University Of AL-Qadisiyah

Available online at www.qu.edu.iq/journalcm

JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS

ISSN:2521-3504(online) ISSN:2074-0204(print)

Strength Pareto Evolutionary Algorithm Based on New Fitness Strategy for Multi Objective Knapsack Problem

Muna Shaker Salman

Middle Technical University, Technical Institute- Suwaira , Iraq

Email: Muna.shaker@mtu.edu.iq

ARTICLE INFO

Article history:
Received: 01 /09/2022
Rrevised form: 30/09/2022
Accepted : 04 /10/2022
Available online: 1 /12/2022

Keywords:

Multi objective Optimization , Evolutionary Algorithms, EMO, SPEA2

ABSTRACT

problems. Many different experiments have shown that the evolutionary algorithm Strength Pareto Evolutionary Algorithm 2 (SPEA2) outperforms other approaches, and this makes it an excellent candidate for inclusion in the final design. According to Strength, one of SPEA2's core concepts, the population is sorted into niches based on the Pareto Front idea. With regard to outcomes this technique has a flaw that is mitigated by adding a fitness density estimator. Weakness in Strength is addressed with an approach called Strength by objective, which aims to include solution who do not dominate or are dominated by others inside the process. In this paper, the results will show a clear superiority of the proposed method comparing with the original method in solving the multi-objective Knapsack problem using three sizes of the Knapsack and the problem size of 750 items which is generated randomly. The comparison results between the proposed method and the other algorithms show outperformance of the Proposed algorithm using the dominance inductor showed the superiority of the proposed algorithm by a percentage of more than 1 %, Thanks to a highly diverse population and the inclusion of solutions that can be improved, the (SPEA2) algorithm's performance has been vastly enhanced by the Objective sorting mechanism

MSC..

https://doi.org/ 10.29304/jqcm.2022.14.4.1096

1. Introduction

A multi-objective optimization issue is one in which the goal is to optimize numerous functions in the same search space, but no weight or priority may be assigned to any of the functions. Also, in many circumstances, these roles are in conflict with one another. ' Schaffer F2 (Figure 1) is a good example of a function that may be used to find

∗Corresponding author

Email addresses:

Communicated by 'sub etitor'

the same real number X that minimizes the values of both $h(\mathrm{x})$ and $g(\mathrm{x})$ equation (1) . If $h(\mathrm{x})$ travels toward its global minimum, $g(\mathrm{x})$ moves away from its global minimum, and vice versa, as may be observed at first look [1].

NOMENCLATURE

Aradius of

Bposition of

Cfurther nomenclature continues down the page inside the text box

$$f(x) = \begin{cases} g(x) = x^2 & \rightarrow \text{Global minimum} \quad g(0) = 0 \\ h(x) = (x-2)^2 & \rightarrow \text{Global minimum} \quad h(2) = 0 \end{cases} \qquad (1)$$

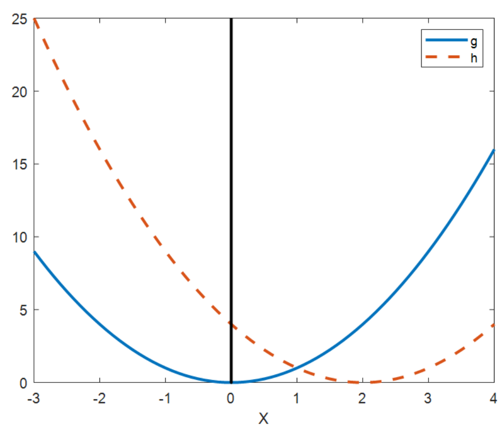


Figure. 1. Schaffer f2

Let's look at the following comparison:

$$\text{for} \quad X = 0 \rightarrow f(0) = (0{,}4)$$
$$\text{for} \quad X = 2 \rightarrow f(2) = (4{,}0)$$

Neither alternative is preferable if we adhere to the idea of equal importance for all goals. Therefore, the Pareto Front is generally a set when solving a multi-objective issue. In this collection, only those solutions that can't be improved further by any other means are included. Non-dominated solutions are another name for these pareto-optimal solutions. The most popular solutions are those that, when combined with another, improve on all of their goals. To solve multiobjective optimization problems, the second version of Strength Pareto Evolutionary Algorithm (SPEA2) [2] has been widely employed. SPEA [1] has been demonstrated to be superior to other techniques, such as NSGA-II [3], which is one of the first possibilities to examine when presenting an optimization issue of this kind. In comparison to previous multi-objective evolutionary algorithms, SPEA2 has a few key differences, two of which are particularly important for assessing an individual's quality or fitness:

- A approach for assigning raw fitness or raw fitness, referred to as Strength in this study, that takes into consideration for each solution how many solutions dominate (strength) and how many others are dominated (dominance), is described in detail in the next section (blows received).

