,661

Optimization of Many Objectives with Intervals Applying the MOEA/D Algorithm

Lorena R. Rosas-Solórzano, Claudia G. Gómez-Santillán, Nelson Rangel-Valdez, Laura Cruz-Reyes, Fausto A. Balderas-Jaramillo and José A. Brambila-Hernández

***Abstract*—Project Portfolio Selection (PPS) is a major strategic decision problem faced by any organization. PPS decides how to invest resources into projects subject to a decision process influenced by multiple conflicting criteria. The portfolio's compromise to the organization's well-being has an uncertainty that directly affects a decision maker's preferences (DM). MOEA/D is a well-known approach to tackle multicriteria optimization problems, and it is still open for the development of strategies to handle uncertainty on its search process. This work proposes I-MOEA/D, a new method based on a MOEA/D approach, to deal with DM's uncertainty in costs and benefits of portfolios' projects. The proposed novel features include (a) handling large numbers of objectives; (b) a method to generate the initial population; and (c) handling the uncertainty of resources, costs, and benefits through intervals. An experiment compared I-MOEA/D against the state-of-the-art I-NSGA-II algorithm in instances with two to fifteen objectives. Results demonstrate the competitiveness of I-MOEA/D by improving the quality of solution of I-NSGA-II in most instances.**

***Index Terms*—Decision making, uncertainty, multi-objective optimization, mathematics of intervals, project portfolio problem.**

1. Introduction

ORGANIZATIONS usually address Project Portfolio Selection (PPS) aided by a decision-maker (DM). The DM and the decision analyst often must provide information on portfolio values; however, such information might be incomplete, causing a condition of uncertainty. The PPS has distinct solutions in state-of-the-art works; however, there is still a lack of research that handles uncertainty in the PPS, even fewer using intervals [1][2][3]. An interval is a range used to represent unclear projects' values defined for organizational resources, e.g., benefits, costs, requirements, times, synergies, partial support, among others.

Manuscript received on September 14, 2020, accepted for publication on November 24, 2020, published on December 30, 2020.

L. R. Rosas- Solórzano, C. G. Gómez-Santillán, L. Cruz-Reyes, F. A. Balderas-Jaramillo, and J. A. Brambila-Hernández are with the Postgraduate & Research Division, National Mexican Institute of Technology/Madero Institute of Technology, 89440, Madero, Tamaulipas, Mexico (e-mail: {g13071208, claudia.gomez, lauracruzreyes, austo.balderas}@itcm.edu.mx, g02070820@cdmadero.tecnm.mx).

N. Rangel-Valdez is a Researcher of CONACYT, attached to Postgraduate & Research Division, National Mexican Institute of Technology/Madero Institute of Technology, 89440, Madero, Tamaulipas, Mexico (e-mail: nelson.rangel@itcm.edu.mx).

The DM is responsible for selecting the portfolio that best meets the organizational objectives. However, to carry out this activity, he or she faces difficulties in solving the PPS due to: the exponential complexity of the optimization problem, the number of involved objectives, the lack of information about the exact contribution of the projects to the portfolio, and the imprecise knowledge of the requirements or resources needed to complete the projects and their availability. The improper modeling of the previous difficulties can lead to portfolios that can affect the institutions' interests.

Carazo [4] defines a project as a temporary, unique, and unrepeatable process that pursues a specific set of objectives, which, when combined, will impact the vi-sion and mission of the organizations. A portfolio is a set of projects that, carried out in a given period, share a series of resources, among which there may be rela-tionships of complementarity, incompatibility, and synergies produced by sharing costs and benefits derived from the implementation of more than one project at a time [5].

The proper selection of projects for a portfolio can ben-efit any organization based on the DM's objectives. Therefore, a portfolio decision analysis can help the DMs select a subset of an extensive set of projects through modeling, considering relevant constraints, preferences, and inaccuracies in the information [6].

Currently, some state-of-the-art strategies to tackle with PPS and uncertainty are: fuzzy sets [7][8][9][10][11][12][13], interval analysis [14][15][16][17][18] and probability distributions [19]. Some authors have also introduced modifications in the portfolio to minimize uncertainty and model the DM attitude to risk [20] [21][22].

This paper proposes the analysis of an algorithm based on intervals as the solution to PPS under uncer-tainty. The approach uses an MOEA family algorithm (Multi-Objective Evolutionary Algorithms) that solves problems with many objectives and of interest to the scientific community. Specifically, we worked with the MOEA/D (Multi-Objective Evolutionary Algorithm based on Decomposition) algorithm. The proposed MOEA/D, denoted I-MOEA/D for Interval Multi-Objective Evolutionary Algorithm based on Decomposition, includes novel features as intervals to manage uncertainty and adequate handling of many-objective optimization problems. A developed experi-ment shows the performance of I-MOEA/D against I-NSGA-II, an algorithm from the scientific literature ba-sed on NSGA-II (Non-Dominated Sorting Genetic Algo-rithm II) that also handles uncertainty [10].

