Enhancing Parallel Coordinates and RadVis Visualizations Using Single- and Multi- objective Optimization

by
Khiria Aldwib

A thesis submitted to the
School of Graduate and Postdoctoral Studies in partial
fulfillment of the requirements for the degree of

Master of Applied Science in Electrical and Computer Engineering

Faculty of Engineering and Applied Sciences
University of Ontario Institute of Technology (Ontario Tech University)
Oshawa, Ontario, Canada
April 2021

Thesis Examination Information

Submitted by: Khiria Aldwib

Master of Applied Science in Electrical, Computer, and Software Engineering

Thesis title:
Enhancing Parallel Coordinates and RadVis Visualizations Using Single- and Multi- objective Optimization

An oral defense of this thesis took place on April 12, 2021 in front of the following examining committee:

Examining Committee:

Chair of Examining Committee Dr. Ying Wang

Research Supervisor Prof. Shahryar Rahnamayan

Examining Committee Member Dr. Masoud Makrehchi

Examining Committee Member Dr. Amir Rastpour

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

Abstract

Data visualization is crucial to discover hidden patterns and relationships in high-dimensional datasets; visualization is an essential branch in data analytics applied in science and engineering fields. This thesis has targeted two state-of-the-art methods from two powerful families of visualization techniques: one with dimension reduction, Radial Coordinate Visualization (RadViz), and the other without dimension reduction, for instance, Parallel Coordinates Plot (PCP). In improving these techniques, evolutionary algorithms have been utilized to determine the optimal ordering of coordinates by considering single- and multi-objectives; using this concept, a smart mutation operator has been proposed and tested comprehensively. In order to investigate the performance of visualization proposed schemes, a benchmark dataset has been proposed, and objective and subjective assessments have been conducted. This investigation shows that the optimal ordering of coordinates can influence crucially visualization results. This thesis's findings can be utilized to enhance other large-scale visualization techniques used in visual-data analytics areas.

Keywords— Visualization; Parallel Coordinates Plot; Pareto-front; Single- and Multi-optimization Algorithms; Radial Coordinate Visualization

Author's Declaration

I hereby declare that this thesis titled, "Enhancing Parallel Coordinates and RadVis Visualizations Using Single- and Multi- objective Optimization " consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Ontario Institute of Technology (Ontario Tech University) to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize University of Ontario Institute of Technology (Ontario Tech University) to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

Khiria Aldwib

_____

Statement of Contributions

Part of the work described in Chapter3 has been published as:

1: Aldwib, A. A. Bidgoli, S. Rahnamayan, and A. Ibrahim. "Proposing a Pareto- VIKOR Ranking Method for Enhancing Parallel Coordinates Visualization". In: 2019 14th International Conference on Computer Science Education (ICCSE). IEEE. 2019, pp. 895–902.

2: K. Aldwib, S. Rahnamayan, and A. Ibrahim. "Enhancing Parallel Coordinates Visualization Using Genetic Algorithm with Smart Mutation". In: 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE. 2020, pp. 3746–3752.

I performed the majority of the experiments and writing of the manuscript.

Acknowledgements

First, I would like to thank Allah for giving me the strength to take my master's degree. Also, I am proud of my country, Libya, to support me financially, even though it is in a difficult economic situation. I hope that Libya will gain strength and flourish one day.

It is my honor to thank Professor Shahryar Rahmahany for supervising me in my master's degree in the thesis-based program and supporting me emotionally and educationally during my journey. He oriented me in terms of my thesis, gave me kind support during my pregnancy, and again when my father passed away last year. My special thanks to Dr. Amin Ibrahim, who was very kind and friendly during my research trip. Furthermore, I gratefully acknowledge Dr. Azam Asilian Bidgoli for offering me the opportunity to work with her.

I would like to express my gratitude to my family, especially my mother and my father. I hope my mother will recover her health soon after her surgery. My thanks to my husband, who stood by me during my journey. Thanks to my children. When I saw how much they grew, I realized how much time I have spent on my studies. I hope they will forgive me for the times when I was missing due to my studies. Last, I would like to thank my special friends, Dr. Hanan Heiba, Dr. Sharareh Kiani Harchegani, and the NICI research group for their support, guidance, and feedback.

Contents

List of Figures

List of Tables

List of Abbreviations

GA	**G**enetic **A**lgorithm
NDS	**N**on-dominated **S**orting
NSGA-II	**N**on-dominated **S**orting **G**enetic **A**lgorithm **II**
NSGA-III	**N**on-dominated **S**orting **G**enetic **A**lgorithm **III**
PCP	**P**arallel **C**oordinates **P**lot
RadVis	**Rad**ial Coordinates **V**isualization
NICI	**N**ature **I**nspired **C**omputational **I**ntelligence **L**ab
MCDM	**M**ulti-**c**riteria **D**ecision **M**aking

Chapter 1

Introduction

This chapter aims to explain the data visualization concept, the categorization of visualization techniques, the thesis contribution, its scope, and organization.

1.1 The Concept and Categorization of Data Visualization Techniques

Data visualization has a history dating back hundreds of years [2]. Thus, visualization plays a crucial role in discovering hidden patterns and relationships in large-scale data as part of the visual data analytic task. It can be defined as how the data is presented so the user can interpret it. For instance, in archaic civilizations, various visualized representation techniques were used such as drawing, stone engraving to document information. Thus, It can be visualized by other people after many years. This is how humans recorded history and knowledge [3]. Lately, data visualization with computer software has allowed for creation of graphical visualization tools such as charts, graphs, bar charts, tables, and diagrams. Computer science has played a critical role in visualizing data, which is used in many areas such as the medical field. One of the recent widely used applications for visualization of different types of data is on "large high-resolution displays" [4]. Besides, in [5], Hrabovskyi et al. tried to improve the data visualization methods used in multi-media applications. On the one hand, the advancement of using algorithms in computer science and engineering helps the customers to explore and understand the data in various and fast ways. The PCP is used to visualize high-dimensional data sets without using the reduction technique. The RadVis plot uses a reduction technique to reduce the M-dimensional space to 2 D space. But it is known that, each tool has its weakness. Optimization algorithms can help to solve some of

these issues. For instance, the PCP may deteriorate due to clutter and many crossing lines between adjacent axes. Also, the RadVis suffers from overlapping data points, which may hide patterns in the data. Consequently, the order of coordinates can affect extremely the output of visualization schemes in an extreme way, so achieving an optimal ordering based on the targeted objective can play an essential role in enhancing visualization techniques. This thesis obtains optimality of the order coordinates in the PCP and the RadVis to improve them when viewed as combinatorial problems.

The Categorization of Visualization Techniques: The amount of data has increased quickly because the more frequent use of apps in many fields. Therefore, the data needs to be visualized in every area in science and engineering. The datasets need to be explored, cognized, and processed, especially in dimensional- and large-scale datasets where visualization becomes complex. Therefore, different techniques have been used for data visualization to discover hidden patterns. The visualization methods can be classified as geometric, Icon-based, Pixel-oriented, Hierarchical, Graph-based, Dynamic, and Hybrid. The most well-known geometric forms are used for low and high-dimensional-dataset visualizations, such as scatter plot matrices. The scatter plot matrices in [6] contains a group of metrics used to show the relationships between bivariable; each matrix visualizes the relationships between two variables. All of these matrices plot 2D space, which helps to analyze the datasets. Hyperslice in [7] is used to visualize scalar functions that have many variables. The high-dimensional processes are analyzed the same way. Therefore, Hyperslice can interpret data in a representation of the orthogonal matrix of a two dimensional slice. However, data visualization techniques can also be classified into leading families with dimensional reduction techniques such as projection pursuit, Prosection Views techniques, and Radial coordinates visualization (RadVis) and without dimensional reduction, for instance, the scatter plot and parallel coordinates plot (PCP). The following paragraph presents some of these techniques.

The projection pursuit technique in [8] is used to project the high dimensional to the low dimensional dataset using the normal distribution. For instance, in [9], Petrakos et al. used projection pursuit VR environment to visualize remotely sensed data, and they tried to use linear projections to project high to low dimensional data. Prosection Views technique [10] is a combined technique of orthogonal projections which maps N-dimension to 2D or 3D dimensional space and selections. This means k-section is the intersection of n-dimension

protective subspace with its respective n-space. The PCP technique proposed by Inselberg in 1985 [11] and developed as a tool in 1990. It displays multi-dimensional data in Euclidean plane R^n [12] similar to Cartesian coordinates. However, N-dimension are introduced as verticals lines such as $\overline{x}_1, \overline{x}_2, ..., \overline{x}_N$. A line connects every point with the corresponding axis, so N $y - axis$ are produced. However, only the connected lines of the data points between adjacent axes are shown as bold polylines in the x, y space. Indeed, it contains $N - 1$ pair neighboring coordinates. These lines can produce many crossing lines and clutter regions. An example of the PCP using the red wine dataset with 12 dimensions and 1599 samples from the UCI machine learning repository [13] is demonstrated in Fig. 1.1 by indicating the clutter and crossing lines.

The RadVis [14] is mapping the M-dimensional datasets to 2D dimensional space using a non-linear mapping. Each data point maps to a 2D space by putting the M-dimension data in the corresponding place on a circle's circumference with springs. This spring corresponds to the data point. However, this mapping technique is prone to be faced with overlapping data points, as shown in Fig. 1.2. This data, called the a1-raw dataset, is about the Gesture Phase Segmentation from the UCI machine learning repository [13] with 15 dimensions. It shows 1747 samples after the RadVis method was applied to it. Consequently, this thesis selects two state-of-the-art methods, the RadVis and the PCP, with and without dimension reduction techniques, respectively, in order to the coordinates' order.

1.2 Research Contribution

This thesis has the following five contributions:

Contribution 1) The first method is to improve the PCP based on rearranging its coordinates' order using an exhaustive search for low-dimensional-dataset visualization; otherwise, using evolutionary algorithms. The PCP suffers from cluttering, overlap, and crossing lines in Fig. 1.1, especially in large-scale and big-dataset visualizations. Therefore, this contribution improves PCP by using exhaustive search. This method is called a Pareto-VIKOR Ranking method for enhancing parallel coordinates visualization [15]. It is a ranking model of reordering coordinates in the PCP based on multi-criteria using exhaustive search and comparison using two or more evaluation metrics. The non-dominated sorting algorithm is a well-known method for comparing vectors in a multi-criteria space based on dominance, which classifies

solutions into diffident levels called Pareto. On each Pareto, there are one or several vectors (solutions). Besides, when the ranking vectors have several levels on the same Pareto, the multi-criteria decision-making measure is necessary. The VIKOR is utilized for this purpose, and works based on a vector's distance to an ideal point. Furthermore, the visualization benchmark datasets are proposed to investigate the success method besides other real and multi-objective datasets.

Contribution 2) The second contribution uses the problems decomposability property to provide a smart mutation operator for single objective evolutionary algorithms [16]. It swaps two axes with the worst objective values to improve GA's performance for enhancing the PCP. In the experiments, the objective is to minimize the crossing lines between adjacent axes, so they swap the two coordinates in the PCP with top-line crossing values to generate candidate solutions with minimum crossing lines. Besides, the Monte Carlo simulation is utilized to compare the smart mutation with other schemes.

Contribution 3) The third method uses non-dominated sorting algorithms NSGA-II and NSGA-III to conduct a comparative study using the multi-criteria concept and VIKOR ranking for high-dimensional datasets to optimize the coordinates' order in the PCP.

Contribution 4) A subjective assessment of visualization results is conducted on NSGA-III results in the third method to compare the algorithm's results to the users' desires criteria.

Contribution 5): The last contribution enhances the RadVis by minimizing the overlapping data points, as demonstrated in Fig. 1.2. This method improves the RadVis for the low-dimensional-dataset visualizations (D<10) using an exhaustive search; otherwise, it uses an evolutionary algorithm for solving the combinatorial problems. Therefore, the method finds the optimal order of the dimensions around the circumference in RadVis based on clustering quality resulted by k-means clustering algorithm.

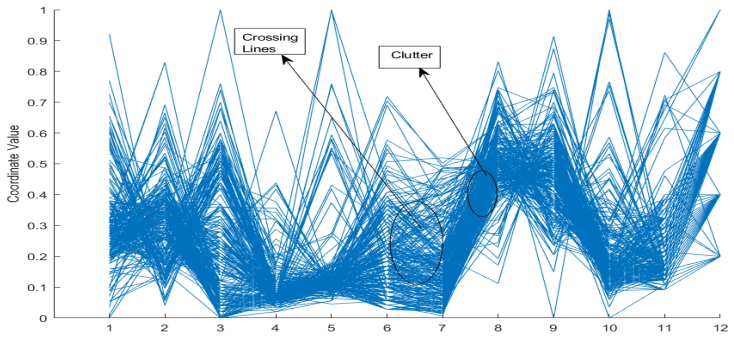


Fig. 1.1: An example to illustrate the PCP drawbacks, such as many crossing lines and clutter regions between the coordinates

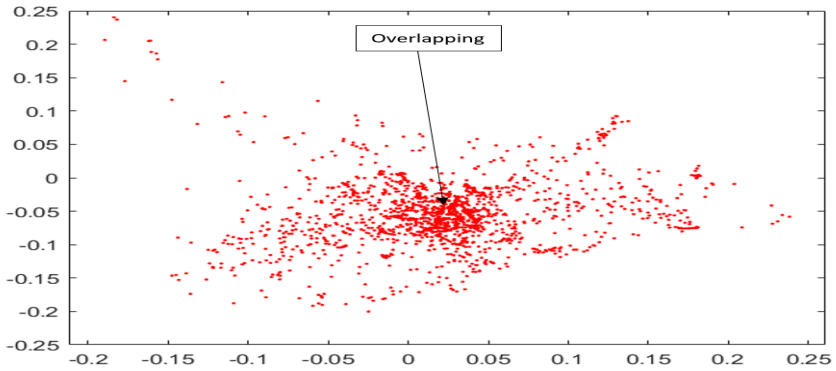


Fig. 1.2: The RadVis indicates the overlapping data points

1.3 The Scope of This Research

This thesis has two main parts; the first one presents three methods to enhance the ordering of PCP coordinates. This part also optimizes the order of coordinates in the PCP so that the viewer can adapt the criterion based on their desire using three methods.This can improve the PCP utilizing more than one measure such as minimizing the crossing lines and maximizing the correlation between adjacent axes using exhaustive search. The second method is to enhance the optimal order in the PCP for high-dimensional-dataset visualizations using a single objective optimizer, GA, with a smart mutation operator to improve the GA's performance. The third one refines the optimal order coordinates' PCP based on multi-criteria using multi-objective algorithms for high-dimensional-dataset visualizations.

The second part of the thesis aims to improve the RadVis that relies on rearranging the coordinates again by utilizing k-means clustering and internal validity measures using exhaustive search for low-dimensional datasets and the GA for high dimensional-dataset visualizations.

1.4 The Organization of the Thesis

The thesis contains the following chapters:

Chapter 2 presents the background review on data visualization techniques, enhancement techniques for the PCP, data visualization and human perception.

Chapter 3 explains methods for enhancing the PCP. The methods applied non-dominated sorting (NDS), multi-criteria decision maker (VIKOR), combinatorial optimization algorithms, and evaluation metrics to evaluate the results.

Chapter 4 demonstrates a technique for enhancing RadVis plot using k-means clustering, genetic algorithm (GA), and internal validity measures.

Chapter 5 provides the summary, conclusion remarks, and future work.

Chapter 2

Background Review

2.1 Introduction

This chapter describes background reviews of with and without dimensional- reduction methods for data visualizations. Also, it represents literature reviews of the PCP, as categorized one of the without dimension reduction-based methods, and the RadVis in the family of dimension reduction-based methods. Therefore, the RadVis and the PCP are utilized for high dimensional-dataset visualizations. The PCP has clutter and many intersection lines between neighboring axes, so many researchers tried to refine the PCP. However, the optimal order of coordinates in the PCP is based on user desire metrics. In this thesis's experiments, the objective is to minimize the crossing lines between axes and maximize the correlation using the two metrics. The RadVis has overlapping data points. This thesis suggests two ways to refine PCP and RadVis by reordering the coordinates.

2.2 Visualization Techniques

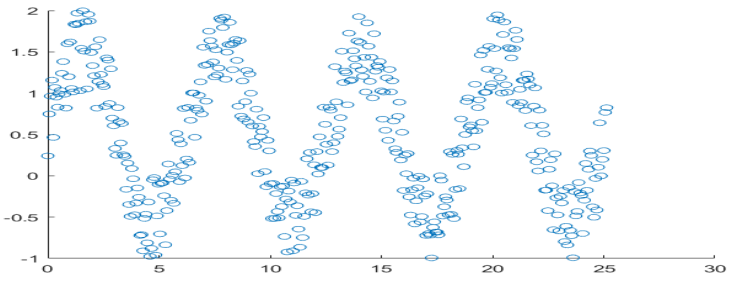
Data visualization is an important task to make information comprehensible and interpretive to the user ; and also discovering hidden patterns and relationships. In [17], Schmidt categorized the visualization tools into five main categories: basic charts and flow charts, matrix, temporal, and hierarchical visualizations. In another way, they can be classified in two big families as with and without dimension-reduction techniques. In the following sections, some of them are explained accordingly.

2.2.1 Visualization without Dimension Reduction

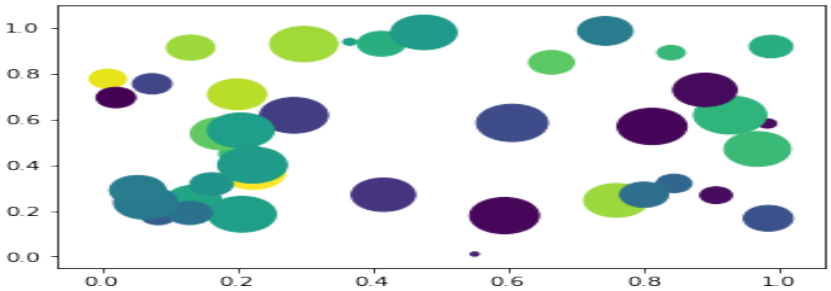
Many visualization techniques have been utilized such as basic charts and the PCP. Basic charts are simple and without dimension reductions, such as scatter plots, pies, Bar-charts,

Histograms and bubble charts.