- When estimating the density of data, a technique known as the *K*-th nearest neighbor is applied, which refines the raw fitness and provides a more precise direction to the search process.

It is possible to break down the raw fitness allocation approach into two parts: Firstly, it establishes the power of each solution, which is equal to the number of other solutions who are controlled by that one. In the second round, each solution will be hit by all of the solutions who are in a position of dominance over him. The force of these hits is determined by the power of the solution delivering them[2].

The total of the hits received equals the individual's raw fitness. The solution with the fewest hits wins, showing that the raw fitness function is to reduce.

This strategy's knowledge reveals a detail. If the majority of solutions do not dominate each other, vast groups of individuals will have the same fitness, making it impossible to classify the population accurately, and the individuals will be picked at random. SPEA2 adds the density estimator to raw fitness, allowing solutions with the same fitness to be separated. [1]

But also, this strategy does not take into account the individuals who are not dominated or dominate others, since they do not give strength to any individual and also do not hit anyone. It can even be said that the existence of these individuals does not affect the fitness allocation strategy of the rest of the population at all. Information is definitely being lost here.[3]

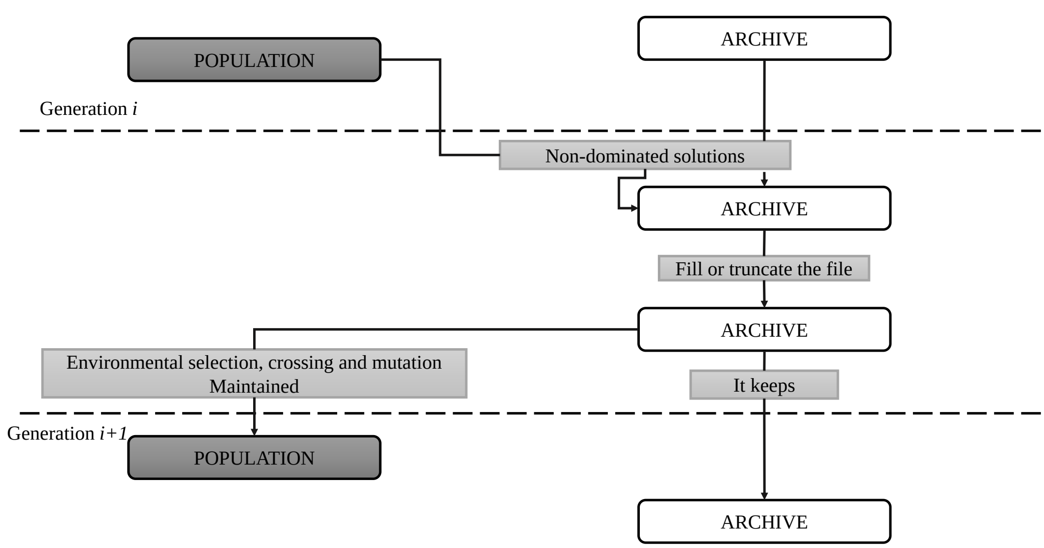


Figure 2. Main algorithm of SPEA2[1]

This heuristic technique presents a simple evolutionary algorithm, shown in Figure 2 that has the following execution thread principal:

1. Calculate the quality or fitness of the Population as well as the external File. n
2. Make changes to the External File.
3. Run a binary tournament using just the components from the External File as a starting point.
4. Cross the champions of the event in order to create a new population.
5. Restart the process from step 1 until the stop condition is reached.

