. perform a binary tournament selection with replacement on A to fill the mating pool;
7.*Variation*. apply crossover and mutation operators to the mating pool;
8.*Replacement*. replace old population by the resulting offspring population.
end

3.6. E-IBEA

The *Indicator Based Evolutionary Algorithm* (IBEA) [13] is a third generation of MOEAs, where selection is based on solution contribution to a certain quality indicator. Its basic idea is to establish a pairwise comparison of solutions by using a binary quality indicator as fitness assignment strategy. Different indicators can be used for such a purpose such as the binary Hypervolume indicator chosen in our case.

More precisely, the algorithm computes the indicator value I of each pair of individuals and assigns to each individual i a fitness value f_i, to be maximized, measuring the loss in quality if it's removed from the current population P.

$$f_i(i) = \sum_{j \in P \setminus \{i\}} -e^{-I(i,j)/k} \tag{27}$$

where k is a scaling factor which has to be set in advance. The purpose of the exponential is to amplify the differences between dominated and non-dominated individuals.

The extension of IBEA algorithm to our fuzzy setting is more simple than the previous algorithms, because all the dominance relations are generalized by means of the quality indicator and also no diversity preservation technique is required. Indeed, knowing that almost all quality indicators are dedicated only to crisp approximations and as in our case every individual is represented by a triplet of values, we propose to compute separately the fitness of the three values. In other words, we assign to the lower, middle and upper values of every solution, a unique scalar value representing its quality or fitness. Next, we suggest to use these fitness values to compare and evaluate a pair of individuals. In fact, the worst individual (with the lowest fitness) is iteratively removed and the fitness values of the remaining individuals are updated. This process, of deleting one by one the worst solutions, is iterated until the reach of the required population size. Finally, as IBEA is sometimes sensitive to the parameter k used to scale the values of indicators, we use adaptive scaling values of k, starting from a value greater than 0 until reaching the best final k value. The extended version of IBEA is given in Algorithm 3.

Algorithm 3: E-IBEA

Input : Population size N
Maximum number of generations G
Fitness scaling parameter k
Output: Pareto set approximation
begin
1.*Initialization*. create a random population P of N fuzzy solutions;
repeat
2.*Fitness Assignment*. scale indicator values and then use them to assign fitness:
- compute the indicator value for each vector of lower, middle and upper bound of the triangular fuzzy objective;
- use adaptive scaling k for each indicator value in interval $[0, 1]$;
- assign scaled values as fitness of fuzzy solution in P.
3.*Environmental Selection*. select the best fuzzy solutions based on their fitness values and copy them into an external population P':
if *size of* P' *is less than* N **then**
- determine the worst (smallest) fitness value of fuzzy individuals;
- remove the individual with worst fitness from the population P;
- set P' with the remaining individuals and update their fitness values;
else
the environmental selection is completed
end
4.*Elitism*. update $P \cup P'$
until *5. Stopping condition. number of generations* $> G$ *is satisfied*
6.*Mating Selection*. perform a binary crowded tournament selection to select parents from P;
7.*Variation*. apply crossover and mutation operators to the mating pool;
8.*Replacement*. replace old population by the resulting offspring population.
end

4. Application on a multi-objective Vehicle Routing Problem

The basic VRP consists in finding a set of optimal routes (with minimum total cost) to be covered by a fleet of identical vehicles in order to serve a number of geographically distributed customers, subject to side constraints such that: (i) all routes start and end at a central depot; (ii) each customer is visited exactly once by only one vehicle. Several variants of this basic problem were extensively studied in the literature, see [44]. However, almost all of them were originally proposed for modeling deterministic routing problems, while only few variants were devoted for problems with uncertain data such as uncertain customers demand, location or travel cost.

4.1. Problem definition

In this paper, we focus on a well-known variant of VRP, the so-called Multi-objective VRP with Time Windows and Uncertain Demands (MO-VRPTW-UD). In this problem, all the data are deterministic excepting the customer demands which are uncertain, meaning that the actual demand is only known when the vehicle arrives at the customer location. The main constraints of MO-VRPTW-UD are the following:

- ***Time windows constraint*** imposes that each customer will be served within its time window which represents the interval of time planned for receiving the vehicle service. This means that, if the vehicle arrives too soon, it should wait until the arrival time of its time window to serve the customer, while if it arrives too late (after the fixed departure time), wasted cost in term of tardiness time appears.

- ***Vehicle capacity constraint*** imposes that the total of customer demands for any route must not exceed the limited vehicle capacity. Moreover, each vehicle with a limited capacity must deliver goods to the customers with the minimum transportation costs in term of traveled distance. Yet, if the capacity constraint of a vehicle is not satisfied, the delivery fails and causes wasted costs.

In our case, two objectives have to be minimized in this problem which are the total traveled distance and total tardiness time. Besides, the customer demands are assumed to be triangular fuzzy numbers. Then, as uncertainty in the inputs leads often to the uncertainty in the outputs, the objective functions in our problem will be consequently affected by the fuzziness of customer demands.

Figure 12 illustrates an example of VRPTW-UD problem, with a central depot, 3 vehicles $(V1, V2, V3)$ having a maximum capacity $Q = 10$ and a set of 8 customers represented by nodes. Each customer $i = 1 \ldots 8$ has an uncertain demand expressed by a triangular fuzzy number $dm = [\underline{dm}, \widehat{dm}, \overline{dm}]$ (For example, the fuzzy demand of customer 1 is $dm_1 = [2, 7, 11]$). Yet, in this case, we cannot often determine if the capacity constraint is satisfied or not, since the customer' demands are fuzzy values. For example, consider the customer 7 with fuzzy demand $dm_7 = [8, 10, 13]$, we cannot check if dm_7