For instance, the scatter plot visualization illustrates each data entry as one point in a two- or three-dimensional space. It correlates data points and their combinations to allow the user to understand and interpret the data; furthermore, discovering hidden relationships and patterns. However, it is limited in term of the number of dimensions scalability. Besides, visualizing the patterns becomes complicated when datasets get larger [18]. For low-scale datasets (D<=3), the scatter plot is utilized to provide a clear visualization as illustrated in Fig. 2.1. However, for large-scale datasets, the visualizations become more challenging. Multi -dimensional data visualization tools are for more than three-dimensional datasets. Moreover, the bubble chart structure arranges the dots of the data points in 2D space as bubbles and shows the different sizes and colors of the bubbles as the third dimension [19] as shown in Fig. 2.1. Other kinds of visualization techniques are applied to visualize high dimensional-datasets, such as the PCP. The PCP is a powerful technique, but it has clutter. However, this thesis attempts to improve the PCP based on the reordering coordinates strategy .



a) Scatter Plot



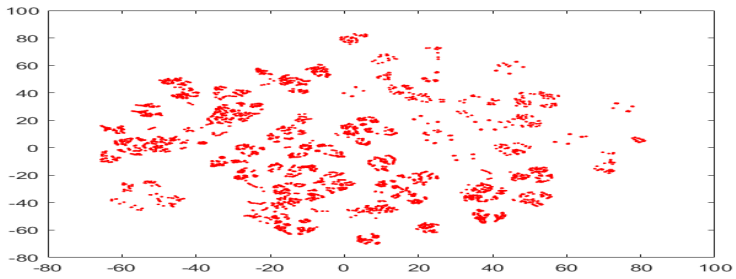
c) Bubble Chart

Fig. 2.1: Examples to demonstrate the Scatter Plot and Bubble Charts

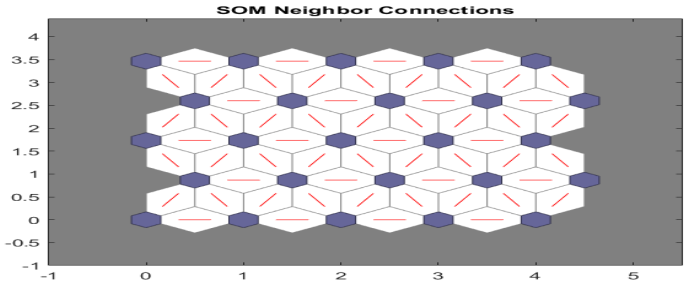
2.2.2 Visualization with Dimension Reductions

Different types of algorithms and tools are proposed to improve the large-scale visualizations with using dimension reduction based methods. One of the utilized methods is Stochastic Neighbor Embedding (SNE) [20]. Unlike other dimension reduction methods, SNE utilizes a probabilistic framework to keep each object's neighboring identities. It uses pairwise dissimilarity techniques to preserve the adjacent identities for each dimension in the low dimensions space. Moreover, the Gaussian distribution is utilized to perform each dimension's center in high dimension and preserves all possible neighbours' probability distribution to each object. The embedding aims to keep this distribution during mapping to the lower space since the same step is applied in low-dimensional space. The aim of matching the distributions from high- to low space is achieved by using minimizing the cost function. The cost function is determined by the sum of Kullback-Leibler divergence between two distributions over all neighbours for each object. Where Kullback-Leibler divergence determines between two probabilities distributions. However, from the SNE method, the t-Distributed Stochastic Neighbor Embedding (t-SNE) approach is created [21], which is an improved variant of SNE and maps m-dimension into two or three-dimensional space with preserving data points neighbourhood property, as shown in Fig. 2.2, a). The t-SNE solves the crowding points problem in SNE. There are two main differences: t-SNE uses the symmetrized cost function and a Student-t distribution instead of the Gaussian type. The local linear embedding system (LLE) method is an unsupervised learning algorithm that maps high-dimensional datasets to lower dimensions and keeps dimensional inputs high by mapping the inputs to manifolds space [22]. Space is called a global axis of lower-dimensional space since the data points become close to an embedded manifold area, an example of LLE plot in Fig. 2.2, c) [23]. The LLE stages can be summarized as follows: first, it finds each data point's neighborhood using k as nearest neighbors. Second, it determines the weights for each point in the embedding space to refer to it in manifold space. Thirdly, it computes the coordinate in low dimension space by mapping the points to one global manifold space using the reconstruction weights matrix. In [24], the supervised learning algorithm (SLLE) for classification, the main achievement of SLLE, finds disjoined manifolds space, which means that each class has a separate fold in low-dimensional space. It locates the neighborhood for each point, which belongs to its class only, not others using Euclidean distance measure. However, SLLE minimizes the

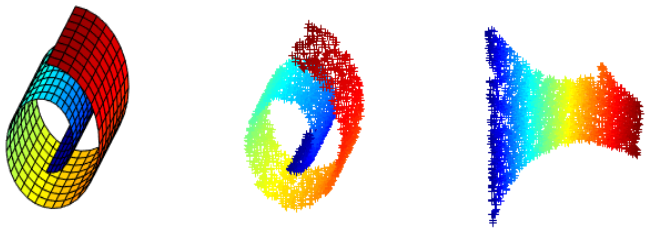
distance among the intraclasses; however, an Enhanced SLLE method (ESLLE) expands the distance among intraclasses using a new proposed distance measure [25]. The Self-organized Map (SOM) [26] uses artificial neural network (ANN) which maps the n-dimension into 2D spaces. This is also an unsupervised learning technique. The input of ANN is the set of data points in the given dimension, which associates with weight vectors for each neuron. A hexagonal or rectangular grid is utilized to represent these nodes in 2D spaces. Therefore, the Unified Distance Matrix (UDM) is used to record each node's average distance to the nearest nodes with a variety of colors for visualization. An example of SOM plot is given in Fig 2.2, b). Consequently, each method has some problems. The PCP suffers from clutter in many intersections among the coordinates. Also, the RadVis has overlapping data points. Therefore, this thesis chooses the RadVis from the dimensional- reduction techniques to be enhanced based on rearranging the dimensions. The method is presented in the following section.



a) t-Distributed Stochastic Neighbor Embedding (t-SNE) Plot



b) Self-organized Map (SOM) Plot



c) Local Linear Embedding (LLE) Plot

Fig. 2.2: Examples to illustrate t-Distributed Stochastic Neighbor Embedding(t-SNE), Self-organized Map (SOM), Scatter Plot Matrix, and Local Linear Embedding (LLE) methods

2.3 The parallel coordinates plot (PCP)

The PCP is a tool used for large-scale visualization without dimension reduction. However, the PCP visualization struggles with some challenges, one is prone to hide the patterns of the information by clutter and/or overlapping, which occur due to the structure of the PCP. The PCP space is an N-dimensional Euclidean space, R^n. All the dimensions are represented as the N vertical axes in 2D space. Each data point in the PCP connects to its adjacent dimension point by a line called polyline. However, only the polylines are plotted out in the 2D space, and the vertical axes are hidden. Typically, crossing lines are high in large-size and

high-dimensional-dataset visualizations in the PCP. Unlike other visualization techniques, the PCP is based on linking each data point between adjacent dimensions by a line [27]. According to [28], crossing lines create clutter between coordinates. So, making these connections creates obfuscations such as overlaps, crossing lines, and clutter, hiding patterns and correlations in the data. Therefore, finding the balance between the minimum number of crossing lines between adjacent axes and the desired correlation among data points can be a major goal. The correlation value is taken over all adjacent axes. It is by calculating pairwise correlation coordinates and take the mean of all. Many techniques have been utilized to enhance the PCP. Some of them are applied with and without dimension reduction, brushing, and hybrid techniques.

One powerful method to improve the PCP visualization is to reorder the coordinates based on the user's desired criteria. Reordering the coordinates in PCP is a combinatorial problem. It aims to enhance the PCP based on an criteria or more. Dasgupta and Kosara introduced a screen- space metric. It gives the best axis order in the PCP by testing the data concerning crossing lines, overlaps, crossing angles, convergence, parallelism, and mutual information using pixel-space histograms to show data distribution. The branch-and-bound algorithm is used to determine the best axis order.

In another interesting work [29], Zhen et al. proposed an algorithm that gives the optimal axis order in the PCP utilized for high-dimensional multi-objective optimization problems depending on the high value of the correlation coefficient among data points. In particular, they use this algorithm to reorder the axes in PCP given vectors of optimal solution objectives based on ascending the highest correlation values among adjacent axes to demonstrate the conflict between these objectives in the PCP. The clarity in the PCP can be negatively affected by clutter and crossing lines, making interpreting patterns inside the data complicated. Accordingly, in [30] Peng et al. identified the clutter as outliers and used axis permutations to reduce the number of outliers among neighboring axes. Further, Liang Fu Lu and Mao Lin Huang proposed a clutter reduction method by minimizing the number of edge crossings between axes by optimal ordering axes [31]. They also introduced an algorithm based on similarity measurement among the data points by combining two algorithms, Non-linear Correlation Coefficient and Single Value Composition SVD, to create a similarity matrix to reorder the axes mathematically. Moreover, in [32], the authors proposed an algorithm using

a new clustering and heuristic-based branch-and-bound approach to tackle axes reordering using a proposed polynomial-time approach. The data point's slope between the dimension axes is determined and clustered to different clusters. It shows the variation between data points by slope. If the slope is positive, they classify it as one cluster. If it is negative, they group it as another cluster. If the slope is zero, it is named a different cluster. In [33], the authors have introduced three techniques to enhance PC visualization by reordering the axes. The first determines the correlation among the features and rearranges the coordinates based on that. The second method depends on combining various characteristics and reordering the PC coordinates, relying on the correlation values. The last one is to determine the negative and positive entropy values for each feature descending PCP axes. In [34], Muhammad Sajjad Akbar et al. proposed an approach to minimize the crossing lines between adjacent axes, which improves the interpretation of PCP by using k-means clustering with reordering the axes to interpret the pattern of the dataset. In another recent work in [35], Blumenschein et al. proposed a new method to reorder the axes in the PCP depending on the dissimilarity technique between adjacent coordinates. Therefore, these methods based on reordering the axes are not flexible in choosing the measures because the criteria are predefined by the user. This thesis aims to provide reordering the coordinates in the PCP based on the user's desired criteria.

Moreover, enhancing the PCP using interactive techniques is widely utilized. For example, Raseman et al. have introduced an interactive visualization tool, Parasel library, to visualize multi-objective decision making for helping web application developers [36]. Also, [37], Jing Yang et al. present "an interactive hierarchical dimension ordering, spacing, and filtering approach called DOSFA". This automatically creates settings to reorder space and filter the coordinates in the PCP to control the dimension management during its process and uses a dataset with over 200 dimensions in their experiment. Therefore, their method enhances high-dimensional visualization techniques effectively. Also, in [38], Chuxuan et al. introduce an interactive visualization tool based on the PCP, which helps the user interpret the high-dimensional dataset and comprehend all the features. However, this required preprocessing dataset to detect the feature of each row in the dataset.

The brushing PCP method is well-known method to improve PCP visualization interactively. In [39], Richard C. Roberts et al. introduced the smart brushing for metadata to guide the

user's perception during the run time of the algorithm to reflect properties of the dataset. Futhermore, clustering techniques has played a rule to improve the PCP. For instance, in [40], Halld'or Janetzko et al. proposed an extension of the PCP to enhance overplotting between adjacent axes in the PCP by using clustering for Soccer dataset to see the distribution of local density utilizing the state-of-the-art visualization techniques.

Furthermore, orientating the polylines in PCP can be useful. In [41], Renata Georgia Raidou et al. have introduced a new technique to augment the visualization in the PCP called the " Orientation enhanced Parallel Coordinate Plots", which helps to reduce the clutter in order to discriminate the patterns between adjacent axes in the PCP. As a result, the visualization on each polyline in the PCP with an appreciative slope improves. Also, the distribution and categorization of the data is an important task to tackle the overlapping in the data. In [40] this work, Janetzko et al. concentrate on the visualizations of the distribution and categorizing the data points. Their previous work focuses on visualizing the global density distribution. However, this work included the data class using the three state of the art algorithm.

Moreover, another method is to bundle the axes in PCP, in [42], Palmas Gregorio and Bachynskyi proposed a method called the edge-bundling method. It utilizes a density clustering technique for each coordinate in the PCP. It is a useful technique to render clusters for each plotline and reduce the rendering time noticeably.

In addition, replacing the coordinates to curves and splatting the lines in the PCP are well-known methods. In [43], Graham and Kennedy introduced a technique to improve the PCP by replacing polylines with curves between adjacent axes that help the user see each element in a limited time. Splatting the Lines in Parallel Coordinates method [44] was proposed by Zhou et al. to improve the PCP. Their framework contains two components, " a polyline splatter for cluster detection and a segment splatter for clutter reduction". It reduces the clutter and discovers the pattern between the polylines.

Lately, in [45], Liangfu Lu et al. proposed an approach for improving the PCP. The method uses the double arc coordinate plot coordinates (DACP). It reduces the thickness around each axis since it has two-arc coordinates for each axis pair.

Consequently, the previous methods are different in improving techniques such as brushing, orientating, reordering, and bundling the axes in the PCP to enhance it. This thesis's proposed ways are different because they optimize the axes' order in the PCP based on the user's

desired criteria and for large-scale visualizations.

2.4 Radial Coordinate Visualization

Radial coordinate visualization (RadVis) is one of several visualization techniques that maps M-dimension into 2D spaces using a nonlinear mapping, Furthermore, RadVis connects M-dimension to a circle by springs as points. Each spring represents one dimension, has a constant value, and maps M-dimensional data to a two-dimensional point [46].For more understanding, [14] Hooke's law is played. The puck is forced for each spring. When it reaches an equilibrium place, the sample $(u_i, v_i)T$ reduces into 2D space. Therefore, each point i $(x_1, ..., x_m)$ maps to $(u_i, v_i)^T$. The forcing accrues for all springs on the puck, so the sum of these acting individual forces will be the force of acting all springs m. If the puck is reached to equilibrium, so the sum of the forces on the puck will be equal to zero. if we denote the dimensions $S_1, S_2, ...S_m$ and the points vector will be $(u_i, v_i)^T$. Thus,

$$\sum_{j=1,m} (S_j - u_i)x_{ij} = 0 \tag{2.1}$$

Where it could be solved giving weights for each point i as:

$$u_i = \sum_{j=1,m} w_{ij}(S_j) \tag{2.2}$$

Where the weights equal to:

$$w_{ij} = \left(\sum_{j=1,m} x_{ij} \right)^{-1} x_{ij} \tag{2.3}$$

Therefore, for each point i, u_i is the mean of the weighted S_j. Then these weights of variable m are normalized. The nonlinear mapping is made using normalization of the data point i between interval $[0, 1]$ to avoid the negative values and maximum and minimum values. These negative values make the mapping of the samples does not organize on the convex hull while it is needed to be regular polygon on it. The normalized methods are the local metric (L-metric) or the global metric (G-metric) can be used. Once the data points are re-scaled, many similar variables will be mapped near the center region, so the overlapping among data points will appear in 2D space.

In Fig. 2.3, Driving performance evaluation dataset (DrivFace) is reduced from 6400 dimensions to 2D using RadVis from the UCI machine learning repository [13]. As indicated in the plot, there is an overlapping in the data points. In recent work [47], Marco et al. presented a new method to mitigate the RadVis' drawbacks using the disposition dimensions in RadVis circumference. This method uses effective heuristics to find the optimal global order of the dimensions around the rim. In other current work [48], Jingjing et al. proposed an algorithm to find the optimal position of the dimensions in RadVis circumference. This method maps the data points using the convex hull and taking the samples' proper order by RadVis reductions. Consequently, this thesis refines the overlapping issue in RadVis by using exhaustive search and GA to optimize the positions of the dimensions' places around the circumferences' RadVis different from previous methods by the internal validity measures for k-means clustering. The internal validity measure is Dunn measure. It is used in the experiments as objective without constraints.

The 3D-RadVis visualization method is used to visualize the solution sets, given multiple objectives. It maps M-dimensions into 3D spaces while keeping possible positions, distributions, and convergence of solution points using hyper-plane reference [49]. The 3D RadVis structure can be summarized in two main points. The first one is calculating the distance between every solution and a reference hyper-plane. Next, using the RadVis reduction to map the locations of M-dimension solutions into 2D space. An example of 3D-RadVis is demonstrated in Fig. 2.4.

The 3D-RadVis Antenna visualization [1] improves the 3D-RadVis method by adding Antennas to each dimension's top. The distributions of the solutions align each objective. Each antenna's location is determined utilizing by the boundary points. The length of the antenna is equal to the maximum distance from the points to the reference hyper-plane. Next the direction of antenna dimensions is determined by multiplying the maximum distance for each point from the reference hyper-plane by the non-dominated solutions and moving them to the top of each antenna ploy. However, increasing the number of dimensions and data points makes understanding relationships between non-adjacent dimensions more difficult.