2. Strength Pareto evolutionary algorithm SPEA2

2.1. The SPEA2 Fitness Assignment Strategy

This is the strong point of heuristics. SPEA2 as it presents a solution assessment system based on dominance between vectors. Each individual has two attributes:

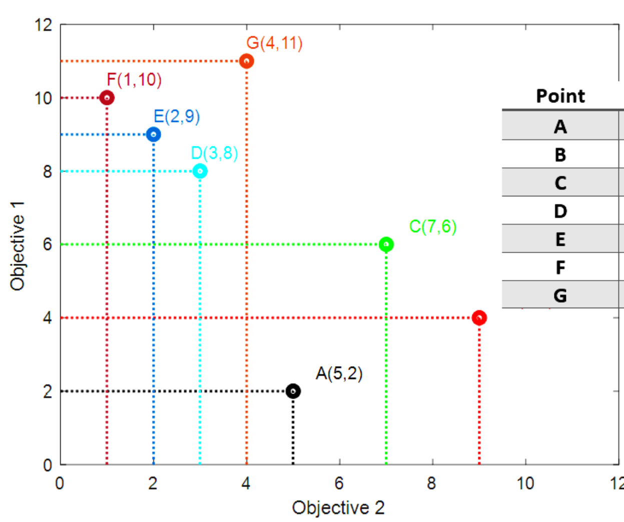
a) Force or Strength: Represents the number of individuals dominated by it.
b) Raw Fitness: It is the sum of the forces of the individuals who dominate it.

Population strengths are first allocated to determine how many other individuals they dominate, and then the fitness of the whole population and the external file is calculated. After that, the raw fitness of each solutions is determined by adding together the strengths of those who outperform him. Fitness is to be minimized, not maximized. Figure 3 shows that no one has dominance over the individuals in the pareto front, hence their raw fitness is 0 . It's possible for a population of people to create huge groups of individuals with the same level of fitness, resulting in an algorithm that selects a big number of individuals with a high degree of unpredictability. Raw fitness is supplemented with the density estimator *K-th* nearest neighbor to smooth out the noise. In the first step, the distance between each solution in the population and an external file in solution space must be determined for each individual.

With this we obtain a distance vector $(\mathrm{dist}(\times))$, which must be ordered from least to greatest. Then the K-th element of the vector is taken (where k = $\sqrt{\text{populationSize} + \text{archiveSize}}$) and it is placed in the following formula:

$$D(i) = \frac{1}{\mathrm{dist}(k) + 2} \qquad (2)$$

The constant 2 that is added in the denominator is to ensure that the density estimator is always less than zero. With this it can be said that every pareto-optimal individual has a fitness lower than 1.



