

moGrams: A Network-Based Methodology for Visualizing the Set of Nondominated Solutions in Multiobjective Optimization

Krzysztof Trawiński, Manuel Chica, David P. Pancho, Sergio Damas, *Member, IEEE*,
and Oscar Cordón, *Senior Member, IEEE*

Abstract—An appropriate visualization of multiobjective nondominated solutions is a valuable asset for decision making. Although there are methods for visualizing the solutions in the design space, they do not provide any information about their relationship. In this paper, we propose a novel methodology that allows the visualization of the nondominated solutions in the design space and their relationships by means of a network. The nodes represent the solutions in the objective space while the edges show the relationships among the solutions in the design space. Our proposal (called moGrams) thus provides a joint visualization of both objective and design spaces. It aims at helping the decision maker to get more understanding of the problem so that (s)he can choose the most appropriate and flexible final solution. moGrams can be applied to any multicriteria problem in which the solutions are related by a similarity metric. Besides, the decision maker interaction is facilitated by modifying the network based on the current preferences to obtain a clearer view. An exhaustive experimental study is performed using four multiobjective problems with a variable number of objectives to show both usefulness and versatility of moGrams. The results exhibit interesting characteristics of our methodology for visualizing and analyzing solutions of multiobjective problems.

Index Terms—Decision making, design space, multiobjective optimization, social network analysis (SNA), visualization.

I. INTRODUCTION

MOST of the real-world problems have a multicriteria nature (i.e., they include several conflicting criteria) [1]. Multiobjective optimization methods are designed for solving

Manuscript received December 17, 2015; revised July 1, 2016 and October 24, 2016; accepted December 6, 2016. Date of publication January 16, 2017; date of current version January 15, 2018. This work is supported by Spanish Ministerio de Economía y Competitividad under the NEWSOCO project (ref. TIN2015-67661), including European Regional Development Funds (ERDF). This paper was recommended by Associate Editor Q. Zhang.

K. Trawiński is with Novelti, 28012 Madrid, Spain (e-mail: krzysztof@novelti.io).

M. Chica is with the Department of Computer Science, Open University of Catalonia, 08018 Barcelona, Spain (e-mail: manuel.chica.serrano@gmail.com).

D. P. Pancho is with eWhere Unit, Treelogic, Parque Tecnológico de Asturias, 33428 Llanera, Spain (e-mail: david.p.pancho@treelogic.com).

S. Damas is with the Department of Software Engineering, University of Granada, 18071 Granada, Spain (e-mail: sdamas@ugr.es).

O. Cordón is with the Department of Computer Science and Artificial Intelligence, and the Research Center on Information and Communication Technologies, University of Granada, 18071 Granada, Spain (e-mail: ocordova@decsai.ugr.es).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCYB.2016.2642886

and producing a set of nondominated solutions when tackling these problems [2], [3]. Decision makers aim at selecting the best possible solution from the set of returned alternatives. To do so, the decision maker has to compare the alternative solutions and that is a difficult and time-consuming task. In evolutionary multiobjective optimization (EMO) [2], [3] some authors define a global framework considering multicriteria decision making (MCDM) as a conjunction of three components: 1) search; 2) preference tradeoffs; and 3) visualization [4].

A helpful visualization process in EMO should provide a deeper problem understanding and new useful information regarding the alternatives. The decision maker should receive as much relevant information as possible provided by powerful methods and techniques. A good visualization enables her/him to obtain more insights of the problem and the different solutions to identify differences and similarities before coming up with the final decision [5]. In particular, the flexibility, i.e., the ease to change one solution into another in the decision space, is an important property to respond to frequent environmental changes in many managerial and operation research problems in which the information is uncertain [6]. Additional information about the flexibility of the nondominated solutions will be really worthy for the decision maker.

There is an increasing number of studies demonstrating that visualization together with optimization can promote design innovations and provide decision makers with an improved understanding of the problem [7], [8]. The visualization of solutions in multiobjective optimization is an active research area [9], [10] and most of the works are devoted to visualization of the Pareto front in the objective space (see [11]–[13]). Walker *et al.* [9] focused on how to represent nondominated solutions for many-objective problems (i.e., those having more than three objectives). The interest in this kind of optimization problems is rapidly increasing in the evolutionary computation community [14]. However, just a few proposals deal with the visualization of the multiobjective solutions in the design space (e.g., self organization maps [15], heatmaps [16], cloud visualization [17], or hyper-space diagonal counting [18]). Up to our knowledge, there is no proposal providing the decision maker with insights about the relationships existing among the solutions in the decision space with the aim of helping the decisor to understand the problem and assisting her/him to choose the final solution.

In this paper, we aim at proposing a novel methodology, called moGrams, for visualizing and analyzing multiobjective solutions in order to facilitate the MCDM process. Our novel framework represents the nondominated solutions to multiobjective problems, obtained by any optimization process, in the form of a network. This representation has the advantage of showing a clear view of the solutions relationship in the design space, which becomes a complementary information to that provided by Pareto fronts in the objective space. Moreover, moGrams provide objective space information of the solutions in a joint visualization of both design and objective spaces.

A moGram is a weighted network where its nodes represent nondominated solutions and its edges represent design space relationship between the solutions. A similarity metric is used to generate the weighted edges of the network. This metric is defined by the decision maker and is specific to the multiobjective problem. moGrams are thus a generic methodology that applies to any multicriteria problem where solutions can be related by a similarity metric.

Our network-based proposal enhances the similarity analysis of the solutions in the decision space. This analysis facilitates the evaluation of the flexibility of every solution together with the objective space information. For instance, when several solutions having similar objective values exist, the decision maker can analyze how flexible they are and will be able to choose the most connected solution, represented by a node within the network. The reason is that the most connected node could be easily transformed into other adjacent nodes. These nodes are similar solutions in the design space where changes between them are less costly for decision makers.

moGrams use social network analysis (SNA) techniques to improve the visual information from the network [19]. In particular, the Pathfinder pruning algorithm [20], [21] reduces the complexity of the network by only representing the most salient connections, thus only preserving the most important information and making its analysis easier. In order to clearly visualize the network, a force-directed layout is used, namely the Kamada–Kawai algorithm [22]. moGrams is also an interactive visualization framework where the decision maker has the possibility to obtain a clearer view of the network according to her/his preferences.

In order to show the benefits and versatility of moGrams we perform a complete experimentation on four multiobjective problems with a variable number of objectives. These MCDM application cases come from different research fields and we chose them because of their different features (e.g., their decision variables are very diverse, from combinatorial optimization to real-valued parameters). The first application case is the time and space assembly line balancing problem (TSALBP) [23], [24] which deals with the joint optimization of the number of stations and their area when configuring an industrial line for assembling products. The second considers the overproduce-and-choose strategy [25] for classifier ensembles (CEs) [26], [27] and it aims at obtaining CEs with a low number of base classifiers, while keeping a good accuracy [28], [29]. The third application case focuses on the inductive query by example (IQBE) [30]

TABLE I
PROPOSALS CONSIDERING THE DESIGN SPACE VISUALIZATION
OF MULTIOBJECTIVE SOLUTIONS

Reference(s)	Visualization method	Supplementary algorithms	Joint visualization
[17]	Cloud visualization	-	✗
[34]	Graph morphing	-	✓
[18], [35]	HSDC	-	✓
[15], [36]	SOM	ANOVA	✓
[16]	Heatmap	Clustering	✓
[15], [37]	Synchronous visualization	-	✗
[12]	Spatial coordinates, size, shape, color	-	✗
[38]	Level diagrams	Classifier	✗
[39]	Isomap	-	✓
[9]	Heatmap	Spectral seriation	✓
[40]	Scatterplot and parallel coordinate plot	-	✗

paradigm that allows a system to automatically derive queries for a specific information retrieval (IR) system, defined as a multiobjective problem with a conflicting precision-recall tradeoff [30]. Finally, we apply moGrams to a classical, continuous multiobjective optimization problem, DTLZ4 [31], to analyze the scalability of the framework for a multiobjective problem with four objectives and a hundred of nondominated solutions.

The rest of this paper is structured as follows. Section II reviews the state of the art in MCDM visualization, the flexibility concept, and a background of SNA. Section III provides a description of our novel visualization proposal. In order to show the utility of moGrams, Sections IV–VII present the four application cases (TSALBP, CEs, IQBE, and DTLZ4, respectively). Finally, Section VIII outlines concluding remarks and future research works.