This document is structured as follows: Section 2 pro-vides some background on Multi-Objective PPS; also, it presents the proposed I-MOEA/D, initialization fun-ction, and random instance generator. Section 3 descri-bes the experiment conducted to validate I-MOEA/D and the results; besides, it provides the analysis that demonstrates the proposed strategy's advantages. Fina-lly, Section 4 summarizes the main conclusions drawn from the research.

2. Proposed Solution

2.1 Multi-objective Project Portfolio Selection

Until today, Multi-objective Project Portfolio Selection (or just PPS) has distinct approaches that solve it [4] [5]. A solution is a portfolio composed of one or more projects. A project is a series of activities related to each other to reach a specific objective, which consumes resources. A formal definition of PPS with uncertainty (UPPS) is the following.

Let the binary vector $\vec{x} = \langle x_1, x_2, \dots, x_p \rangle$ of size p be a portfolio, where p is the available projects, $x_i = 1$ or $x_i = 0$ represents whether or not a project i is in the portfolio, respectively. Let $c(\vec{x})$ and $f(\vec{x}) = \{f_1(\vec{x}), f_2(\vec{x}), \dots, f_m(\vec{x})\}$ be the portfolio cost and fitness. Let B be the budget available to form the portfolio. Finally, let $A = \{A_1, A_2, \dots, A_a\}$ and $R = \{R_1, R_2, \dots, R_r\}$ bounds over specific areas and regions of interest that must be satisfied by the portfolio. If $c(.)=[\underline{c}, \overline{c}]$, $f(.)=[\underline{f}, \overline{f}]$, $B=[\underline{B}, \overline{B}]$, $A_i=[\underline{A}, \overline{A}]$, and $R_i=[\underline{R}, \overline{R}]$ are intervals defined by a lower $\underline{l}$ and upper $\overline{u}$ bounds then equations 1 and 2 formalize the definition of UPPS.

$$\max f(\vec{x}) = \{f_1(\vec{x}), f_2(\vec{x}), \dots, f_n(\vec{x})\} \tag{1}$$

$$\begin{aligned} \text{Subject to:} \quad & c(\vec{x}) \le B \\ & \underline{A_i} \le A_i(\vec{x}) \le \overline{A_i} \\ & \underline{R_i} \le R_i(\vec{x}) \le \overline{R_i} \\ & \vec{x} \in \{0,1\}^n \end{aligned} \tag{2}$$

where the basic arithmetic and relational operations follow previously defined computations (cf., [15, 20]), e.g. $f_i(\vec{x})$, $c(\vec{x})$, $A(\vec{x})$, and $R(\vec{x})$ are product of a linear combination of the contribution of each project i in the portfolio in fitness, cost, area, or region, respectively.

2.2 Generate Initial Population with Exchange

This section describes the initialization strategy for the population of I-MOEA/D, in the presence of intervals. The process is simple; it chooses a project i as part of the portfolio whenever a random uniform value v lies under a predefined selection threshold selection, β (set to 0.5 for this research work). Following a trial-and-error ap-proach, the algorithm discards those solutions that be-came infeasible in the process. Algorithm 1 shows the pseudocode of the method.

Algorithm 1. Generation of Initial Population

Input:

- β-Threshold selection
- *m*-Objectives number
- *p*-Total projects
- *a*-Number of areas
- *r*-Number of regions

Output:

- Initial population

```
0.  x⃗ = {1, 1, …,1}
1.  while (! Feasibility (x⃗)) do
2.    x⃗ = {0, 0, …,0}
3.    for each i ∈ {1,2, …, p} do
4.      r=random (0,1)
5.      If (r < β) then
6.        x_i = 1
7.        c (x⃗) += c(i)
8.        for each j ∈ {1, …, m} do
9.          f_i(x⃗) += f_j(i)
10.       end
11.       for each j ∈ {1, …, a} do
12.         A_i(x⃗) += A_j(i)
13.       end
14.       for each j ∈ {1, …, r} do
15.         R_i(x⃗) += R_j(i)
16.       end
17.     end
18.   end
19. end
20. return x⃗
```

Line 1 uses function Feasibility(.) to ensure a feasible solution; it validates the restrictions of the UPPS of budget, area, and region.

The algorithm tests each project for inclusion into the portfolio in Lines 4 and 5. Whenever the condition is satisfied the costs, and values for objectives, areas, and regions are accumulated (Lines 7, 8, 11, and 14, respectively).