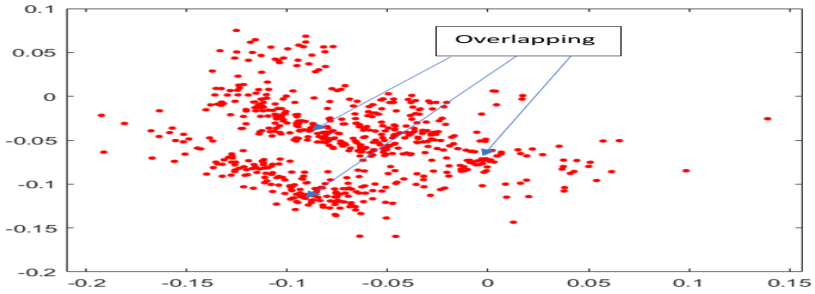


Fig. 2.3: The RadVis Plot

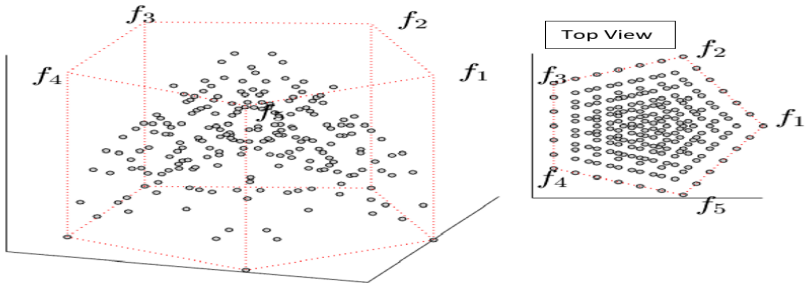


Fig. 2.4: The 3D-RadVis plot, the permission is taken [1]

Chapter 3

Enhancing the PCP

3.1 Introduction

The PCP has clutter and too many intersection lines between adjacent axes in large-size-high-dimensional data visualizations. Enhancing the PCP by the optimal ordering of its axes is a combinatorial decomposable problem. Moreover, having the optimal order of the axes in the PCP is a critical task and can be classified as a multi-objective combinatorial problem. Therefore, multi-objective optimization algorithms have been widely used to solve different problems, such as the traveling salesman problem (TSP), and optimize the order in the PCP in this chapter. Consequently, this chapter presents three methods to optimize the order of coordinates in the PCP. The first method is titled Proposing a Pareto-VIKOR Ranking Method for Enhancing Parallel Coordinates Visualization. This method finds the best order axes in the PCP using an exhaustive search, Non-dominated Sorting (NDS), Multi-criteria Decision Maker VIKOR [50, 51] for more than one metric. In the experiment, the metrics minimize the crossing lines and maximize the correlation between the neighboring axes. The second method optimizes the coordinates order in the PCP using an evolutionary optimization algorithm, the genetic algorithm with a proposed smart mutation operator for single-objective combinatorial problems. This method uses one metric as fitness to find the optimal order of axes in the PCP, minimizing the crossing lines adjacent axes. However, any objective metric could be used based on application and user preference. The third method uses the Non-dominated sorting algorithms, NSGA-II and NSGA-III to solve multi-objective problem that is optimizing the order of coordinates in the PCP based on the user's criteria. The metrics are used as objectives, minimizing the crossing lines and maximizing the correlation between neighboring axes in the PCP.

3.2 Proposing a Pareto-VIKOR Ranking Method for Enhancing Parallel Coordinates Visualization

In this section, the two algorithms non-dominated sorting (NDS) and (VIKOR), are employed to enhance the PCP for lower dimensional datasets (D<=10). This method is named a multi-metric Pareto-VIKOR ranking (PVRPCP) method. An exhaustive search is utilized to find the best order of coordinates in the PCP by calculating the using two metrics; crossing lines between adjacent axes and correlation between data points as fitness values. Therefore, in the first, the fitness values are determined for all possible coordinate permutation axes in the PCP using the given matrices. Next, NDS is utilized to determine the non-dominated solutions. By using NDS, more than one solution of each front have been obtained. In this case, in order to show how much the best order improves visualization compared to the worst-case and the original-case scenarios based on the user's desired metrics, the VIKOR is utilized to rank the solutions for each of the first and last front solutions. Several multi-dimensional benchmarks are designed to investigate the effect of reordering coordinates in the PCP. In addition to author-created benchmarks, several multi-objective function benchmarks and real-world datasets are engaged to evaluate the method.

3.2.1 Multi-criteria Comparison

Many real-world optimization problems have two or more conflicting objectives [50, 51]. The definition of the optimality is not as simple as the single-objective optimization. It is necessary to make a trade-off between objective values. There are some well-known concepts to compare two entities in terms of multiple criteria. Dominance is one of them which is defined as follows. If $x = (x_1, x_2, ..., x_d)$ and $\acute{x} = (\acute{x}_1, \acute{x}_2, ..., \acute{x}_d)$ are two vectors in a multi-objective space in which the minimization of all objectives is desirable, x dominates $\acute{x}$ ($x \prec \acute{x}$) if and only if:

$$\forall i \in \{1, 2, ..., d\}, x_i \leq \acute{x_i} \wedge \exists i \in \{1, 2, ..., d\} : x_i < \acute{x_i} \tag{3.1}$$

This defines optimality for vectors in multi-criteria space. For a minimization problem, vector x is better than $\acute{x}$ if it is not bigger than $\acute{x}$ in any of the criteria and it has at least one smaller value in one of the criteria; i.e., it is better at least in one of them. Fig. 3.1 shows he dominance

concept in a minimization problem. Two objectives should be minimized, so as it is presented in the Fig. 3.1. The vectors that are at the bottom left region of x_1 have values smaller than x_1 in both criteria and they dominate x_1. Similarly, x_1 dominates vectors in top right region of x_1. Other vectors (bottom right and top left) are not comparable with x_1, because they are better only in one criterion and worse or the same in another one. In a set of vectors, all vectors

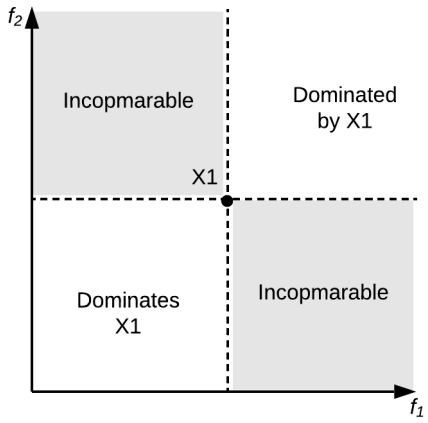


Fig. 3.1: Domination Concept for a Minimization bi- Objectives/Metrics.

which are not dominated by others are called non-dominated vectors; these create the Pareto front set. These vectors can be interpreted as top-rank ones in the set. In order to classify other vectors in different levels, the non-dominated sorting algorithm (NDS) is utilized. This approach sorts a set of vectors into ordered subsets based on the Pareto dominance which works as follows. As it is mentioned, all non-dominated solutions in the first rank are called Pareto front sets, which are F_j,$j = 1 = 1, 2...., k$. In order to identify the second rank of individuals F_2, these non-dominated vectors are removed process the rest of the set in the same way. The non-dominated solutions of this step make the second front F_2. Then the second ranked individuals will be removed to identify the third Pareto $F_j = F_j - F_1 - F_2 + ..., F_K$ and store them to avoid duplication in each pair solutions. This process will continue until all individuals are grouped into different fronts. The steps of the algorithm are presented in Algorithm 1. Fig. 3.2 also shows the non-dominated vectors as the first PF and the remaining data in the following ranks. As it is expected from the NDS process, maybe there is more than one vector in each rank, so in order to sort them within the same rank, a multi-criteria decision making procedure is needed to utilize .

Input : POP:Population of candidate solutions, NP: Population size
Output: Perato fronts rank of each solution

while $NP \neq 0$ **do**
  for $i \leftarrow 1$ **to** NP **do**
    $n_i = 0$;
    for $j \leftarrow 1$ **to** NP **do**
      `// Calculating the number of`
      `// solutions that dominate`
      `// POP(i)`
      if $POP(j) \prec POP(i)$ **then**
        $n_i = n_i + 1$
      end
    end
    if $n_i = \emptyset$ **then**
      $F_i = F_i \bigcup POP(i)$
    end
  end
  `// Temporarily removing Pareto front`
  `// from population to compute next`
  `// fronts`
  $POP = POP - F_i$;
  $NP = NP - size(F_i)$;
end

Algorithm 1: Pseudo-code for NDS algorithm

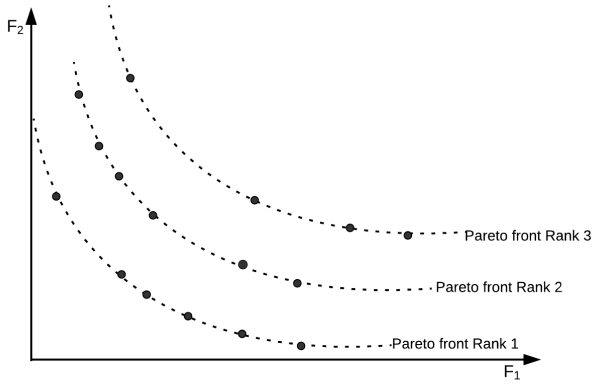


Fig. 3.2: Several fronts of a two-criteria Minimization Problem.

3.2.2 VIKOR: A multi-criteria Decision-making Measure

Multi-criteria decision making (MCDM) is a process to rank and select from a set of candidate solutions with conflicting and non-commensurable criteria [52]. This method can be applied to individuals on the same Pareto rank to determine a ranking list of these individuals. The VIKOR method was introduced as one applicable technique to implement within MCDM. It ranks the multi-criteria individuals based on the particular measure of closeness to the ideal solution. Consider $X = x_1, x_2, x_3, ..., x_N$ as a set of candidates which should be ranked by VIKOR in a minimization MCDM. In order to compute the distance between each candidate, $x(i)$, and an ideal point, the value of j as the objective value, $f_j(x_i)$, should be normalized as follows:

$$f_j(x_i) = \frac{f_j(x_i)}{\sqrt{\sum_{k=1}^{N} f_j^2(x_k)}} \tag{3.2}$$

Each variable for ideal point, f^*, can also be defined as the minimum value of each objective, and j=1,2...,N.

$$f^* = \{f_1^*, f_2^*, ..., f_N^*\} \tag{3.3}$$

Where

$$f_j^* = \min_{i=1}^{N}(f_j(xi)) \tag{3.4}$$

The worst vector is defined as follows:

$$f^- = \{f_1^-, f_2^-, ..., f_N^-\} \tag{3.5}$$

Where

$$f_j^- = \max_{i=1}^{N}(f_j(xi)) \tag{3.6}$$

VIKOR is a weighted sum of two kinds of distance between each point and the ideal point. They are normalized Manhattan (S) and the Chebyshev (R) distances, which are calculated as follows:

$$S(x_i) = \sum_{j=1}^{N} w_j \frac{f_j(x_i) - f_j^*}{f_j^- - f_j^*} \tag{3.7}$$

$$R(x_i) = \max_{j=1}^{N} \frac{f_j(x_i) - f_j^*}{f_j^- - f_j^*} \tag{3.8}$$

Where the weight w_j denotes the significance of the objective function f_j as imposed by the decision maker.

Finally, the VIKOR index for each individual, x_i , is defined as a combination of S and R as follows, where α is determined by the weight of each dissimilarity measure.

$$Q(x_i) = \alpha \frac{S(x_i) - S^*}{S^- - S^*} + (1 - \alpha) \frac{R(x_i) - R^*}{R^- - R^*} \tag{3.9}$$

$$S^* = \min_{i=1}^{N} S(x_i) \quad \text{and} \quad S^- = \max_{i=1}^{N} S(x_i) \tag{3.10}$$

$$R^* = \min_{i=1}^{N} R(x_i) \quad \text{and} \quad R^- = \max_{i=1}^{N} R(x_i) \tag{3.11}$$

Finally, individuals are ranked based on the VIKOR index in ascending order. Less VIKOR indicates shorter distance to an ideal point. It means the corresponding individual has a higher rank.

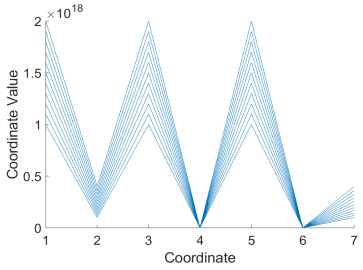
3.2.3 The Proposed Method

In this section, the evaluation metrics, the PCP coordinates ranking, and the utilized datasets are introduced.

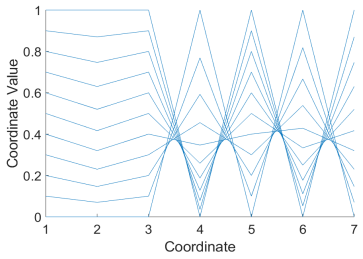
Evaluation Objectives/ Metrics: One of the metrics for getting best visualization for the PCP reduces of the number of crossing lines between adjacent axes by a better ordering the coordinate in the PCP. For this reason, each data point is compared with another one on the same axis with the adjacent axis points based on the following conditions in Eq. 3.12. For instance, if two data points are provided in two neighboring axes with the values (2,3) in the first axis and (3,1) values on adjacent axis, one crossing line between them is calculated. The crossing lines be taken into account only between neighboring dimensions according to [28], so calculating the crossing lines is considered as follows .

$$L = \sum_{d=1}^{D-1} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \begin{cases} 1 & l_{d,i} > l_{d,j} \wedge r_{d+1,i} < r_{d+1,j} \vee \\ & l_{d,i} < l_{d,j} \wedge r_{d+1,i} > r_{d+1,j} \\ 0 & \text{Otherwise} \end{cases} \tag{3.12}$$

This is applied for all possible coordinate permutations, where n is the number of data points, D is the number of dimensions, r and l are adjacent axes. There are some cases of datasets visualization where the crossing lines can be visualized on the coordinates as clutter in the PCP, but they are not actually there; for instance, created benchmark with 7D in Fig. 3.3 (a), when the dataset is not normalized, the crossing lines between neighboring axes cannot be seen. However, if the dataset is normalized, the crossing lines can be visualized clearly in Fig. 3.3 (b), Where is number of crossing lines (L=220) are the same in both figures. The second metric is maximizing the correlation between data points among dimensions.



(a) Non-normalized created benchmark, L=220



(b) Normalized created benchmark, L=220

Fig. 3.3: Comparing the visualization of crossing lines at non-normalized and normalized created benchmark.

Pearson's correlation coefficient is used [53]. It refers to the relationships between data points and dimensions. The interval values between -1,1 indicate the high correlation among data points, but 0 value indicates a weak correlation. First, the correlation coefficient value is determined for the data points between each pair of adjacent axes and then the mean of the absolute of correlation values. Spearman's rank correlation coefficient can be used for non-linear correlation data points, as given in Eq. 3.13.

$$\rho(X,Y) = \frac{\mathrm{cov(X,Y)}}{\sigma_X \sigma_Y} \tag{3.13}$$

Where X and Y are neighboring axes, cov(X,Y) is the covariance between variables, and σ_X and σ_Y are the standard deviation of datasets at each dimension. This is done for all possible permutations. Alternatively, the Pearson's rank correlation coefficient can be utilized for linear correlation data points in Eq. 3.14 :

$$r = \frac{\mathrm{S_{xy}}}{\sqrt{S_{xx}\, S_{yy}}} \tag{3.14}$$

Where

$$S_{xx} = \sum_{i=1}^{n}(x_i - \overline{x}), S_{yy} = \sum_{i=1}^{n}(y_i - \overline{y}) \tag{3.15}$$

Where n is the number of data points, and S_{xy}, S_{yy} and S_{xx} are the average of the data points among axes. Thus, these metrics are used as objectives in the experiments of this method.

The Parallel Coordinate Axes Ranking: The next step for the evaluation is using non-dominated sorting. Then, the solutions are ranked on each Pareto front (PF) rank using VIKOR. This obtains the best and worst order in the PCP in terms of minimization of the crossing lines and maximization of correlation among data points. For a better understanding, the multi-metric PVRPCP method starts with measuring all possible permutations in axes PCP by two metrics for any datasets, crossing lines between neighbouring axes and correlation coefficient among data points. As it is obvious, the PCP with high correlation and fewer crossing lines should be the top-ranked PCP. At first, NDS computes different levels of Pareto ranking based on these two criteria as objectives. As mentioned before, each rank of Pareto includes one or more individuals with the same significance in Pareto's ranking point of view. So, decision-making approaches are needed to sort individuals on the same Pareto. In this experiment, VIKOR as an MCDM measure is utilized to rank the PCP on the same Pareto level. All the data are sorted based on two concepts, Pareto dominance and then VIKOR indicator. Consequently, a new ranking method (PVRPCP) for the PCP is provided, which is explained in this section.

The Utilized Data Visualizations: In this experiment, multiple case studies are conducted to investigate the efficacy of the PVRPCP method. The data in each dataset are normalized before applying the proposed method to find the best ranking of coordinates, however in resulted figures, the exact values of features are presented. Designing benchmarks for evaluating methods is a challenging task in data science because it is based on the users' desired

criteria. In order to evaluate the proposed method, the effect of the order of the PCP is clearly seen. Thus, in the first case study, datasets are designed by the author and referred to as created benchmarks, there are two designed datasets based on the users' desire criteria. In Fig. 3.4, designed datasets based on the utilized metrics are presented. In a seven-dimensional created dataset, the dataset is designed so that the crossing lines number is equal to zero between axes 1 to 3, while axes 3 to 7 should give a high correlation coefficient value. The design PCP benchmarks are intended to present how significantly the order of coordinates affects evaluation measures. The created eight-dimensional dataset in Fig. 3.4 is designed to offer the users' desire in terms of the utilized metrics. A dataset is designed and then the PVRPCP method is applied to obtain a better visualization in terms of both metrics. Therefore, both created benchmarks datasets designed using some equations based on the authors' desire criteria as it explained.

In the second case study, real-world datasets are used. One of them is the Diabetes database [54] containing data about women of Pima Indian heritage who are age 21 years or older, and the other one is the Breast Tissue Dataset [55]. 100 data points are selected from each (1 to 100 points).

In the third case study, multi-objective benchmarks are used of the following benchmark problems: DTLZ5 [56], MaF2, MaF4, MaF10, and MaF13 [57] to visualize the optimal Pareto front. Theses benchmarks have various properties. MaF2 has Pareto-biconcave shape solutions and MaF4 has concave shape and multi-modal PFs, neither giving an optimal solution for any subset. MaF10 has mixed and biased PFs. MaF13 has concave shape, unimodal, non-separable, and degenerate PFs [57].

