IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 2 September 2024, accepted 23 September 2024, date of publication 25 September 2024,
date of current version 9 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3467997

METHODS

Improvement Strategies for Visualizing Solution Sets in Many-Objective Optimization Problems

CHRISTIAN VON LÜCKEN, URIEL PEREIRA, ENRIQUE JAVIER DÁVALOS, (Member, IEEE), AND FABIO LÓPEZ-PIRES
Facultad Politécnica, Universidad Nacional de Asunción, San Lorenzo 111421, Paraguay
Corresponding author: Christian von Lücken (clucken@pol.una.py)

This work was supported in part by the CONACYT under Grant PINV18-949.

ABSTRACT In real-world multi-objective optimization, dealing with many objectives and a large number of solutions is a common challenge that complicates data visualization and analysis. This study aims to simplify decision-making by analyzing tools to better explore Pareto optimal solutions in many-objective scenarios, integrating clustering, filtering, and ranking with existing graphics techniques. The dynamic combination of these tools should reduce complexity and highlight significant patterns in the data set, allowing decision-makers to tailor the visualization to their specific needs and preferences. Central to the approach presented in this work is the innovative application of shape-based clustering to organize the solution set and the use of this clustering to define distinct types of filters. Additionally, ranking methods originally proposed to enhance search in many-objective evolutionary algorithms are used here to identify the best solutions based on predefined criteria in combination with other techniques. The efficacy of the proposed integrated approach was evaluated using an application developed with this aim and considering a five-objective problem as a case study. The analysis suggests that using these combined strategies aids interactive visual exploration, effectively reducing solution volume and improving data understanding, potentially facilitating decision-making tasks.

INDEX TERMS Multi-objective optimization, many-objective optimization, data visualization, shape-based clustering, decision-making, filtering methods, ranking methods, interactive visual exploration.

I. INTRODUCTION

Multi-objective optimization problems (MOPs) involve more than one conflicting objective function, which should be minimized or maximized, such that an improvement in one criterion often comes at the expense of at least one another, creating inherent trade-offs. Consequently, MOPs typically yield a set of compromise solutions rather than a single optimal solution known as the Pareto Optimal set. The representation of these solutions in the objective space is termed the Pareto Optimal Front.

When a set of viable solutions contains many alternatives, decision-makers must analyze and evaluate them to identify the one that best fits their needs. Visual analysis plays a crucial role in this process, as the human brain is better at processing visual information and quickly identifying patterns [1], [2], [3]. By visualizing the obtained solutions, decision-makers can thoroughly explore alternative solutions. This exploration allows them to estimate the range of solutions for each objective, gain insights into the location and shape of the Pareto front, assess the conflicts and trade-offs between objectives, and select the most suitable solutions from the available options [2].

Many-objective optimization problems (MaOPs) are MOPs involving four or more objectives. Several visualization methods are available for inspecting solution sets in many-objective optimization problems. A survey by von Lücken et al. [4] broadly categorized these methods into three types: (i) those that display objectives in groups of two or three at a time; (ii) those that show all objectives simultaneously in a single graph; and (iii) those that apply dimensional reduction techniques to the objective space

The associate editor coordinating the review of this manuscript and approving it for publication was Mauro Gaggero.

before plotting, aiding in quantitative analysis. Additionally, Filipič and Tučar [3] classify these methods into two groups: one showing actual original or normalized values and another transforming these values into alternative representations for visualization. Popular methods in the first group include the Scatter Plot Matrix, which displays objectives in groups, and Parallel Coordinates [5], [6], which represents the complete set of objectives in a single graph. In the category of transformed values, Radial Coordinate Visualization or RadViz [7] is notable for its ability to display the complete set of objectives in a single graph.

As the number of problem objectives increases, so does the complexity and the number of alternative solutions. This escalation in complexity often leads to problems such as data overlap, difficulty discerning clear patterns, and other challenges individuals face when the data to visualize increases [8]. To mitigate these challenges, researchers have proposed refined visualization techniques. These enhancements to simplify and clarify data presentation include adjusting visualization axes [9], introducing three-dimensional representations [10], reducing the number of objectives [11], and mapping higher-dimensional variable spaces into lower dimensions [12]. Designers of these strategies aim to help users navigate and understand the intricate data landscapes typical in many-objective optimization problems more effectively [13].

Previous works have explored clustering techniques for visual analysis in many-objective optimization [11], [13], [14], and the use of filtering by value range to enhance visualization [15], [16]. Several evolutionary algorithms have incorporated various ranking methods to search for solutions, which differ from the traditional non-dominance ranking. These methods often reflect the decision-maker's (DM) preferences during the search process. However, integrating these ranking methods with display techniques still needs to be explored, highlighting potential research areas in visualizing many-objective optimization solutions.

This work introduces a method that integrates clustering, filtering, and ranking strategies with established visualization techniques to enhance the analysis and representation of solutions in many-objective optimization problems. The approach employs Shape-based clustering [17] as the basis for nuanced solution selection, enabling the definition of various filtering methods based on this clustering. Additionally, it utilizes K-means [18] and provides the ability of manual classification to complement and refine the selection process based on defined criteria. The framework also includes filtering methods to categorize solutions by group attributes and specific value ranges. It employs two ranking approaches: *favour* and ϵ*-preferred*, to aid decision-makers in evaluating and prioritizing solutions effectively.

This study illustrates the enhancement proposal using an application specifically developed with this aim, which is available upon request. Considering the solutions of a many-objective optimization problem named Water [19], the developed application served to examine the possible impact of using these strategies in various sequences, reflecting the diverse approaches a decision-maker might take during the visual exploration. Interestingly, the results indicate that the strategic combination of these methods can effectively reduce the volume of data presented, tailoring it more precisely to the specific needs of the decision-maker. This finding highlights the usefulness of our approach in simplifying complex data sets, thereby aiding in more efficient decision-making in the context of MOPs.

This article's organization is as follows: Section II presents some existing visualization methods related to this work, providing an overview and evaluation of their relevance and application in many-objective optimization problems. Section III introduces and details the proposed enhancement strategies: clustering, rankings, filters, and their integration to improve the visual analysis and decision-making in many-objective optimization. Section IV illustrates these proposed strategies' practical application and results through a case study based on a specific many-objective problem, demonstrating their effectiveness and utility. The last section summarizes the conclusions drawn from this study and outlines potential avenues for future work, suggesting ways to extend and refine the methodologies presented.

II. RELATED VISUALIZATION METHODS

Scatter Plot is one of the most popular display methods in multi-objective optimization due to its simplicity of implementation and interpretation. This two-dimensional graphic effectively represents solutions by using points plotted along perpendicular axes. However, its effectiveness diminishes when dealing with more than two dimensions, as the Scatter Plot is inherently limited to two-dimensional data representation. By varying the size and color of the points, one can integrate additional dimensions into the standard Scatter Plot, thus transforming it into a Bubble Chart. This modification enables the representation of up to five dimensions, providing a more comprehensive view of the data [2]. The Bubble Chart, therefore, enhances the Scatter Plot's utility in visualizing complex, multi-dimensional data sets typical in multi-objective optimization.

The Scatter Plot Matrix is a grid layout of individual scatter plots, each showing objectives in pairs or with a reduced set of them, ignoring all the vector dimensions other than the ones considered in each subplot. This fast, simple, and robust method makes it a valuable tool for initial data analysis. However, its practicality decreases with an increase in the number of dimensions as the matrix becomes cluttered with numerous rows and columns. A significant limitation of the Scatter Plot Matrix is its inability to provide a complete view of all objective relationships simultaneously, as it only plots pairs of objectives. In scenarios with many objectives, dimensional reduction techniques are employed to eliminate redundant objectives before plotting. While this approach helps manage complexity, it runs the risk of omitting potentially valuable information, as it simplifies the objective set.

Another notable display method is RadViz [7], which employs a physics-based approach. In this method, objectives are evenly spaced around the circumference of a unit circle, and each objective is conceptually attached to the solutions or vectors through 'springs.' The strength of these springs is proportional to the solution's performance in the corresponding objective. A solution's position within the circle represents a balance point where all these spring forces reach equilibrium. Consequently, solutions closer to a particular objective on the circle's perimeter indicate a higher performance in that objective compared to others. Solutions that perform uniformly across all objectives are positioned near the circle's center, effectively representing their balanced nature. RadViz excels in preserving the distribution of vectors, offering a unique perspective on their relative performance. 3D RadViz [10] is a transformation of RadViz that consists of adding a new dimension to the method; this dimension is the distance from the individuals to a hyperplane that passes through the extreme vectors of the set.

The Parallel Coordinates method is a visualization technique that represents each vector in a multi-dimensional space by connecting segments across parallel axes. Each axis corresponds to one dimension, and the position of a vertex on the i-th axis represents the vector's i-th coordinate. Unlike the Scatter Plot that uses orthogonal axes, the Parallel Coordinates method employs parallel axes, making it possible to represent sets of solutions with any number of dimensions on a two-dimensional plane. This method is advantageous because it can reveal conflicting or harmonious relationships between pairs of adjacent objectives, as noted in [5]. Various techniques were developed to enhance this visualization to strategically order the axes, thereby highlighting the relationships between different objectives more clearly [9], [20].