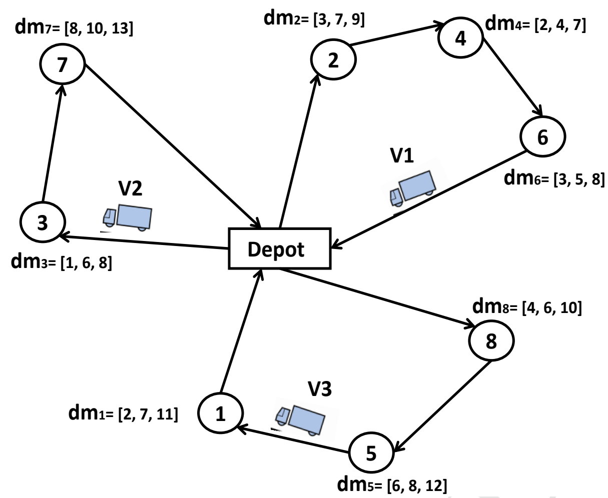


Figure 12: Example of MO-VRPTW-UD

is lower, equal or higher than $Q = 10$ in order to estimate the transportation costs in terms of time spent and traveled distance. At this level, we propose to verify separately the capacity constraint satisfaction for each value of the triangular demand. Then, based on the three situations found for the triplet of demand values, the traveled distance will be calculated three times and so obtained as triangular fuzzy number $D = [\underline{D}, \widehat{D}, \overline{D}]$. In addition, knowing that the travel time for a given route depends mainly on its traveled distance to serve customers and as the distance in our case is a triangular fuzzy variable, the time variable will be also disrupted by this fuzzy form and consequently the tardiness time will be also obtained as triangular fuzzy number $T = [\underline{T}, \widehat{T}, \overline{T}]$.

4.2. Experimental design

The extended E-SPEA2, E-NSGAII and E-IBEA algorithms were applied on the MO-VRPTW-FD problem and implemented in C++ using the ParadisEO framework, especially with the MOEO module under Linux [21]. For a fair comparison and evaluation, all tested algorithms share the same base parameters such as the chosen variation operators, the initial population, the random seed, etc.

4.2.1. Benchmarks

To the best of our knowledge, there is no common benchmark available in the literature for stochastic VRPs. Hence, to evaluate our model with fuzzy customer demands, we create a new benchmark for MO-VRPTW-FD by adapting the problem instances provided by [45], which is a popular reference for evaluating most methods designed for VRPTW. Our interest is focused on the larger benchmark of 100-customers that includes a total of 56 different problems. Each problem is composed of 100 geographically distributed customers, a unique depot and a fleet of homogenous vehicles having a same capacity Q=60 and a constant speed fixed to 1 (distance unit/time unit). The travel times between customers is proportional to the corresponding *Euclidean distances* between them. According to the customer distribution and time-windows size, these problems are classified into 6 categories, namely C1, C2, R1, R2, RC1, RC2, where (C) corresponds to a distribution of customers in clusters, (R) corresponds to a random distribution of customers and (RC) corresponds to customers located partly in clusters and partly randomly. The notations 1 and 2 indicate the size of the time windows (1 for closely time window and 2 for large time window).

Thereafter, we propose to adapt the 56 Solomon's problem to our fuzzy context by applying the methodology described in Figure 13:

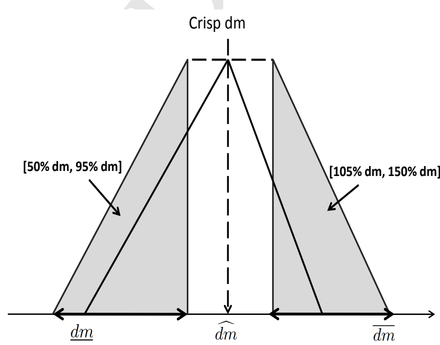


Figure 13: Fuzzy sampled demand

The basic idea of this methodology is to generate for each deterministic instance its fuzzy sampled version, in which each crisp demand values is replaced by its corresponding fuzzy value. The kernel value ($\widehat{dm}$) for each triangular fuzzy demand dm is firstly kept the same as the crisp demand value dm_i of the current instance. Then, the lower ($\underline{dm}$) and upper ($\overline{dm}$)

bounds of this triangular fuzzy demand are uniformly sampled at random in the intervals $[50\%dm, 95\%dm]$ and $[105\%dm, 150\%dm]$, respectively. The new fuzzy instances are labeled with names preceded by the word "Fuzz" like Fuzz-C101, Fuzz-R101, Fuzz-RC101, etc.

4.2.2. Experimental protocol

To assess the performance of our algorithms, we have conducted a set of experiments that can be divided into 2 tests:

(i) First, we aim to examine the ability of new algorithms to tolerate fuzziness versus their crisp versions.

(ii) Second, we aim to compare the quality of generated front approximations of proposed algorithms.

In these experiments, we have used the 56 fuzzy instances sampled uniformly at random from the classical Solomon's instances. Each fuzzy instances is tested on the following six algorithms executed 30 times: C-SPEA2, C-NSGAII, C-IBEA denote respectively the crisp algorithms considering only the core (i.e. the most plausible value) of fuzzy demands, E-SPEA2, E-NSGAII and E-IBEA are the extended algorithms considering the triangular fuzzy representation of demands (i.e. the triplet of values) and propagating fuzziness to the objective functions. For each algorithm, a set of 30 runs per instance was performed with random initial populations of size=100 evolving across 1000 generations; crossover rate of 0.8 and mutation rate of 0.1; final scaling parameter $k = 0.05$. Thereafter, with 6 algorithms tested on 56 instances and repeated 30 runs, we have done 6×56×30=10080 runs. Hence, we obtained, for every test instance, 30 sets of optimal solutions that represent Pareto front approximations. Each solution shows the minimum total traveled distance and total tardiness time for an efficient vehicles route. However, the solutions obtained for the crisp algorithms are represented by a set of exact numbers, while for the new extended algorithms the solutions returned are vectors of triangular fuzzy numbers.