II. BACKGROUND

A. Visualization of Multicriteria Solutions

There are multiple proposals for visualizing the multiobjective and many-objective Pareto front solutions in [9]. Probably, the most common method for visualizing multiobjective solutions is a scatterplot [32]. One of the most popular choices for many-objective solutions is the use of parallel coordinate plots [33]. Recent reviews [9], [10] show that most of the works in the literature are devoted to the visualization of the Pareto front solutions in the objective space.

Nevertheless, the visualization of multicriteria solutions in the design space did not receive much attention. We present a summary of the existing proposals in Table I. The first and second columns include the reference and visualization method used, respectively. The third column indicates the additional algorithms used to either obtain a clearer visualization (e.g., hierarchical clustering, spectral seriation) or provide additional numerical information (e.g., ANOVA, data mining classification). Only some proposals considered a joint visualization for objective and design spaces (fourth column). None of them were focused on the similarities between the solutions in the design space and none of them used a network as a tool for visualization of the multiobjective solutions.

Self-organizing maps (SOMs) attracted attention as a novel means for visualization of both objective and design space. SOM projects multidimensional data on a 2-D map without any information loss. In [15] and [36], SOM was applied to find the tradeoffs between objective spaces, relationships between objective spaces and design variables. Pryke *et al.* [16] introduced a heatmap for presenting both objective and parameter space in the same view. A hierarchical clustering was used to alleviate the transparency issue. Walker *et al.* [9] also used heatmaps for MCDM visualization and improved their readability by using spectral seriation which rearranges the objective or parameter space views on the heatmap.

Kudo and Yoshikawa [39] used a concept of Isomap to visualize the geometric distance of solutions in the design space considering their distance in the objective space. By doing so, they were able to capture the relationships between design and objective spaces. Agrawal *et al.* [18], [35] presented an intuitive visualization methodology for multidimensional optimization problems by using the hyper-space diagonal counting which enables a lossless mapping of dimensions. The proposed method dealt with more than three objectives and also with design space visualization.

There are some proposals providing a visualization of both objective and decision spaces in separate windows such as cloud visualization [17] and synchronous visualization [37]. The visualization frameworks VIDEO [12] and EVOLVE [40] followed the same idea. They used standard techniques such as spatial coordinate axes, color, Kriging mapping in VIDEO; and scatterplot with parallel coordinate plot in EVOLVE.

Besides, we can highlight other interesting proposals where authors presented each objective and design variable in separate windows (level diagrams) [38] or showed several 3-D visualizations with fixed values of some design variables (graph morphing) [34].

B. Flexibility in MCDM

The ease to change one solution into another in the decision space (named flexibility) might be critical in some real-world scenarios. Tools that help the decision maker in this crucial task will be really valuable. A formal definition of flexibility is given as follows.

Multiobjective optimization can be described by $\min_{x \in T} F(x)$, where $F(x) = [f_1(x), \dots, f_m(x)]$ is the m -dimensional function to be optimized and T is the set of constraints applied to the decision variables of the problem x : $T = \{x \in \mathbb{R}^n : g(x) = 0, h(x) \geq 0\}$. Let us assume $X_1 = (x_{11}, x_{12}, \dots, x_{1n})$ is a vector of n -decision variables and a Pareto-optimal solution to a multiobjective problem. X_1 is already implanted in a fully operational environment.

Given a new Pareto-optimal solution X_2 , the managerial switching cost of replacing X_1 by X_2 (noted as $C(X_1, X_2)$) includes the economic, managerial, and all other related costs to keep on efficiently solving the multiobjective problem in the same environment using X_2 instead of X_1 . The need of such a replacement is rather common in many fields. For instance, changes in demand when manufacturing different products

require at least the analysis and implementation of mechanisms to face the likely change of the current operational solution due to a changing demand [6].

The ϵ -neighborhood of a solution X can be formulated as $N_\epsilon(X) = \{X_i \in T : C(X, X_i) < \epsilon\}$. Given two Pareto-optimal solutions X_1 and X_2 , X_1 is more flexible than X_2 , noted by $X_1 \rightsquigarrow X_2$, iff $|N_\epsilon(X_1)| > |N_\epsilon(X_2)|$, where $|\cdot|$ is the cardinality function.

C. Social Network Analysis

SNA techniques have shown their capability to achieve high quality and schematic visualizations of the network-based representations from different fields, from psychology [20] to scientometrics (e.g., analysis of large scientific domains [41]). Among others, SNA techniques are useful to solve two tasks.

1) *Network Scaling*: Networks are usually dense and scaling is necessary to obtain structures revealing the underlying organization, maintaining all the nodes but only the most important relations. Three predominant SNA alternatives to accomplish this task are considered in [42]. The first method discards edges with weights below a given threshold [43]. This approach, although easy to implement, does not consider intrinsic structure of the underlying network. Thus, the transformed network may not properly reflect the structure of the original one. The second method extracts a minimum spanning tree of the network [44], already used in the EMO literature [45]. This guarantees a fixed number of edges (a number of nodes minus one), although it may lose important information. The last method provides constraints on paths and removes the edges which do not satisfy them. The pathfinder algorithm [20], [21] is one of the most popular, known for its mathematical properties for preserving the triangular inequality. Some of these properties are edges conservation, modeling symmetrical/asymmetrical relationships, subnetworks maintenance, and representation of the most salient relationships.

2) *Network Drawing*: There are different SNA methods for automatic network visualization. Force-based or force-directed algorithms are the most widely used algorithms for drawing networks in the area of information science [46]. Their aim is to locate the nodes of a network in a 2-D or 3-D space so that all the edges are approximately of equal length and there are as few crossing edges as possible by trying to obtain the most aesthetically pleasing view. Kamada-Kawai [47] and Fruchterman-Reingold [48] are the most representative methods of the network drawing family.

III. VISUALIZATION METHOD FOR REPRESENTING MULTIOBJECTIVE SOLUTIONS IN THE DESIGN AND OBJECTIVE SPACES

In this section, we present our visualization framework. We first define and describe the generation of moGrams in Sections III-A and III-B, respectively. Then, we show the decision maker interaction in Section III-C.

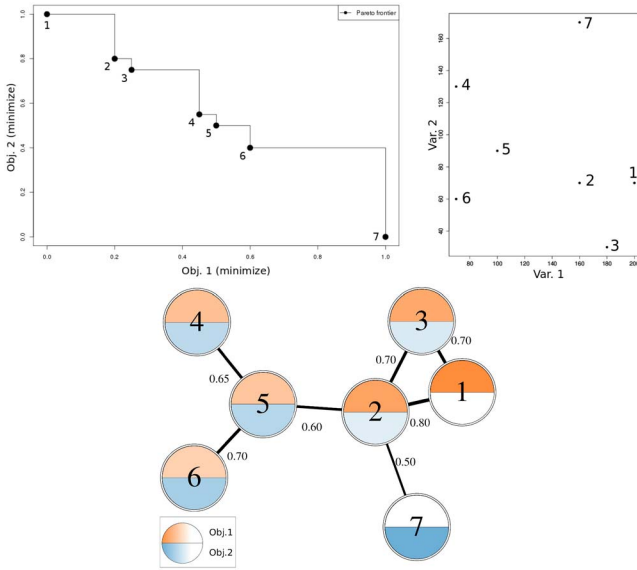


Fig. 1. Pareto front approximation (top left), the solutions in the design space (top right), and the corresponding moGram (bottom).

A. moGrams Definition

A moGram is a network, that jointly presents the solutions of a multiobjective problem in the objective space and their relationship in the design space. In particular, every node of the network represents a nondominated solution in the objective space and its edges represent design space relations between the solutions. There are as many nodes as nondominated solutions in the available Pareto-optimal set. The formal definition of the moGram is as follows.

Definition 1: A moGram is defined as a tuple $G = (N, A)$ where N is the set of nodes representing the n nondominated solutions and A is the adjacency matrix. $A(i, j) \in [0, 1]$. $A(i, j) \neq 0$ means there is an edge between nodes i and j . When $A(i, j) \rightarrow 1$ there is a more significant relationship in the design space between the solutions represented by i and j , according to a given similarity metric.

B. moGrams Generation

We use an illustrative example to facilitate the comprehension of the moGrams generation (see Fig. 1). The illustrative example corresponds to a bi-objective minimization problem for which a Pareto set with seven nondominated solutions has been obtained. The example shows the visualization of the Pareto front approximation, distribution of the two design variables, and the generated moGram. The generation of a moGram is divided into the following four phases.