Lastly, the Star Coordinates method [21] extends the Scatter Plot to higher dimensions. It depicts objectives as axes radiating from a central point in a circular arrangement. Initially, these axes are of equal length and uniformly spaced, ensuring that each attribute contributes equally to the data representation. To enhance the analysis, users can apply various transformations, such as scaling the axes to adjust the prominence of specific attributes or altering the angles between axes to modify the interrelationships among attributes. These transformations allow for a dynamic and customizable approach to understanding complex, multi-dimensional datasets.

III. PROPOSED IMPROVEMENT STRATEGIES

Although various visualization methods exist, the challenge of efficiently simplifying the visual exploration of solutions in scenarios with more than three objectives persists. This paper introduces a novel approach to address this gap, combining Parallel Coordinates with clustering, ranking, and filtering techniques. The objective is to organize information effectively in high-dimensional spaces, facilitating decision-makers' exploration and visual analysis of solution sets with numerous objectives. The following section details these proposed techniques, illustrating how their integrated application can significantly enhance the decision-making process in complex many-objective optimization scenarios.

A. CLUSTERING

Clustering algorithms perform an unsupervised classification of unlabeled data, aiming to categorize a dataset into distinct groups or clusters. The primary objective of clustering is to group elements with similar characteristics while ensuring that elements from different clusters are as dissimilar as possible. In multi-objective optimization, this classification aids decision-makers in exploring and understanding solutions at a broader level, focusing on group characteristics rather than individual data point attributes. This approach aligns with the natural human inclination to comprehend new objects or phenomena by identifying and comparing features based on similarity or dissimilarity, a concept often referred to as proximity [22]. Moreover, clustering is critical in identifying unusual patterns within a dataset, such as notable disparities in cluster sizes or the presence of outlier elements.

Several studies explored using clustering to visually depict solutions in multi-objective optimization scenarios [15], [23], [24], [25], [26], [27]. A prominent example is the Self-organizing Map (SOM) [24], an unsupervised learning algorithm that maps multi-dimensional input data to a two-dimensional output space, preserving data's topological relationships. This attribute of SOM makes it particularly effective for visualizing complex sets of solutions. In [27], authors present an interpretable self-organizing map (iSOM) method that produces a more simplistic mapping of higher-dimensional variable spaces into two dimensions. Additionally, some proposals integrate clustering with RadViz [10] and Parallel Coordinates [15], [25], [28]. These integrative approaches foster a more structured and insightful visual representation of solutions, equipping decision-makers to identify underlying patterns and associations that might not be immediately apparent within complex, high-dimensional data sets.

Among the various clustering algorithms, this work considers the K-means algorithm [29] for visual exploration for its simplicity and effectiveness. The K-mean algorithm aims to partition a set of m-dimensional vectors in $\mathbb{R}^m$ into k distinct groups by finding optimal central points in the space and minimizing the mean squared distance from each point to its nearest center. This method's simplicity and low computational complexity has led to its widespread application in various fields [30], such as machine learning, computer vision, and market segmentation.

Besides the K-means, clustering solutions by the shape of objective vectors [17], [25] is considered here both as a practical method for grouping solutions as well as to define filtering methods indicating user preferences, as Section III-C explains. To determine the shape of a real vector $y = y_1,$

TABLE 1. Example of obtaining the shape of objective vectors in a solution set.

Normalized Objectives S_i	Ordered version $o(S_i)$	Shape $\pi(S_i)$
$S_1 = [0.9; 0.1; 0.2; 0.3]$	$[0.1; 0.2; 0.3; 0.9]$	$[2, 3, 4, 1]$
$S_2 = [0.8; 0.3; 0.2; 0.9]$	$[0.2; 0.3; 0.8; 0.9]$	$[3, 2, 1, 4]$
$S_3 = [0.2; 0.4; 0.5; 0.8]$	$[0.2; 0.4; 0.5; 0.8]$	$[1, 2, 3, 4]$
$S_4 = [0.6; 0.5; 0.4; 0.2]$	$[0.2; 0.4; 0.5; 0.6]$	$[4, 3, 2, 1]$
$S_5 = [0.1; 0.6; 0.9; 0.7]$	$[0.1; 0.6; 0.7; 0.9]$	$[1, 2, 4, 3]$
$S_6 = [0.1; 0.5; 0.6; 0.9]$	$[0.1; 0.5; 0.6; 0.9]$	$[1, 2, 3, 4]$

$\ldots, y_m \in \mathbb{R}^m$, we define the pair $(o(y), \pi(y))$ where $o(y)$ is the ordered version of the elements y_i, and $\pi(y)$ is the permutation of the indices $1, \ldots, m$ achieving this ordering. A formal definition of the shape of a vector in $\mathbb{R}^m$ is as follows.

Definition 1 (Shape of a Vector in $\mathbb{R}^m$ [17]): given a vector $y \in \mathbb{R}^m$, a permutation $\pi(y) = \pi_1, \ldots, \pi_m, \pi_i \in 1, \ldots, m$, is the shape of y iff:

$$y_{\pi_i} \leq y_{\pi_j}, \forall i < j \tag{1}$$

To compare the values of the different objectives, these must be on the same scale. Therefore, normalization of objective values is carried out prior to determining the vector shape. Table 1 shows a set of solutions and their shapes for a minimization problem with $m = 4$ objectives.

Algorithm 1 outlines the shape-based clustering approach to categorize solutions into distinct groups. The algorithm starts by accepting an input set of solutions, denoted as S. This set contains multiple solutions, each with a set of objective values. Then, the objective values for each solution S_i are normalized. Normalization ensures consistency when dealing with values across different scales, preventing any objective from dominating others due to its range or scale. Following normalization, the algorithm identifies the shape of each vector S_i, leading to the recognition of γ distinct shapes.

Once these shapes are determined, the algorithm establishes γ empty clusters, denoted as $G_1, G_2, \ldots, G_\gamma$ corresponding to each unique shape. These clusters are to group solutions based on their shapes. For each unique shape sh_j in Sh, the algorithm goes through the solutions and assigns each solution $S_i \in S$ to a cluster G_j if the shape of S_i matches the current shape sh_j. This process ensures that the algorithm assigns every vector to a cluster corresponding to its shape.

Algorithm 1 Shape-Based Clustering Algorithm

```
Receive the set of solutions S
Normalize objective values of each solution S_i in S
For each S_i in S obtain its shape π_{S_i} = π(S_i)
Determine the set Sh = sh_1, sh_2, ..., sh_γ composed
  with the γ available shapes
Initialize G_1, G_2, ..., G_γ groups as empty
for each sh_j do
    G_j = {S_i | π_{S_i} = sh_j; S_i ∈ S}
end for
```

A distinctive feature of this method is the construction of an m-dimensional shape-preserving representative vector for each cluster by averaging the values of each dimension across all vectors in the cluster. This dual representation ensures that the representative vector reflects the average values of the objectives and mirrors the shape of all vectors in the cluster. Such a comprehensive representation allows decision-makers to quickly understand each cluster's quantitative and qualitative characteristics, facilitating more nuanced and efficient decision-making processes.

TABLE 2. Example of clustering solutions using the shape-based clustering.

Group	Shape	Objective Vectors
G_1	$[2, 3, 4, 1]$	$S_1 = [0.9; 0.1; 0.2; 0.3]$
G_2	$[3, 2, 1, 4]$	$S_2 = [0.8; 0.3; 0.2; 0.9]$
G_3	$[1, 2, 3, 4]$	$S_3 = [0.2; 0.4; 0.5; 0.8]$ $S_6 = [0.1; 0.5; 0.6; 0.9]$
G_4	$[4, 3, 2, 1]$	$S_4 = [0.6; 0.5; 0.4; 0.2]$
G_5	$[1, 2, 4, 3]$	$S_5 = [0.1; 0.6; 0.9; 0.7]$

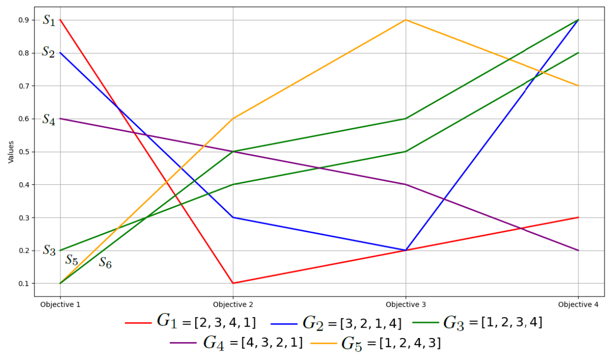


FIGURE 1. Parallel coordinates plot representing the shape-based clustering corresponding to the solutions in Table 2. The colors of the lines represent the shapes of the vectors.

Table 2 provides an example of clustering based on the shapes of solutions from the previous example. Figure 1 illustrates this clustering using a Parallel coordinates plot, where each line represents a vector. The lines' colors correspond to the shapes of the vectors, meaning that vectors with the same shape share the same color. This visual representation facilitates the comparison of multiple variables and highlights patterns and similarities among vectors with identical shapes, corresponding to the solutions in Table 2.