Feasibility requires the addition and relational ≤ operations. Given two interval numbers $\boldsymbol{E} = [\underline{E}, \overline{E}]$ and $\boldsymbol{D} = [\underline{D}, \overline{D}]$, the result of $\mathbf{C} = \mathbf{E} + \mathbf{D}$ can be computed as $\mathbf{C} = [\underline{E} + \underline{D}, \overline{E} + \overline{D}]$. In the other hand, the relational operation $\boldsymbol{D} \le \boldsymbol{E}$ can be estimated using the relational quotient defined by equation 3.

$$p_{ED} = \frac{\overline{E} - \underline{D}}{(\overline{E} - \underline{E}) + (\overline{D} - \underline{D})} \tag{3}$$

Based on the relational quotient, equation 4 defines the possibility measure of $Poss(\boldsymbol{D} \le \boldsymbol{E})$ used to express the desired relationship between the intervals **D** and **E**. This work establishes that $Poss(\boldsymbol{D} \le \boldsymbol{E}) \ge 0.5$.

$$Poss(\boldsymbol{D} \le \boldsymbol{E}) = \begin{cases} 1 & \text{if } p_{ED} > 1, \\ p_{ED} & \text{if } 0 \le p_{ED} \le 1, \\ 0 & \text{if } p_{ED} \le 0 \end{cases} \tag{4}$$

Figure 1 depicts the process performed by Algorithm 1 in the construction of a portfolio. This figure shows an array with 25 cells representing a portfolio and the 25 potential projects. Each cell also represents an iteration and its randomly generated value. Note that the shadow cells correspond to those where the random value lay under the threshold β = 0.5. The process is repeated as many solutions the initial population of I-MOEA/D has.

1	2	**3**	4	5	**6**	7	8
0.89	0.59	**0.16**	0.76	0.68	**0.42**	0.73	0.96

9	10	**11**	12	**13**	14	15	16
0.09	0.53	**0.38**	0.71	**0.12**	0.83	0.88	0.65

17	18	19	**20**	21	22	23	**24**
0.99	0.79	0.55	**0.36**	0.68	0.62	0.92	**0.28**

25
0.64

Selected random value

Fig. 1. Graphic representation of the iterative process of Algorithm 1 to build a portfolio.

While Figure 1 shows the process of selecting projects, Figure 2 shows the binary representation required by I-MOEA/D. The portfolio must be feasible, and all the constraints of costs, areas, and regions must be satisfied.

1	2	3	4	5	6	7	8	9
0	0	1	0	0	1	0	0	1

10	11	12	13	14	15	16	17	18
0	1	0	1	0	0	0	0	0

19	20	21	22	23	24	25
0	1	0	0	0	1	0

Fig. 2. A binary vector representing a solution (or portfolio) used by I-MOEA/D. Here the value 1 means that the project is part of the portfolio and 0 otherwise.

2.3 Multi-objective Evolutionary Algorithm Based on Decomposition with Intervals (I-MOEA/D)

MOEA/D is a technique proposed by Zhang and Hui [23]. This algorithm consists of decomposing a multi-objective optimization problem into several sub-problems that are optimized simultaneously. I-MOEA/D is a variant of MOEA/D that solves PPS with uncertainty; it implements evolutive operators to handle intervals. The intervals represent a mean of expression of uncertainty in objectives values, costs, and resources. Algorithm 2 shows the general pseudocode of the proposed strategy. I-MOEA/D gives an external population (*EP*) containing the non-dominated solutions found during the optimization process.

Algorithm 2. I-MOEA/D

Input:

- MOP= Multi-objective Optimization Problem
- N= Population size
- p=Number of projects
- m= Number of objectives
- T= Neighborhood size of the weight vectors
- MaximumEvaluations= Number of Generations

Output:

EP= External population

0. W = ReadWeightsVector()
1. EP= ϕ
2. Calculate_Euclidean_Distance (V)
3. SortVector ()
4. **Population=GenerationOfInitialPopulation()**
5. Initialize_Z (Population)
6. Generations=0;
7. While (Generations < MaximumEvaluations)
8.   *i*=1
9.   For each $i \in N$ do
10.   [p1, p2]=**SelectionByTournament(Population,B(i), T)**
11.   *offspring*= CrossoverOnePoint (p, [p1, p2])
12.   *offspring*= GeneMutation (p, *offspring*)
13.   *offspring*=**ImprovementGeneMutation**(*offspring*)
14.   **UpdateZ** (**$\vec{z}$**, *f(offspring)*)
15.   **UpdateEP** (**EP**, *f(offspring)*)
16.   end
17. Generations++
18. end

The binary vectors $\vec{x} = \langle x_1, x_2, \ldots, x_p \rangle$ encode one portfolio or solution provided by the algorithm. Such vectors are chromosomes in the evolutive approach, and the vectors' indexes of the array are alleles denoting distinct projects.