- 1) *Design Space Visualization:* According to the managerial switching cost defined in Section II-B, in this paper $C(X_1, X_2)$ is given by the similarity metric

$$\begin{aligned} \text{Sim} &: \mathbb{R}^n \times \mathbb{R}^n \rightarrow [0, 1] \\ \text{Sim}(X_1, X_2) = 1 &\iff C(X_1, X_2) = 0 \\ \text{Sim}(X_1, X_2) = 0 &\iff C(X_1, X_2) = \infty \end{aligned} \quad (1)$$

where X_1 and X_2 are two Pareto-optimal solutions. X_1 and X_2 are *similar* solutions iff $\text{Sim}(X_1, X_2) \simeq 1$.

To build a moGram, the similarity metric must be defined to represent the relationship between all the p nondominated solutions and it is typically specific for each multiobjective problem. Sim provides the degree of similarity in the design space of a pair of solutions of a multiobjective problem. The computation of this metric for all the p solutions ends up with a p -square symmetric matrix S

$$S = \begin{pmatrix} 1 & \dots & \text{Sim}(X_1, X_p) \\ \text{Sim}(X_2, X_1) & \dots & \text{Sim}(X_2, X_p) \\ \dots & \dots & \dots \\ \text{Sim}(X_p, X_1) & \dots & 1 \end{pmatrix}. \quad (2)$$

A complete network is generated from this matrix S where the weight of the network edge between solutions X_i and X_j corresponds to the value from the position (i, j) of S . Equation (3) presents the S matrix obtained from calculating the similarity values for all the seven nondominated solutions of our illustrative example (see Fig. 1). Notice that the highest values are presented in bold and represent the most similar solutions. For a better visualization, the thickness of each edge of the network is proportional to the similarity value between the two solutions represented by the linked nodes. For instance, see in the example of Fig. 1 that the edge between nodes 1 and 2 is thicker than the edge between nodes 2 and 7. Edges also present the numerical labels of the similarity values from S

$$S = \begin{pmatrix} 1 & \mathbf{0.8} & \mathbf{0.7} & 0.1 & 0.1 & 0.1 & 0.1 \\ \mathbf{0.8} & 1 & \mathbf{0.7} & 0.1 & \mathbf{0.6} & 0.1 & \mathbf{0.5} \\ \mathbf{0.7} & \mathbf{0.7} & 1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 1 & \mathbf{0.65} & 0.1 & 0.1 \\ 0.1 & \mathbf{0.6} & 0.1 & \mathbf{0.65} & 1 & \mathbf{0.7} & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & \mathbf{0.7} & 1 & 0.1 \\ 0.1 & \mathbf{0.5} & 0.1 & 0.1 & 0.1 & 0.1 & 1 \end{pmatrix}. \quad (3)$$

- 2) *Objective Space Visualization:* moGrams provide information regarding the objective space of the solutions obtained by using different colors and opacity. The number of nodes of the moGram corresponds to the number of nondominated solutions. Every node is divided into m sectors of the same size, where m is the number of objectives in the optimization problem. Every sector is assigned a different color whose opacity is proportional to the quality of the nondominated solution in the objective represented by that sector. The color intensity may range from fully transparent color (corresponding to the worst objective value) to fully opaque color (related to the best objective value). moGrams thus allow the decision maker to get a detailed insight of the solution not only in the design space but also in the objective space. In the illustrative example (Fig. 1), each node is divided in half (180° slices). We assign the orange color to the first objective (the upper half of the node), while the blue color represents the second objective (the lower half of the node). For example, solution 1 has a very strong orange color in the upper half and is white (the blue color is fully transparent) in the lower half. Thus, we

Algorithm 1: Pathfinder (S, q)

```

input :  $S$  /* Original network */
          $q$  /*  $q$  constrains the number of
indirect nodes checked ( $2 \leq q \leq p-1$ ) */
output:  $S'$  /* Resulting scaled network */
begin
   $S' \leftarrow S$ 
  for  $i = 0$  to  $q$  do
    foreach node  $a$  in  $S'$  do
      foreach node  $b$  in  $S'$  do
         $ps \leftarrow$  obtain all paths between ( $a, b$ )
         $S' \leftarrow$  eliminate edges producing
          costly paths ( $S', ps$ )
  return  $S'$ 

```

infer that this node has a high fitness value (i.e., good quality) for objective 1 and a low fitness value (i.e., low quality) for objective 2. As we see in the Pareto front visualization (Fig. 1 (top left)), this solution 1 is located in the left upper extreme of the Pareto front approximation.

- 3) *Network Scaling*: Depending on the number of solutions in the Pareto set approximation, the resulting complete network can be dense and difficult to analyze. Therefore, the pathfinder network scaling method [21] is applied to S in order to prune the network and maintain only the most significant edges. Pathfinder algorithm has two important characteristics: a) it retains the most important edges and b) when applied to a connected network, it provides a connected pruned network (without any isolated node). Notice that, this specific scaling algorithm deals with the scalability of the problem, as pathfinder is appropriate to large social networks composed of hundreds to thousands of nodes and it drastically reduces the number of connections between the nodes. Algorithm 1 briefly shows how pathfinder operates.
- 4) *Network Drawing*: Force-based algorithms are suitable to graphically visualize networks, as already mentioned in Section II-C. We make use of Kamada–Kawai [47] (Graphviz implementation) because it has been proved very effective in combination with pathfinder [49]. Fig. 1 shows the final network of the example where all the solutions are conveniently placed. Kamada–Kawai (shown in Algorithm 2) represents networks following aesthetic criteria by trying to maximize the use of the available space and minimize their global energy—attractive forces are used to attract pairs of endpoints of the graph’s edges toward each other, while simultaneously repulsive forces are used to separate all pairs of nodes.

C. Decision Maker Interaction

moGrams allow the decision maker to interact with the generated network to get more insights about her/his MCDM problem. For instance, the decision maker can adjust the similarity values between the nodes to better compare the

Algorithm 2: Kamada–Kawai (S', k)

```

Input :  $S'$  /* Scaled network */
          $k$  /* Criteria of stable layout */
output:  $G$  /* Graphical representation */
begin
   $G \leftarrow$  place nodes at random ( $S'$ )
   $e \leftarrow$  calculate energy of nodes ( $G$ )
  while not stable layout ( $k, e$ ) do
     $G \leftarrow$  relocate node with the highest energy ( $G, e$ )
     $e \leftarrow$  calculate energy of nodes ( $G$ )
  return  $G$ 

```

different solutions of the problem. Thanks to scaling the network by using a given range of similarity values, the moGram can provide a clearer view of the relationships between nodes because the thickness of their edges will consequently change, allowing the decision maker to better observe the similarities of the solutions represented by those nodes.

Additionally, our framework allows us to remove nodes from the visualization or zoom in. Then, only those nodes with specific characteristics are shown to better focus the analysis. This is interesting to better scale moGrams for problems having Pareto set approximations with a high cardinality. Also, the decision maker can decide to focus the analysis by removing information about any of the existing objectives. Although information about the objective space is normally needed to make a decision about any multiobjective problem, the decision maker could just show the decision space view within the moGram. In this case, all the nodes of the network will have one single and uniform color.

IV. APPLICATION CASE 1: ASSEMBLY LINE BALANCING

An assembly line consists of a set of m workstations and n tasks which divide the manufacturing of a production item. All of them require an operation time and additional information such as their linear area. One usual and difficult problem, called assembly line balancing, is to determine how these n tasks are assigned to m stations fulfilling the line restrictions. Different multiobjective methods were applied to assembly line balancing [24], [50] but firms need to respond to changes in the environment and then, they need flexible assembly line solutions [6], [51]. moGrams can be used to study the flexibility when replacing one assembly line configuration by a more appropriate one when the environment changes.

A. Problem Definition

One assembly line balancing problem variant is the multiobjective TSALBP- m/A which searches for an assembly line configuration to jointly minimize the number of stations of the assembly line (m) and their line area (A) for a given product cycle time (c) [23], [24], [52]. The formal description of TSALBP- m/A is as follows.