Point	Strength	fitness
A	0	2
B	1	0
C	1	0
D	0	3
E	0	3
F	0	3
G	3	0

Figure 3. Fitness mapping used by SPEA2

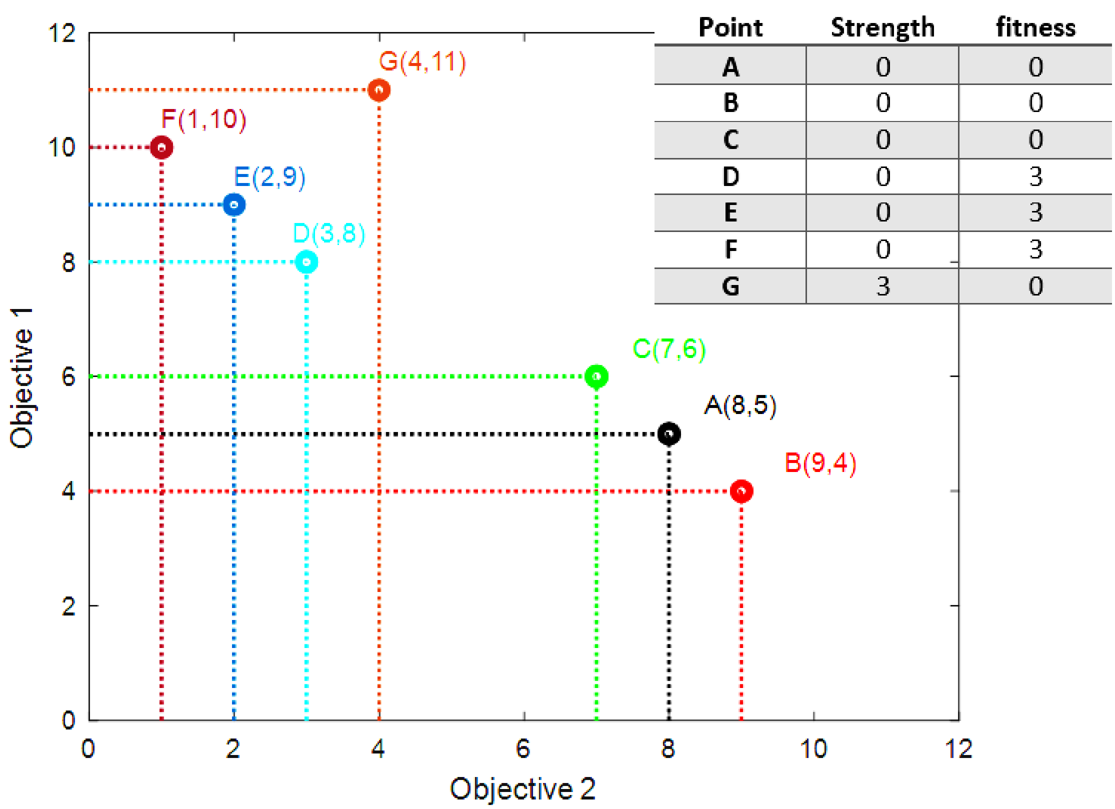
2.2. Updating the External Archive

The External Archive is a list of the best solutions that have been achieved up to the current generation. The update process for this external file complies with the following steps:

a) The non-dominated solutions of the current Population and the external File are calculated.
b) A new external file is generated with the solutions obtained in the previous step.
c) If the size of the new External File exceeds a preset size, a truncation operator based on the Cartesian distance of vectors is used. If the file size is less than this size, the External File is filled with the best individuals in the Population, based on the goodness or fitness.

2.3 *The Binary Tournament (With Replacement)*

The external Archive is the only source of information for the selection process. The population for the next iteration of the algorithm is made up of the people created by the crosses. It is a single-point crossover with random points that is utilized in the crossover operator .



Point	Strength	fitness
A	0	0
B	0	0
C	0	0
D	0	3
E	0	3
F	0	3
G	3	0

Figure. 4. A detail in the fitness assignment of SPEA2

3. *Strength by Objective*

Individuals A, C, B, and C in Figure 4 do not dominate or be dominated by anybody else. As a result, it has little physical strength or fitness (0). There is a dominant connection with at least one person in the population, although this is rare.

Individuals A, B, and C are entirely disregarded by the algorithm when the raw fitness of all individuals is determined. Thus, even if we took away those points, the raw fitness of the remaining members of the population would stay unchanged. As a result, it is reasonable to assume that the algorithm has some information loss, as some people are unable to alert the rest of society of its existence. [7]

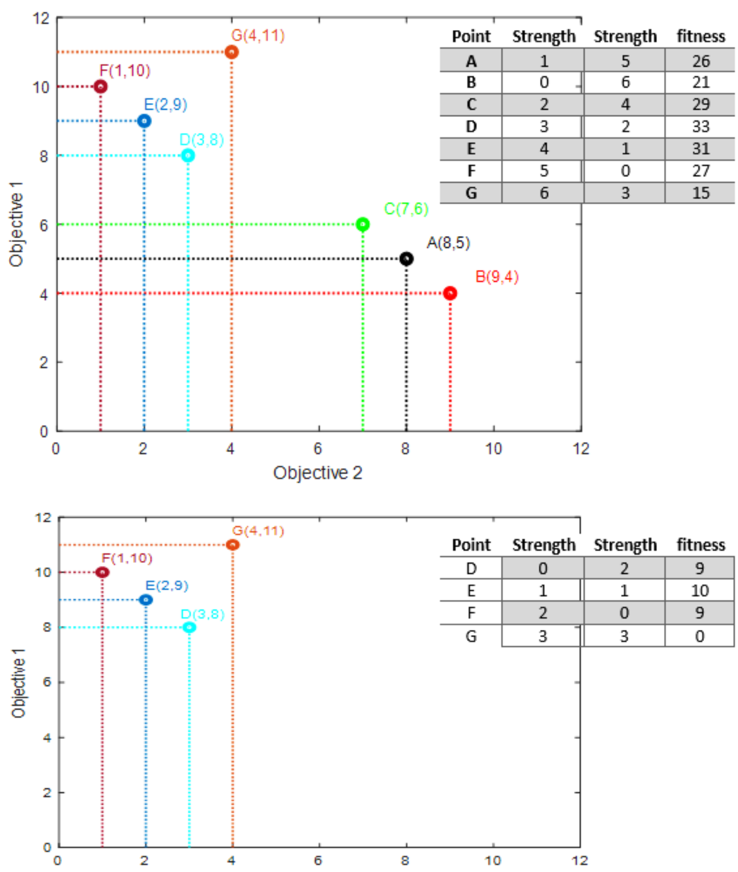
Strength by Objective is a new fitness allocation strategy that seeks to minimize the weaknesses presented by Strength, addressing these two problems:

- Large groups of individuals with the same fitness.
- Individuals who are ignored by the algorithm.