The methods of I-MOEA/D that distinguish from implementations of other MOEA/D are five, appearing in bold in Algorithm 2 (Lines 2, 10, 13-15). These methods are the initialization function, the selection operator, the repair/improve operator, and the Z vector and EP set update. The remaining section provides a detailed description of the methods.

Initialization Phase. In the first stage, the I-MOEA/D algorithm makes the weight vector's initial set (Line 0) and initializes EP to empty (Line 1). It also initializes the weight vectors $\{w_1, w_2, \ldots, w_N\}$, computes the Euclidean distance among them and initializes their vectors' neighborhood $B(i)$, $1 \leq i \leq N$ (Lines 2 to 4). The Algorithm 1 fills the initial solution vector B of size N and associates each solution B_i to a weight vector w_i. Then, the neighborhood $B(i)$ of a weight vector w_i contains the closest weight vectors index by Euclidean distance. Finally, the initialization phase fills the vector Z, the vector of best objective values found in the search process (Line 5). The vector B corresponds to the initial population. The main loop of the I-MOEA/D begins having as stop criterion a maximum number of evaluations previously defined (Line 7).

budget, areas, regions, and for projects their costs, objectives values, and the area and region where they belong.

The generator creates an interval budget in Line 0 based on the input budget *B*. Figure 4 shows an example of the definition of such intervals.

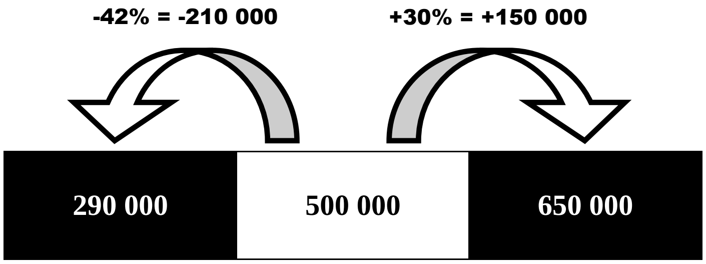

Fig. 4. Budget parameter

From Lines 1 to 6 the generator creates the values for the areas of the instance. Here, it uses the budget *B* to define appropriate limites to areas' values (Lines 1 and 2). After that, it randomly chose values withing those limites as the bounding values of each area (Lines 4 and 5).
From Lines 7 to 12 the generator assigns values to the regions in a similar fashion as done in the areas; i.e., it uses the budget to define appropriate maximum limits, and with them randomly chose values to bound the distinct regions.

The next step in the generator is the definition of values for the projects. Lines 14 to 26 perform this task. First the area and regions are randomly chosen in lines 14 and 15. After that, the cost of the project is created within the limits provided as inputs (Lines 16 to 17). From Lines 18 to 26 the process generates the values for the objectives of a project. It uses two strategies, one based on the costs of the project (Line 20), and the other based on the limits for the objectives established as input (Line 22). With the value *o* the generator creates an interval for the objective using 80% of it as lower bound and 120% as upper bound.

3. Experimentation

This section contains a series of experiments aimed to validate the quality of the I-MOEA/D compared with the I-NSGA-II algorithm [18].

The configuration of the experimental design took into account the following details: a) the size of the set of tested UPPS instances was 7; b) the project set involved was always of cardinality 100; c) the number of objectives involved in the cases varied according to {2,3,4,8,9,13,15}; d) the proposed random generator shown in section 2.4 generated the instances. Concerning the algorithms, the population size was 100, the stop criterion was after 500 generations, and the crossover and mutation operators considered a probability of 100%.

The algorithm test environment was implemented in the Java programming language and ran on a computer with the following features: 2.20 GHz Intel Core i5 CPU, 4 GB RAM, and Windows 10 Operating System.

Algorithm 6. PPS with intervals instance generator

Input:

- $B \leftarrow$ Budget (No interval)
- $\{m, p, a, r\} \leftarrow$ Numbr of objectives, projects, areas, and regions
- $[\underline{c}, \overline{c}] \leftarrow$ Project Costs extreme limits (No Intervals)
- $[\underline{m}, \overline{m}] \leftarrow$ Objectives extreme limits

Output:

- $[\underline{B}, \overline{B}] \leftarrow$ Budget as interval
- $\{[\underline{a_1}, \overline{a_1}], [\underline{a_2}, \overline{a_2}], \ldots, [\underline{a_a}, \overline{a_a}]\} \leftarrow$ Limits of each areas i
- $\{[\underline{r_1}, \overline{r_1}], [\underline{r_2}, \overline{r_2}], \ldots, [\underline{r_r}, \overline{r_r}]\} \leftarrow$ Limits of each region r
- $\{\{C_1, A_1, R_1\}, \ldots, \{C_p, A_p, R_p\}\} \leftarrow$ Cost, Area and region for each project p
- $\{[\underline{f_{1p}}, \overline{f_{1p}}], [\underline{f_{2p}}, \overline{f_{2p}}], \ldots, [\underline{f_{mp}}, \overline{f_{mp}}]\}$Benefit from the m objectives of each project p (in intervals)