- 1) *Decision Variables*: Given a set of n tasks with their temporal and spatial attributes, t_j and a_j , and a precedence

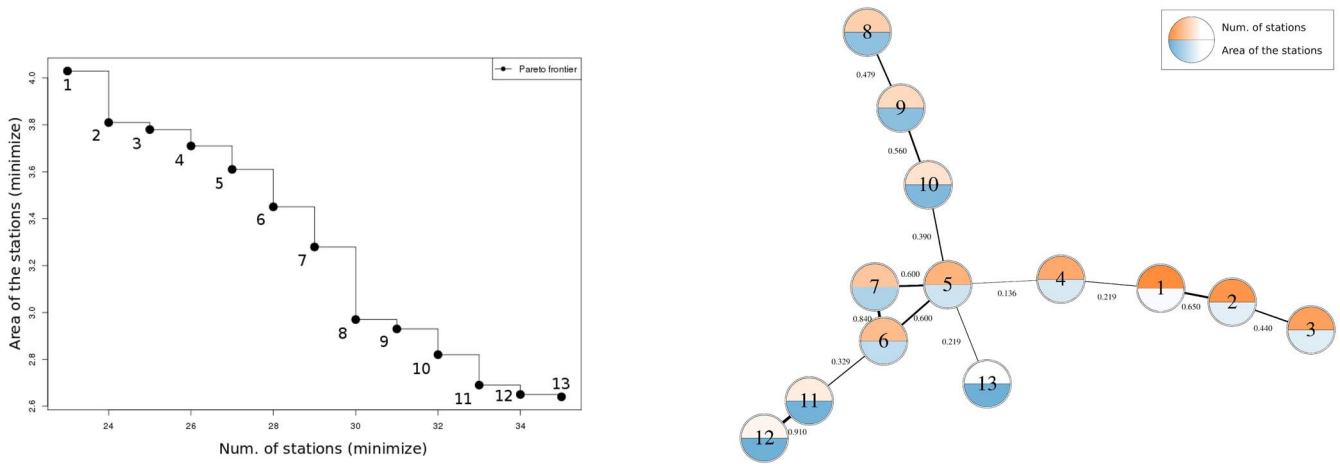


Fig. 2. Pareto front (left) and moGram (right) generated for an instance of the TSALBP problem.

graph, each task must be assigned to just one station k such that all the precedence constraints, workload time, and required area are greater than the global available cycle time c and area A . A solution for the problem is the assignment of all the tasks to the different stations of the line. Each decision variable x_{jk} is then a binary variable taking value 1 if task j is assigned to station k , and 0 otherwise.

- 2) *Objectives*: Two objectives to be minimized are considered, the number of stations m of the assembly line and their area A , defined by (4) and (5), where UB_m is the upper bound of the number of stations (usually equal to the number of tasks n)

$$\text{Min } m = f^0(x) = \sum_{k=1}^{UB_m} \max_{j=1,2,\dots,n} x_{jk} \quad (4)$$

$$A = f^1(x) = \max_{k=1,2,\dots,UB_m} \sum_{j=1}^n a_j x_{jk}. \quad (5)$$

B. Similarity Metric

As said, two assembly line configurations ψ_1 and ψ_2 are characterized by the assignment of n tasks to m_1 and m_2 workstations, respectively. Their station workloads are $(S_1^{\psi_1}, \dots, S_{m_1}^{\psi_1})$ and $(S_1^{\psi_2}, \dots, S_{m_2}^{\psi_2})$. Initially, a similarity index for each station k for the two configuration lines is calculated by

$$\text{Sim}(S_k^{\psi_1}, S_k^{\psi_2}) = \frac{2|S_k^{\psi_1} \cap S_k^{\psi_2}|}{|S_k^{\psi_1}| + |S_k^{\psi_2}|}. \quad (6)$$

Taking into account the latter station similarity index, we define the similarity index $\text{Sim} \in [0, 1]$ in (7) to measure how different two line configurations ψ_1 and ψ_2 are in m stations, where $m = \max\{m_1, m_2\}$. Two assembly line configurations are completely similar when $\text{Sim}(\psi_1, \psi_2) = 1$

$$\text{Sim}(\psi_1, \psi_2) = \frac{1}{m} \sum_{k=1}^m \text{Sim}(S_k^{\psi_1}, S_k^{\psi_2}). \quad (7)$$

C. moGrams Analysis and MCDM Implications

In Fig. 2, we can observe the Pareto front approximation and the moGram generated from the 13 nondominated solutions obtained for a TSALBP instance (420 tasks in a real automotive assembly line (see [6] for more details). The orange color represents the first objective, the number of stations m , while the blue color shows the second objective, the area of the configuration line A . Each edge has a label with its similarity value and its thickness depends on it.

The obtained moGram allows us to get the following observations from the visualization of the network.

- 1) The existing edges present very diverse similarity values, 0.91 being the highest value, whereas 0.136 being the lowest one. The highest values (0.91 and 0.84) are obtained between nodes 11, 12 and 6, 7, respectively. In contrast, the edge between nodes 4 and 5 (0.136) obtains the lowest value. The edges between nodes 1, 4 and 5, 13 also have a low similarity value (0.219).
- 2) We can distinguish three subnetworks based on the high similarity values between some pairs of network nodes. That happens at the top of the network (nodes 8–10), at the bottom-right corner (nodes 1–3), and also in the center of the network (nodes 5–7).
- 3) The subnetwork considering nodes 5–7 (in the center) is an interesting case. It is a fully connected subnetwork with strong connections. Nodes 6 and 7 exhibit the highest similarity value (0.84), while their edges with node 5 have the same similarity (0.6).

From this moGram, a decision maker can get important insights about the TSALBP solving. For instance, the following items are problem-specific conclusions drawn from the visualization.

- 1) Node 1 exhibits the lowest number of stations (represented by a fully opaque orange color). According to its edges, this solution can be transformed into solution 2 and also 4. Notice that transition to node 4 is more costly, due to the lower similarity value.
- 2) Solutions 12 and 13 characterize the lowest area (second objective reaches a fully opaque color). The former is well-connected to its neighbor (the thick edge toward

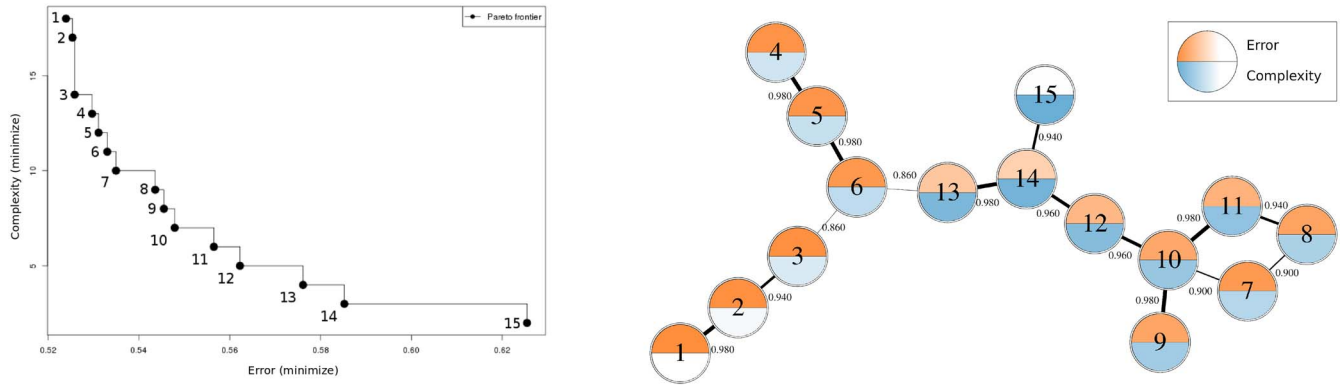


Fig. 3. Pareto front (left) and moGram (right) generated for the CE problem.

node 11) demonstrating very similar characteristics in the design space. On the other hand, solution 13 is directly connected with solution 5, even when having contrasting objective values.

- 3) The most central node is solution 5. It can be easily transformed to solutions 6 and 7; and it is also linked to other nodes (nodes 4, 10, and 13) with different characteristics in the objective space. Although the similarity values between solution 5 and the latter nodes are low, it is a very useful information. It shows the flexibility of solution 5 if the configuration line needs to be transformed into others with different objective values.

V. APPLICATION CASE 2: CLASSIFIER ENSEMBLES

CEs, also called multiclassification systems, are machine learning methods able to obtain better performance than a single classifier when dealing with complex classification problems [26], [27]. The overproduce-and-choose strategy [25] tries to generate the best performing CE by a subsequent selection to extract the optimized classifiers subset to compose the final CE. Therefore, these methods aim at determining the optimal ensemble size by considering a tradeoff between their accuracy and complexity. This problem has a multiobjective nature and different EMO algorithms have been applied to improve the accuracy-complexity tradeoff of the final CE [28], [29]. Depending on the problem demands, a decision maker could need to convert one CE to either a more accurate or to a less complex one. For instance, when having a more limited computational resources, a model with lower complexity is preferred. We apply moGrams to help when obtaining flexible CEs solutions and visually analyze the relationship in the design space of the CE solutions of the Pareto set approximation.

A. Problem Definition

The CE selection multiobjective problem tackled in this paper is formally described as follows [28], [29].