3.2.4 The Experimental Results

This section presents three conducted studies to obtain the success of the PVRPCP method. They used different kind of datasets. In case study 1, the created Benchmarks datasets are utilized, which are designed based on the desired criteria. In case study 2, some real-world datasets are used. In case study 3, multi-objective functions datasets, which are the results of multi-objective optimization algorithm NSGA-II are employed. Consequently, all of these cases are to obtain the best order of the coordinates in the PCP based on two metrics as objectives, which minimize the crossing and maximize the correlation between adjacent axes. In the multi-objective optimization, the objectives are in conflict and it is a forceful task to

minimize one objective and maximize other objective. That is why one metric maximizes the correlation, and another one minimizes the number of crossing lines for NSD in these experiments.

Case Study 1: Investigation of the Proposed Approach on Created Benchmarks for Visualization: In case study 1, the PVRPCP method is applied on created benchmark datasets. The result of the first dataset, which has seven dimensions, has been demonstrated in Fig. 3.4. At the beginning of the process, NDS is applied to all possible permutations. Then VIKOR is applied to rank the subset solutions on all levels of Pareto fronts in order to rank all possible orders of coordinates. Even if this is done based on the users' desire criteria, it can be better visualized based on the metrics in the PCP. On the one hand, as it is presented in the best rank, the crossing lines number (L) decreases by about a quarter of the original number of crossings, from L= 220 to L= 55, and the correlation is improved as well, from 0.97 to 0.99. This order of coordinates gives better visualization of values of feature values with less clutter and better demonstration of data. On the other hand, the worst rank of visualization, i.e., the last rank solution on the last Pareto front, is demonstrated visually and numerically in Fig. 3.4. In this case, there is an increase in the number of crossing lines, reducing the correlation value.

In the second experiment, whose the results are presented in Fig. 3.4, the created benchmark has eight dimensions. The number of crossing lines among dimensions decreases to about L= 3,000 from the original to the best rank solution. However, the correlation value remains the same because of the conflict between metrics but is still sufficiently good.

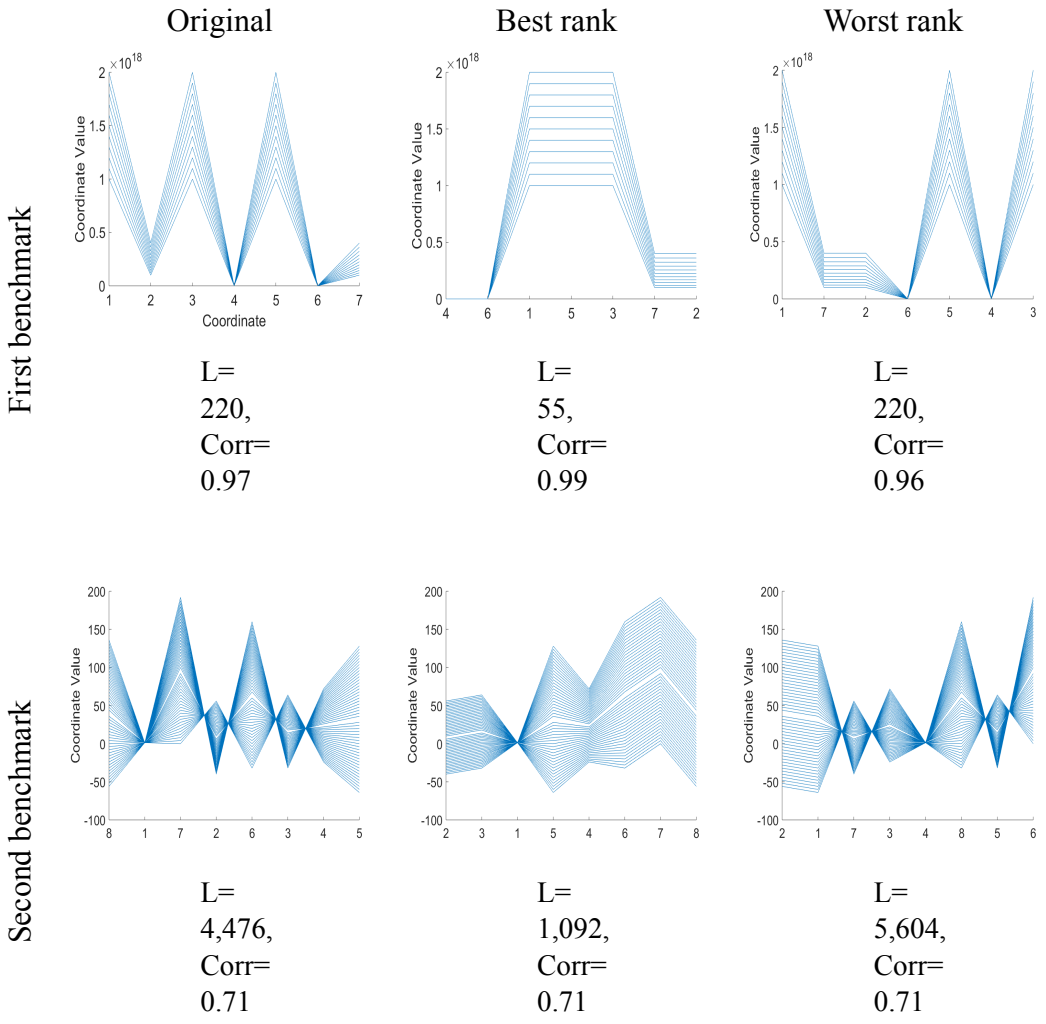


Fig. 3.4: Case Study 1: The Original design, the Best Order based on the VIKOR ranking in the Pareto front, and the Worst ranks in the last Pareto using NDS and VIKOR of the PCP of the Created Benchmarks, where L indicates the number of crossing lines and Corr indicates the correlation value

Case Study 2: Real World Dataset Visualizations: In case study 2, the PVRPCP method is applied on real-world datasets. The Diabetes dataset visualization is in Fig. 3.6. As in Case Study 1, first, the multi-metric PVRPCP is applied to obtain the first rank PFs. This is demonstrated in Fig. 3.5, where the non-dominated solutions are shown and the best solution is indicated by the large hexagon. In this figure, the visualization of best rank is demonstrated with L= 9,985 and the correlation value of 0.34. Reducing the number of crossing lines by

more than L= 2,000 from the original visualization results in reduced clutter and better correlation than in the original and the worst case. In addition, the clutter increased in the last rank. Therefore, the practitioners should choose the criteria that need to see in the dataset visualizations. Thus, it will show different results. This method is flexible in choosing the criteria.

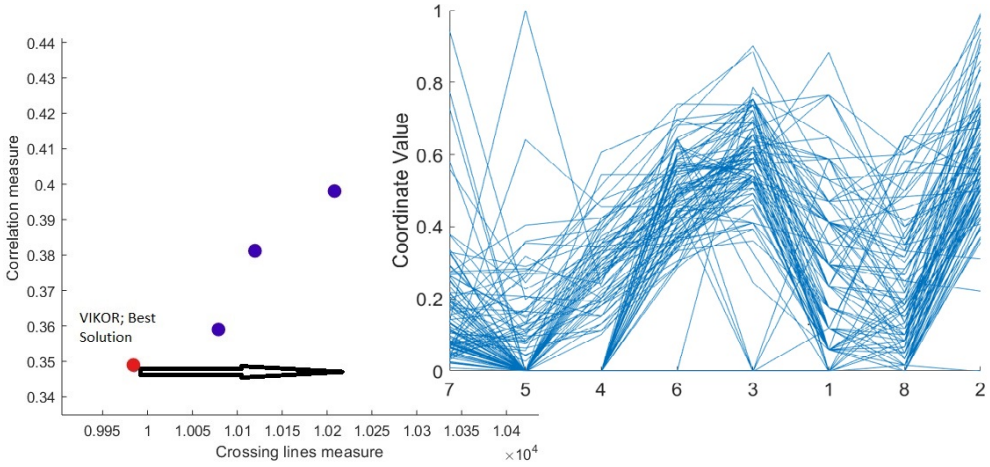


Fig. 3.5: Diabetes dataset first PF rank; The hexagon indicates the top solution selected by VIKOR

In the other real dataset, Breast Tissue data in Fig. 3.6, an improvement is shown in terms of both metrics on the first solution in PFs compared to the original and the worst cases, both numerically and visually. In numerical terms, crossing lines are reduced from L= 11,400 to L=7,341 and the correlation value is raised from 0.56 to 0.70. Visually, the clutter has been reduced in the first rank visualization compared to the other two cases.

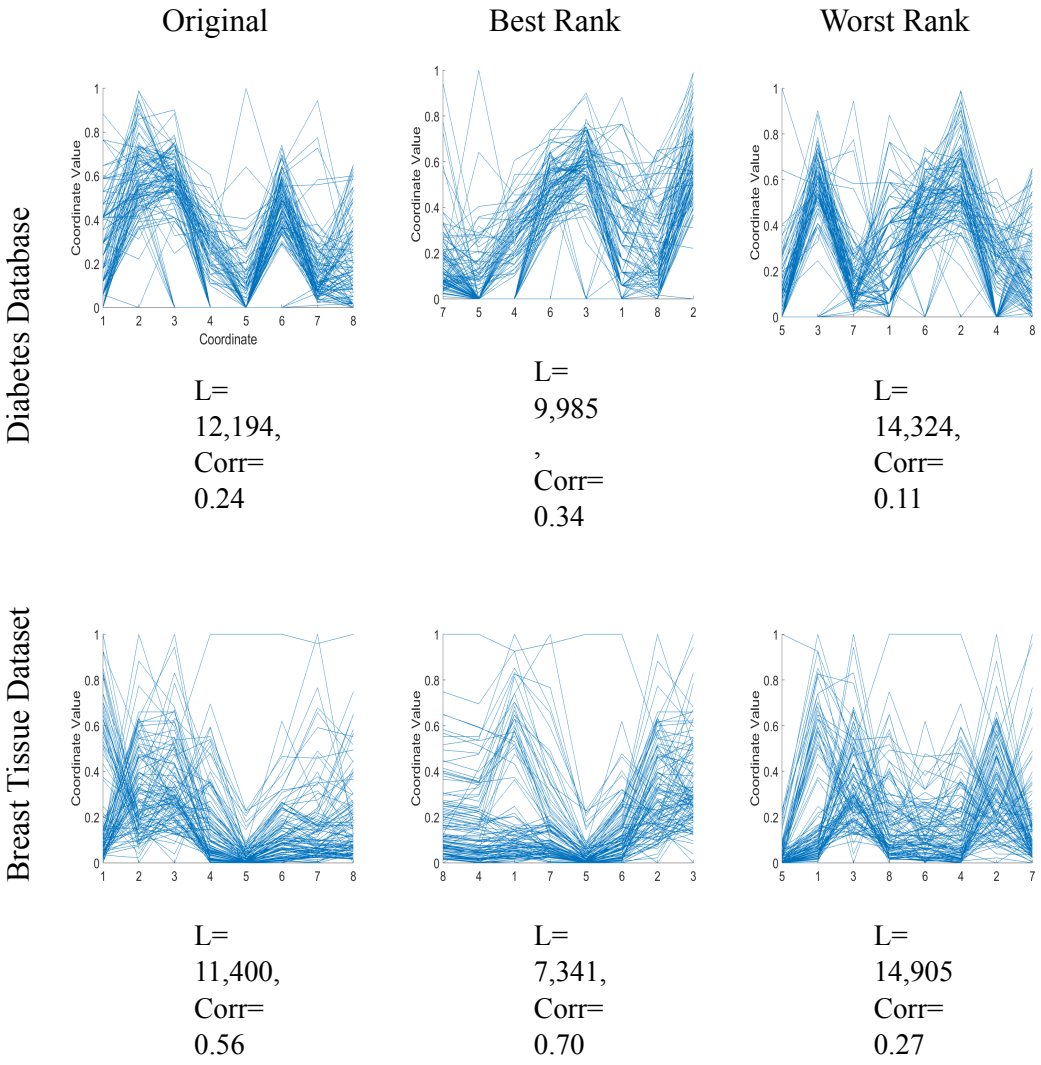


Fig. 3.6: Case Study 2: The Original, the Best, and the Worst Ranks of Parallel Coordinates Visualization of the Real-world Datasets, Namely, Diabetes Database and Breast Tissue Datasets, where L indicates the number of crossing lines and Corr indicates the correlation value

Case Study 3: Multi-Objective Functions: In this case study, the PVRPCP method is applied to several multi-objective benchmark dataset results, which are solved by NSGA-II. It compared the original visualization for all considered benchmarks (DTLZ5, MaF2, MaF4, MaF10, and MaF13) to the best and worst cases based on the same two metrics. Interestingly, in Fig. 3.8 the optimal solution visualization for DTLZ5 is one of the non-dominated solutions in the first PF. This means that the optimal solution could be one of the designed multi-objective benchmarks. In contrast, in the last rank, the number of crossing lines is twice

of those in the original, while the value of correlation is somewhat lower when compared to the original and the best PF visualizations.

In MaF10 in Fig. 3.8 visualization, crossing lines are minimized in the first point on PF by approximately L=2,300, providing clear improvement in visualization between the 2nd and the 4th axis, and the correlation value increased a little as well. However, in the last visualization of rank, there is a cluttered area at the bottom of all axes, which means the number of crossing lines increased by around L= 1,000 from the original PCP.

Furthermore, in MaF13 in Fig. 3.8, the difference between the best order, the original and the worst one can be clearly seen. In the best rank, from the 2nd to the 5th axis there are not many crossing lines, and although the correlation decreased, it did not decrease significantly. In another visualization MaF2 in Fig. 3.8, as a consequence of the conflicting objectives, the number of crossing lines reduced numerically. However, there was not much visual differences. On the other hand, in the last point rank, the clutter demonstrates the increase in the number of crossing lines.

Similar to MaF10, in MaF4 in Fig. 3.8, crossing lines in the areas between the 1st and the 3rd, and the 4th and the 6th axis are fall off in the first rank. However, in the last rank visualization, the clutter is high in the area between the 2nd and the 4th axis, while the correlation values remain the same since the metrics are in conflict. Therefore, in multi-objective optimization, the objectives are in conflict. That explains why it is hard to decrease number of crossing lines and increase correlation.

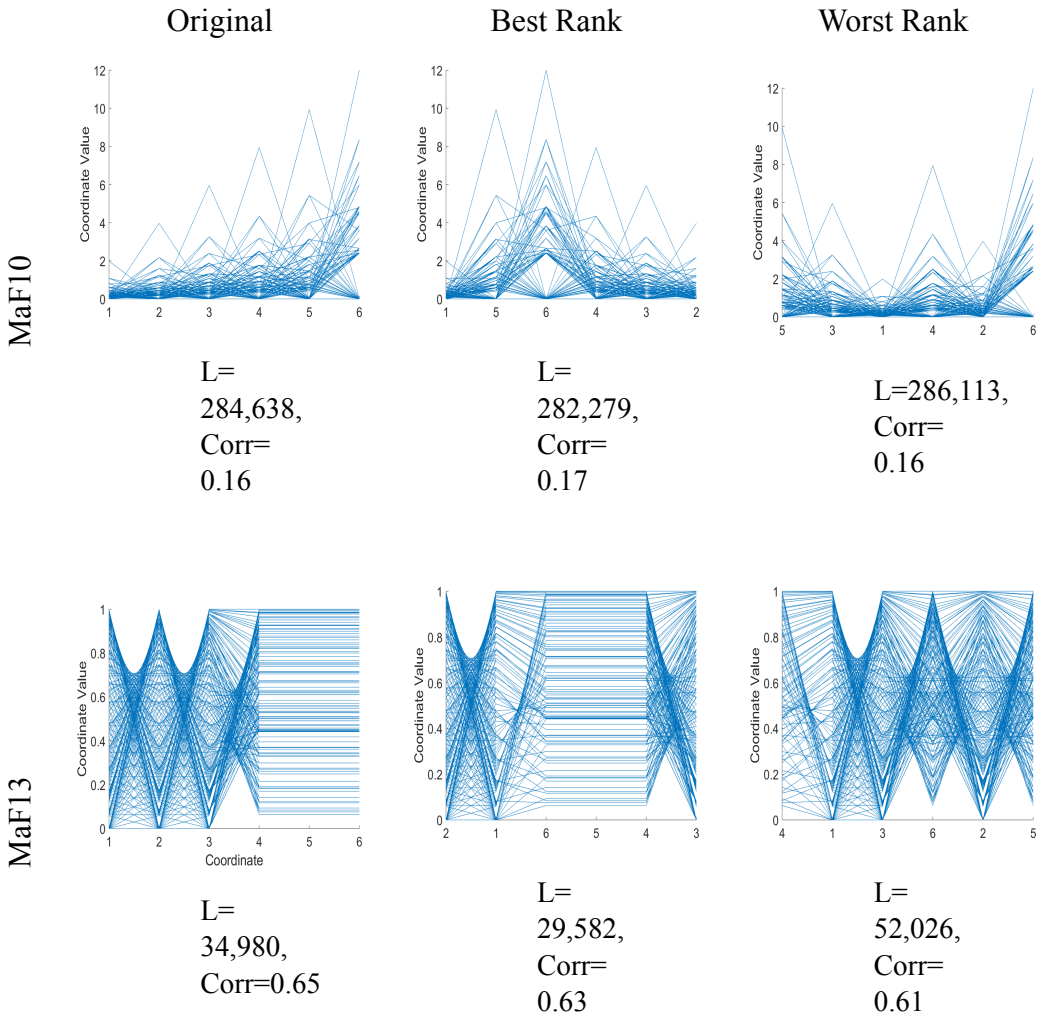


Fig. 3.7: Case Study 3: The Original, the Best, and the Worst Ranks of Parallel Coordinates Visualization of Multi-objective Optimization Benchmarks, Namely, MaF10, and MaF13, where L indicates the number of crossing lines and Corr indicates the correlation value.