B. RANKINGS

Besides clustering, this work applies ranking techniques commonly used in many-objective optimization algorithms to improve the visual exploration and analysis process. The goal is to help decision-makers efficiently evaluate, compare, and prioritize solutions based on specific criteria. With this aim, this work examines the ranking method proposed by Drechsler et al. [31] based on the Satisfiability Class Ordering (SCO) and the *favour* relation, as well as a similar ranking method based on SCO and the *ϵ-preferred* relation [32]. Besides providing a relevant ranking of solutions, as noted in [31], these ranking methods achieve very low run times due to their efficient graph-based representation. We consider

TABLE 3. Example of *favour* relationships between pairs of solutions. In this table, '1' at row s_i, column s_j indicates $s_i \prec_{favour} s_j$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1	0	1	1	0	1	1
s_2	0	0	0	0	0	1
s_3	0	0	0	0	0	1
s_4	0	0	0	0	1	1
s_5	0	0	0	0	0	0
s_6	0	0	0	0	1	0

these characteristics to make them particularly suitable for interactive and iterative analysis.

The SCO method offers a framework for classifying solutions at different levels according to a defined relationship. The algorithm entails constructing a directed graph of the solutions, with each node representing a solution and a directed edge between two nodes indicating that one solution dominates the other. In the event of cycles, the method groups nodes into a single node. Then, the resulting graph determines a partial order of the solutions.

Favour and ϵ*-preference* relations are helpful in many-objective problems to consider an additional layer of information beyond non-dominance, facilitating the identification of subtle differences in their performance. For instance, the *favour* relation assesses which of two solutions performs better across a greater number of objectives, even if both are non-dominated. Similarly, the *epsilon-preference* [32] relation introduces a threshold to consider performance differences between solutions as significant. What follows are formal definitions and examples of these relations.

Definition 2 (Favour Relation [31]): Given two solutions $s, s' \in S$ with their respective returns $y = F(s), y' = F(s')$, it is said that s dominates with the relation *favour* to s', denoted as $s \prec_{favour} s'$, if and only if:

$$n_b(y, y') > n_b(y', y) \tag{2}$$

where

$$n_b(y, y') = |\{y_i | y_i < y'_i, \forall i \in [1, m]\}| \tag{3}$$

Table 3 shows the result of applying the *favour* relationship between each pair of solutions in the set we used in the example in Table 1. In Table 3, a one in the row corresponding to s_i and the column of s_j represents that there is a relationship of the type $s_i \prec_{favour} s_j$. Utilizing the data from this table, applying the SCO process facilitates the ordering of solutions. As a result of this ordering, the consequent ranking is illustrated in Figure 2.

Based on the *favour*, the *epsilon-preference* relation [32] compares two solutions taking into account the number of objectives in which a solution has better performance than the other giving a threshold. When solutions tie, the *favour* relationship determines which solution is better.

Definition 3 (Relationship ϵ-Exceed [4], [32]): Let s y $s' \in S$, $y = F(s), y' betwosolutions = F(s')$, a specified vector of limit values $\epsilon = (\epsilon_1, \ldots, \epsilon_m)$, s is said to dominate with

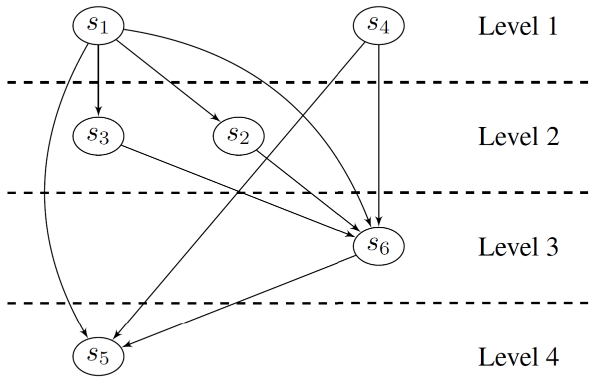


FIGURE 2. Ranking using SCO and the *favour* relationship between each pair of solutions.

TABLE 4. Example of the relationship $s \prec_{\epsilon-exceed} s'$ between pairs of solutions, a "1" in a cell at row s_i and column s_j, represents $s_i \prec_{\epsilon-exceed} s'_j$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1	0	1	1	0	1	1
s_2	0	0	0	0	1	0
s_3	0	0	0	0	1	0
s_4	0	1	0	0	1	0
s_5	0	0	0	0	0	0
s_6	0	0	0	0	1	0

TABLE 5. Example of the relationship $\epsilon -$ *preferred* between pairs of solutions. The relationship $s_i \prec_{\epsilon-pref} s_j$ is represented by "1" at row s_i and column s_j.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1	0	1	1	0	1	1
s_2	0	0	0	0	1	1
s_3	0	0	0	0	1	1
s_4	0	1	0	0	1	1
s_5	0	0	0	0	0	0
s_6	0	0	0	0	1	0

relation ϵ*-exceed* to s', denoted as $s \prec_{\epsilon-exceed} s'$ iff:

$$\begin{aligned} &|\{i | y_i < y'_i \wedge |y_i - y'_i| > \epsilon_i, \forall i \in [1, m]\}| > \\ &|\{i | y'_i < y_i \wedge |y'_i - y_i| > \epsilon_i, \forall i \in [1, m]\}| \end{aligned} \tag{4}$$

The definitions 2 and 3 are combined to form the relationship ϵ*-preferred* as follows:

Definition 4 (ϵ-Preferred Relation [4]): Given two solutions s and $s' \in S$, it is said that s dominates with relation ϵ*-preferred* to s', denoted as $s \prec_{\epsilon-pref} s'$ ssi:

$$s \prec_{\epsilon-exceed} s' \vee (s' \not\prec_{\epsilon-exceed} s \wedge s \prec_{favour} s') \tag{5}$$

Consider again solutions from Table 1 and $\epsilon_i = 0.2$ $\forall i \in [1, 4]$. To determine the ϵ*-preferred* between each pair of solutions, the solutions are first compared considering $\prec'_{\epsilon-exceed}$ as Table 4 shows where a 1 in a cell in row i and column j indicates a relationship of the type $s_i \prec_{\epsilon-exceed} s_j$.

Finally, using the information in Tables 4 and 3 in conjunction with Equation 5, the relationship $\epsilon -$ *preferred* can be obtained between each pair of solutions in Table 5.

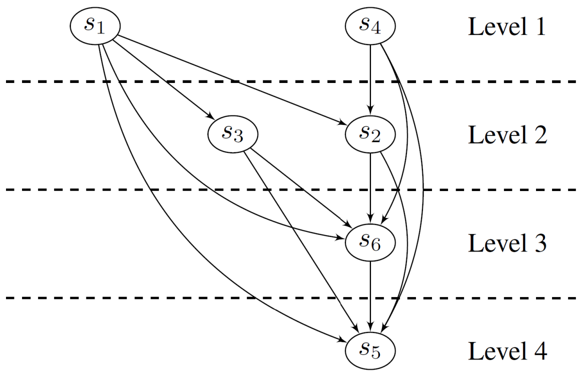


FIGURE 3. *S* **ranking using SCO and** ϵ**-*preferred* with** $\epsilon_i = 0.2 \forall i \in [1, 4]$.

TABLE 6. **Example of filtering solutions with objective 2 within the range [0.3, 0.5].**

Filter	F_{R_1}
Parameters	$j = 2$ $\alpha = 0.3$ $\beta = 0.5$
Results	$S_2 = [0.8; 0.3; 0.2; 0.9]$ $S_3 = [0.2; 0.4; 0.5; 0.8]$ $S_4 = [0.6; 0.5; 0.4; 0.2]$ $S_6 = [0.1; 0.5; 0.6; 0.9]$

Figure 3 shows the classification of solutions considered in the examples, using SCO and relation ϵ-*preferred*. Note that applying the relation ϵ - *preferred* with $\epsilon_i = 0 \ \forall i \in [1, m]$ is equivalent to applying SCO with the relation *favour*.

C. FILTERS

Filtering data for visualization simplifies the analysis by reducing the volume of information. A commonly used filter is based on the range of values that can be defined as follows.

Definition 4 (Filter by a Range of Values):

$$F_R = \{s_i | \alpha \leq S_i^j \leq \beta, \forall s_i \in S\}$$
$$1 \leq j \leq m$$
$$\alpha \leq \beta \tag{6}$$

where S_i^j is the value at position j of the performance vector S_i, α and β are the value constraints of the filter.

Table 6 illustrates the application of a filter based on a specific range of values, targeting solutions whose value in the second objective ($j = 2$) falls within the interval $[\alpha = 0.3; \beta = 0.5]$. All the values of S_i are normalized within the range $[0, 1]$.

Although range-based filters are helpful and easy to use, this study advocates using filters based on previously described clustering strategies. Since clustering categorizes solutions into distinct groups, filter by groups is defined as follows.

Definition 5 (Filter by Groups):

$$F_G = \{s_i | s_i \in G\} \tag{7}$$

where G represents the group to be filtered.

A filter proposed in this work based on the shape of the solutions is the priority filter.

Definition 6 (Filter by Priority Based on Shape): can be defined as:

$$F_P = \{S_i | \tau(S_i, o_i) > \tau(S_i, o_j), \forall s_i \in S\}$$
$$1 \leq i, j \leq m$$
$$i \neq j \tag{8}$$

where $\tau(S_i, o_i)$ and $\tau(S_i, o_j)$ are the positions of the shape $\pi(S_i)$ for objectives o_i y o_j, respectively.

Table 7 shows an example of applying a priority filter based on the shape, having as priority objective two over objective number 3.