This fitness allocation strategy addresses the problem in a more granular way, taking into account the quality of individuals in each of the functions to be optimized [9].

Strength by Objective uses attributes equivalent to those of SPEA2 for each individual, although they are calculated differently:

1. Strength per target: This attribute indicates how many individuals in the population the individual defeats on a particular target. Note that there is a force for each objective to be optimized.
2. Hits received: Every time an individual is defeated by another on a target they receive a hit. The magnitude of this blow depends on the strength of the victor for that goal. This attribute is the sum of the blows received by an individual [6].



Point	Strength	Strength	fitness
A	1	5	26
B	0	6	21
C	2	4	29
D	3	2	33
E	4	1	31
F	5	0	27
G	6	3	15

Point	Strength	Strength	fitness
D	0	2	9
E	1	1	10
F	2	0	9
G	3	3	0

Figure 5. Strength by Objective [8]

Algorithm.1 the calculation of Strength [8]

Strength calculation method (Individual A)

For each individual B of the population and the Archive

For each objective i to optimize

If (A. vector Solution $[i]$ is better than B. vector Solution $[i]$)

$A.force\ [i]\ ++;$

First, the forces of the individuals are calculated (algorithm 1). For this, each one of them is compared with the rest of the population and it is determined how many individuals' beats in each objective to be optimized. Then, how many blows an individual receives are determined by comparing each individual with the rest of the population (algorithm 2). In each comparison, if the individual beats another in the population on a target, he hits him with whatever force he has for that target. The goodness or fitness of an individual is the amount of hits received. The lower the fitness, the better the individual.

Algorithm.2 Fitness allocation algorithm

Hitting Method (Individual A)

For each Individual B of the population and the file

For each objective i to optimize

If (A .vector Solution $[i]$ is better than B.vector Solution $[i]$)

$B.StrokesReceived = B.StrokesReceived + A.force\ [i];$

The whole population is involved in this approach, since each person must fulfill the criteria of either losing or winning in a particular aim to be a part of the process. With this method, it is also less likely that big groups of people would have the same fitness level.

Strength By Objective establishes a new social hierarchy, which is an important aspect to consider. A, B, C, and G in figure 5 all belong to the pareto front, yet their fitness is radically different without the requirement for any additional technique, such as the density estimator, to be implemented.

4. Strength vs Strength by Objective

The way to determine if the new proposal addresses the problem correctly is by subjecting it to an experimental comparison. SPEA2's Strength fitness allocation strategy will be used as a point of comparison.

4.1 The Algorithm

An implementation of SPEA2 was made following the specifications of the work of Zitler , Leummans and Thile, which can use Strength or Strength by Objective as a raw fitness allocation strategy. It is important to note that both strategies will not use the density estimator in order to make an exclusive comparison between raw fitness.

4.2. Test description

Both strategies were used to solve three approaches to the "Knapsack problems" (combinatorial problem) with 750 elements whose values and weights are randomly generated. Individuals are represented with a binary chromosome. The parameters are specified in Table 1.

Table 1. Parameters of the tests to be carried out (Knapsacks problems)

	Two Knapsack	***Three*** Knapsack	***Four*** Knapsack
Population size	**250**	**300**	**400**
Archive size	**250**	**300**	**400**
Mutation	**0.006**	**0.006**	**0.006**
Generations	**Fifty**	**Fifty**	**Fifty**

4.3. Measurement variables

- Area covered: This criterion indicates the amount of area of the solution space covered by the given pareto front. In a two-dimensional space, the area covered will be equal to the area of the union of the rectangles formed by each of the vectors of the pareto front and the origin (0,0). This criterion can be canonically extended to N dimensions. Figure 5 shows an example.