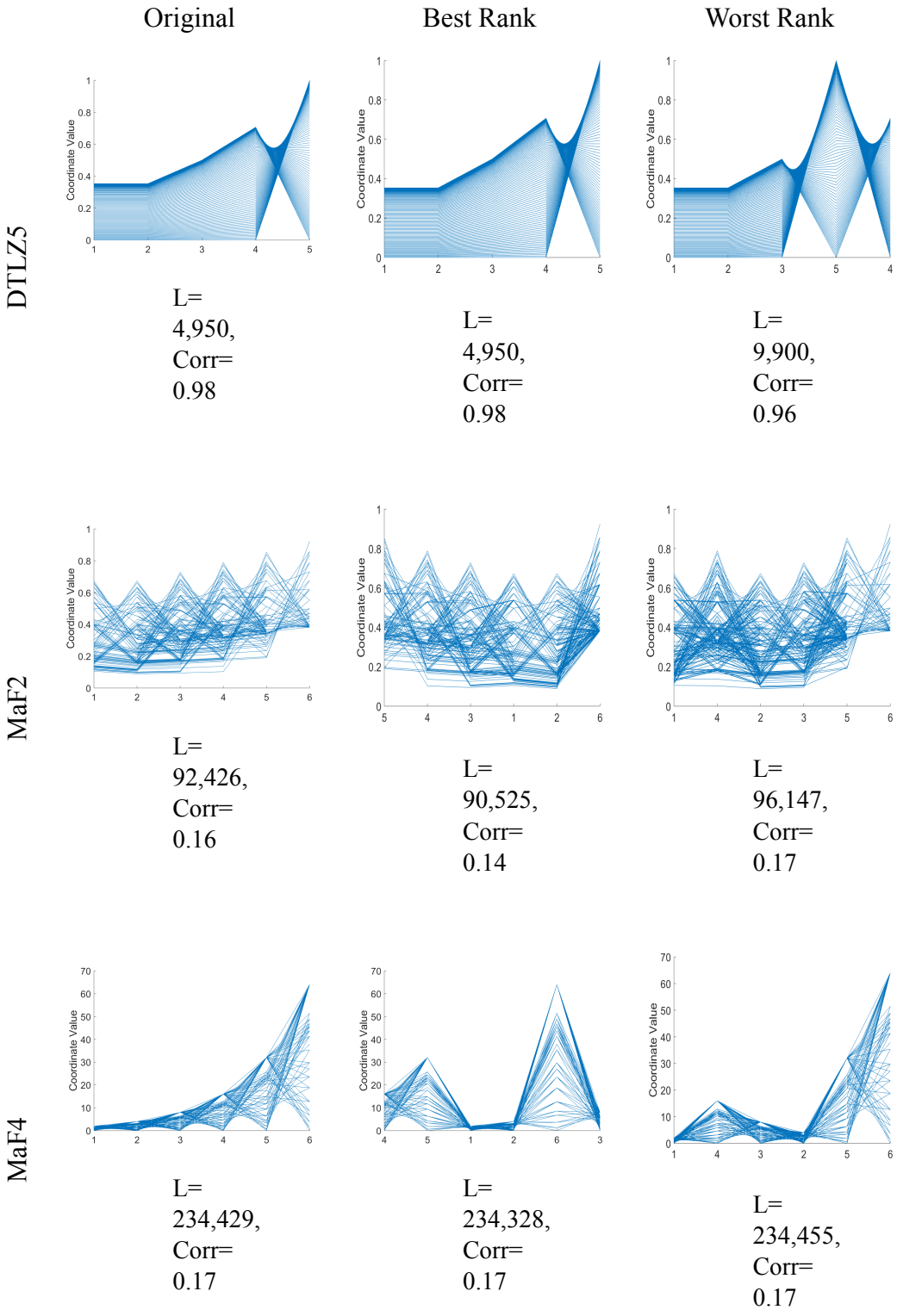


Fig. 3.8: Case Study 3: the Original, the Best, and the Worst Ranks of Parallel Coordinates visualization of multi-objective optimization benchmarks, Namely, DTLZ5, MaF2, MaF4

3.3 Enhancing Parallel Coordinates Visualization Using Genetic Algorithm with Smart Mutation

The PCP has problems visualizing large-scale datasets, such as high clutters produced from numerous intersection lines between neighboring axes. Many researchers have conducted techniques to refine the PCP. For instance, reordering adjacent axes in the PCP technique is a useful procedure to reduce the clutter. Therefore, finding the optimal coordinate order acquisition can be classified as a combinatorial optimization problem. However, in high-dimensional datasets, the optimization algorithms may face difficulty in dealing with this issue. In this section, the smart mutation operator sets out to enhance the performance of the Genetic Algorithm (GA) by finding the optimal order of axes in the PCP based on reducing the numerous intersection lines with a brief GA description. However, the user has the choice to select any objective function. The Monte Carlo simulation is utilized to compare and test the smart mutation with the other three schemes to be successfully used as a new mutation scheme to improve the performance of GA. Moreover, several experiments are used to find an optimal coordinate order in PCP to visualize the datasets with various numbers of samples and dimensions. Furthermore, this section explains GA for combinatorial optimization algorithms.

3.3.1 GA

GA is one of the well-known evolutionary algorithm proposed by Holland in 1992 [58]; GA is a commonly utilized method for solving many optimization problems. In addition, GA is an appropriate method to solve the combinatorial optimization problem as an NP-hard problem. For instance, it is used to solve the traveling salesman problem (TSP) [59]. Re-arranging the axes in PCP to enhance visualization is also a combinatorial optimization problem that can be solved by GA. A combinatorial GA works with a set of individuals, namely population, which are a group of different orders of variables. For instance, in a 6-dimensional dataset, 6! possible orders of coordinates are presented in the PCP from which the population are randomly selected. During the optimization process, the new individuals, called offsprings, are generated using crossover and mutation operators. Using the crossover operators, the two genes are selected based on the high fitness value of parents. In addition, the mutation operator

is applied to some genes of new offspring to obtain the genetic diversity of the population. Finally, the fitness values are calculated for the population members, so that GA chooses the top individuals for the following generation. The process of producing new generations continues until a termination criterion such as a predefined numerous of iteration or acceptable fitness error is met.

3.3.2 The Proposed Method

This section explains the utilized evaluation metric to enhance the ordering of the PCP axes. In addition, the detail of the GA with proposed smart mutation operator is provided.

Evaluation Metric as Objective Function Since the PCP suffers from clutter as the main problem, the optimal order of the coordinates in the PCP is an effort that can enhance the PCP by reducing the intersection lines between neighboring coordinates. As in the previous method 3.12, the crossing lines are calculated among each pair of points on neighboring axes, so the number of crossing lines is determined in the PCP visualization is considered as the minimization objective for GA.

GA with a Smart Mutation: In this study, GA is applied to improve the PCP using a new mutation scheme called the smart mutation, which has a better performance in optimizing the order of the coordinates. At the beginning of the algorithm, the population is initialized by uniform randomly selected over are different axes orders in the PCP. For instance, if the number of dimensions is 15, 15! possible permutations are made for the order of axes, so only a subset of them is chosen to provide the initial population of GA. Then at each step, the crossover operator is applied on the whole population to generate new individuals (crossover point called offspring). Next, the only smart mutation is applied by swapping two axes that have the worst pairwise fitness values instead of doing a random mutation, i.e., the highest number of crossing lines. This mutation leads to obtain better-ordered coordinates that have better fitness values. If a random swap is applied to the coordinates that maybe have good fitness values, it might degrade the qualified order. While by considering decomposability property of this problem, the two genes are selected intelligently to be mutated. In this study, the objective metric is to decrease the crossing between neighboring coordinates in the PCP, thus the two axes that have the maximum crossing lines numbers are selected for swapping. The objective value is calculated by the sum of the crossing line numbers among pairwise axes, thus this is a decomposable function which is a combination of multiple sub-functions.

Therefore, the algorithm is capable to change the orders in such way that minimizes the sub-functions. An example explains assigning the number of crossing lines to each axis. If a seven-dimensional dataset is provided, the two edge axes fitness values are the sum of two values of intersecting lines; the first edge value is between the first and second axes, and the last axis is between the two last axes (6th and 7th axes). For a middle axis, the fitness value is also the sum of two values; the first value is between the corresponding axis and the left neighbor, and the second value is between the corresponding axis and the right neighbor axis, such as (2,3) and (3,4) axes. The pause-code is in algorithm 2.

Input : POP:Population of candidate solutions, $PopSize$: Population size, Pop is the population, $MaxIter$ is the number of generation,
Output: $OffSp$: is offspring, $FV1$is the fitness value for current OffSp, $FV2$ is the fitness value for current parent
Initialization the POP randomly with PopSize
for $g \leftarrow 1$ **to** $MaxIter$ **do**
  for $i \leftarrow 1$ **to** $PopSize$ **do**
    each individual in the Pop i do evaluate f_i
    for current population (FV2)
  end
  for $pc \leftarrow 1$ **to** $(PopSize/2)$ **do**
    Apply crossover
  end
  for $pcs \leftarrow 1$ **to** $(PopSize)$ **do**
    Do only the smart mutation by selecting the two axes have the worst FV1, swap and calculate the $FV1$ **for** $OffSp$
    if *(*$FV2$*)* $<=$ *(*$FV1$*)* **then**
      Pop_i=Pop_i
    else
      Pop_i= $Offsp_i$
    end
  end
  Select the best fitness value
end

Algorithm 2: Pseudo-code of Genetic Algorithm with a smart mutation

3.3.3 The Experimental Results

This section introduces a brief description of the visualized datasets. In addition, a Monte Carlo simulation is conducted on a smart mutation and different mutation schemes to investigate and test the intelligent mutation's effectiveness to be used confidently in order to

improve the fitness value in GA. Also, the consequences of GA with the intelligent mutation is presented.

Visualized Datasets: The datasets are different from the previous method because this method uses for high-dimensional-dataset visualizations. Nine commonly used datasets are employed with various number of dimensions as shown in the Table 3.1 to evaluate the performance of the proposed method. The 1985 Auto Imports Database and Cardiac Arrhythmia datasets are from the UCI machine learning repository [13]. The Red and White dataset is from [60]. In addition, Birds, Breast Cancer, and Emotion datasets are taken from [61].

Dataset	Number of Dimensions	Number of Samples
Birds Train Test	160	322
Breast Cancer	30	569
Red Wine	12	1599
White Wine	12	4898
1985 Imports	15	205
1985 Imports	26	205
Cardiac Arrhythmia	50	279
Cardiac Arrhythmia	279	279
Emotion Train	78	782

Table 3.1 Details of Visualized Datasets Used as Benchmarking Datasets.

Monte Carlo Simulation on Smart Mutation: The Monte Carlo Simulation is conducted to estimate unknown solution by using the principles of inferential statistics. The main contribution of simulating the smart mutation by Monte Carlo is to investigate its success before using it in enhancing the performance of GA and comparing it with different schemes. In order to simulate the application of smart mutation and comparing it with other schemes, a repetitive process in a predefined number of iterations is performed. At each iteration, a random order of axis is generated for a dataset, then the fitness values of the generated order are computed before and after applying the smart mutation. Finally, if there is an improvement in fitness value when the smart mutation is applied, it counts as success. In order to

monitor the success rate of improvement, the number of successes is divided by the max iteration number. As mentioned previously, the metric in our experiment is used to minimize the number of crossing lines between adjacent coordinates, so the worst fitness values are the two axes that have the highest number crossing lines between neighboring axes. Furthermore, different schemes are compared using the Mont Carlo simulation to evaluate the proposed mutation. The first scheme is the proposed smart mutation, which was explained in the previous section. The second scheme of mutation swaps the axis that has the maximum crossing lines with a random chosen coordinate and is called a max-random swapping mutation. The third mutation scheme is to swap two end coordinates. In the last scheme, called a rand-rand swapping mutation, two randomly selected axes are swapped. Table 3.2 represents the mean, standard deviation, the minimum, and the ratio of improvement (RI) of the fitness values for all schemes. As shown in the Table 3.2, the RI of the swapping based on the smart mutation has a higher percentage in comparison to the other three mutation schemes. In addition, the second scheme (swapping one axis that has the maximum fitness value with a randomly selected axis) has the second rank because one of the swapped axes has the worst fitness value, i.e., partially smart. As it is expected, swapping axes randomly demonstrates the worst RI values.

Max-Max Swapping Mutation					
Dataset	D	Mean	Std	Min	RI (%)
Birds Train Test	160	1.05E+07	3.52E+05	8.92E+06	**80.35%**
Breast Cancer	30	1.63E+06	8.65E+04	1.27E+06	**76.79%**
White Wine	12	5.71E+07	3.10E+06	4.73E+07	**70.440**%
1985 Imports	15	1.16E+05	1.27E+04	7.68E+04	**75.929 %**
1985 Imports	26	1.35E+05	1260076845	9.36E+04	**76.81%**
Arrhythmia	50	5.09E+05	6.92E+04	2.57E+05	**73.95%**
Arrhythmia	279	3.31E+06	1.72E+05	2.62E+06	**91.29%**
Emotion	78	9.54E+06	2.54E+05	8.61E+06	**81.98 %**

Max-Rand Swapping Mutation					
Dataset	D	Mean	Std	Min	RI(%)
Birds Train Test	160	1.05E+07	3.38E+05	9.14E+06	78.560%
Breast Cancer	30	1.66E+06	8.25E+04	1.29E+06	62.74%
White Wine	12	5.81E+07	3.1385e+06	4.80E+07	55.95
1985 Imports	15	1.21E+05	1.12E+03	7.97E+04	63.10%
1985 Imports	26	1.39E+05	1.17E+04	9.86E+04	65.85%
Arrhythmia	50	5.06E+05	6.92E+04	2.42E+05	70.97%
Arrhythmia	279	3.32E+06	1.73E+05	2.61E+06	82.8%
Emotion	78	9.59E+06	2.46E+05	8.67E+06	67.82%

Edge-Edge Swapping Mutation					
Dataset	D	Mean	Std	Min	RI (%)
Birds Train Test	160	1.05E+07	3.36E+05	9.19E+06	50.54%
Breast Cancer	30	1.67E+06	8.33E+04	1.33E+06	49.74%
White Wine	12	5.86E+07	3.23E+06	4.88E+07	50.49%
1985 Imports	15	1.24E+05	1.17E+04	7.68E+04	49.25 %
1985 Imports	26	1.42E+05	1.20E+04	9.81E+04	49.22%
Arrhythmia	50	5.21E+05	7.03E+04	2.60E+05	49.67%
Arrhythmia	279	3.35E+06	1.73E+05	2.63E+06	48.71%
Emotion	78	9.64E+06	2.44E+05	8.71E+06	49.71%

Rand-Rand Swapping Mutation					
Dataset	D	Mean	Std	Min	RI(%)
Birds Train Test	160	1.05E+07	3.36E+05	9.09E+06	50.12%
Breast Cancer	30	1.67E+06	8.33E+04	1.32E+06	50.96%
White Wine	12	5.85E+07	3.26E+06	4.82E+07	50.7 %
1985 Imports	15	1.24E+05	1.17E+04	8.32E+04	49.37%
1985 Imports	26	1.42E+05	1.42E+05	9.82E+04	50.330%
Arrhythmia	50	5.21E+05	6.96E+04	2.57E+05	49.48 %
Arrhythmia	279	3.35E+06	1.73E+05	2.62E+06	49.6%
Emotion	78	9.64E+06	2.45E+05	8.68E+06	50.01%

Table 3.2 Monte Carlo Simulation Results for investigating the smart mutation (Max-Max Swapping Mutation) in Comparison to Three Different Mutations; Max-Rand, Edge-Edge, and Rand-Rand Swapping Mutations

The Results of GA with Smart Mutation: In order to evaluate the smart mutation applied in GA, several datasets are visualized using the proposed scheme. Since GA is a stochastic algorithm, it is run 31 times independently, then the min, max, standard deviation, and the mean of the best fitness values resulted in all runs are provided. In table 3.3, the results of various schemes are presented.

The first applied scheme (Scheme 1) is random mutations by giving different probabilities to three strategies: 1) exchanging two random genes, 2) flipping a random gene, 3) exchanging two pieces of the chromosome. The probabilities of applying the three strategies on the population are set to 0.1, 0.2, and 0.8, respectively. In the second scheme (Scheme 2), the smart mutation is used with a probability of 0.1 to apply on the population while the first two strategies of Scheme 1 are applied on the remaining of population with probabilities of 0.2

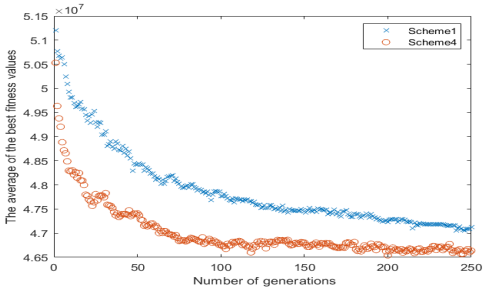
and 0.8, respectively. In the third scheme (Scheme 3), the smart mutation is applied for the whole population, and at the same time, random mutations are provide with the chances of 0.2 and 0.8 as same as in Scheme 1 to be applied. In the last scheme (Scheme 4), the smart mutation is employed for the entire population.