TABLE 7. **Representation of the result of applying the filter by priority based on the way where it is specified that objective 2 has priority over objective 3.**

Filter	F_{P_1}
Parameters	$i = 2$ $j = 3$
Results	$S_1 = [0.9; 0.1; 0.2; 0.3]$ $S_3 = [0.2; 0.4; 0.5; 0.8]$ $S_5 = [0.1; 0.6; 0.9; 0.7]$ $S_6 = [0.1; 0.5; 0.6; 0.9]$

TABLE 8. **Example that demonstrates the result of applying the combined use of the filter by value F_{R_2} and the filter by priority based on the form F_{P_1}.**

Filter	$F_{R_2} \wedge F_{P_1}$
Results	$S_1 = [0.9; 0.1; 0.2; 0.3]$

Because filters result in sets of solutions, these can be combined using basic set operations such as union or intersection.

Table 8 shows an example of considering filters F_{R_2} and F_{P_1} simultaneously, i.e., the intersection, thus providing solutions whose value in the fourth objective is within the range $[\alpha = 0.0; \beta = 0.5]$ with objective number 2 as priority before number 3.

Taking into account the shape of the solutions, this work introduces the concept of a "position-based filter." This filter allows for the selection of solutions where a specific objective occupies a particular position or falls within a range of positions. For example, using this position-based filter, one could isolate all solutions in which objective 3 occupies the first or second position based on the shape.

IV. IMPLEMENTATION AND RESULTS

To illustrate the advantages of the proposed enhancement strategies, a visualization tool has been developed in *Java*. Available upon request, this tool features Parallel Coordinates and RadViz as its primary visualization techniques. Rather

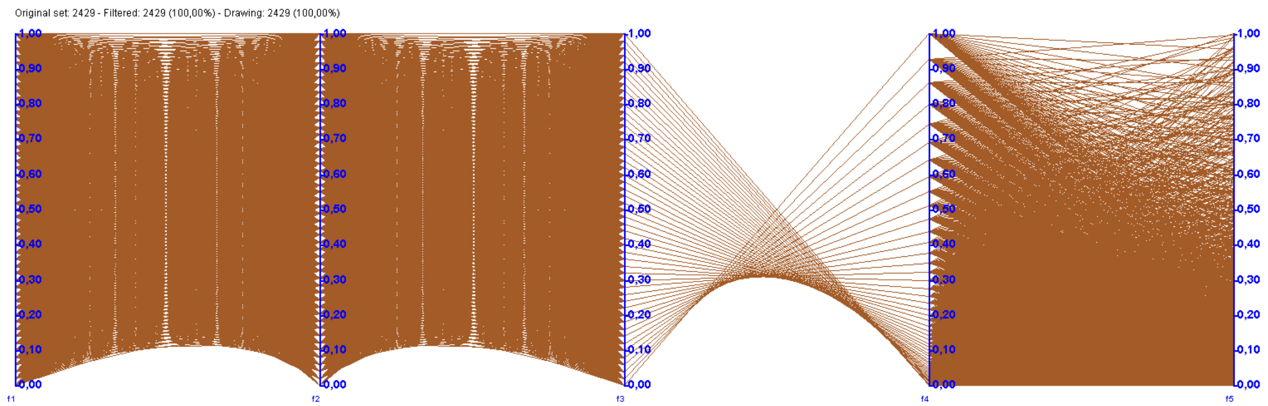


FIGURE 4. Representation of the original set of the problem Water using the parallel coordinates method.

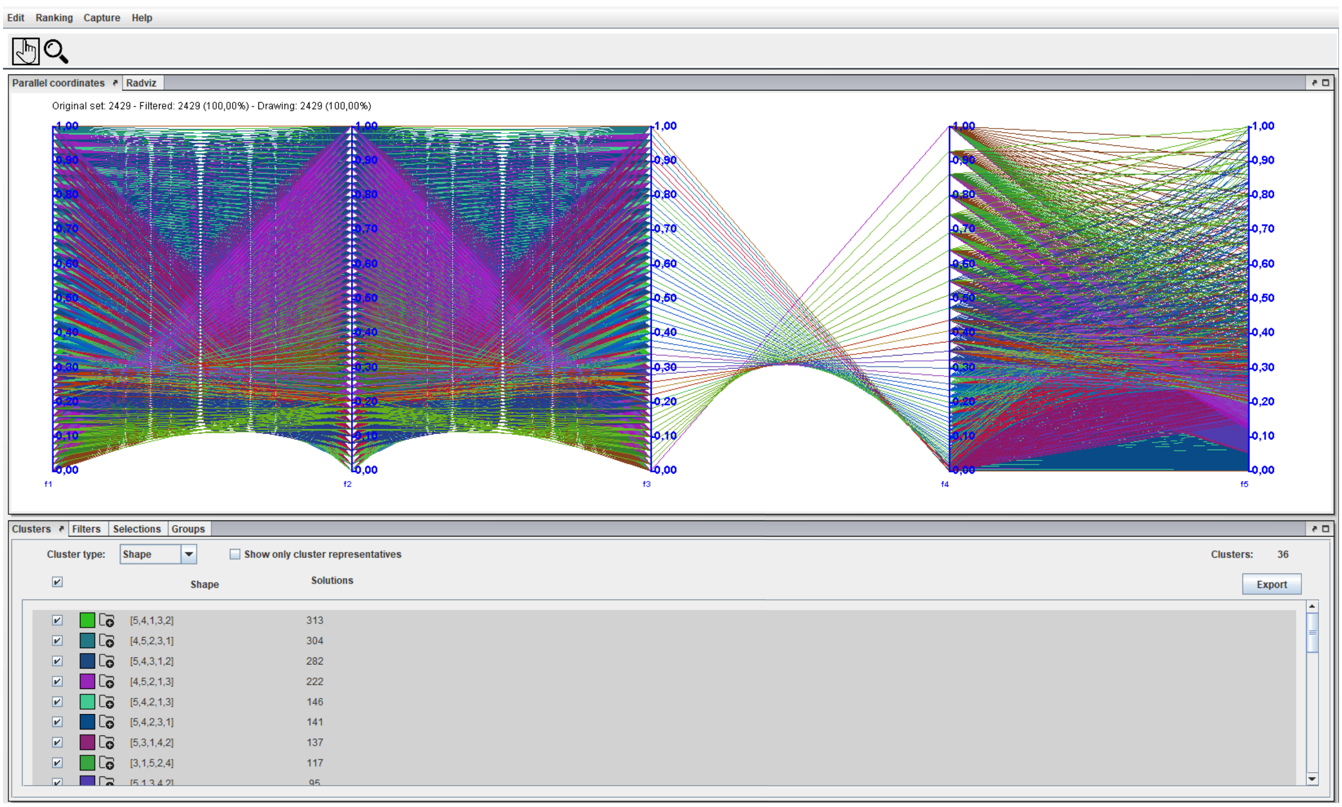


FIGURE 5. Application interface showing the clusters based on the shape of the solution set of the problem Water using the parallel coordinates method.

than detailing the application specifics, this section offers a comprehensive overview of how the discussed methods can support visual analysis. The primary focus is showcasing results through the parallel coordinates method and on the utility of integrating shape-based clustering with filters and ranking to uncover deeper insights.

As a case study, this work considers the solution set of the Water problem [19] available in [33]. It is a many-objective problem considering the simultaneous optimization of the following objectives within a river drainage system:

- f1: cost of the drainage network.
- f2: cost of storage infrastructure.
- f3: cost of treatment infrastructure.
- f4: expected cost of flood damage.
- f5: expected economic loss due to floods.

Additionally, the referenced problem mandates adherence to constraints, detailed in [19].

The developed application produced Figure 4, which shows the solution set using the Parallel Coordinates method before applying the proposed improvement strategies. Due to the varied scales across the problem's objectives, the application normalizes the data for comparability. The large volume of data, encompassing 2429 solutions, results in significant overlap among the lines, hindering practical

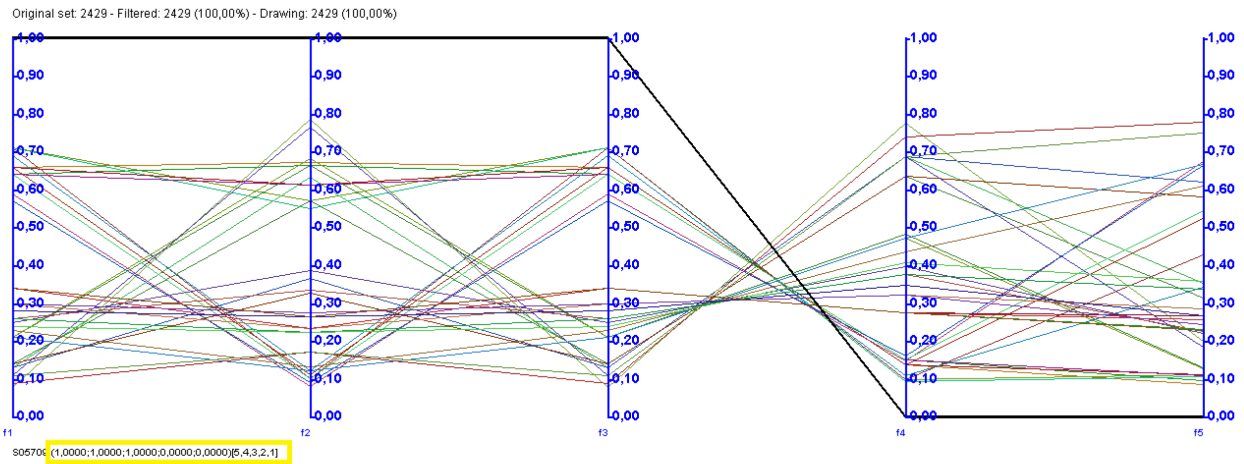


FIGURE 6. **Visualization of 36 representative vectors for each cluster, derived from shape-based clustering. The interactive feature of the tool allows users to explore individual vector characteristics, as demonstrated by the cursor placed over the vector (1.0, 1.0, 1.0, 0.0, 0.0), with details shown in the yellow box.**