- Dominance: Given two sets against pareto A and B, the percentage of dominance of A / B indicates the percentage of solutions of B that are dominated by at least one of the solutions of A.

4.4. Results

These results are the average obtained from ten (10) independent runs of the algorithm for each approach.

- **Covered area**

Table 2. Average results of covered area

Strategy	***Two*** Knapsack	***Three*** Knapsack	***Four*** Knapsack
Strength	**6.62 E+10**	**1.42 E+16**	**3.04 E+21**
Strength by Objective	**6.58 E+10**	**1.41 E+16**	**2.93 E+21**
Difference	**4.5 E+8**	**1.25 E+14**	**1.06 E+20**

- **Dominance**

Table 3. Average dominance results

Strategy	***Two*** Knapsack	***Three*** Knapsack	***Four*** Knapsack
Strength/ Strength by Objective	**45.38 %**	**28.25%**	**21.50%**

Strength by Objective/ Strength	**44.50 %**	**29.38%**	**62.50%**
Difference	**0.88%**	**1.13%**	**41.00%**

4.5. Analysis

The results obtained in the experimental test put Strength by Objective in a very good position. Using the two measurement variables, it can be seen that Strength by Objective improves its performance with respect to Strength by increasing the number of targets. From the average covered area (Table 2) it is observed that Strength remained above Strength by Objective by a slight difference. With this it can be said that both strategies behave the same. But dominance reveals that there is a difference. In Table 3, it can be observed how Strength by Objective begins to gain ground over Strength by increasing the number of objectives, reaching the point of obtaining a difference of 41 percentage points for the problem of four backpacks (Four Knapsacks).

6. Conclusion

For multi-objective algorithms, "Strength By Objective" is a fitness assignment strategy that takes into account an individual's quality for each function to be optimized. To avoid losing the valuable information that each individual represents, this new proposal builds on the success of SPEA2 by creating a new classification of populations that requires all members of the population to work together on their evaluation. But Strength by Objective also ensures the information spreads, since one person can sway the opinions of others by setting up a win-lose dynamic in any given objective. Based on experimental comparisons of the two strategies, we can say that the new proposal improves upon SPEA2 by increasing the number of functions to optimize, and that in the worst case, both strategies behave similarly

References

[1]. Zitzler, Eckart, and Lothar Thiele. " Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach." *IEEE transactions on Evolutionary Computation* vol. 3, no.4,257-271,(1999), 10.1109/4235.797969.

[2]. Zitzler, Eckart, Kalyanmoy Deb, and Lothar Thiele. "Comparison of multiobjective evolutionary algorithms: Empirical results." *Evolutionary computation* vol.8,no.2 ,173-195, (2000), 10.1162/106365600568202.

[3]. Deb, Kalyanmoy, et al. "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II." *International conference on parallel problem solving from nature*. Springer, Berlin, Heidelberg, 2000.

[4]. Deb, Kalyanmoy, et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II." *IEEE transactions on evolutionary computation* vol.6,no.2 , 182-197, (2002), 10.1109/4235.996017.

[5]. Mansour, Imen Ben, Mathieu Basseur, and Frédéric Saubion. "A multi-population algorithm for multi-objective knapsack problem." *Applied Soft Computing* vol. 70,814-825,(2018), https://doi.org/10.1016/j.asoc.2018.06.024

[6]. Luo, Rong-juan, Shou-feng Ji, and Bao-lin Zhu. "A Pareto evolutionary algorithm based on incremental learning for a kind of multi-objective multidimensional knapsack problem." *Computers & Industrial Engineering* , vol. 135: 537-559, (2019),https://doi.org/10.1016/j.cie.2019.06.027

[7]. Osawa, R., Watanabe, S., Hiroyasu, T., & Hiwa, S. (2019, December). Performance Study of Double-Niched Evolutionary Algorithm on Multi-objective Knapsack Problems. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 1793-1801). IEEE, 10.1109/SSCI44817.2019.9003130.

[8]. Zapotecas-Martínez, S., & Menchaca-Méndez, A. (2020, July). On the Performance of Generational and Steady-State MOEA/D in the Multi-Objective 0/1 Knapsack Problem. In *2020 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1-8). IEEE, 10.1109/CEC48606.2020.9185715.

[9]. Jiang, S., & Yang, S. (2017). A strength Pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization. *IEEE Transactions on Evolutionary Computation*, *21*(3), 329-346.