The experimental results show more significant improvement in terms of decreasing the crossing lines between neighboring coordinates for Scheme 4 in comparison to other schemes so that the mean of the number of crossing lines in Scheme 4 is lower than in other schemes. Therefore, for the birds dataset, which is a large-scale dataset with 260 dimensions, as it is showed, the mean of crossing lines of Scheme 4 is lower than other schemes. It drops from 8.67E+06 in Scheme 1 to 7.70E+06 . This means approximately a million crossing lines decreased. In the second dataset, the performance on Scheme 4 is the same. Moreover, it can also be seen also that the minimum and the maximum number of crossing lines decrease in using Scheme 4 compared to other schemes. From the third dataset, the mean crossing lines has a diminished value using Scheme 4 compared to the other mutation schemes. In the fourth experimental dataset, the mean of crossing is reduced by close to 1.00E+06, and the std is lower than in other schemes. The improvement is also significantly observed from other datasets too, such as Arrhythmia, a large-scale dataset with 279 dimensions. Consequently, the results indicate better enhancement using Scheme 4 to find an optimal order for the PCP. On the one hand Scheme 1 results in the worst performance which using the random mutations in GA. In order to demonstrate the efficiency of the proposed method, the performance plot of the smart mutation in comparison to Scheme 1 is represented in Fig. 3.9 in which the mean value of fitness during different generations are plotted. As it is indicated, the mean of crossing lines in Scheme 4 has decreased remarkably in comparison to Scheme 1 so that it converges into a better candidate solution and consequently better visualization. Thus, [62] the Wilcoxon rank sum test which is a non-parametric alternative between two sets indicates the winner (minimum value of the mean values) between them. This test is conducted between the Scheme 1 and Scheme 4, as indicated in the last row in Table.3.3. $(w/t/l)$ are the winner, tie, and loser. 9 out of 9 datasets are the winner (w) of the for Scheme 4. Moreover, two datasets are visualized in Fig. 3.10 to compare the schemes visually and numerically simultaneously. As it can be seen for both datasets, the Auto imports dataset with 15 dimensions and the Arrhythmia dataset with 20 dimensions,the result is a number of diminished crossing lines in

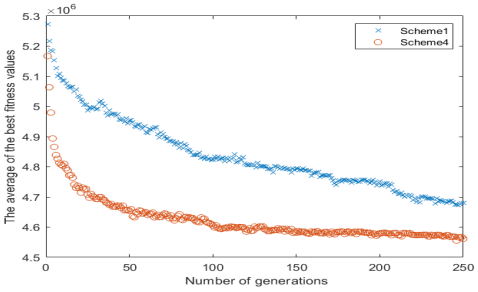
the order of axes in Scheme 4 in comparison to the original order (taken from the source of dataset) and order by Scheme 1. For instance,in the Auto imports dataset, the crossing lines are minimized by around 5.00E+03. In addition, it is indicated visually that as a result, there is less clutter in Scheme 4, especially between the last coordinates from 11th to 15th.

Dataset	D	Measures	Scheme 1	Scheme 2	Scheme 3	Scheme 4
Birds Train Test	160	Min	8.28E+06	7.60E+06	8.40E+06	**7.05E+06**
		Max	1.16E+07	1.13E+07	1.14E+07	**9.73E+06**
		Std	2.11E+05	2.30E+05	1.69E+05	**2.62E+05**
		Mean	8.67E+06	8.04E+06	8.71E+06	**7.70E+06**
Breast Cancer	30	Min	5.13E+06	1.07E+06	1.13E+06	**1.06E+06**
		Max	8.57E+06	1.86E+06	1.94E+06	**1.78E+06**
		Std	1.45E+05	4.12E+04	5.06E+04	**4.64E+04**
		Mean	5.47E+06	1.18E+06	1.28E+06	**1.14E+06**
Red Wine	12	Min	4.33E+06	4.34E+06	4.46E+06	**4.38E+06**
		Max	7.43E+06	7.24E+06	7.49E+06	**6.75E+06**
		Std	2.00E+05	1.40E+05	1.64E+05	**1.E+05**
		Mean	4.74E+06	4.61E+06	4.73E+06	**4.57E+06**
White Wine	12	Min	4.55E+07	4.51E+07	4.56E+07	**4.51E+07**
		Max	6.73E+07	6.71E+07	6.79E+07	**6.27E+07**
		Std	1.45E+06	1.40E+06	1.11E+06	**1.33E+06**
		Mean	4.73E+07	4.73E+07	4.73E+07	**4.70E+07**
1985 import	15	Min	4.37E+04	4.45E+04	4.56E+04	**4.10E+04**
		Max	1.03E+05	1.10E+06	1.16E+05	**9.88E+04**
		Std	4.17E+03	4.22E+03	4.47E+03	**1.67E+03**
		Mean	5.42E+04	5.34E+04	5.58E+04	**4.46E+04**
1985 mport	26	Min	4.95E+04	5.36E+04	4.57E+04	**4.43E+04**
		Max	1.24E+05	1.18E+05	1.15E+05	**1.11E+05**
		Std	5.40E+03	5.26E+03	3.13E+03	**3.83E+03**
		Mean	6.23E+04	6.27E+04	5.47E+04	**5.08E+04**
Arrhy-thmia	50	Min	5.43E+03	5.62E+04	3.62E+03	**4.52E+03**
		Max	2.21E+04	1.25E+05	2.01E+04	**2.03E+04**
		Std	1.15E+03	3.52E+03	1.07E+03	**7.81E+02**
		Mean	8.07E+03	6.29E+04	6.80E+03	**5.70E+03**
Arrhy-thmia	279	Min	6.72E+04	6.78E+04	6.48E+04	**6.46E+04**
		Max	1.04E+05	1.07E+05	1.01E+05	**9.84E+04**
		Std	2.92E+03	2.44E+03	1.01E+05	**3.98E+03**
		Mean	7.17E+04	7.26E+04	6.91E+04	**7.24E+04**
Emo-tion Train	78	Min	8.04E+06	7.58E+06	8.08E+06	**7.33E+06**
		Max	1.04E+07	1.01E+07	1.04E+07	**9.38E+06**
		Std	1.39E+05	1.54E+05	1.46E+05	**1.49E+05**
		Mean	8.36E+06	7.89E+06	8.32E+06	**7.75E+06**
(w/t/l)						(9/0/0)

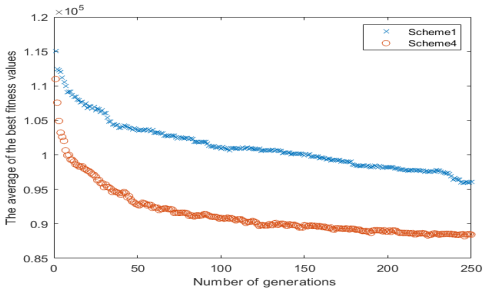
Table 3.3 GA with Smart Mutation in Comparison with Three Schemes, Namely, Scheme 1, Scheme 2, Scheme 3, and Scheme 4



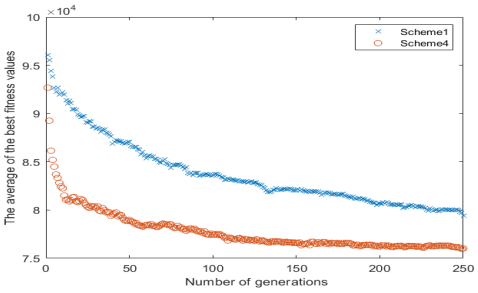
White Wine Dataset_D=12



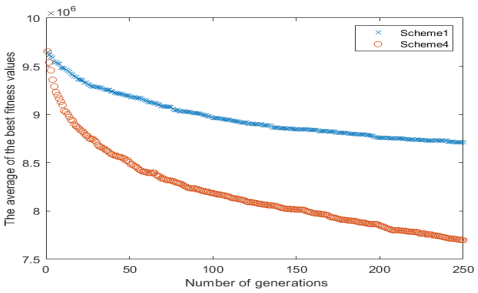
Red Wine Dataset_D=12



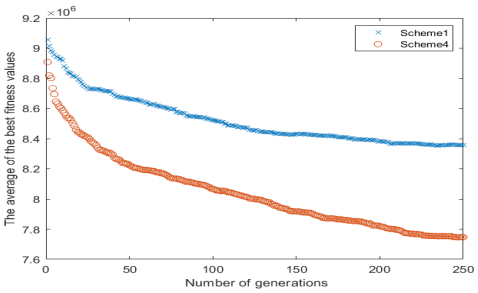
Imports_85 Dataset D=26



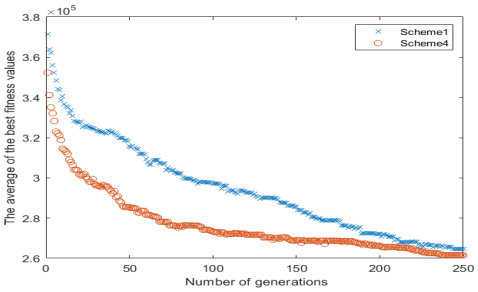
Imports_85 Dataset D=15



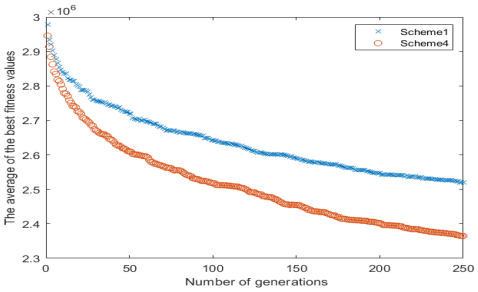
Birds Dataset D=160



Emotion Train Dataset_D=78



Arrhythmia Dataset D=50



Arrhythmia Dataset D=278

Fig. 3.9: Performance plots of GA with smart mutation in comparison with the random mutation.

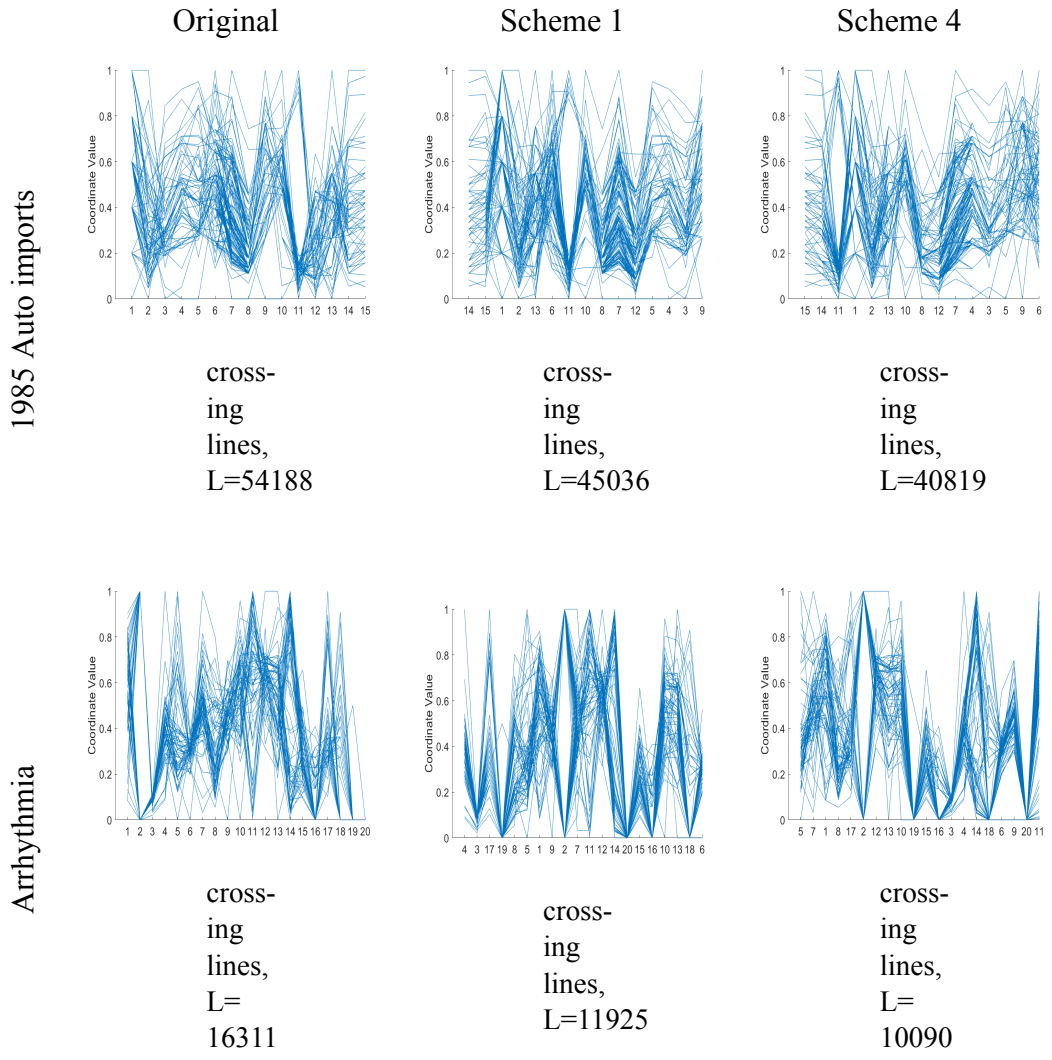


Fig. 3.10: The PCP Visualization Using the Original Order of the Axes, the order resulted by Scheme 1, and Scheme 4 for Two Different Datasets.

3.4 Enhancing the PCP Using Multi-objective Evolutionary Algorithms

The high-dimensional-dataset visualizations require useful visualization tools to help reviewers understand and analyze the data. The PCP is a powerful tool, but it has some drawbacks, such as clutter and many intersecting lines between its axes. While the request of having more than one objective in any problem has increased, the PCP has been refined to have more than one criterion using optimization algorithms below. In this part of the thesis,

the NSGA-II is applied using two measurements to improve reordering-of-the-coordinates-technique for high-dimensional-dataset visualizations. Moreover, the NSGA-III i s utilized to have the optimal order of the axes in the PCP based on two criteria. The used measures minimize the numbers of crossing lines and maximize the correlation between their axes that were used in previous methods. Both performance methods are compared using well-known datasets. Moreover, given that human perception is the main crucial decider in what is better visualization than optimization algorithms solution, several experiment results of NSGA-III were shown to have a fair comparison between both solutions.

3.4.1 NSGA-II

NSGA was invented in 1994 [63] , then NSGA-II in 2001 [64] and it developed to the fast elastic algorithm in 2002 [65]. NSGA-II is one of the well-known multi-objective evolutionary algorithms. Therefore, it provides multi-Pareto front solutions for multi-objective problems, such as NP-hard problems by using NDS and crowding distance to find out the different levels of PF solutions. As the NSD is explained in the previous section in algorithm 1, it sorts the set of vectors into ordered subsets dependent on the Pareto dominance approach. Diversity preservation is required, so crowded tournament selection operates to create diversity between non-dominated solutions and spread the solutions. For a deeper understanding, NSGA-II is based on generating a random population of P_t, which depends on the range and constraints of the problem and is sorted based on the NSD concept. Next, the offspring is generated by using tournament selection, crossover, and mutation. Then, the current population and the offspring Q_t are integrated. In the next step, the combination of parent and offspring solutions $R_t = P_t \cup Q_t$ is sorted based on NDS-concept in 1 and selecting the best individuals. After this, applying the crowd distancing in the beginning, the distances of all individual solutions are initialized to zero. After that, all individuals are sorted in the population to each objective function in ascending order using $i \prec_n$. Thirdly, then, the infinite distance value is assigned as the boundary of the solutions. Finally, the individual distance values with corresponding objective values are added up, as shown in algorithm 4 based on this Eq. Besides, each objective value has to be normalized before applying the crowding distance.

$$S(i).dist = S(i).dist + (S(i+1), m) - S(i-1, m))/(S_m^{max} - S_m^{min}) \tag{3.16}$$

Where $S(i).dist$ refers to the m objective values of i in the set S, (S_m^{max} and S_m^{min}) are the minimum and maximum of m objective values from the set i. Furthermore, the $P_t + 1$ of

Input : S=Non-dominated set, M: Number of objective ,$N_S = |S|$: Number of non-dominated solutions, $S(i).dist = 0$ where is $i = 1, 2.....N_s$: initialize the distance
Output:

for $m \leftarrow 1$ **to** M **do**
  $S = sort(S, m)$ `// sorting non-dominated solution according to each objective`
  $S(1).dist = 0 = S(N_s).dist = \infty$ `// Assign to the corner points to the infinity`
  for $i \leftarrow 2$ **to** $N_s - 1$ **do**
    $S(i).dist = S(i).dist + (S(i+1), m) - S(i-1, m))/(S_m^{max} - S_m^{min})$ `// The boundary of points is selected from other points`
  end
end

Algorithm 3: Pseudo-code for Crowding Distance Algorithm

Pop size is selected by sorting the solutions in descending order using crowded comparison operator. The crowded-comparison operator $i \prec_n$ controls the selection part in the algorithm. This is made by assigning there two contributes for each individual i in the population, the non-domination rank $i(rank)$ and crowding distance $i(distance)$. The choice of having optimal PF can be defined as partial order as follows.

$i \prec_n j$

if $i(rank) < j(rank)$

or $i(rank) = j(rank)$

and $i(distance) > j(distance)$

This selection procedure employs between every two different solutions of non-dominated ranks. The solution with lower value is the better rank. When the two solutions are in the same rank, they assign to be in a lesser crowded region. Consequently, the point of highest distance value at the same level in PF is the best solution. In the end, the new population Pt + 1 is utilized to generate the next offspring of Qt + 1 by applying tournament selection, recombination, and mutation.