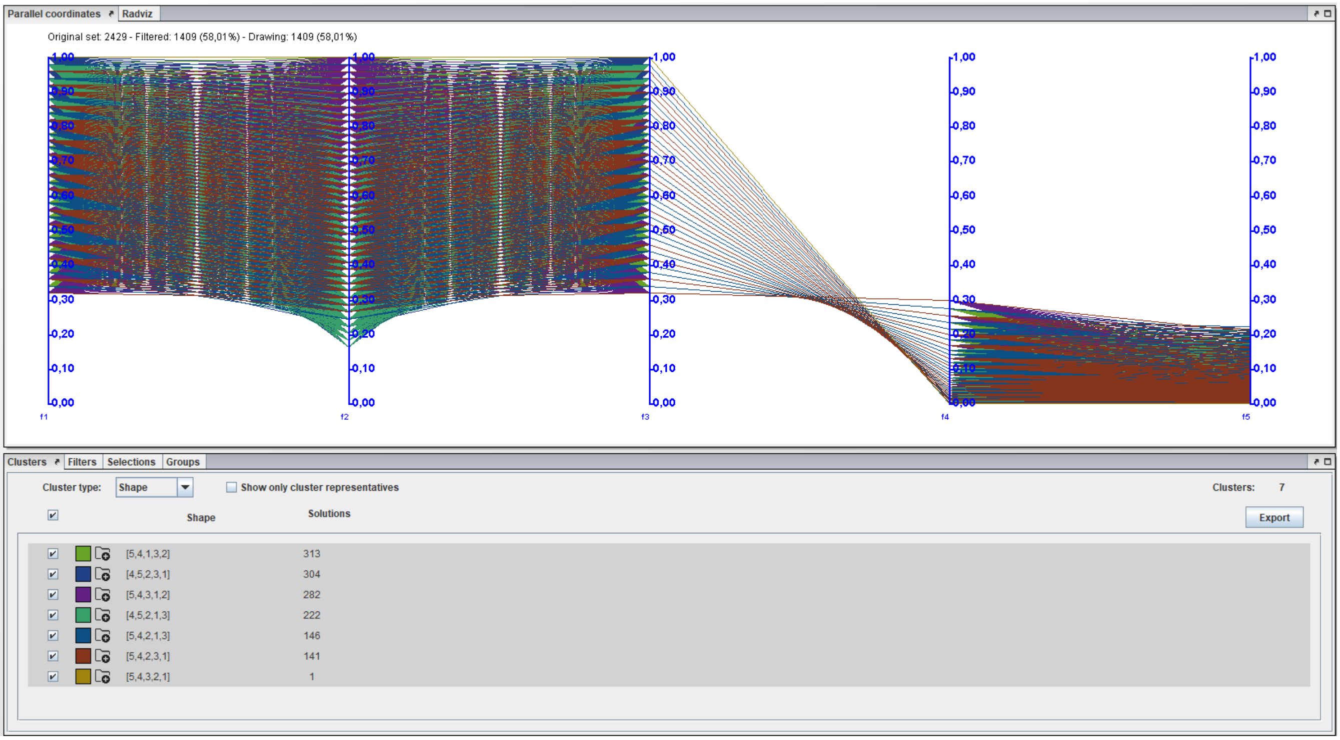


FIGURE 7. **Visualization using Parallel Coordinates of 1409 solutions filtered by position-based criteria, showing solutions where objectives f4 (expected cost of flood damage) and f5 (expected economic loss due to floods) are prioritized.**

data interpretation. This issue is visible in the figure, where the dense aggregation of solutions prevents the clear identification of distinct patterns or trends.

To address the data set visualization challenges, we initially employed clustering based on the shape of the objective vectors over the set of solutions. Figure 5 showcases the application's interface, displaying the solutions via a Parallel Coordinates graph where those vectors with the same shape have identical colors. Below the figure is a panel with multiple tabs where it is possible to configure clustering and filter alternatives. The menu provides options for editing the figure and solution ranking. Among other alternatives, the edit menu offers to display only some objectives, reorder and rename the objectives, and maintain the original set of solutions in the background.

In the clusters tab, it is possible to select between k-means, shape-based, or no clustering options. In this case, as shape-based clustering is used, a scrollable window lists the shapes, showing the number of solutions for each group, their assigned color, and checkboxes. Figure 5 indicates that the solutions are categorized into 36 distinct groups out of a potential 120, with group sizes ranging from 1 to

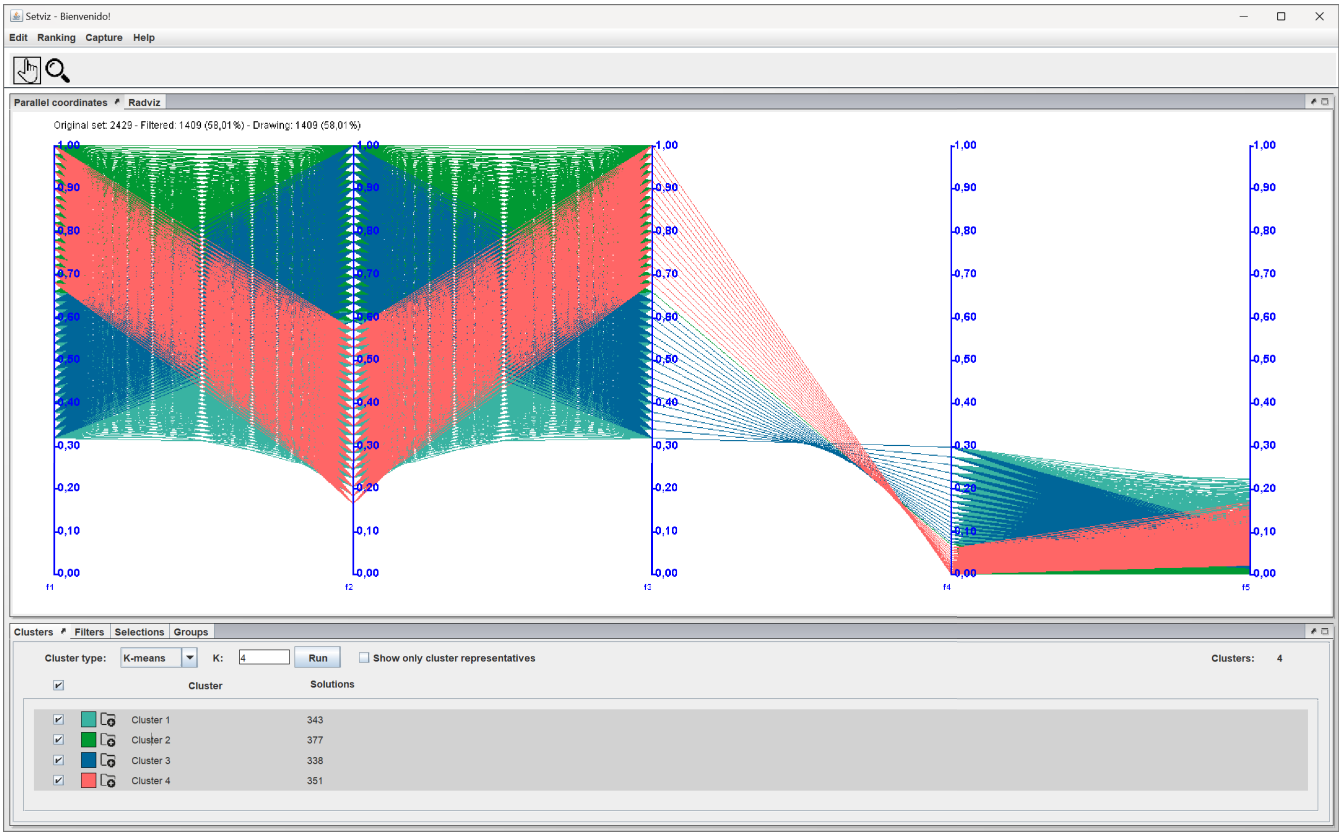


FIGURE 8. **Further refinement of the filtered solutions from Figure 7 using k-means clustering, resulting in four distinct groups. This demonstrates the depth of analysis possible by combining position-based filtering with k-means clustering to uncover patterns and trade-off characteristics within the solution space.**