Examples of front approximations found for the problem instance Fuzz-C101 using each of algorithms E-SPEA2, E-NSGAII and E-IBEA are shown respectively in Figures 14, 15 and 16. The illustrated fronts represent a set of triangular fuzzy solutions. For instance, the bold triangle in Figure 14 represents a solution with minimum total distance (the green side) equal to $[2413, 2515, 2623]$ and total tardiness time (the red side) equal to $[284312, 295280, 315322]$.

In order to evaluate and compare the quality of the generated front approximations for every test instance, we use:

- *Unary hypervolume indicator* I_H to compare the approximations quality of each extended algorithm with the classical one. This indicator is to be maximized.

- *Binary hypervolume difference* I_H^- *and additive* ϵ*-indicator* $I_{\epsilon+}$ to assess the performance of proposed algorithms. Both indicators are to be minimized.

Yet, as these indicators are performed only on exact approximation samples, we propose to defuzzify the triangular fuzzy solutions of E-SPEA2, E-NSGAII and E-IBEA algorithms by computing their expected values (see Equation 24). As the computation of hypervolume and epsilon indicators usually require a reference set $Z_n^\star$ (or a reference point z_{ref} for the unary case), we propose to follow the experimental steps given in [46]: We first consider $Z_n^\star$ as the set of non-dominated solutions extracted from the union of all front approximations. Then, we compute $z^{\max} = (z_1^{\max}, z_2^{\max})$, where $z_1^{\max}$ and $z_2^{\max}$ denote the upper bounds of both objective functions in the whole non-dominated fronts. The reference point used for the unary hypervolume I_H is fixed to $z_{ref} = (1.05 \times z_1^{\max}, 1.05 \times z_2^{\max})$.

Subsequently, by applying the quality indicators with respect to the reference set (or reference point), we transform our solutions to samples with I-values scalars. In this way, we reach a single I-value for each test run per algorithm. For instance, according to the maximum hypervolume value found with the unary hypervolume metric, we can evaluate the quality of every algorithm outputs (i.e. higher I_H value indicates a better front approximation). Thereby, by analyzing the change in the set of I_H values provided for 30 runs per test instance, we can realize whether the extended algorithm (E-SPEA2, E-NSGAII and E-IBEA) are capable to tolerate the fuzziness comparing with their crisp versions (C-SPEA2, C-NSGAII and C-IBEA).

Afterwards, the binary I_H^- and $I_{\epsilon+}$ indicators are applied to assess the performance of proposed algorithms (w.r.t the reference set). Similarly, since 30 runs per algorithm have been performed, we obtain 30 hypervolume differences and 30 epsilon measures for each test instance. Once all these I-values are computed, we need to use a statistical analysis in order to compare them and so obtain valid statements about their quality. Since the algorithms share the same parameters for all the runs such as the initial population, the random seed, the variation operators, etc, the resulted approximations can be considered as *matched samples*. To this end, we use the Wilcoxon-signed rank test with a P-value=0.5% for a pairwise comparison between them. Consequently, for every test instance and according to the indicator under consideration (i.e. I_H^- or $I_{\epsilon+}$), this statistical test indicates if the approximation samples obtained by a given algorithm are significantly better than the ones of another algorithm, or if there is no significant difference between both. Notice that, all the experimental tests have been conducted using the PISA [1] performance assessment tool suite [47].

4.2.3. Computational results

Figures 17, 18 and 19 summarize the unary hypervolume results for the six implemented algorithms by using box-plots, such that each box presents 30 hypervolume values from the 30 runs of a corresponding algorithm on one test instance. Then as we have a total of 56 tested fuzzy instances, we present in these figures only the box-plots for 12 sampled fuzzy instances such as: *Fuzz-C101*, *Fuzz-C104*, *Fuzz-C201*, etc. The box-plots for other instances are completely similar and exhibit the same trend.

More precisely, by examining the behaviour of plotted boxes between the accumulated hypervolume values, we can assert the quality of our extended algorithms E-SPEA2, E-NSGAII, E-IBEA and compare them with the crisp algorithms C-SPEA2, C-NSGAII and C-IBEA. Taking a look at the Figure 17, we can intuitively compare the C-SPEA2 and E-SPEA2 algorithms based on their boxes of hypervolume values. In fact, it is not difficult to realize that the boxes of crisp C-SPEA2 are very large, in the sense that their hypervolume values vary from instance to instance (approximately from 0.3 to 1.1). Also, we can remark that the E-SPEA2 boxes are better (higher) than those of C-SPEA2, because they are less variable (varying slightly from 0.6 to 1.2) and look identical for all the illustrated instances.

In the same way, a comparison between the C-NSGAII and E-NSGAII algorithms is shown in Figure 18. Indeed, from the illustrated box-plots, we can easily observe that the boxes of the crisp algorithm are larger than those of the extended one. Otherwise, the E-SPEA2 has high and less dispersed hypervolume values varying from 0.5 to 1.2, whereas the C-NSGAII has hypervolume values varying from 0.2 to 1.1.

Furthermore, according to the box-plots in Figure 19, we can observe that there is no significant differences between the ”largeness” or variation

[1] http://www.tik.ee.ethz.ch/pisa/assessment.html

of C-IBEA and E-IBEA boxes. Indeed, the hypervolume values of C-IBEA (varying between 0.4 and 1.1) are not very large and near to the spread of E-IBEA hypervolume values (varying between 0.5 and 1.1). This could be explained by the fact that both algorithms already use the hypervolume indicator as fitness assignment strategy.

These remarks leads us to conclude that for all the sampled fuzzy instances, the new algorithms E-SPEA2, E-NSGAII and E-IBEA are less sensitive to fuzziness and so converge better than their crisp versions C-SPEA2, C-NSGAII and C-IBEA respectively. This may be explained by the fact that taking into account all the three vertices of a triangular fuzzy demand instead of only consider the most plausible demand in the crisp algorithms, provide more accurate approximations and consequently a better estimate of the algorithm quality.