Input : P: Initialization a random uniform distribution of the population , Pop_Size: Population size. $MaxIter$: number of generation
Output: Q: Offspring

Initialization the population
for $i \leftarrow 1$ **to** Pop_Size **do**
| Evaluating the fitness values and constraints for each objective
end
Do ranking (level) based Pareto dominate concept
Generating offspring based on tournament selection, recombination, and mutation
for $t \leftarrow 1$ **to** $MaxIter$ **do**
| $R_t = P_t \cup Q_t$ `// Combine parent and offspring population`
| S = Fast non-dominated-sort (R_t)`// all fronts for Combination of parent and offspring solutions`
| $i = 1$, $P_t + 1 = \phi$
| **until** $R_(t+1) = P_(t+1) + |S_i| <= N$ `// until the parent population is filled`
| crowding-distance-assignment for (S_i) $R_t = P_t \cup S_i$ `// Including non dominated solutions in the parent`
| $i = i + 1$, `// Check the next front`
| Sort ($S_i \prec n$) `// Sort in descending order using` $\prec n$
| $P_t + 1 = P_t + 1 \cup S_i$ [1:(N-(|P_t+1|)]
| $Q_t + 1$ = make new population $P_t + 1$ based on tournament selection, recombination, and mutation
end

Algorithm 4: Pseudo-code for NSGA-II

3.4.2 NSGA-III

NSGA-III is used to solve many-objective optimization problems. The NSGA-III strategy is similar to the NSGA-II procedure [66] [67]. The only difference is in the selection procedure, replaced by the crowding distance operator in NSGA-II compared to some approaches in NSGA-III. The first approach includes the classification of the population into non-dominated solutions into different levels. It used the MCDM, which is explained in previous sections [68]. All non-dominated solution front levels from level 1 to level l from the population are included in S_t. If the $|S_t|= N$, no more operations are needed, and $P_t + 1 = S_t$. However, if $|S_t| > N$, all fronts from level 1 to the level $(l - 1)$ are chosen. $P_t = \cup_{i=1}^{l-1} F_i$ and the rest of them are $K = N - (P_t + 1)$ population members of selecting the last front level l. NSGA-III uses this one as the reference points on the hyper-plane concept, which is used to obtain diversity in produced solutions. They can be predefined by the user or structures. However, the reference point can be changed. As an example of structured reference points, in [69], Das et al. use a systematic approach, which replaces the reference points on the hyperplane. Each objective value is adaptively normalized depending on the member of S_t. Then each member in the population is joined with a reference point and a reference line on the hyper-plane and connected to the reference point with the origin. Therefore, all population members of S_t with the reference points with their reference lines are the nearest to the member of the population in the normalized objective space. Then, the members of the population $P_t + 1 = S_t$ St is counted and associated with a reference point. If there is no member associated with the reference point, calculating the smallest perpendicular space between each member is needed, and $P_t + 1$ is added to each reference. Therefore, all the members in the population that joined many reference points are randomly chosen from F_t and added to $P_t + 1$. The Niche procedure counts the population members associated with many reference points or without reference points by using $P_t + 1 = S_t / F_l$. After that, it repeats the stages until the desired population size is reached. The algorithms' procedures are in algorithm 5. Associate each member **s** of S_t with a reference point:

3.5 The Proposed Method

This method has a similar aim to the PVRPCP method, namely, to get the optimal order of the coordinates in the PCP based on more than one user's desired criteria as objective, but for

Input : H: Generate reference points structured reference points Z^s or supplied aspiration, $MaxIter$= number of generation, points Z^a, P: Initialize population (uniform distribution)
Output: P_{t+1}
for $t \leftarrow 1$ **to** $MaxIter$ **do**
  $S_t = \emptyset$
  $i = 1$
  Q_t=Recombination+Mutation (P_t)
  $R_t = P_t \cup Q_t$
  Rt=Non-dominated solutions `// All level fronts` $F1, F2, ...Fl$
  repeat
    $S_t = S_t \cup F_i$ and $i = i + 1$
  until $|S_t| >= N$`// until the parent population is filled`
  ;
  $F_i = F_l$ `// Last front is included`
  if $|S_t| = N$ **then**
    $P_{t+1} = S_t$, break
  else
    $P_t = \cup_{j=1}^{l-1} F_j$
    choose points from $F_l : K = N - |P_{t+1}|$
    Normalize the objectives and generate reference set Z^r :
      Normalize(f^n, S_t, Z^r, Z^s, Z^a)
    Associate each member **s** of S_t with a reference point:
      [$\pi(s), d(s)$]=Associate(S_t,Z^r)%$\pi(s)$: closest reference point, d: distance between **s** and $\pi(s)$
    Compute niche count of reference point: $j \in Z^r$:
      $p_j = \sum_{s \in S_t / F_l} (\pi(s) = j ? 1 : 0)$
    Choose K members one at a time from F_l to construct $P_t + 1$: Niching $(K, p_j, \pi, d, Z^r, F_l, P_{t+1})$
  end
end

Algorithm 5: Pseudo-code of NSGA-III

high dimension datasets. Therefore, it uses multi- or many-objective optimization algorithms, such as NSGA-II and NSGA-III to find the axes' best order in PCP. It uses the same two criteria used in the PVRPCP method to minimize crossing lines and maximize the correlation between neighboring axes in the PCP, which are used as objectives. Several datasets are utilized to discover the success of the proposed method. Also, the two algorithms' results are compared to see which one gives better results in terms of the given criteria. Furthermore, subjective (human-based) assessments of the results are provided to investigate the proposed method's effects.subsection Evaluation Metrics

The evaluation metrics have to be in conflict to be used for the proposed method that is why the number of crossing lines is minimized and the correlation values are maximized as objectives in this experiment. These measures are used to optimize the order's coordinates in the PCP, utilizing NSGA-II and NSGA-III. The explanation of both criteria are presented in the first of the PVRPCP evaluation metrics. NSGA-II and NSGA-II are used for a fair comparison in the results.

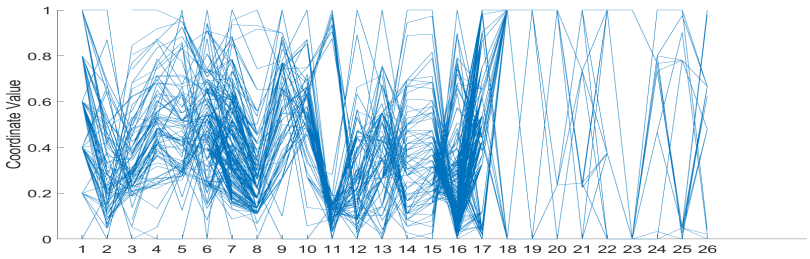
3.5.1 The Experimental Results

This section provides the used datasets as well as the results for NSGA-II and NSGA-III in comparison.

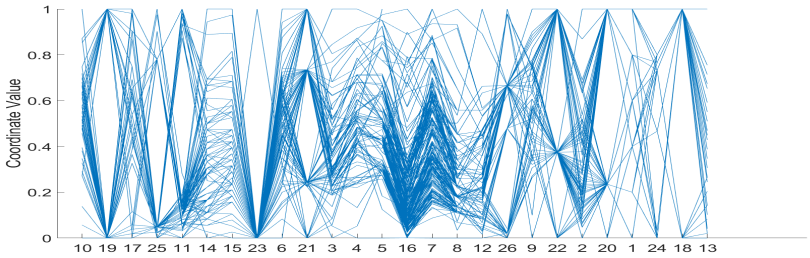
Utilized Dataset Visualizations: The datasets include various dimensions, as shown in the Table 3.1 to evaluate the method. The 1985 Auto Imports Database and Cardiac Arrhythmia datasets are taken from the UCI machine learning repository [13]. These datasets are used with different dimensions; the Auto Imports Database with 15 and 26, and the Cardiac Arrhythmia ones with 50-, 75-, and 100-dimension number (ND).

NSGA-II and NSGA-III Results: In these experimental results, NSGA-II and NSGA-III are applied to improve the PCP for high-dimensional data (D>10) and multi-criterion. In this experiment, the evaluation metrics minimize the crossing lines and maximize the correlation between adjacent axes in PCP. Thus, the NSGA-II and NSGA-III gives the first PF, which has a set of solutions. Then, VIKOR ranking selects the best solution from the first PF set. In the first case study in Fig. 3.11, the results of the 26-dimensional import dataset using NSGA-II and NSGA-III methods show improvement in terms of both criteria. The number of crossing lines decreased L=81037 and L=80639, respectively. Also, the correlation value

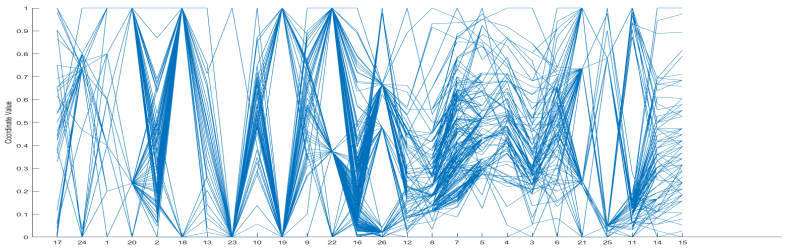
increased from 0.343 to 0.459 compared to the original plot. Moreover, in the Arrhythmia dataset, $D = 50$ in Fig. 3.12, $D = 75$ in Fig. 3.13, and $D = 100$ in Fig. 3.14, both criteria have improved due to using the two methods. In the Arrhythmia dataset D=50, the number of crossing lines dropped dramatically from 38406 in the original to 1324 in NSGA-II and 1633 in NSGA-III results. Therefore, in the Arrhythmia dataset D=100, by utilizing NSGA-II and NSGA-III, both criteria have significantly improved. The number of crossing lines decreased from original order= 53256 to 204 and 352, respectively. Also, the correlation values rose from 0.3 to 0.4 approximately. Moreover, it can be visually seen how the number of crossing lines reduced. For instance, in the Arrhythmia dataset D= 75 result, the crossing lines are fewer than in the original plot in NSGA-II and NSGA-III results.



Original plot, L= 141990, Corr=0.34

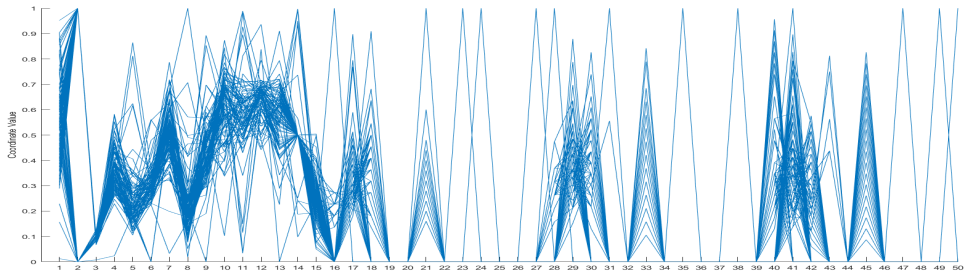


Best Order,NSGAII, L=60953, Corr=0.45

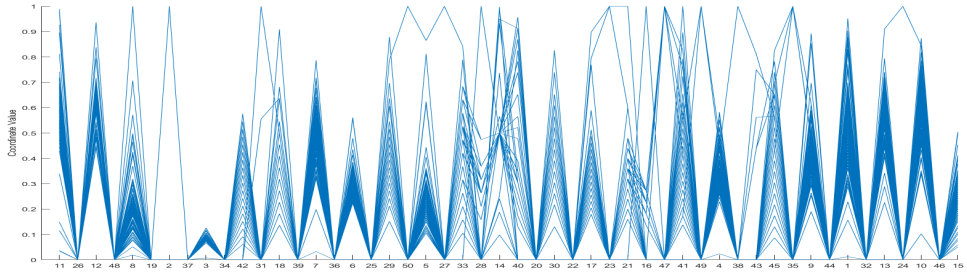


Best Order, NSGA-III, L=61361, Corr=0.46

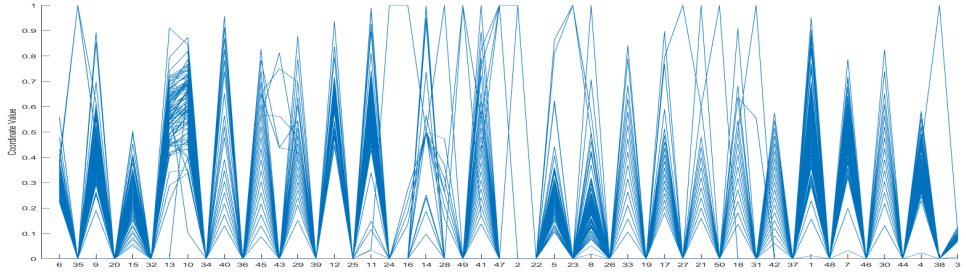
Fig. 3.11: NSGA-II and NSGA-III RESULTS: the Original and the best of Parallel Coordinate Visualization of the Imports Data D=26, where L indicates the number of crossing lines and Corr indicates the correlation value



Original plot, L= 38406 , Corr=0.33

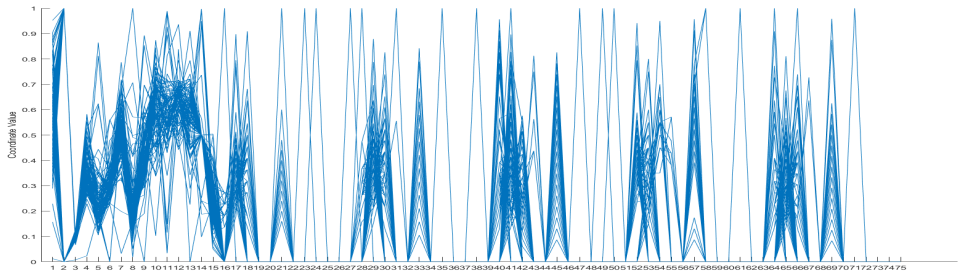


Best Order, NSGA-II, L=1324, Corr=0.38

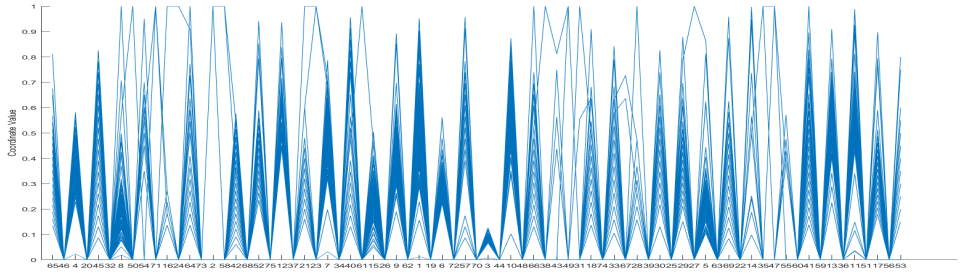


Best Order, NSGA-III, L=1633, Corr=0.37

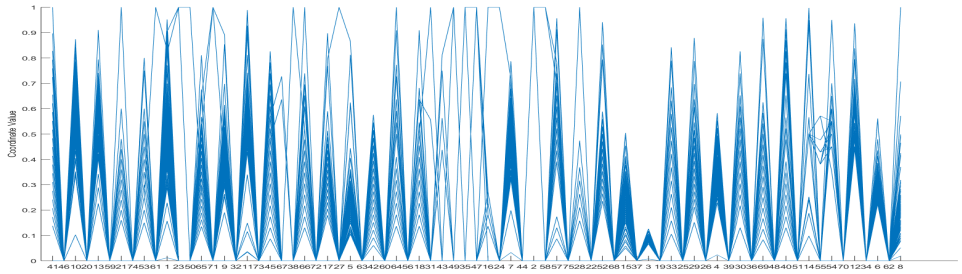
Fig. 3.12: NSGA-II and NSGA-III RESULTS: the Original and the Best of Parallel Coordinate Visualization of the Arrhytimia Datasets D=50, where L indicates the number of crossing lines and Corr indicates the correlation value



Original plot, L= 45315, Corr=0.36

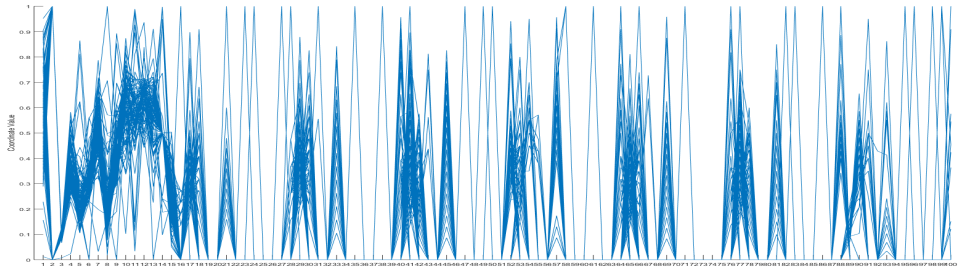


Best Order, NSGA-II, L=199, Corr=0.42

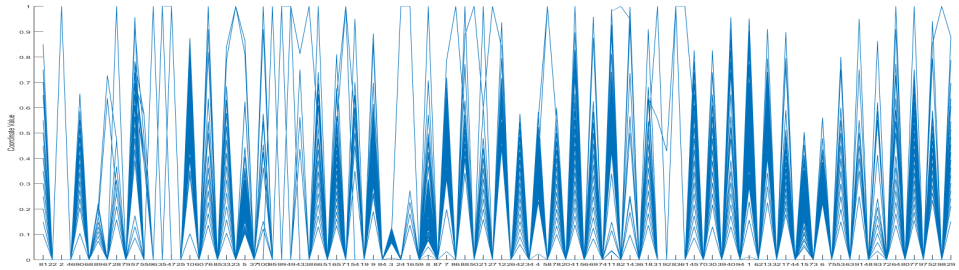


Best Order, NSGA-III, L= 414, Corr=0.41

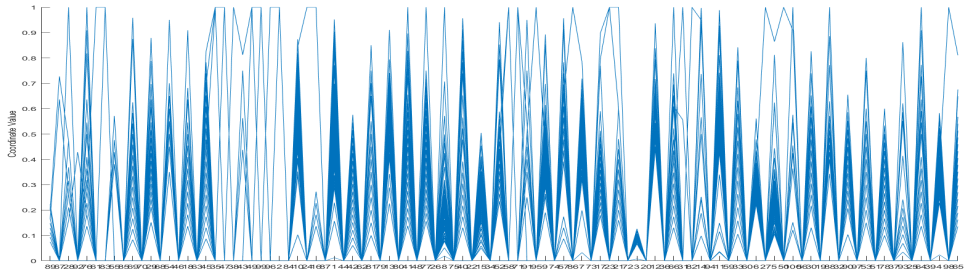
Fig. 3.13: NSGA-II and NSGA-III RESULTS: the Original and the Best, of Parallel Coordinate Visualization of the Arrhythmia Datasets D=75, where L indicates the number of crossing lines and Corr indicates the correlation value