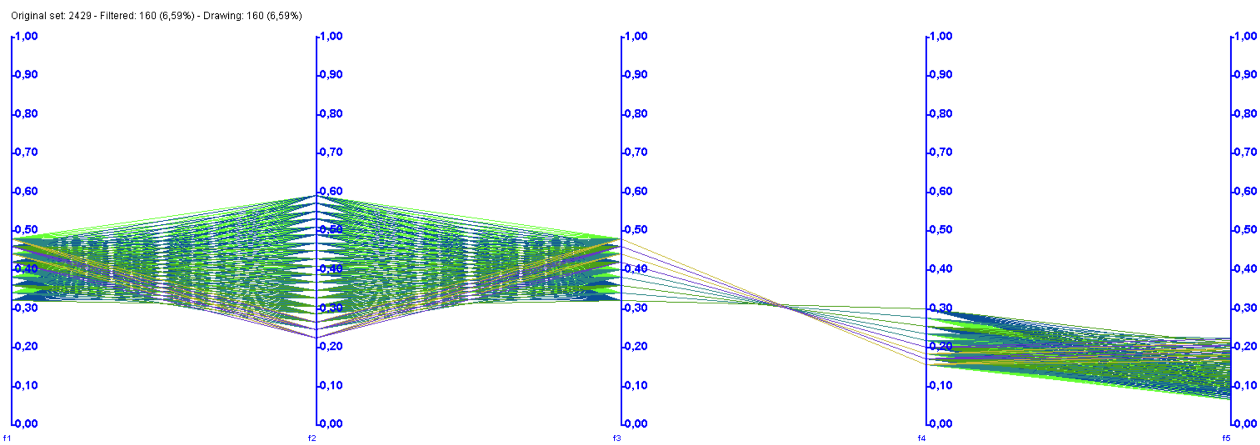


FIGURE 9. **Visualization of the solution set after applying filters for the normalized performance cost of the drainage network (f1) is less than 0.5, and the cost of storage infrastructure (f2) is less than 0.6.**

313 elements. Checkboxes serve to implement the filter by grouping the solutions to visualize; this feature simplifies the exploration of large datasets and enhances analytical efficiency by allowing users to focus on specific clusters.

As outlined earlier, the shape-based clustering method ensures that the average vector representing the elements within each cluster maintains a consistent shape. Figure 6 displays 36 representative vectors corresponding to each cluster. This visualization provides insight into the relationships between the proposed solutions' objectives, such as identifying groups with significantly divergent objective values and sets where the objectives seem more balanced. For example, in Figure 6, by visual inspection, it is possible to note that f1 and f3 appear to be correlated, whereas these objectives contradict f4. An interactive feature of the developed tool allows indicate the shape and value of each vector by positioning the cursor over it. In Figure 6, for example, the cursor is over vector (1.0, 1.0, 1.00.0, 0.0), and

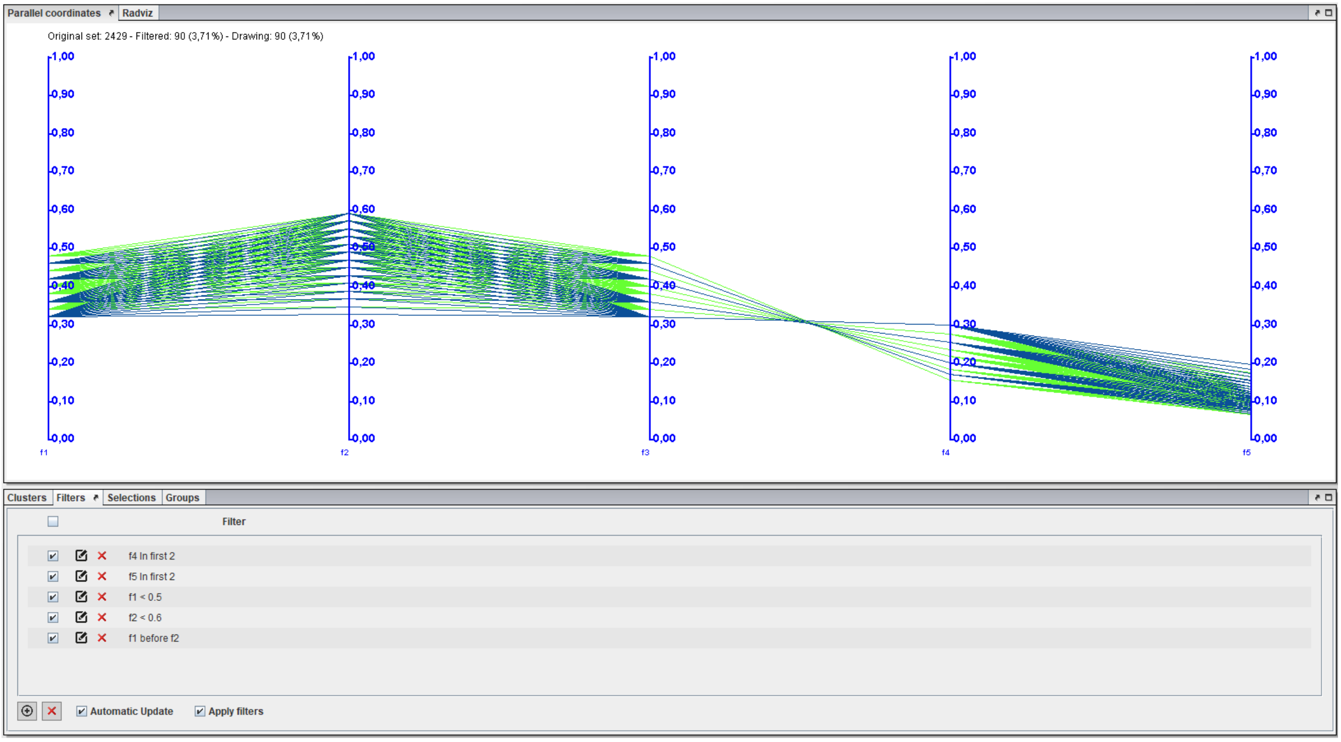


FIGURE 10. **Further analysis of the filtered solutions from Figure 9, applying a priority filter based on shape focusing on those where the cost of the drainage network (f1) has better normalized value over the cost of storage infrastructure (f2).**

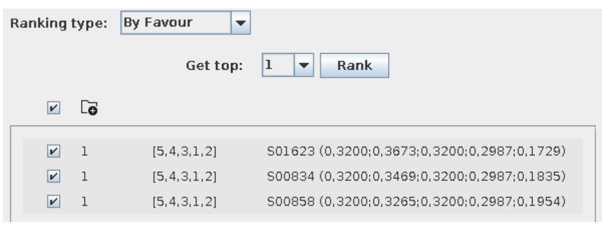


FIGURE 11. **Solutions belonging to the best level of the ranking *favour* applied to the filtered set.**

the characteristics of the vector are displayed as the yellow box notes.

When assessing solutions trade-offs, focusing on those that exhibit optimal values for specific objectives becomes relevant. In the context of the problem considered in this study, one example of the many potential areas to explore through detailed visualization analysis is to examine solutions that emphasize the minimization of the expected cost of flood damage and the expected economic loss due to floods—identified as objectives f4 and f5, respectively. To this aim, the application supports the use of various types of position-based filters, enabling to isolate solutions where objectives f4 and f5 occupy the first two positions, represented by solutions with shapes $[5, 4, *, *, *]$ and $[4, 5, *, *, *]$. The result of this filter is in Figure 7, which draws a reduced set with 1409 solutions that fit in 7 groups corresponding to different shapes.

Additionally, it is possible to refine the analysis of the solutions displayed in Figure 7. For example, in Figure 8, k-means is used to classify the filtered solutions into four groups since the elbow method indicated that this value might be optimal. Clustering filtered solutions by k-means aids in highlighting patterns and trends that might not be immediately apparent and facilitates a more granular analysis, identifying clusters of solutions sharing similar trade-off characteristics. This approach provides insights into how different solutions compare in achieving these objectives. The flexibility to apply k-means clustering to solutions already filtered by position-based criteria (as shown in Figure 7) demonstrates the dynamic nature of the analysis. This layered approach to filtering and clustering enables users to refine their exploration of the solution space iteratively, adapting the analysis to evolving objectives or insights.

Additional filters can be added to produce a detailed analysis of solutions in Figure 7. As an example in Figure 9, the solution set is filtered to consider those solutions with normalized performance cost of the drainage network (f1) and the cost of storage infrastructure (f2) below 0.5 and 0.6, respectively. The result of applying both filters is in Figure 9. The figure displays 160 solutions that meet these criteria, highlighting the effectiveness of using multi-dimensional filters to narrow the solution space.