0. $[\underline{B}, \overline{B}] = [0.58B, 1.3B]$
1. $[\underline{a_l}, \overline{a_l}] = [(0.7 * B)/(1.7a+0.1a^2), (1.27 * B)/(1.7a+0.1a^2)]$
2. $[\underline{a_u}, \overline{a_u}] = [((2.159 +0.127a) *B) / a, ((2.635+0.155a) * B)/a]$
3. **for each** $i \in \{1, 2, \ldots, a\}$ **do**
4. $\underline{a_i} = \underline{a_l}$ + Random $(\overline{a_l} - \underline{a_l})$
5. $\overline{a_i} = \underline{a_u}$ + Random $(\overline{a_u} - \underline{a_u})$
6. **end**
7. $[\underline{r_l}, \overline{r_l}] = [(0.8 * B)/(1.7r+0.1r^2), (1.2 * B)/(1.7r+0.1r^2)]$
8. $[\underline{r_u}, \overline{r_u}] = [((1.02+0.06r) *B)/r, ((2.38 +0.14r) *B)/r]$
9. **for each** $i \in \{1, 2, \ldots, r\}$ **do**
10. $\underline{r_i} = \underline{r_l}$ + Random $(\overline{r_l} - \underline{r_l})$
11. $\overline{r_i} = \underline{r_u}$ + Random $(\overline{r_u} - \underline{r_u})$
12. **end**
13. **for each** $i \in \{1, 2, \ldots, p\}$ **do**
14. A_i= Random(a)
15. R_i= Random(r)
16. $v = \underline{c}$ + Random $(\overline{c} - \underline{c})$
17. $[\underline{C_i}, \overline{C_i}] = [0.99 * v, 1.2 * v]$
18. **for each** $j \in \{1,2, \ldots, m\}$ **do**
19. **if (**Random**.**nextBoolean()**) then**
20. obj = Random $((v - \underline{c}) / (\overline{c} - \underline{c}))$
21. **else**
22. $o = \underline{m}$ + Random $(\overline{m} - \underline{m})$
23. **end**
24. $\underline{f_{ij}}$= 0.8*o
25 $\overline{f_{ij}}$ = 1.1*o
26. **end**
27. **end**

The *number of non-dominated portfolios*, and the *portfolios' cardinality* are the two quality measurements of interest for this work to assess the algorithms' performance for comparison purposes. Equations (5) to (8) show the indicators formed from the previous measurements, where *EP* is the final non-dominated set of algorithm's solutions after 30 independent runs. Let's point out that larger indicator' values represent better performance in an algorithm.

$$I_1 = |EP| \tag{5}$$

$$I_2 = \frac{\sum_{x \in EP} |x|}{|EP|} \tag{6}$$

$$I_3 = \min_{x \in EP}\{|x|\} \tag{7}$$

$$I_4 = \max_{x \in EP}\{|x|\} \tag{8}$$

Table 1 compares I-MOEA/D and I-NSGA-II. Columns 1 and 2 show the instances' names and algorithms, respectively. Columns 3 to 6 shows the indicators' values. Note that the encoded name *oipj* contains the numbers of objectives *i* and projects *j*.

According to Table 1, I-NSGA-II improves I-MOEA/D in the instance with two objectives. The differences range from 5% to 12% in the indicators' observed values; this is a common condition since NSGA-II generally has a good performance in that number of objectives.

Consequently, I- MOEA/D is the clear winner in the remaining instances; its performance differences vary from 69% to 97%. In conclusion, the overall results shown in Table 2 demonstrates that I-MOEA/D improves I-NSGA-II in all the indicators. These results also indicate poor performance of I-NSGA-II in many-objective problems, a condition previously observed.