- 1) *Decision Variables*: A solution to this problem, ce , is coded by a binary string in a way that each binary value cl_k of solution ce is assigned to a specific classifier of the initial pool. Then, an ensemble is defined by $ce = (cl_1^{ce}, cl_2^{ce}, \dots, cl_{n_{cl}}^{ce})$ where n_{cl} is the maximum number

of base classifiers from the initial pool of base classifiers. When the binary value of cl_k is 1, it means that a given classifier is included in the final ensemble, while 0 stands for classifier exclusion.

- 2) *Objectives*: We consider two conflicting objectives to be minimized, defined in (8) and (9): the training error of the CE (accuracy) and the ensemble complexity (Cplx) [28]. The error is the rate of incorrectly classified examples (majority voting rule) among the total number of examples in the training dataset m . Complexity is directly the number of classifiers of the CE (i.e., ce)

$$\text{Min } E_{tra} = f^0(x) = \frac{\#\text{incorrect}}{m} \quad (8)$$

$$\text{Cplx} = f^1(x) = |ce|. \quad (9)$$

B. Similarity Metric

The similarity metric measures the difference between the base classifiers that compose two CE solutions. We use the normalized *Hamming distance* [53] to compute the distance between the two binary strings which internally represent two CE solutions ce_1 and ce_2 . Afterward, this distance is subtracted from 1 to calculate the similarity metric $\text{Sim}(ce_1, ce_2) \in [0, 1]$, formally defined as

$$\text{Sim}(ce_1, ce_2) = 1 - \frac{\sum_{k=1}^{n_{cl}} |cl_k^{ce_1} - cl_k^{ce_2}|}{n_{cl}}. \quad (10)$$

C. moGrams Analysis and MCDM Implications

Following the procedure of Section III, a moGram is generated for an instance of a CE selection multiobjective problem (the abalone dataset; see [28] for more details) with 15 different solutions from the Pareto front approximation (Fig. 3). The orange color is associated with the accuracy, while the blue color represents complexity. Since the edges present akin similarity values (their range is quite narrow), the final moGram visualization was previously adjusted by the user to represent similarities in a clear way.

The generated moGram offers a representation of the solutions in the design space and leads to the following observations. Unlike the previous application case (Section IV), the existing edges present high similarity values (all above 0.86). When looking for the most accurate solution, node 1 is the

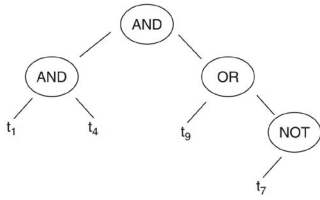


Fig. 4. Example of the representation of a Boolean query solution.

suggested choice. However, it is separated from the rest of the nodes in the design space. So, if a flexible but still accurate solution is desired, then solution 6 is the proposed one. It is connected to solutions with different objective values and can be easily modified to more accurate or less complex solution (solutions 3, 5 and 13). In contrast, node 15 is the one having the least complex CE, but it is also isolated from other nodes, as it has only one edge. Then, it is not the best choice when looking for flexibility. Globally, this moGram shows that node 10 is the most flexible solution and one of those solutions with a good accuracy-complexity tradeoff.

VI. APPLICATION CASE 3: BOOLEAN QUERIES FOR INFORMATION RETRIEVAL

IR is the problem of selecting documentary information from a repository based on search queries introduced by a user [54]. In IQBE [55], a query is automatically generated to describe the content information of a provided set of documents provided. Cordón *et al.* [30] and López-Herrera *et al.* [56] formulated IQBE as a multiobjective problem and tackled it using EMO algorithms, obtaining several queries with a different precision-recall tradeoff in a single run. This tradeoff means that, for making an appropriate IBQE decision, it is necessary to obtain a high fraction of retrieved documents that are relevant (i.e., a high precision) as well as retrieving a high fraction of the relevant documents from the repository (i.e., a high recall) [57]. moGrams can visually represent the solutions to the multiobjective IQBE problem to identify difficulties to transform one query solution to others, if the problem's needs changes.

A. Problem Definition

The IQBE multiobjective problem is defined as follows.

- 1) *Decision Variables*: Boolean queries are query languages composed of query terms and combined by logical operators AND, OR, and NOT [58]. A Boolean query bq is encoded as an expression syntax tree. The terminal nodes are query terms, whereas the inner nodes are the latter logical operators. Each term or operator of the Boolean query is treated as a single entity or character en_{bq} of the query string bq . Hence, the natural representation is to encode query bq as a syntax tree and to work with a genetic programming algorithm to evolve the EMO as done in [56]. Fig. 4 shows a graphical example of a query with four terms and four Boolean operators.
- 2) *Objectives*: Precision (P) and recall (R) are two common measures for the IR system quality [58]. They are the

two conflicting IQBE objectives to maximize. Precision is the ratio between the relevant documents retrieved in response to a query and the total number of documents retrieved. Recall is the ratio between the number of relevant documents retrieved and the total number of relevant documents for the query that exists in the database. Both objectives are defined as

$$\text{Max } P = f^0(x) = \frac{D_{\text{tr}}}{D_{\text{r}}} \quad (11)$$

$$R = f^1(x) = \frac{D_{\text{rt}}}{D_{\text{r}}} \quad (12)$$

where D_{tr} is the number of relevant documents retrieved, D_{r} is the total number of documents retrieved, and D_{rt} is the total number of relevant documents for the query which exist in the database. P and R are defined in $[0, 1]$, being 1 their optimal value.

B. Similarity Metric

The defined similarity metric is the *edit* or *Levenshtein distance* [59] which measures the similarity between two Boolean queries bq_1 and bq_2 . *Levenshtein distance* aims at finding the simplest way to transform one string into another. The transformations of the *Levenshtein distance* are one-step based operations of insertion, deletion, and substitution. *Levenshtein distance* is computed by a dynamic programming algorithm and it is defined by the recurrence of the following equation (see [60] for more details):

$$Ldist(i, j) = \min \begin{cases} Ldist(i-1, j) + 1 \\ Ldist(i, j-1) + 1 \\ Ldist(i-1, j-1) + 1_{(en_{i1} \neq en_{j2})} \end{cases} \quad (13)$$

where $1_{(en_{i1} \neq en_{j2})}$ is the indicator function, equal to 0 when $en_{i1} = en_{j2}$ and to 1 otherwise. Equation (14) shows how the latter distance is normalized and subtracted from 1 to compute the similarity metric $\text{Sim}(bq_1, bq_2)$ between solutions bq_1 and bq_2

$$\text{Sim}(bq_1, bq_2) = 1 - \frac{Ldist(|bq_1|, |bq_2|)}{\max(|bq_1|, |bq_2|)} \quad (14)$$

with $|bq_1|$ and $|bq_2|$ being the total number of entities in Boolean queries bq_1 and bq_2 , respectively.

C. moGrams Analysis and MCDM Implications

Fig. 5 shows the Pareto front approximation and the moGram for the IQBE problem applied to the Cranfield collection [30]. The orange and blue colors of nodes mean the precision and recall, respectively. In the light of this moGram, we can observe the following.

- 1) There are three different subsets or clusters of solutions in the design space.
 - a) Solution 1, which maximizes precision, is far from the rest of solutions conforming the first subset. Although it has seven edges, their similarity values are very low in all the cases.

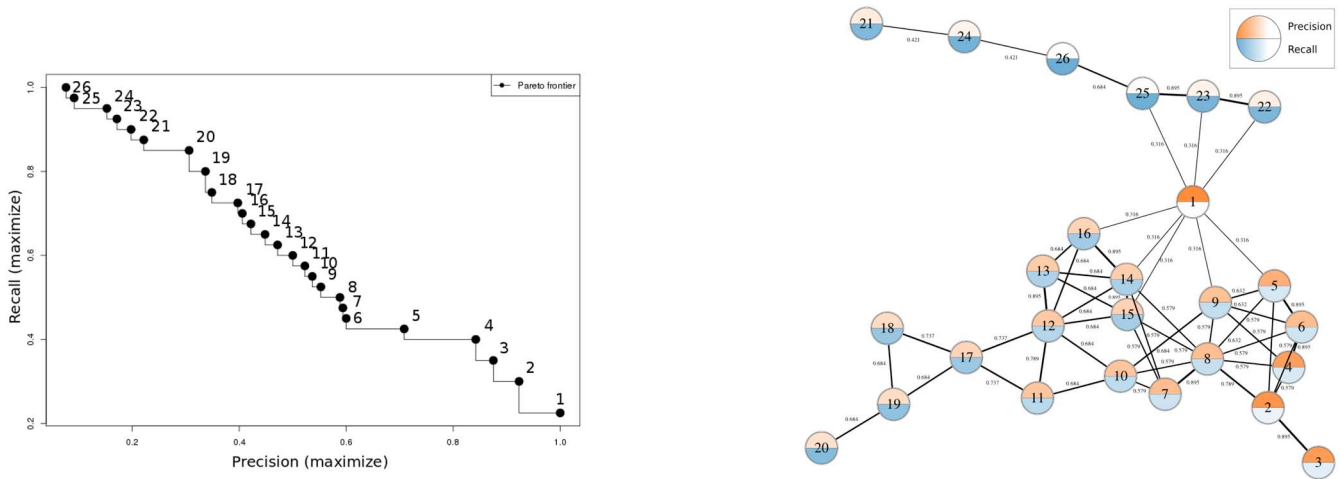


Fig. 5. Pareto front (left) and moGram (right) generated for the Boolean queries problem.