Then, from the filtered solutions Figure 9, it may be interesting to study those solutions having a shape such that f1 is before f2, i.e., a filter by priority where we specify that the first objective is prior to the second objective. After adding this filter, Figure 10 shows the resulting graph. The filtered subset consists of 90 solutions that conform to the specified

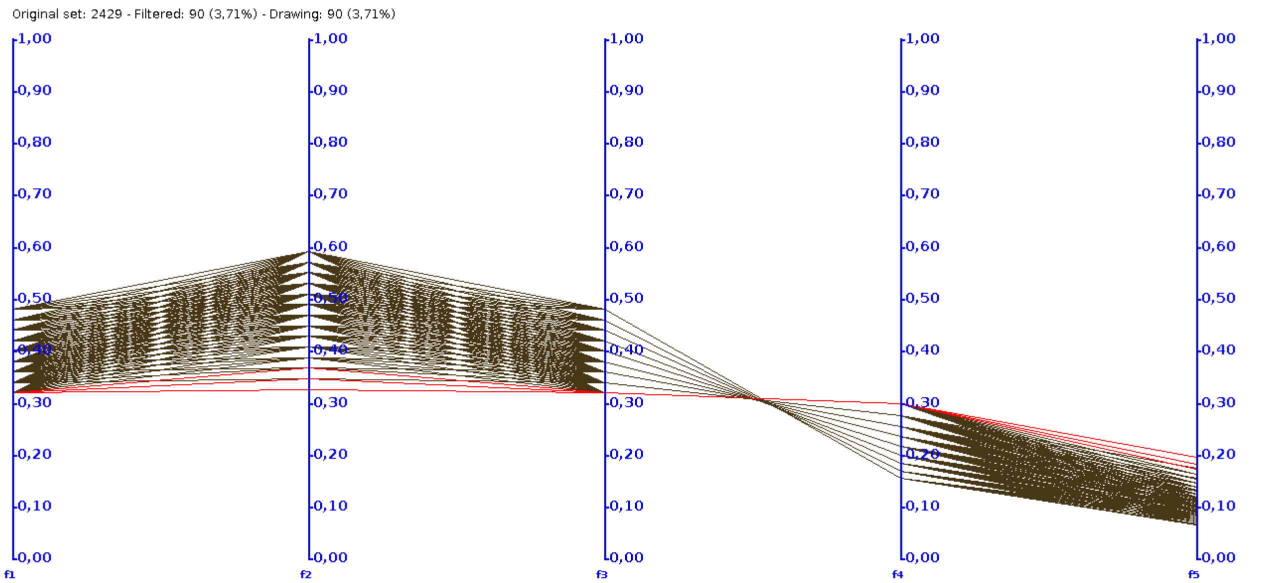


FIGURE 12. Result after applying ranking *favour* on the filtered set where the solutions belonging to the best level are highlighted with red color using the Parallel Coordinates graph.

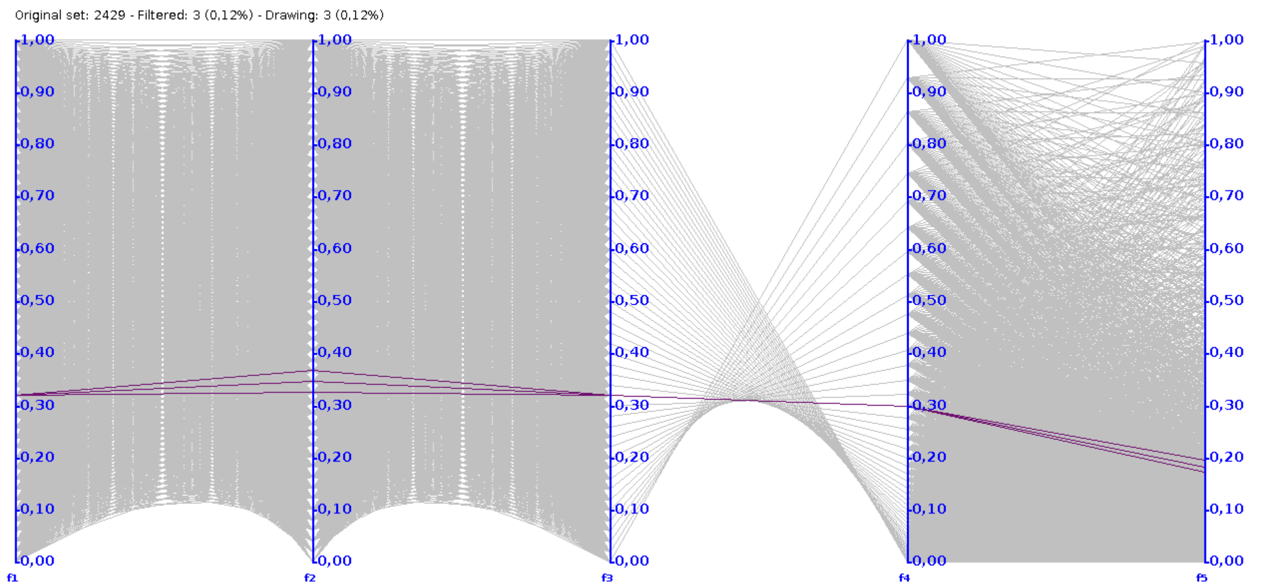


FIGURE 13. Resulting set where the three best solutions found after the visual exploration process are graphed using Parallel Coordinates. In the background, the original set is represented with a lighter color.

shape criteria, demonstrating the impact of prioritization filters on the solution set, which represents approximately 4% of the original set.

Examples in Figure 7, Figure 9, and Figure 10 underscore the capability that using distinct types of filters may help tailor the analysis towards specific optimization goals, thereby providing a focused exploration of solution sets that align with preferred objectives. They mainly show how valuable filters such as group, position, and priority-based filters can be implemented based on the shape idea.

After narrowing down the area of interest, the ranking procedures proposed in Section III-B can be applied to speed up identifying the best-ranked solutions, according to a selected criterion, in the refined set. For example, considering the solutions in Figure 10, using the application interface for ranking, the solutions that are in the first place according to the *favour* ranking can be determined as indicated in Figure 11. These solutions are shown in Figure 12 highlighted in red for clear differentiation along with the filtered subset.

Finally, these three solutions can be added to a group using the tool's functionality to create and assign solutions to custom groups. Subsequently, the filter by groups serves to graph only these solutions. Figure 13 displays the result of filtering such solutions with the original set in the background. The interactive use of the strategies proposed in this work reduced the solution set from 2429 solutions to 3, belonging to only one of the 36 shapes in the original set. decision-makers can adopt other ways of applying the filters and rankings according to their interests.

V. CONCLUSION AND FUTURE WORK

In this paper, we explored the integration of visualization methods with filtering, clustering, and ranking techniques to address the challenges of many-objective optimization, particularly in high-dimensional scenarios. Implementing these strategies within a specially developed visualization tool demonstrates their practicality and effectiveness. The case study based on the Water problem illustrates how the proposed strategies facilitate the exploration and analysis of solutions, highlighting the benefits of integrating shape-based clustering with filters and ranking to uncover deeper insights and guide strategic decision-making.

The developed application displays solutions using Parallel Coordinates and RadViz. In this work, however, only results with parallel coordinates are shown. Regarding display methods, it is important to note that they must meet certain characteristics to implement the proposed operations effectively:

- Display methods should ideally balance computational efficiency with the ability to assist decision-makers in selecting their preferred solution based on specific criteria and needs. Rapid visualization is necessary due to the task's iterative nature, which requires frequent updates to the visual display. However, there are times when compromising speed is acceptable to better assist decision-makers in selecting the solutions that best fit the problem at hand.
- The decision-maker should be able to observe the changes produced in the group of solutions through the applied operations and the visualization method. Therefore, the visual results obtained by the visualization technique must reflect the magnitude of changes induced in the solution group by these operations. This proportional representation ensures that the decision-maker receives an accurate and intuitive understanding of the impact of each operation on the solution set.

Clustering through the novel use of shape-based clustering has shown to be instrumental in breaking down the solution set into manageable subsets, each representing unique solution characteristics. This method, augmented by the traditional K-means algorithm, provides a dual approach to categorize solutions effectively, enabling decision-makers to focus on specific areas of interest within the solution space. Also, shape-based clustering serves to define position and priority-based filters. The systematic application of filters and clustering allows us to uncover deeper insights and guide strategic decision-making.

Ranking methods such as *favour* and *epsilon-preferred* can obtain improved insight into solution quality since they help identify solutions that offer a balanced trade-off across multiple objectives. An additional advantage of the *epsilon-preferred* relation is that adjusting threshold parameters allows the integration of decision-maker preferences directly into the solution classification process in an iterative visualization analysis. Finally, based on the defined mathematical relations, a ranking-based approach provides an objective and systematic way to compare and rank solutions, ensuring that the selection process is transparent and justifiable.

As in the example considered in this paper, rankings can be applied after reducing the solution set to an area that fits the user's interest. This prioritization accelerates the decision-making process by focusing on the most promising solutions. Moreover, with large sets of solutions, these methods can be applied to reduce the dataset to a more manageable size before visual analysis, making it easier for decision-makers to navigate the solution set. This possibility may be essential for handling complex, high-dimensional optimization problems without overwhelming the decision-maker.

In future works, exploring other types of operations that can reduce the set according to the decision-maker's needs may be worth exploring. Additionally, it may be beneficial to analyze the outcome of implementing the proposed operations with other visualization methods. This could involve creating new visualization techniques that consider the operations proposed in this study and the characteristics that should be fulfilled to support them [2].

REFERENCES

[1] B. Dy, N. Ibrahim, A. Poorthuis, and S. Joyce, "Improving visualization design for effective multi-objective decision making," *IEEE Trans. Vis. Comput. Graphics*, vol. 28, no. 10, pp. 3405–3416, Oct. 2022.

[2] J. Hakanen, D. Gold, K. Miettinen, and P. M. Reed, "Visualisation for decision support in many-objective optimisation: State-of-the-art, guidance and future directions," in *Many-Criteria Optimization and Decision Analysis*. Cham, Switzerland: Springer, 2023, pp. 181–212.

[3] B. Filipič and T. Tušar, "A taxonomy of methods for visualizing Pareto front approximations," in *Proc. Genet. Evol. Comput. Conf.*, vol. 4434. New York, NY, USA: Association for Computing Machinery, Jul. 2018, pp. 649–656.