Table 2 and 3 present the performance assessment of our proposed algorithms in solving the MO-VRPTW-FD. These algorithms are compared on 10 random selected fuzzy instances with respect to both binary I_H^- and $I_{\epsilon+}$ indicators and using the Wilcoxon-signed rank test (with a P-value less or equal to 0.05). For each test instance, either the algorithm located at a specific row significantly dominates the algorithm located at a specific column ($\prec$), either it is significantly dominated ($\succ$) or there is no significant difference between both ($\equiv$).

Observing the obtained results, both algorithms E-SPEA2 and E-NSGAII concurrently outperform E-IBEA, with respect to both I_H^- and $I_{\epsilon+}$, for all the tested instances. This may be due to the fact that E-IBEA is more sensitive to the parameters of indicators, while both E-SPEA2 and E-NSGAII are largely independent from the tested indicators. Although the difference between the structure of E-SPEA2 and E-NSGAII algorithms is small (i.e. Both are based on Pareto-dominance approach to guide their search process), E-SPEA2 has usually better convergence to the global Pareto front than E-NSGAII. Notice that, the size and inputs data of sampled instances are not the same; this is what led to some exceptions and different results.

In fact, as shown in the tables, E-SPEA2 performs better than E-NSGAII algorithm for almost all the instances, excepting for some instances: *Fuzz-RC204*, *Fuzz-R110* and *Fuzz-RC208*, where there is no significant difference between both algorithms E-SPEA2 and E-NSGAII algorithms. Another exception is for the *Fuzz-C103* and *Fuzz-RC106* instances, where the E-NSGAII obtained better results than E-SPEA2 with respect to the $I_{\epsilon+}$ indicator. Therefore, E-SPEA2 is clearly the most useful for fuzzy sampled instances since it outperforms the two other algorithms. On the other hand, in terms of running time, E-NSGAII is the fastest algorithm (i.e. requites approximately between 3 and 8 seconds) and this seems natural because it doesn't involve expensive calculations (as in E-IBEA that yields between 5 and 17 seconds) or archive procedure (as in E-SPEA2 that yields between 3 and 13 seconds).

5. Conclusion

Through this work, we have contributed to the design of a generic approach for combinatorial multi-objective problems with fuzzy data, especially expressed by means of triangular fuzzy numbers and propagated to the objective functions. Our major contributions is two-fold: First, we have proposed a novel Pareto approach for ranking the fuzzy outcomes generated in our case. Second, we have introduced a fuzzy extension of three well-known multi-objective evolutionary algorithms. The usefulness of proposed algorithms was illustrated through the resolution of a practical VRP problem and their performance assessment was validated by means of some experimental tests. The computational results were straightforward and encouraging for multi-objective problems confronted with fuzziness. Notice that our implementation are added into the ParadisEO-MOEO platform under Git [2].

As future work, we intend to make a robustness study to analyze in more detail the obtained solutions. We also intend to extend the well-known multi-objective performance indicators (i.e., Hypervolume indicator) to our fuzzy context. Finally, it would be interesting to validate the proposed approach for different fuzzy multi-objective routing problems, in which the fuzziness is expressed by other shapes like trapezoidal fuzzy numbers.

[2] https://gforge.inria.fr/projects/paradiseo/

Table 2: Algorithms comparison according to the I_H^- indicator

Instances	Algorithms	E-SPEA2	E-NSGAII	E-IBEA
Fuzz-C201	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R201	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-C204	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R204	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-RC204	E-SPEA2	-	$\equiv$	$\prec$
	E-NSGAII	$\equiv$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-C103	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R110	E-SPEA2	-	$\equiv$	$\prec$
	E-NSGAII	$\equiv$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R208	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-RC106	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-RC208	E-SPEA2	-	$\equiv$	$\prec$
	E-NSGAII	$\equiv$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-

Table 3: Algorithms comparison according to the $I_{\epsilon+}$ indicator

Instances	Algorithms	E-SPEA2	E-NSGAII	E-IBEA
Fuzz-C201	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R201	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-C204	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R204	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-RC204	E-SPEA2	-	$\equiv$	$\prec$
	E-NSGAII	$\equiv$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-C103	E-SPEA2	-	$\succ$	$\prec$
	E-NSGAII	$\prec$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R110	E-SPEA2	-	$\equiv$	$\prec$
	E-NSGAII	$\equiv$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-R208	E-SPEA2	-	$\prec$	$\prec$
	E-NSGAII	$\succ$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-RC106	E-SPEA2	-	$\succ$	$\prec$
	E-NSGAII	$\prec$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-
Fuzz-RC208	E-SPEA2	-	$\equiv$	$\prec$
	E-NSGAII	$\equiv$	-	$\prec$
	E-IBEA	$\succ$	$\succ$	-

References

[1] C. A. C. Coello, G. B. Lamont, D. A. Van-Veldhuisen, Evolutionary algorithms for solving multi-objective problems, Vol. 5, Springer Science and Business Media, 2007.

[2] Y. Collette, P. Siary, Multiobjective Optimization Principles and Case Studies, Springer Science & Business Media, 2003.

[3] K. Deb, Multi-objective optimization, Search methodologies, Springer, 2014, pp. 403–449.

[4] J. Steele, Probability Theory and Combinatorial Optimization, Siam, 1997.

[5] L. Zadeh, Fuzzy sets, Information and control, 1965.

[6] G. Goncalves, T. Hsu, J. Xu, Vehicle routing problem with time windows and fuzzy demands: an approach based on the possibility theory, Advanced Operational Management, Inderscience (2009) 312–330.

[7] L. Paquete, T. Stutzle, Stochastic local search algorithms for multiobjective combinatorial optimization, Handbook of Approximation Algorithms and Metaheuristics (2007).

[8] J. Teich, Pareto-front exploration with uncertain objectives, Evolutionary Multi-Criterion Optimization (2001) 314328.

[9] P. Limbourg, Multi-objective optimization of problems with epistemic uncertainty, Evolutionary Multi-criterion Optimization (2005) 413–427.

[10] P. Limbourg, E. Daniel, An optimization algorithm for imprecise multi-objective problem functions, Evolutionary Computation (2005) 459–466.