- b) Solutions 2–20 are highly related to each other and form a subset of solutions with good tradeoff between precision and recall but without extreme values (except for solution 2).
 - c) Solutions 21–26 clearly maximize recall at the expense of reducing precision.
- 2) Although solution 1 obtains the best precision, alternatively solution 2 also reaches very good precision values and is much more flexible, highly connected with solutions 3–6 and 8. In case flexibility is required, solution 2 is suggested when looking for high precision.
 - 3) If we look for a solution having high recall, we can choose any of the solutions from 21 to 26. However, solution 25 obtains high recall and it is flexible (it is directly connected to three nodes). Therefore it is one of the best options.
 - 4) Globally, the most flexible solution is 8 as it has connections to other nine alternatives. However, any solution from 2 to 20 is also flexible when needing to be changed to other solutions.

VII. APPLICATION CASE 4: DTLZ4 BENCHMARK PROBLEM

In this section, we explore the use of moGrams over a classical benchmark multiobjective problem: DTLZ4 with four objectives. A more comprehensive description of this problem can be found in [31]. The main goal of this application case is to show the scalability capabilities of the proposed visualization method when tackling an optimization problem with a higher number of objectives and Pareto sets with a large cardinality.

A. Problem Definition

The DTLZ problems are a suite of continuous optimization problems designed to test multiobjective evolutionary algorithms using a generic multiobjective problem generator [31], [61]. In particular, the DTLZ4 problem has a spherical Pareto-optimal front and it is a highly multimodal problem. Its

scalability with respect to both search and objective space is another interesting characteristic.

- 1) *Decision Variables*: The decision variables are composed of 13 real numbers normalized to $[0, 1]$. Mathematically, they are defined as a vector of decision variables $X = (x_1, x_2, \dots, x_{13}), \forall x_i \in [0, 1]$.
- 2) *Objectives*: This optimization problem has four objective functions to be minimized. These objective functions are defined as

$$\begin{aligned}
 f^0(X) &= (1 + g(X)) \cos(x_1^\alpha \Pi/2) \dots \cos(x_3^\alpha \Pi/2) \\
 f^1(X) &= (1 + g(X)) \cos(x_1^\alpha \Pi/2) \dots \sin(x_3^\alpha \Pi/2) \\
 f^2(X) &= (1 + g(X)) \cos(x_1^\alpha \Pi/2) \sin(x_3^\alpha \Pi/2) \\
 f^3(X) &= (1 + g(X)) \sin(x_1^\alpha \Pi/2)
 \end{aligned} \tag{15}$$

where $g(X) = \sum_{x_i \in X} (x_i^\alpha - 0.5)^2$ and $\alpha = 100$.

B. Similarity Metric

We use the normalized Euclidean distance metric to relate each pair of solutions, represented by real-valued arrays X_1 and X_2 . Then, this distance is subtracted from 1 to calculate the similarity metric $\text{Sim}(X_1, X_2) \in [0, 1]$ as shown in the following equation:

$$\text{Sim}(X_1, X_2) = 1 - \sqrt{\frac{(x_{11} - x_{21})^2 + \dots + (x_{1n} - x_{2n})^2}{n}} \tag{16}$$

C. moGrams Analysis and MCDM Implications

A Pareto front with 100 nondominated solutions was obtained for DTLZ4 problem. The top left plot of Fig. 6 presents a view of the Pareto front for the first two objectives while the top right plot shows the design space distribution of these solutions for the first two decision variables. Fig. 6 (bottom) shows the generated moGram for these 100 solutions. We present the complete moGram and zooms of its different parts. The orange, blue, yellow, and purple colors represent the value of the four objective functions $f^0(X)$ to $f^3(X)$, respectively. Notice that our approach succeeds to represent the information associated to the four objectives in a single graph, in addition

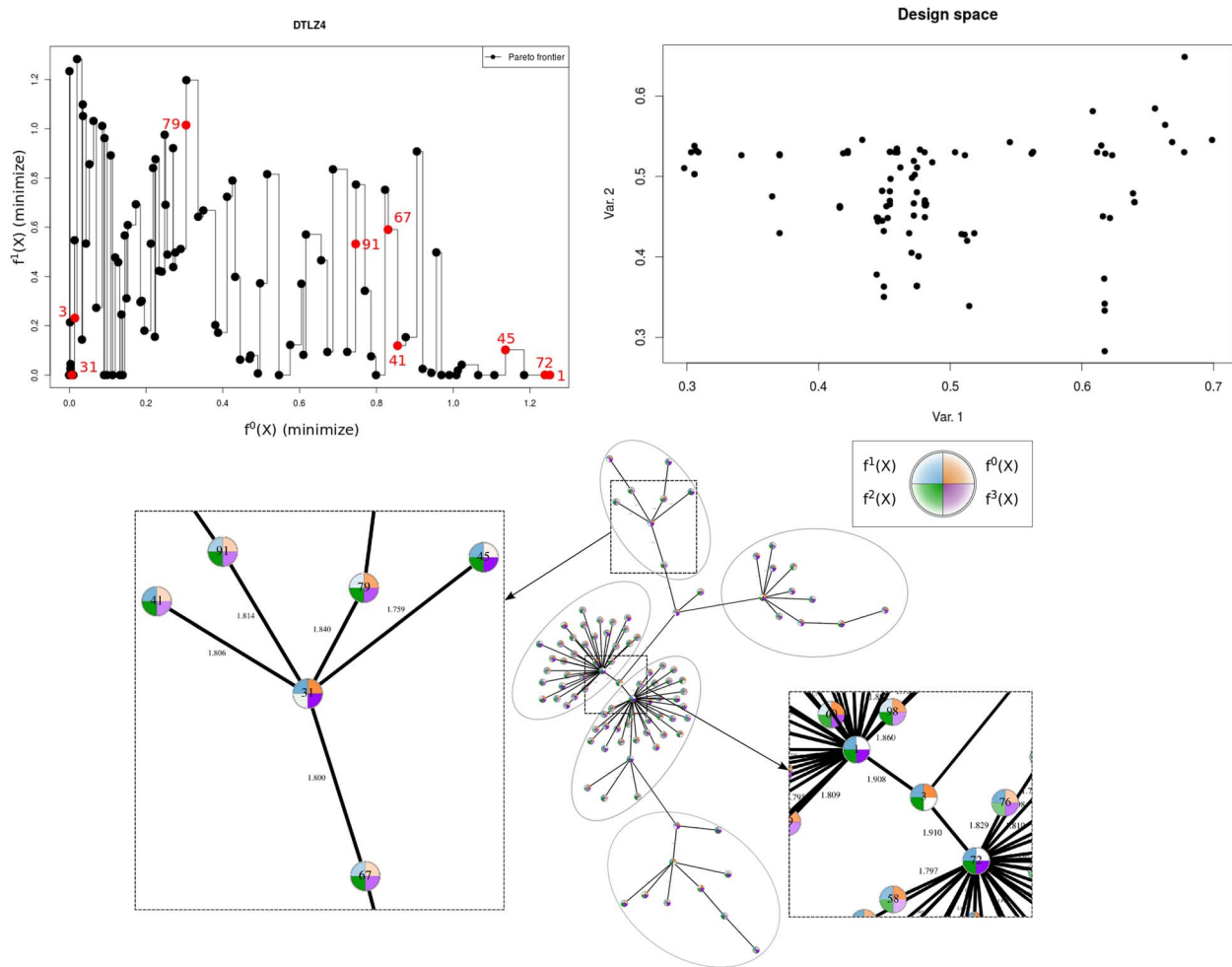


Fig. 6. Pareto front showing the first two objectives (top left), distribution of the solutions in the first two decision variables (top right), and moGram (bottom) generated for the DTLZ benchmark problem.

to the design space relations. From the moGram visualization we can obtain the following observations.