[4] C. von Lücken, B. Barán, and C. Brizuela, "A survey on multi-objective evolutionary algorithms for many-objective problems," *Comput. Optim. Appl.*, vol. 58, no. 3, pp. 707–756, Feb. 2014.

[5] M. Li, L. Zhen, and X. Yao, "How to read many-objective solution sets in parallel coordinates [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 88–100, Nov. 2017.

[6] A. Inselberg and B. Dimsdale, "Parallel coordinates: A tool for visualizing multi-dimensional geometry," in *Proc. 1st IEEE Conf. Vis.*, Oct. 1990, pp. 361–378.

[7] Z. He and G. G. Yen, "Visualization and performance metric in many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 386–402, Jun. 2016.

[8] S. Liu, D. Maljovec, B. Wang, P.-T. Bremer, and V. Pascucci, "Visualizing high-dimensional data: Advances in the past decade," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 3, pp. 1249–1268, Mar. 2017.

[9] L. Zhen, M. Li, R. Cheng, D. Peng, and X. Yao, "Adjusting parallel coordinates for investigating multi-objective search," in *Simulated Evolution and Learning* (Lecture Notes in Computer Science), vol. 10593. Cham, Switzerland: Springer, 2017.

[10] A. Elewah, A. A. Badawi, H. Khalil, S. Rahnamayan, and K. Elgazzar, "3D-RadViz: Three dimensional radial visualization for large-scale data visualization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2021, pp. 1037–1046.

[11] R. Ding, H.-B. Dong, G.-S. Yin, J. Sun, X.-D. Yu, and X.-B. Feng, ''An objective reduction method based on advanced clustering for many-objective optimization problems and its human-computer interaction visualization of Pareto front,'' *Comput. Electr. Eng.*, vol. 93, Jul. 2021, Art. no. 107266.

[12] L. Selvarajan, K. Venkataramanan, A. Nair, and V. P. Srinivasan, ''Simultaneous multi-response Jaya optimization and Pareto front visualization in EDM drilling of $MoSi_2$-SiC composites,'' *Expert Syst. Appl.*, vol. 230, Nov. 2023, Art. no. 120669.

[13] H. Smedberg and S. Bandaru, ''Interactive knowledge discovery and knowledge visualization for decision support in multi-objective optimization,'' *Eur. J. Oper. Res.*, vol. 306, no. 3, pp. 1311–1329, May 2023.

[14] Y. Sato, K. Izui, T. Yamada, and S. Nishiwaki, ''Data mining based on clustering and association rule analysis for knowledge discovery in multiobjective topology optimization,'' *Expert Syst. Appl.*, vol. 119, pp. 247–261, Apr. 2019.

[15] M. Garza-Fabre, G. T. Pulido, and C. A. C. Coello, ''Ranking methods for many-objective optimization,'' in *MICAI 2009: Advances in Artificial Intelligence*. Berlin, Germany: Springer, 2009, pp. 633–645.

[16] Plotly. *Parallel Coordinates Plot*. Accessed: Aug. 1, 2024. [Online]. Available: https://plot.ly/python/parallel-coordinates-plot/

[17] C. von Lücken, C. Brizuela, and B. Barán, ''Clustering based parallel many-objective evolutionary algorithms using the shape of the objective vectors,'' in *Evolutionary Multi-Criterion Optimization*. Cham, Switzerland: Springer, 2015, pp. 50–64.

[18] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, ''An efficient k-means clustering algorithm: Analysis and implementation,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[19] K. Musselman and J. Talavage, ''A tradeoff cut approach to multiple objective optimization,'' *Oper. Res.*, vol. 28, no. 6, pp. 1424–1435, Dec. 1980.

[20] M. Ankerst, S. Berchtold, and D. A. Keim, ''Similarity clustering of dimensions for an enhanced visualization of multidimensional data,'' in *Proc. IEEE Symp. Inf. Vis.*, Oct. 1998, pp. 52–60.

[21] E. Kandogan, ''Star coordinates: A multidimensional visualization technique with uniform treatment of dimensions,'' in *Proc. IEEE Symp. Inf. Vis., Late Breaking Hot Topics*, Oct. 2000, pp. 9–12.

[22] A. N. Brzezińska and W. Ogiegło, ''The similarity of the similarity measures in the context of clustering algorithms for categorical data,'' *Proc. Comput. Sci.*, vol. 225, pp. 4511–4520, Jan. 2023.

[23] E. Bertini, L. Dell'Aquila, and G. Santucci, ''SpringView: Cooperation of radviz and parallel coordinates for view optimization and clutter reduction,'' in *Proc. Coordinated Multiple Views Exp. Vis. (CMV)*, Jul. 2005, pp. 22–29.

[24] S. Obayashi and D. Sasaki, ''Visualization and data mining of Pareto solutions using self-organizing map,'' in *Evolutionary Multi-Criterion Optimization*. Berlin, Germany: Springer, 2003, pp. 796–809.

[25] C. von Lücken, A. D. F. Prieto, A. M. F. V. Y. Aragon, and B. Barán, ''Shape-based visual analysis of solutions for multiobjective optimization problems,'' in *Proc. XLIII Latin Amer. Comput. Conf. (CLEI)*, Sep. 2017, pp. 1–9.

[26] W. J. Raseman, J. Jacobson, and J. R. Kasprzyk, ''Parasol: An open source, interactive parallel coordinates library for multi-objective decision making,'' *Environ. Model. Softw.*, vol. 116, pp. 153–163, Jun. 2019.

[27] D. Nagar, P. Ramu, and K. Deb, ''Visualization and analysis of Pareto-optimal fronts using interpretable self-organizing map (iSOM),'' *Swarm Evol. Comput.*, vol. 76, Feb. 2023, Art. no. 101202.

[28] M. Blumenschein, X. Zhang, D. Pomerenke, D. A. Keim, and J. Fuchs, ''Evaluating reordering strategies for cluster identification in parallel coordinates,'' *Comput. Graph. Forum*, vol. 39, no. 3, pp. 537–549, Jun. 2020.

[29] K. P. Sinaga and M.-S. Yang, ''Unsupervised K-means clustering algorithm,'' *IEEE Access*, vol. 8, pp. 80716–80727, 2020.

[30] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, ''K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data,'' *Inf. Sci.*, vol. 622, pp. 178–210, Apr. 2023.

[31] N. Drechsler, R. Drechsler, and B. Becker, ''Multi-objective optimisation based on relation favour,'' in *Evolutionary Multi-Criterion Optimization* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2001, pp. 154–166.

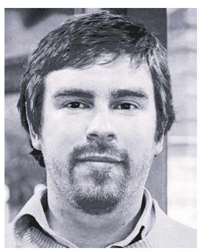
[32] A. Sulflow, N. Drechsler, and R. Drechsler, ''Robust multi-objective optimization in high dimensional spaces,'' in *Evolutionary Multi-Criterion Optimization*. Berlin, Germany: Springer, 2007, pp. 715–726.

[33] jMetal. *JMetal Web Site*. Accessed: Aug. 1, 2024. [Online]. Available: https://jmetal.sourceforge.net/

CHRISTIAN VON LÜCKEN was born in Asunción, Paraguay, in 1975. He received the degree in informatics engineering from the Catholic University of Asunción, in 2001, the master's degree in systems engineering, in 2005, and the Ph.D. degree in informatics from Universidad Nacional de Asunción, in 2016. His primary research interests include methods of computational intelligence, bio-inspired computing, and multi-objective optimization. Since 2009, he has been a Senior Member of the Association for Computer Machinery (ACM). He was distinguished as the Outstanding Young Person of Paraguay by the International Junior Chamber, in 2012.

URIEL PEREIRA was born in Asunción, Paraguay, in May 1995. He received the bachelor's degree in informatics engineering from Universidad Nacional de Asunción, in 2021. He is currently the Lead Product Developer Manager of OneHQ. He has relevant experience as a Backend Developer with the Ruby and Rails framework and PostgreSQL. He also has experience in Elixir and functional languages in general. His main research interests include cybersecurity and data science.

ENRIQUE JAVIER DÁVALOS (Member, IEEE) received the degree in electromechanics engineering, the M.Sc. degree in systems engineering, and the Ph.D. degree in computer sciences from Universidad Nacional de Asunción, Paraguay, in 1987, 2010, and 2018, respectively. He was with IBM World Trade Corporation as Global Services Coordinator, from 1987 to 2004, and PSLine S.A., from 2004 to 2008. Since 2009, he has been a full-time Professor and Researcher with the Facultad Politécnica, Universidad Nacional de Asunción. His research interests include optical networks, network security, and multi-objective optimization.



FABIO LÓPEZ-PIRES received the master's degree in networks and data communications and the Ph.D. degree in computer science. He is a Computer Engineer from Universidad Nacional de Asunción, Paraguay. His research lines are focused on interdisciplinary computational sciences, mainly in cloud computing, multi-objective optimization, computational intelligence, and machine learning. He was a recipient of the Paraguayan National Science Prize 2018 and the Dr. Andrés Barbero Prize 2018, awarded by the Scientific Society of Paraguay. Additionally, he has garnered awards in the Latin American Doctoral Thesis Competition 2018 and in the Latin American Master's Thesis Competition 2014 by the Latin American Center for Computer Studies (CLEI).