TABLE 1
COMPARISON OF I-MOEA/D AND I-NSGA-II BY QUALITY INDICATORS

Instance	Algorithm	I_1	I_2	I_3	I_4
o2p100	I-NSGA-II	63	57	56	58
	I-MOEA/D	55	54	53	55
o3p100	I-NSGA-II	526	39	37	40
	I-MOEA/D	4556	44	40	46
o4p100	I-NSGA-II	139	58	57	59
	I-MOEA/D	449	68	67	68
o8p100	I-NSGA-II	585	38	35	40
	I-MOEA/D	21327	46	41	47
o9p100	I-NSGA-II	579	41	38	42
	I-MOEA/D	27863	45	40	47
o13p100	I-NSGA-II	521	52	49	54
	I-MOEA/D	5123	63	61	64
o15p100	I-NSGA-II	677	34	31	37
	I-MOEA/D	21417	47	41	48

To provide further insights, we compute the relative differences among the indicators measured for I-MOEA/D and I-NSGA-II. For this purpose, equation 9 defines a metric to calculate the percentage of improvement achieved by the winner algorithm for the given indicator I^k_j, where j is the indicator and k=1 if the algorithm is I-MOEA/D or k=2 if it is I-NSGA-II. A winner algorithm has the highest indicator value. Tables 2 and 3 summarizes the results obtained from this metric for instances with objectives 3 to 15.

$$\mathrm{Diff}\left(I_j^1, I_j^2\right) = \begin{cases} 100\left(\dfrac{I_j^1 - I_j^2}{I_j^1}\right), & \text{if } I_j^1 > I_j^2 \\ 100\left(\dfrac{I_j^2 - I_j^1}{I_j^2}\right), & \text{otherwise} \end{cases} \tag{9}$$

The results from Tables 2 and 3 shows that I-MOEA/D improves all the indicators measures with respect to I-NSGA-II in percentual ranges that vary in [69, 97], [8, 27], [7, 24], and [10,13] for the indicators I_1, I_2, I_3, and I_4, respectively. These results tell that I-MOEA/D obtains more non-dominated solutions and portfolios with a greater number of projects, which is desirable.

TABLE 2
PERCENTAGE DIFFERENCE OF NON-NOMINATED PORTFOLIOS AND AVERAGE CARDINALITY

Instance	Diff ($I^1{}_1$, $I^2{}_1$)	Diff ($I^1{}_2$, $I^2{}_2$)
o3p100	88%	11%
o4p100	69%	14%
o8p100	97%	17%
o9p100	97%	8%
o13p100	89%	17%
o15p100	96%	27%

TABLE 3
PERCENTAGE DIFFERENCE IN MINIMUM AND MAXIMUM CARDINALITY OF PORTFOLIOS

Instance	Diff ($I^1{}_3$, $I^2{}_3$)	Diff ($I^1{}_4$, $I^2{}_4$)
o3p100	7%	13%
o4p100	14%	13%
o8p100	14%	14%
o9p100	5%	10%
o13p100	19%	15%
o15p100	24%	22%

Finally, Table 4 compares the dominance proportion per algorithm. For this purpose, a set *EP** combines the final sets EP^1 and EP^2; this new set is the final non-dominated front. Then, we calculate the number of solutions of EP^1 and EP^2 that appear in *EP**. Let's note that EP^1 and EP^2 correspond to the final non-dominated fronts EP of I-MOEA/D and I-NSGA-II, respectively. Column 3 contains the number of non-dominated solutions still appearing in EP*. Column 4 reports the number of solutions that became dominated after integration.

TABLE 4
DOMINANCE PROPORTION AMONG I-MOEA/D E I-NSGA-II

Instance	Algorithm	Total, non-dominated portfolios	Dominated portfolios
o2p100	I-NSGA-II	0	63
	I-MOEA/D	55	0
o3p100	I-NSGA-II	441	85
	I-MOEA/D	4556	0
o4p100	I-NSGA-II	0	139
	I-MOEA/D	449	0
o8p100	I-NSGA-II	1	584
	I-MOEA/D	21327	0
o9p100	I-NSGA-II	546	33
	I-MOEA/D	27863	0
o13p100	I-NSGA-II	0	521
	I-MOEA/D	5123	0
o15p100	I-NSGA-II	1	676
	I-MOEA/D	21417	0

Considering the information of Table 4, all the solutions of I-MOEA/D remain non-dominated. Moreover, it turns out that they dominate several solutions provided by I-NSGA-II and, in some cases, all of them (instances 2 and 13). These results corroborate the excellent performance of I-MOEA/D to solve UPPS over I-NSGA-II with many objectives.

The experimental design concluded with an analysis of the statistical differences of the observed results. Particularly, a Wilcoxon's test [27] validated the difference in the indicator I_1. The considered sample was the number of non-dominated portfolios of each of the 30 runs of an instance. The test utilized a significance level of 5%. The null hypothesis was "**H_0=***The medians of the differences between the two group samples are equal*". Table 5 summarizes the results.