- 1) The DTLZ4 moGram offers a salient and clear view of the nondominated solutions. We can distinguish five areas of more related solutions, marked with oval shapes, where one solution has a central role in each of them.
- 2) The views of the Pareto front do not provide interesting information. This is a common problem when dealing with more than two or three objectives.
- 3) Solutions 1, 3, and 72 are the most central nodes. Solutions 1 and 72 can be easily transformed to other solutions with different characteristics in the design space. Also, solution 3, which links solutions 1 and 72, has different characteristics than solutions 1 and 72. It should be remarked that solutions 1 and 72 are almost considered equal in the objective space, as observed in the Pareto front views of Fig. 6, and also are quite similar in the design space, although they are not directly connected but close in the moGram.
- 4) Solution 31 seems to be a very good solution optimizing objectives 0, 1, and 3. Furthermore, it is connected to three different kind of solutions.
 - a) Intermediate solutions that have a good tradeoff optimizing the four objectives (solutions 41, 67, and 91).

- b) A solution that optimizes all but objective 1 (solution 79).
- c) A solution that optimizes all but objective 0 (solution 45).

This case study allows us to illustrate the potential of moGrams when dealing with a high number of objectives. Due to its characteristics, it is easy to uncover the backbone of the set of related solutions and also zoom and study sets of solutions in detail.

VIII. CONCLUSION

We have proposed a novel methodology for visualizing and analyzing nondominated solutions for multiobjective problems. It provides the decision maker with a graphical exploration of the design space. In particular, it shows the relationships between the obtained solutions using a network. moGrams presents several interesting characteristics: it gives a clear insight into the design space, it incorporates a joint visualization of both design and objective spaces, and it is a generic and scalable approach able to deal with problems with more than three objectives. The decision maker interaction is also possible to better analyze the network of solutions.

Our proposal is based on SNA techniques for constructing, scaling, and drawing the network. The moGrams methodology

was applied to four MCDM applications coming from very different research fields such as time and space assembly line balancing, CEs, Boolean queries for IR, and a continuous optimization problem. We have shown the capabilities of moGrams and how they can help the decision maker when analyzing the composition of the Pareto set approximation and choosing the final solution.

Therefore, we think moGram is a powerful visualization technique, which aids in the understanding of the problem and the flexibility of the solutions represented by their similarities in the design space. Clusters of solutions and/or the most flexible solutions are easily detected thanks to the network-based visualization. This representation of the relationships between solutions in a network makes moGrams a novel approach in the multicriteria visualization literature. However, we should also notice that moGrams generation is not direct and could require more effort than other existing multicriteria visualization methods. But in our view, this additional effort is valuable as it can help with the understanding of the complex set of solutions to a multiobjective problem. In addition, moGrams generation follows a systematic approach based on two well defined methods, pathfinder pruning and Kamada–Kawai layout. Hence, it can be largely automatized by just defining the specific similarity metric for the tackled multiobjective optimization problem.

There are several future research lines that can be investigated. First, our idea is to apply SNA metrics [62] to automatically analyze moGrams by using its network-based properties. We will specifically focus on centrality metrics which provide structural local measurements for each node of the network. These metrics share the common objective of identifying key actors in social networks and they are considered as a measure of centrality or prestige [19]. Examples of these metrics are the closeness or betweenness of the nodes.

Another opportunity is to apply community discovery algorithms to obtain clusters of solutions directly from the network [63]. Many complex networks have been shown to have and detecting those structures is very important for understanding the organization and function of networks [64]. We will also investigate the use of machine learning techniques to obtain associations between the design space and the objective space for all the solutions of a problem. All the latter methods can automatically provide the decision maker with more information about the best-related solutions to the multiobjective problem in terms of the design space. This posterior automatic analysis is also one of the advantages of our moGrams methodology. Finally, we would like to create an open Web tool for the generation and analysis of moGrams.

REFERENCES

- [1] V. Chankong and Y. Y. Haimes, *Multiobjective Decision Making: Theory and Methodology*. New York, NY, USA: North-Holland, 1983.
- [2] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed. New York, NY, USA: Springer, 2007.
- [3] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.
- [4] P. P. Bonissone, "Research issues in multi criteria decision making (MCDM): The impact of uncertainty in solution evaluation," in *Proc. 12th Int. Conf. Process. Manag. Uncertainty Knowl. Based Syst. (IPMU)*, Málaga, Spain, Jun. 2008, pp. 1409–1416.
- [5] K. Miettinen, "Survey of methods to visualize alternatives in multiple criteria decision making problems," *OR Spectr.*, vol. 36, no. 1, pp. 3–37, 2014.
- [6] M. Chica, J. Bautista, Ó. Cerdón, and S. Damas, "A multiobjective model and evolutionary algorithms for robust time and space assembly line balancing under uncertain demand," *Omega*, vol. 58, pp. 55–68, Jan. 2016.
- [7] P. J. Fleming, R. C. Purshouse, and R. J. Lygoe, "Many-objective optimization: An engineering design perspective," in *Evolutionary Multi-Criterion Optimization*. Heidelberg, Germany: Springer-Verlag, 2005, pp. 14–32.
- [8] G. M. Stump, M. Yukish, T. W. Simpson, and E. N. Harris, "Design space visualization and its application to a design by shopping paradigm," in *Proc. ASME Design Eng. Tech. Conf. Design Autom. Conf.*, Chicago, IL, USA, 2003, pp. 795–804.
- [9] D. J. Walker, R. M. Everson, and J. E. Fieldsend, "Visualizing mutually nondominating solution sets in many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 2, pp. 165–184, Apr. 2013.
- [10] T. Tušar and B. Filipič, "Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the projection method," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 225–245, Apr. 2015.
- [11] A. V. Lotov, V. A. Bushenkov, and G. K. Kamenev, *Interactive Decision Maps: Approximation and Visualization of Pareto Frontier (Applied Optimization)*, vol. 89. New York, NY, USA: Springer, 2004.
- [12] J. B. Kollat and P. Reed, "A framework for visually interactive decision-making and design using evolutionary multi-objective optimization (VIDEO)," *Environ. Model. Softw.*, vol. 22, no. 12, pp. 1691–1704, 2007.
- [13] O. Kurasova, T. Petkus, and E. Filatova, "Visualization of Pareto front points when solving multi-objective optimization problems," *Inf. Technol. Control*, vol. 42, no. 4, pp. 353–361, 2013.
- [14] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "Test problems for large-scale multiobjective and many-objective optimization," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–14, Aug. 2016.
- [15] S. Obayashi, S. Jeong, and K. Chiba, "Multi-objective design exploration for aerodynamic configurations," in *Proc. AIAA Fluid Dyn. Conf. Exhibit*, vol. 4666. Toronto, ON, Canada, 2005.
- [16] A. Pryke, S. Mostaghim, and A. Nazemi, "Heatmap visualization of population based multi objective algorithms," in *Evolutionary Multi-Criterion Optimization (LNCS 4403)*, S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, Eds. Heidelberg, Germany: Springer-Verlag, 2007, pp. 361–375.
- [17] J. Eddy and K. E. Lewis, "Visualization of multidimensional design and optimization data using cloud visualization," in *Proc. ASME Design Tech. Conf. Design Autom. Conf.*, Montréal, QC, Canada, 2002, pp. 899–908.
- [18] G. Agrawal *et al.*, "Intuitive visualization of Pareto frontier for multi-objective optimization in n-dimensional performance space," in *Proc. 10th AIAA/ISSMO Multidiscip. Anal. Optim. Conf.*, Albany, NY, USA, 2004, pp. 1523–1533.
- [19] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
- [20] D. W. Dearholt and R. W. Schvaneveldt, "Properties of pathfinder networks," in *Pathfinder Associative Networks: Studies in Knowledge Organization*, R. Schvaneveldt, Ed. Norwood, NJ, USA: Ablex, 1990, pp. 1–30.
- [21] R. W. Schvaneveldt, F. T. Durso, and D. W. Dearholt, "Network structures in proximity data," *Psychol. Learn. Motivation Adv. Res. Theory*, vol. 24, pp. 249–284, Dec. 1989.
- [22] F. de Moya-Aneón *et al.*, "Visualizing the marrow of science," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 58, no. 14, pp. 2167–2179, 2007.
- [23] M. Chica, Ó. Cerdón, S. Damas, and J. Bautista, "Multiobjective constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search," *Inf. Sci.*, vol. 180, no. 18, pp. 3465–3487, 2010.
- [24] M. Chica, Ó. Cerdón, and S. Damas, "An advanced multiobjective genetic algorithm design for the time and space assembly line balancing problem," *Comput. Ind. Eng.*, vol. 61, no. 1, pp. 103–117, 2011.
- [25] D. Partridge and W. B. Yates, "Engineering multiversion neural-net systems," *Neural Comput.*, vol. 8, no. 4, pp. 869–893, 1996.