[11] O. Bahri, N. B. Amor, E.-G. Talbi, New pareto approach for ranking triangular fuzzy numbers, in: Springer (Ed.), International Conference on Information Processing and Management of Uncertainty (IPMU'14), 2014a, pp. 264–273.

[12] O. Bahri, N. B. Amor, E.-G. Talbi, Optimization algorithms for multi-objective problems with fuzzy data, in: IEEE (Ed.), Computational Intelligence in Multi-Criteria Decision-Making (MCDM'14), 2014b, pp. 194–201.

[13] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, in: 8th International Conference on Parallel Problem Solving from Nature, Springer, Birmingham UK, 2004, pp. 832–842.

[14] E.-G. Talbi, Metaheuristics: From Design to Implementation, John Wiley & Sons, 2009.

[15] E. Zitzler, L. Thiele, L. Laumans, et al., Performance assessment of multiobjective optimizers: an analysis and review, Evolutionary Computation (2003) 117–132.

[16] G. Petrone, Optimization under uncertainty: theory, algorithms and industrial applications, PHD thesis, Università degli Studi di Napoli Federico II, cited on pages 77-122 (2011).

[17] J. Zhou, F. Yang, K. Wang, Multi-objective optimization in uncertain random environments, Fuzzy Optimization and Decision Making, Springer 13 (4) (2014) 397–413.

[18] H. Levy, Stochastic dominance: Investment decision making under uncertainty, Springer, 2015.

[19] E. Hughes, Evolutionary multi-objective ranking with uncertainty and noise, Evolutionary Multi-Criterion Optimization (2001) 329–343.

[20] M. Basseur, E. Zitzler, Handling uncertainty in indicator-based multiobjective optimization, Computational Intelligence Research (2006) 255–272.

[21] A. Liefooghe, L. Jourdan, E.-G. Talbi, Indicator-based approaches for multiobjective optimization in uncertain environments, Uncertainty and Robustness in Planning and Decision Making (URPDM) (2010).

[22] K. Deb, H. Gupta, Searching for robust pareto-optimal solutions in multi-objective optimization, Evolutionary Multi-Criterion Optimization 34 (10) (2005) 150–164.

[23] C. Barrico, C. H. Antunes, Robustness analysis in multi-objective optimization using a degree of robustness concept, Evolutionary Computation (2006) 1887–1892.

[24] M. Koppen, R. Vicente-Garcia, A fuzzy scheme for the ranking of multivariate data and its application, in: IEEE Annual Meeting of the Fuzzy Information, Processing NAFIPS'04, Vol. 1, 2004, pp. 140–145.

[25] R. C. Silva, A. Yamakami, Definition of fuzzy pareto-optimality by using possibility theory, in: IFSA/EUSFLAT Conference, citeseer, 2009, pp. 1234–1239.

[26] M. Hendriks, M. Geile, T. Basten, Pareto analysis with uncertainty, in: 9th International Conference on Embedded and Ubiquitous Computing (EUC), 2011, pp. 189–196.

[27] G. Zhang, D. Ruan, Multi-Objective Group Decision Making: Methods Software and Applications with Fuzzy Set Techniques, World Scientific, 2007.

[28] T. Ross, Fuzzy logic with engineering applications, John Wiley & Sons, 2010.

[29] B. Bouchon-Meunier, et al., On the choice of membership functions in a mamdani-type fuzzy controller, Citeseer.

[30] S. Heilpern, Representation and application of fuzzy numbers, Fuzzy sets and systems, Elsevier 91 (2) (1997) 259–268.

[31] J. Zhao, B. Bose, Evaluation of membership functions for fuzzy logic controlled induction motor drive, in: IEEE 28th Conference of the Industrial Electronics Society IECON 02, Vol. 1, 2002, pp. 229–234.

[32] R. Boukezzoula, S. Galichet, L. Foulloy, MIN and MAX operators for fuzzy intervals and their potential use in aggregation operators, IEEE Transactions on Fuzzy Systems 15 (6) (2007) 1135–1144.

[33] C. Chen, A new approach for ranking fuzzy numbers by distance method, Fuzzy Sets and Systems 95 (3) (1998) 307–317.

[34] T. Chu, C. Tsao, Ranking fuzzy numbers with an area between the centroid point and original point, Computers and Mathematics with Applications 43 (1) (2002) 111–117.

[35] Z. Wang, F. Tian, A note of the expected value and variance of fuzzy variables, International Journal of Nonlinear Science 9 (4) (2010) 486–492.

[36] A. Kaufmann, M. Gupta, Fuzzy Mathematical Models in Engineering and Management Science, Elsevier Science Inc., 1988.

[37] J. Yao, K. Wu, Ranking fuzzy numbers based on decomposition principle and signed distance, fuzzy Sets and Systems 116 (2) (2000) 275–288.

[38] S. Abbasbandy, B. Asady, Ranking of fuzzy numbers by sign distance, Information Sciences 176 (16) (2006) 2405–2416.

[39] R. Ezzati, T. Allahviranloo, M. Khezerloo, An approach for ranking of fuzzy numbers, Expert Systems with Applications 43 (1) (2012) 690–695.

[40] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization, Evolutionary Computation 6 (2) (2000) 182–197.

[41] E. Zitzler, E. Laumans, L. Thiele, SPEA2: Improving the strength pareto evolutionary algorithm, Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse.

[42] A. Liefooghe, Methodes pour l'optimisation multiobjectif: Approche cooperative, prise en compte de l'incertitude et application logistique, PHD thesis, Universit de Lille1, Lille, cited on pages 13-20 (Decembre 2009).

[43] C. Bertoluzza, C. Corral, A. Salas, On a new class of distances between fuzzy numbers, Mathware Soft Computing (1995) 71–84.

[44] P. Toth, D. Vigo, The vehicle routing problem, Siam, 2002.

[45] M. Solomon, Algorithms for the vehicle routing and scheduling problem with time window constraints, Operations Research 35 (1) (1987) 254–265.