TABLE 5
WILCOXON TEST

Instance	***p value***	**Result**
o2p100	0.57746866	Is accepted H0
o3p100	0.04311445	Is rejected H0
o4p100	0.04311445	Is rejected H0
o8p100	0.04311445	Is rejected H0
o9p100	0.04311445	Is rejected H0
o13p100	0.04311445	Is rejected H0
o15p100	0.04311445	Is rejected H0

The results from Table 5 show that there are significant differences in 6 instances, and based on information from Table 1, the differences favor I-MOEA/D. Let us point out I-NSGA-II performed better than I-MOEA/D only in the instance *"o2p100"*. However, there is no significant statistical difference in their performance. In conclusion, the overall performance of I-MOEA/D improves largely that of I-NSGA-II in the selected instances of UPPS.

4. CONCLUSIONS

This article proposes a new evolutionary strategy called I-MOEA/D. The main features of this algorithm are the use of intervals to express uncertainty and handling many objectives. A comparison in performance between I-MOEA/D and I-NSGA-II (a state-of-the-art approach) assessed the relevance of our approach. In equal experimental conditions under a controlled environment, the results show that I-MOEA/D outperforms I-NSGA-II [18], demonstrating the significance of I-MOEA/D.

The I-MOEA/D requires at least to modify the genetic operators, the repair/improve operator, the update methods of ideal objectives values and population, in order to integrate the use of intervals properly. The strategy required the definition of some interval operators to perform arithmetic, relational and dominance operations. The dominance operator appears with the definition of relational operators for comparison.

The observed results show that I-MOEA/D and I-NSGA-II solve UPPS. However, with increasing objectives, the performance of I-MOEA/D improves that of I-NSGA-II, as expected, particularly in the analyzed instances with number of objectives varying from two to fifteen. The results show that with an increasing number of objectives, I-MOEA/D return solutions with better quality.

Finally, the number and diversity of solutions offered by I-MOEA/D are large. This is a good condition in contrast to I-NSGA-II because it means that I-MOEA/D approximates the Pareto front better. However, it is interesting to ask if the search process of I-MOEA/D can include DM's preferences. If the latter is possible, then, a narrower set of solutions could be delivered to the DM, based on his/her priorities. Hence, the proper incorporation of preferences in the search process of I-MOEA/D represents an attractive research area for future developments.

ACKNOWLEDGMENTS

Authors thanks to CONACYT for supporting the projects from (a) Cátedras CONACYT Program with Number 3058. (b) CONACYT Project with Number A1-S-11012 from Convocatoria de Investigación Científica Básica 2017–2018 and CONACYT Project with Number 312397 from Programa de Apoyo para Actividades Científicas, Tecnológicas y de Innovación (PAACTI), a efecto de participar en la Convocatoria 2020-1 Apoyo para Proyectos de Investigación Científica, Desarrollo Tecnológico e Innovación en Salud ante la Contingencia por COVID-19. (c) Lorena R. Rosas-Solórzano would like to thank CONACYT for the support number 960719

REFERENCES

[1] G. Rivera, C. Gómez, L. Cruz, R. García, F. Balderas, E. Fernández, & F. López. "Solution to the social portfolio problem by evolutionary algorithms." *International Journal of Combinatorial Optimization Problems and Informatics* 3, 2 (2012): 21-30.

[2] C. Gomez, L. Cruz-Reyes, G. Rivera, N. Rangel-Valdez, L. Morales-Rodriguez, & M. Perez-Villafuerte. "Interdependent Projects selection

with preference incorporation." *New Perspectives on Applied Industrial Tools and Techniques*. Springer, Cham, 2018. 253-271. https://doi.org/10.1007/978-3-319-56871-3_13

[3] S. Bastiani, L. Cruz-Reyes, E. Fernandez., C. Gómez, & G. Rivera. "An ant colony algorithm for solving the selection portfolio problem, using a quality-assessment model for portfolios of projects expressed by a priority ranking." *Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization*. Springer, Cham, 2015. 357-373, https://doi.org/10.1007/978-3-319-17747-2_28

[4] A. Carazo, T. Gómez, J. Molina, A. Hernandez, F. Guerrero, R. Caballero, "Solving a comprehensive model for multiobjective project portfolio selection", *Computers & Operations Research*, pp. 630-639, 2010, https://doi.org/10.1016/j.cor.2009.06.012

[5] G. Fox, N. Baker, "Project Selection Decision Making Linked to a Dynamic Environment", *Management Science*, 1272-1285, 1985, https://doi.org/10.1287/mnsc.31.10.1272

[6] A. Salo, J. Keisler, A. Morton, "Portfolio Decision Analysis. Improved methods for resource allocation", *International Series in Operations Research & Management Science*, In A. Salo et al. (Eds), vol 162, Springer, New York, 3-27, 2011.

[7] F. Perez and T. Gomez, "Multiobjective project portfolio selection with fuzzy constraints", *Annals of Operations Research* 245(1–2) 7–29, 2016 https://doi.org/10.1007/s10479-014-1556-z.