- [26] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. Hoboken, NJ, USA: Wiley, 2004.
- [27] P. Lim, C. K. Goh, and K. C. Tan, "Evolutionary cluster-based synthetic oversampling ensemble (ECO-Ensemble) for imbalance learning," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–12, Jun. 2016.
- [28] K. Trawiński, A. Quirin, and O. Cerdón, "A study on the use of multi-objective genetic algorithms for classifier selection in furia-based fuzzy multiclassifiers," *Int. J. Comput. Intell. Syst.*, vol. 5, no. 2, pp. 231–253, 2012.
- [29] K. Trawiński, O. Cerdón, A. Quirin, and L. Sánchez, "Multiobjective genetic classifier selection for random oracles fuzzy rule-based classifier ensembles: How beneficial is the additional diversity?" *Knowl. Based Syst.*, vol. 54, pp. 3–21, Dec. 2013.
- [30] O. Cerdón, E. Herrera-Viedma, and M. Luque, "Improving the learning of Boolean queries by means of a multiobjective IQBE evolutionary algorithm," *Inf. Process. Manag.*, vol. 42, no. 3, pp. 615–632, 2006.
- [31] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proc. Congr. Evol. Comput. (CEC)*, vol. 1. Honolulu, HI, USA, 2002, pp. 825–830.
- [32] J. M. Chambers and B. Kleiner, "10 graphical techniques for multivariate data and for clustering," in *Classification Pattern Recognition and Reduction of Dimensionality* (Handbook of Statistics), vol. 2, P. R. Krishnaiah and L. N. Kanal, Eds. Amsterdam, The Netherlands: Elsevier, 1982, pp. 209–244.
- [33] A. Inselberg, *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications* (Advanced Series in Agricultural Sciences). New York, NY, USA: Springer, 2009.
- [34] E. H. Winer and C. L. Bloebaum, "Development of visual design steering as an aid in large-scale multidisciplinary design optimization. Part II: Method validation," *Struct. Multidiscip. Optim.*, vol. 23, no. 6, pp. 425–435, 2002.
- [35] G. Agrawal, K. Lewis, and C. Bloebaum, "Intuitive visualization of hyperspace Pareto frontier," in *Proc. 44th AIAA Aerosp. Sci. Meeting Exhibit*, Reno, NV, USA, 2006, pp. 8783–8796.
- [36] S. Jeong, K. Chiba, and S. Obayashi, "Data mining for aerodynamic design space," *J. Aerosp. Comput. Inf. Commun.*, vol. 2, no. 11, pp. 452–469, 2005.
- [37] M.-J. Jeong, T. Kobayashi, and S. Yoshimura, "Multidimensional visualization and clustering for multiobjective optimization of artificial satellite heat pipe design," *J. Mech. Sci. Technol.*, vol. 21, no. 12, pp. 1964–1972, 2007.
- [38] X. Blasco, J. M. Herrero, J. Sanchís, and M. Martínez, "A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization," *Inf. Sci.*, vol. 178, no. 20, pp. 3908–3924, 2008.
- [39] F. Kudo and T. Yoshikawa, "Knowledge extraction in multi-objective optimization problem based on visualization of Pareto solutions," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Brisbane, QLD, Australia, 2012, pp. 1–6.
- [40] M. Kubota, T. Itoh, S. Obayashi, and Y. Takeshima, "EVOLVE: A visualization tool for multi-objective optimization featuring linked view of explanatory variables and objective functions," in *Proc. 18th Int. Conf. Inf. Vis. (IV)*, Paris, France, Jul. 2014, pp. 351–356.
- [41] B. Vargas-Quesada and F. de Moya-Anegón, *Visualizing the Structure of Science*. Heidelberg, Germany: Springer-Verlag, 2007.
- [42] C. Chen and S. Morris, "Visualizing evolving networks: Minimum spanning trees versus pathfinder networks," in *Proc. IEEE Symp. Inf. Vis.*, Seattle, WA, USA, 2003, pp. 67–74.
- [43] M. Zizi and M. Beaudouin-Lafon, "Accessing hyperdocuments through interactive dynamic maps," in *Proc. ACM Eur. Conf. Hypermedia Technol.*, Edinburgh, U.K., 1994, pp. 126–135.
- [44] S. Noel, C. H. Chu, and V. Raghavan, "Visualization of document co-citation counts," in *Proc. IEEE Symp. Inf. Visualisation*, London, U.K., 2002, pp. 691–696.
- [45] M. Li, S. Yang, J. Zheng, and X. Liu, "ETE: A Euclidean minimum spanning tree-based evolutionary algorithm for multi-objective optimization," *Evol. Comput.*, vol. 22, no. 2, pp. 189–230, 2014.
- [46] S. G. Kobourov, "Force-directed drawing algorithms," in *Handbook of Graph Drawing and Visualization*, R. Tamassia, Ed. Boca Raton, FL, USA: CRC Press, 2012, ch. 12.
- [47] T. Kamada and S. Kawai, "An algorithm for drawing general undirected graphs," *Inf. Process. Lett.*, vol. 31, no. 1, pp. 7–15, 1989.
- [48] T. M. J. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Softw. Pract. Experience*, vol. 21, no. 11, pp. 1129–1164, 1991.
- [49] D. P. Pancho, J. M. Alonso, O. Cerdón, A. Quirin, and L. Magdalena, "FINGRAMS: Visual representations of fuzzy rule-based inference for expert analysis of comprehensibility," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 6, pp. 1133–1149, Dec. 2013.
- [50] M. Chica, O. Cerdón, S. Damas, and J. Bautista, "A multiobjective GRASP for the 1/3 variant of the time and space assembly line balancing problem," in *Trends in Applied Intelligent Systems* (Lecture Notes in Artificial Intelligence), vol. 6098. Heidelberg, Germany: Springer-Verlag, Jun. 2010, pp. 656–665.
- [51] M. Chica, Ó. Cerdón, S. Damas, and J. Bautista, "A robustness information and visualization model for time and space assembly line balancing under uncertain demand," *Int. J. Prod. Econ.*, vol. 145, no. 2, pp. 761–772, 2013.
- [52] M. Chica, Ó. Cerdón, S. Damas, and J. Bautista, "Multiobjective memetic algorithms for time and space assembly line balancing," *Eng. Appl. Artif. Intell.*, vol. 25, no. 2, pp. 254–273, 2012.
- [53] R. W. Hamming, "Error detecting and error correcting codes," *Bell Labs Syst. Tech. J.*, vol. 29, no. 2, pp. 147–160, 1950.
- [54] R. A. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. Harlow, U.K.: Addison-Wesley, 1999.
- [55] H. Chen, G. Shankaranarayanan, L. She, and A. Iyer, "A machine learning approach to inductive query by examples: An experiment using relevance feedback, ID3, genetic algorithms, and simulated annealing," *J. Amer. Soc. Inf. Sci.*, vol. 49, no. 8, pp. 693–705, 1998.
- [56] A. G. López-Herrera, E. Herrera-Viedma, and F. Herrera, "A study of the use of multi-objective evolutionary algorithms to learn Boolean queries: A comparative study," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 60, no. 6, pp. 1192–1207, 2009.
- [57] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann, 2005.
- [58] C. J. V. Rijsbergen, *Information Retrieval*, 2nd ed. Boston, MA, USA: Butterworth, 1979.
- [59] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Phys. Doklady*, vol. 10, no. 8, pp. 707–710, 1966.
- [60] G. Navarro, "A guided tour to approximate string matching," *ACM Comput. Surveys*, vol. 33, no. 1, pp. 31–88, Mar. 2001.
- [61] V. A. Shim, K. C. Tan, and H. Tang, "Adaptive memetic computing for evolutionary multiobjective optimization," *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 610–621, Apr. 2015.
- [62] P. J. Carrington, J. Scott, and S. Wasserman, *Models and Methods in Social Network Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
- [63] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, nos. 3–5, pp. 75–174, 2010.
- [64] H. Chang, Z. Feng, and Z. Ren, "Community detection using dual-representation chemical reaction optimization," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–14, Sep. 2016.

Authors' photographs and biographies not available at the time of publication.