[46] J. Knowles, L. Thiele, E. Zitzler, A tutorial on the performance assessment of stochastic multiobjective optimizers, Tik report 214 (2006) 327–332.

[47] S. Bleuler, M. Laumanns, L. Thiele, E. Zitzler, PISAa platform and programming language independent interface for search algorithms, in: International Conference on Evolutionary Multi-Criterion Optimization EMO'03, 2003, pp. 494–508.

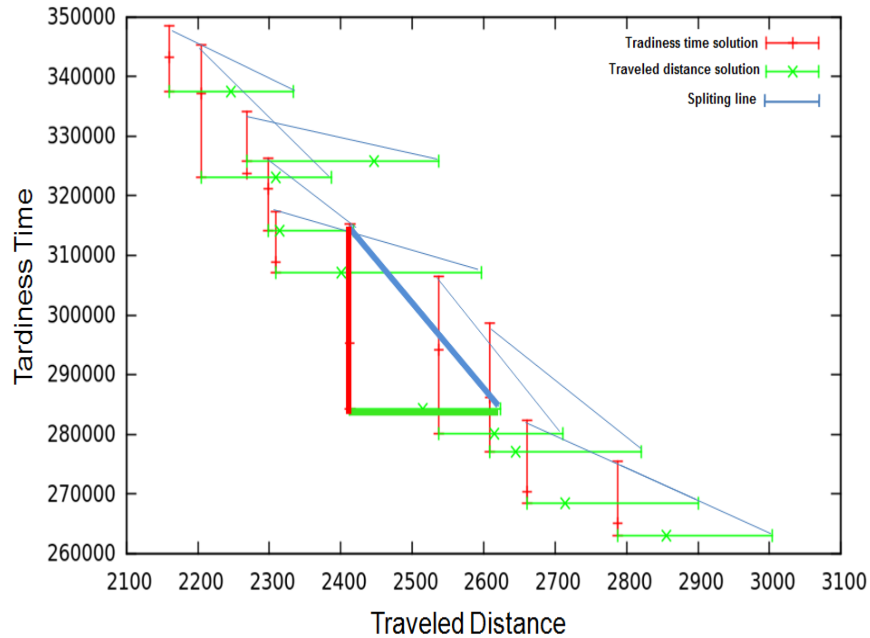


Figure 14: Example of E-SPEA2 solutions for Fuzz-C101

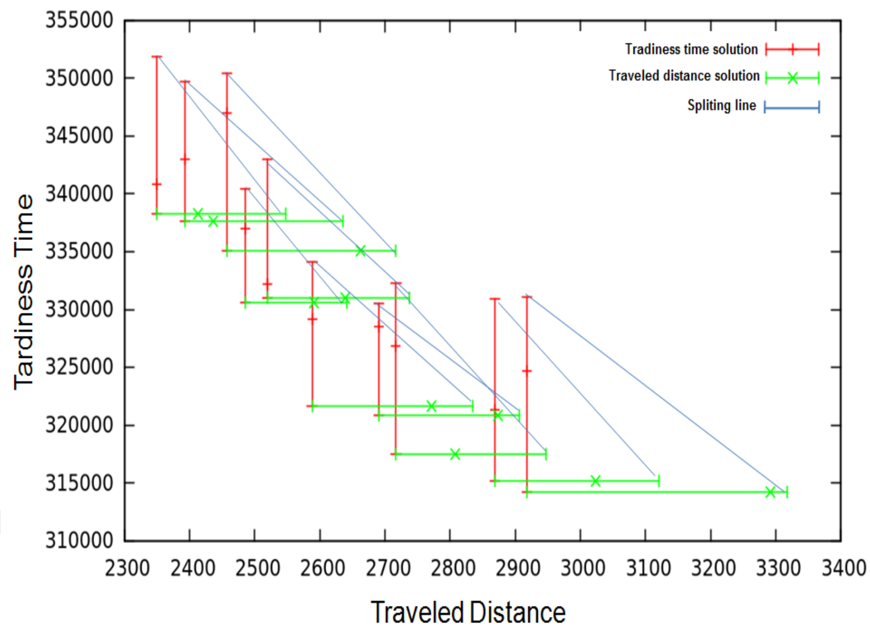


Figure 15: Example of E-NSGAII solutions for Fuzz-C101

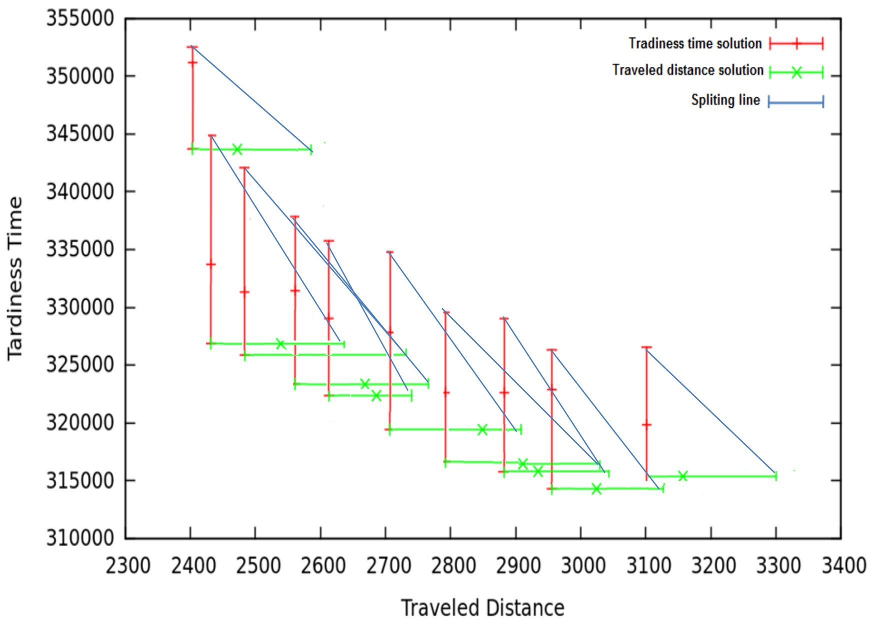