[8] C. Carlsson, R. Fuller, M. Heikkila and P. Majlender, "A fuzzy approach to R&D portfolio selection", *International Journal of Approximate Reasoning* 44(2) 93–105, 2007 https://doi.org/10.1016/j.ijar.2006.07.003.

[9] H. Ke and B. Liu, "Project scheduling problem with mixed uncertainty of randomness and fuzziness", *European Journal of Operational Research* 183(1) 135–147, 2007, http://doi.org/10.1016/j.ejor.2006.09.055.

[10] T. Hasuike, H. Katagiri and H. Ishii, "Portfolio selection problems with random fuzzy variable returns", *Fuzzy Sets and Systems* 160(18), 2579–2596, 2009, https://doi.org/10.1016/j.fss.2008.11.010.

[11] K. K. Damghani, S. Sadi-Nezhad and M. B. Aryanezhad, "A modular decision support system for optimum investment selection in presence of uncertainty: Combination of fuzzy mathematical programming and fuzzy rule based system", *Expert Systems with Applications* 38(1) 824–834, 2011, http://doi.org/10.1016/j.eswa.2010.07.040.

[12] X. Huang, "Optimal project selection with random fuzzy parameters", *International Journal of Production Economics* 106(2) 513–529, 2007, https://doi.org/10.1016/j.ijpe.2006.06.011.

[13] D. Kuchta, "Use of fuzzy numbers in project risk (criticality) assessment", *International Journal of Project Management* 19(5) 305–310, 2001, http://doi.org/10.1016/S0263-7863(00)00022-3.

[14] T. Fliedner and J. Liesio, "Adjustable robustness for multi-attribute project portfolio selection", *European Journal of Operational Research* 252 931–946, 2016, http://doi.org/10.1016/j.ejor.2016.01.058.

[15] J. Liesio, P. Mild and A. Salo, "Preference programming for robust portfolio modeling and project selection", *European Journal of Operational Research*, 181(3), 1488–1505, 2007, http://doi.org/10.1016/j.ejor.2005.12.041.

[16] J. Liesio, P. Mild and A. Salo, "Robust portfolio modeling with incomplete cost information and project interdependencies", *European Journal of Operational Research* 190(3), 679–695, 2008, https://doi.org/10.1016/j.ejor.2007.06.049.

[17] F. Balderas, E. Fernandez, C. Gomez and L. Cruz-Reyes, "Metaheuristic robust optimization of project portfolios using an interval-based model of imprecisions", *International Journal of Combinatorial Optimization Problems and Informatics* 7(3) 101–118, 2016, Available at https://www.ijcopi.org/index.php/ojs/article/view/32.

[18] F. Balderas, E. Fernandez, C. Gómez, L. Cruz, Rangel, N., Morales, M.: "A grey mathematics approach for evolutionary multi-objective metaheuristic of project portfolio selection", *Fuzzy logic augmentation of neural and optimization algorithms: theoretical aspects and real applications*, Springer Cham, 379-388, 2018.

[19] A. Toppila and A. Salo, "Binary decision diagrams for generating and storing non-dominated project portfolios with interval-valued project scores", *European Journal of Operational Research* 260(1) 244–254, 2017, http://doi.org/10.1016/j.ejor.2016.12.019.

[20] M. Rabbani, M. A. Bajestani and G. B. Khoshkhou, "A multi-objective particle swarm optimization for project selection problem", *Expert Systems with Applications* 37(1) (2010) 315–321, https://doi.org/10.1016/j.eswa.2009.05.056.

[21] A. L. Medaglia, S. B. Graves and J. L. Ringuest, "A multiobjective evolutionary approach for linearly constrained project selection under uncertainty", *European Journal of Operational Research* 179, 869–894, 2007, https://doi.org/10.1016/j.ejor.2005.03.068

[22] F. Balderas, E. Fernandez, C. Gómez., G. Rivera, & L. Cruz-Reyes. "Uncertainty modelling for project portfolio problem using interval analysis." *International Journal of Combinatorial Optimization Problems and Informatics* 7, 3 (2016): 20-27.

[23] Q. Zhang, L. Hui, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", *IEEE Transactions on Evolutionary Computation*,712-731, 2007, https://doi.org/10.1109/TEVC.2007.892759

[24] B. Miller and D. Goldberg, "Genetic Algorithms, Tournament Selection, and the Effects of Noise". *Complex Systems* 9: 193-212, 1995.

[25] J. Holland. Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press, 1975.

[26] R. Díaz. Optimización de estructuras reticuladas planas de madera mediante algoritmos genéticos. Valdivia, Chile, 2014.

[27] F. Wilcoxon. "Comparaciones individuales por métodos de clasificación". *Boletín biométrico*,80-83, 1945.