Figure 16: Example of E-IBEA solutions for Fuzz-C101

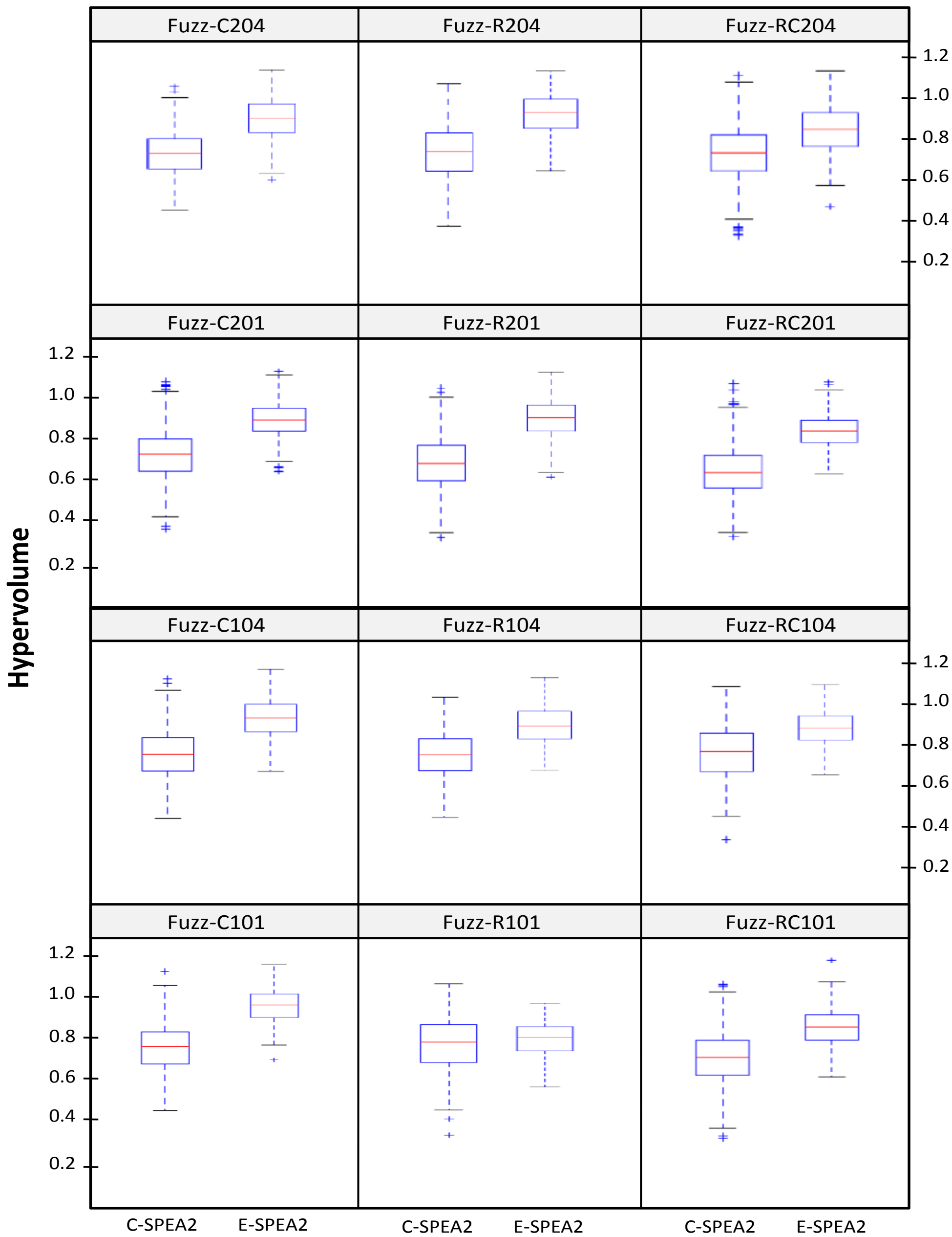


Figure 17: Hypervolume results of C-SPEA2 and E-SPEA2

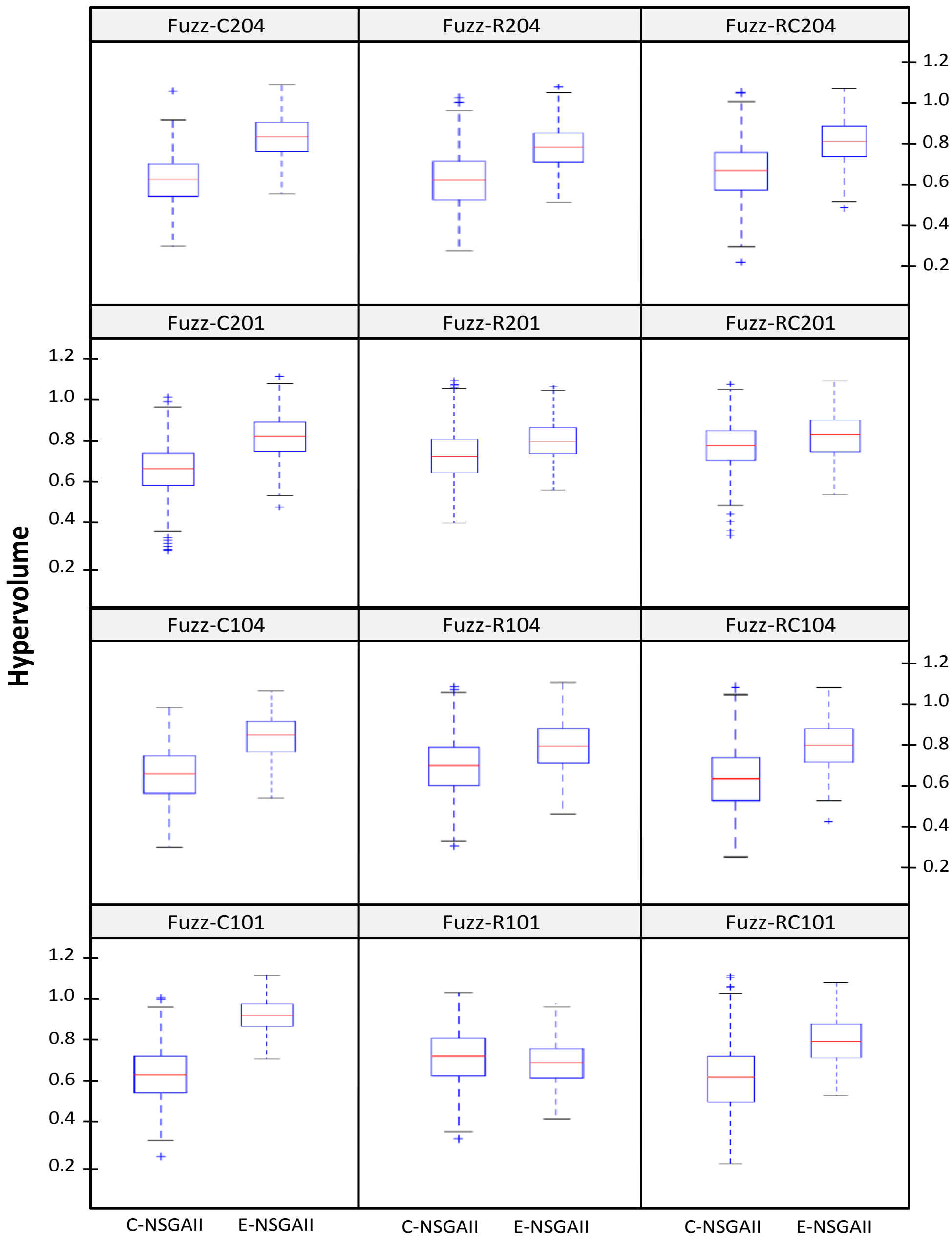


Figure 18: Hypervolume results of C-NSGAII and E-NSGAII

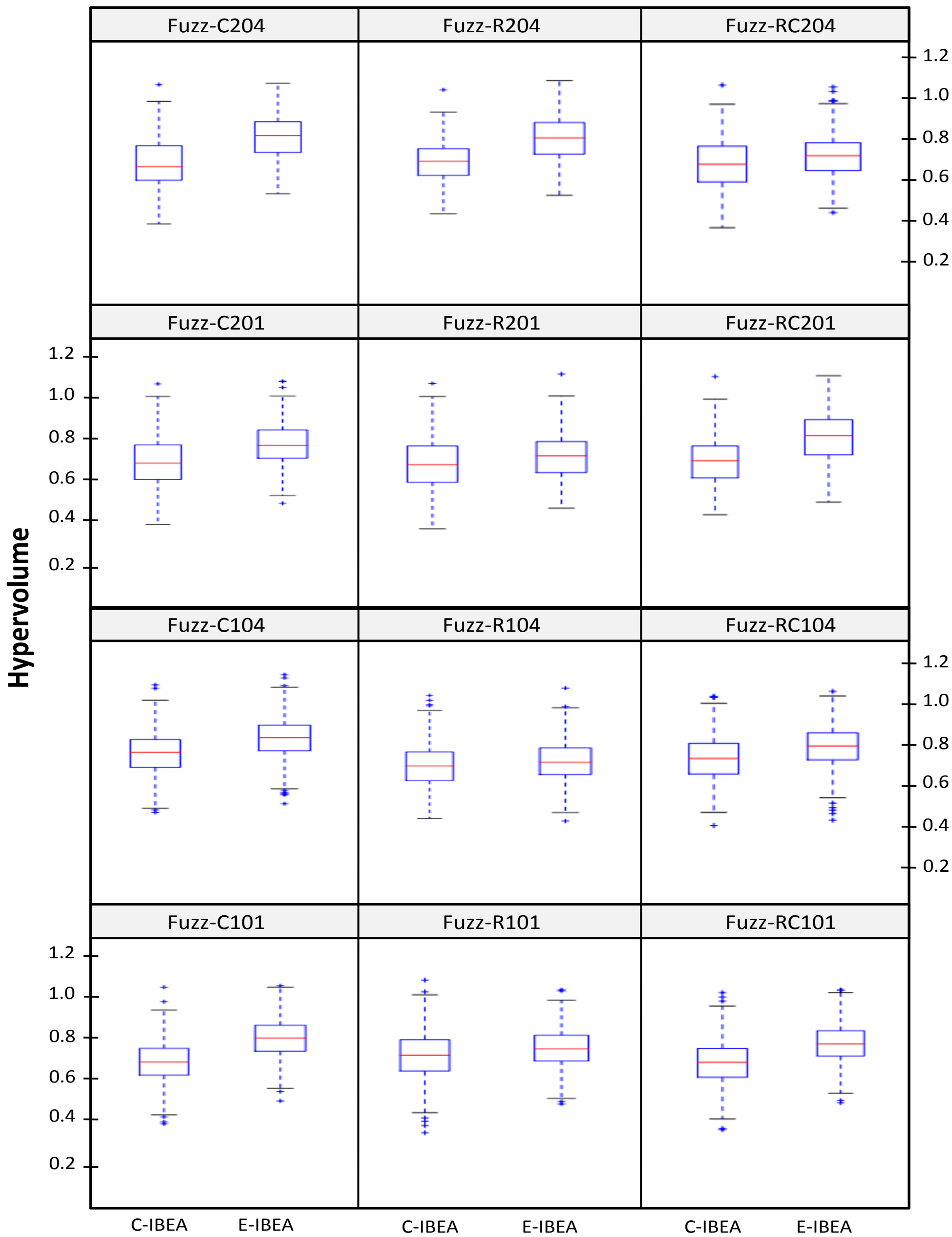


Figure 19: Hypervolume results of C-IBEA and E-IBEA