Original plot, L= 53256, Corr=0.37



Best Order, NSGA-II, L=204, Corr=0.42

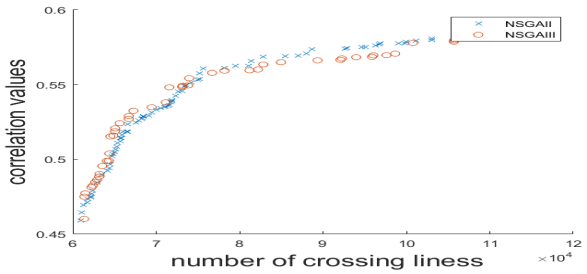


Best Order, NSGA-III, L=352, Corr=0.41

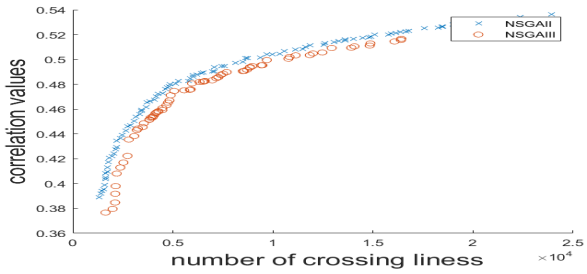
Fig. 3.14: NSGA-II RESULTS: the Original and the Best of Parallel Coordinate Visualization of the Arrhythmia Dataset D=100, where L indicates the number of crossing lines and Corr indicates the correlation value

Pareto-Front Comparison (NSGA-II and NSGA-III): In the experimental results, as shown in the comparison performance plot of NSGA-II and NSGA-III in Fig. 3.15, the NSGA-II PF gives better coverage than in NSGA-III since NSGA-II has been used to solve multi-objective problems. However, a statistical test is applied for an in-depth comparison. Hypervolume (HV) is utilized [70], where a high mean HV value and lower stander variation (Std) indicate that one of the algorithms performs better than the other with scale interval of 95%. The Wilcoxon rank-sum test is applied. This indicates that both NSGA-II and NSGA-III performance are tied (t) and comparable in Table 3.4, where is (w) is the winners, (t) is the

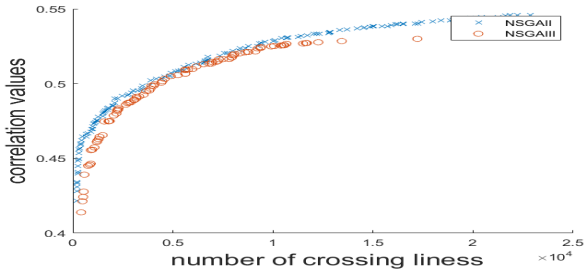
tiers , and (l) is the number of losers. The choice of using the proposed methods is flexible. If more than three metrics enhance the PCP, NSGA-III is more suitable to optimize the order of coordinates in the PCP.



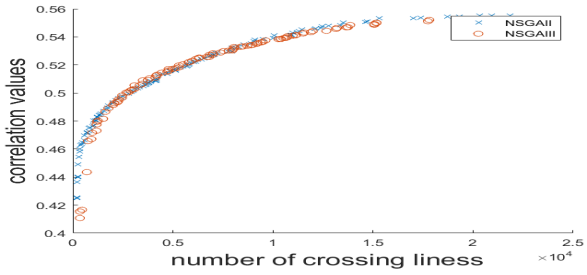
Imports dataset D= 26



the Arrythmia datasets D=50



the Arrythmia datasets D=75



the Arrythmia datasets D=100

Fig. 3.15: The PF Comparison Results of NSGA-II and NSGA-II

Dataset	D		NSGA-II	NSGA-III
1985 Import	26	Mean	4.52E+03	**4.62E+03**
		Std	**2.08.1E+02**	2.46E+02
Arrhy-thmia	50	Mean	**5.34E+03**	4.68E+03
		Std	2.53E+02	2.46E+02
Arrhy-thmia	75	Mean	3.021E+03	**3.15E+03**
		Std	**1.80E+02**	2.46E+02
Arrhy-thmia	100	Mean	**3**.41E+03	**4.68E+03**
		Std	**2.081e+02**	2.46E+02
(w/t/l)				(0/4/0)

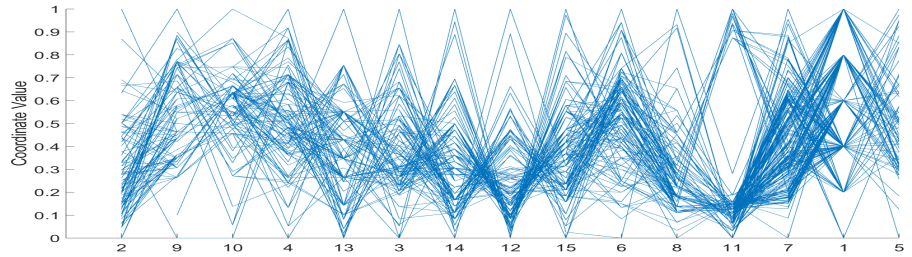
Table 3.4 Comparison NSGA-II and NSGA-III Performances Using HV Measure

3.5.2 Data Visualization and Human Perception

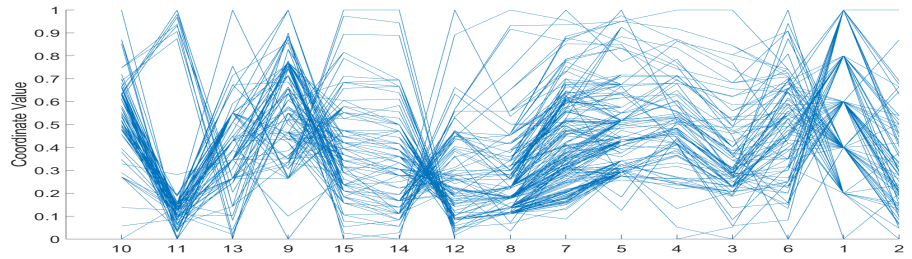
Humans' perception helps discover the knowledge and analyze data visualization and the ways data visualization tools could be improved. In [71], Ward et al. indicated that the humans' perception connects directory to humans' senses, which are watching, smelling, tasting, hearing, and touching. Therefore, the human brain makes decisions based on these senses to discover the information pattern and knowledge. They also emphasize that humans' perception is related to the activities, analyzing the surrounding environment's data. For instance, in some fields such as the medical field, humans need to draw conclusions after viewing the data in a visualized form, such as a tree or a graph. Based on the doctor's analysis, the patient gets medical treatment. Therefore, human perception is used in this chapter to compare human decisions with the optimization algorithms' solutions.

Subjective (human-based) Assessments of Results: This experiment's main contribution is to identify whether the optimization algorithms give the user the most appropriate answer. Therefore, human perception is used to offer the knowledge of having the best order coordinates in the PCP and the worst case based on the utilized criteria. The aim is to minimize the crossing lines and maximize the correlation values between its neighboring axes.

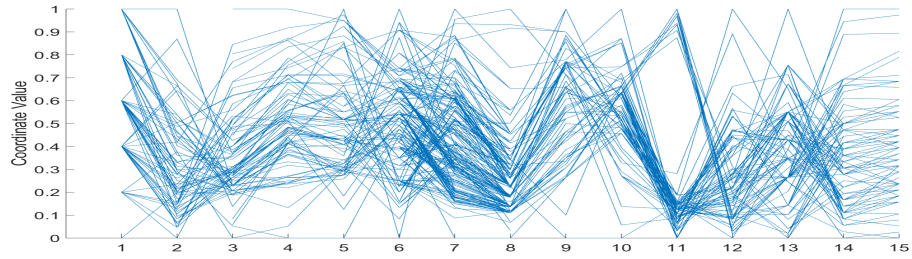
Thus, the results of NSGA-III are used to compare human perception decisions and the optimal and worst solutions from the optimization algorithms, given that NSGA-III had been utilized before NSGAII was compared to NSGA-II results. These data visualization plots of different order axes in the PCP are chosen as the original and the PF solutions' (second extreme, the worst, and best) visualization plots, which are selected based on VIKOR. The original plot is the main order axes in the PCP dataset. The second extreme plot maximizes the correlation between adjacent axes, and the optimal plot is the best solution in the first PF rank. The worst case plot is shown in the worst solution in the last PF rank. The deciders are a human, so the four different order visualization plots wer given to a group of people, without telling them which are the best, the second extreme point, the worst case, or the original PCP. A brief description was provided regarding what they should mark, minimizing the crossing lines and maximizing the correlation between neighboring as 1, 2, 3, and 4 markers. The people were asked to rank them as the best, the worst, and the second-best and the dataset's original orders. The organization of them were unknown to them. The first one is A, which is the worst case, B is the second extreme point case, C is the original plot, D is the best order as in Figs. 3.16, 3.17, 3.18, and 3.19. They were asked to sort the order from the best of good correlation and minimize the crossing lines between neighboring axes in the PCP visually to the worst scenario. Thus, D had the highest number marked 34 scores overall in the Table 3.5, which is the best order in terms of two good correlations and minimum crossing lines in NSGA-III results. Seven people chose the 15 D dataset as the first and the optimal order (D) compared to the said five people who said it should be the second optimal and nobody said it should be in the worst or the third case. Also, in the 26D dataset results in the Table 3.5, the best order (D) had the highest seven votes out of twelve. Therefore, the algorithms give the optimal order in the PCP based on the users' desired criteria. Similar to the 15D and 26D datasets results, 50D and 75D datasets had the highest votes to D plot for the best in terms of both given criteria. For the 100D dataset results, this was marked as number one (D). The optimal plot took nine votes. Moreover, the worst order A plot and the second scheme plot B had similar scores overall, 28 and 27 votes for all datasets results. Therefore, the algorithms give the optimal order in the PCP based on the users' desired criteria.



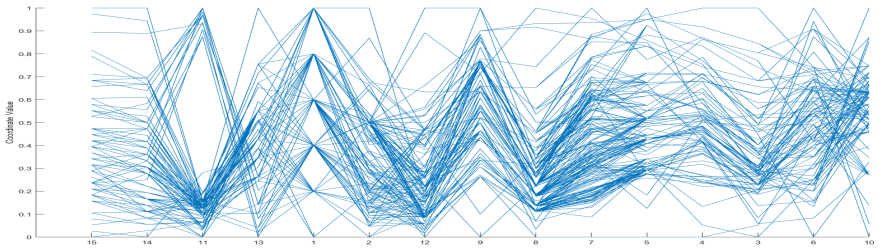
A



B

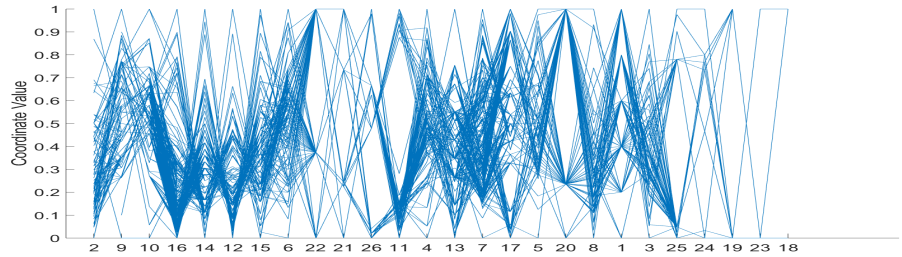


C

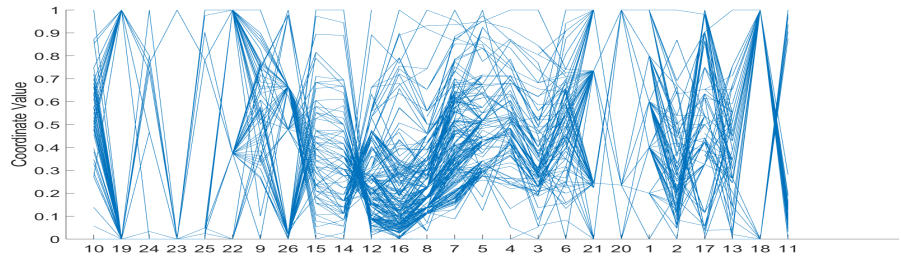


D

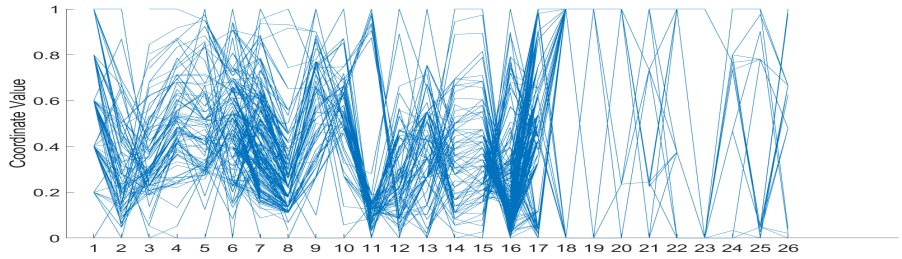
Fig. 3.16: Human Perception Results: Imports Dataset, D=15



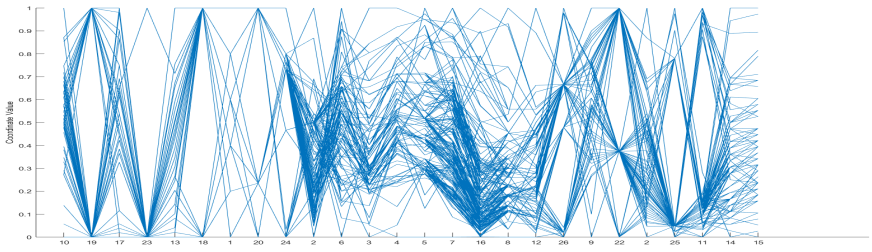
A



B

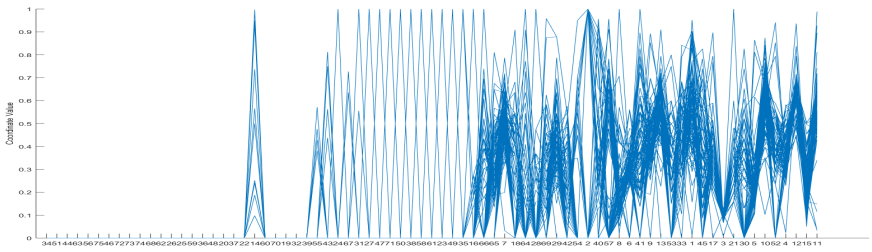


C

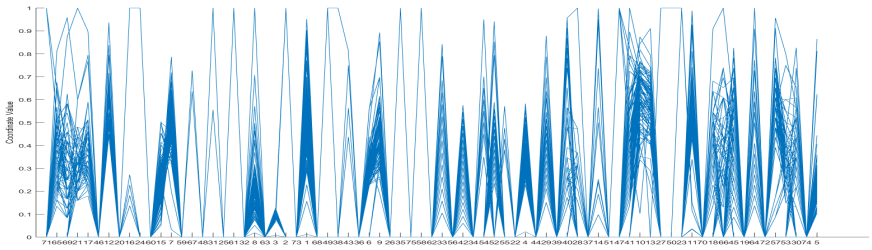


D

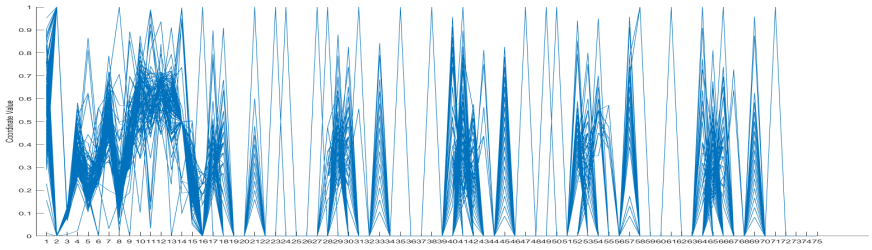
Fig. 3.17: Human Perception Results: Imports Dataset, D=26



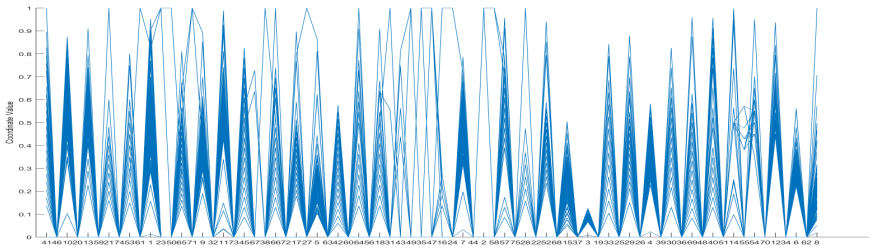
A



B

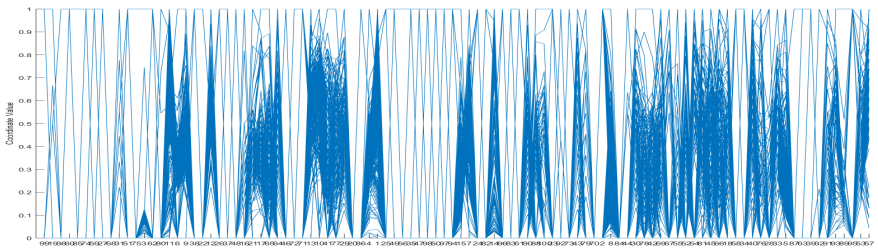


C

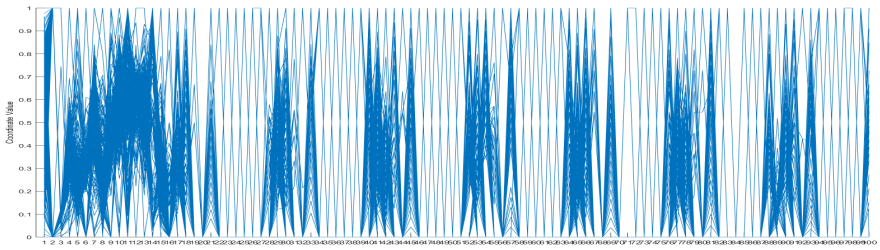


D

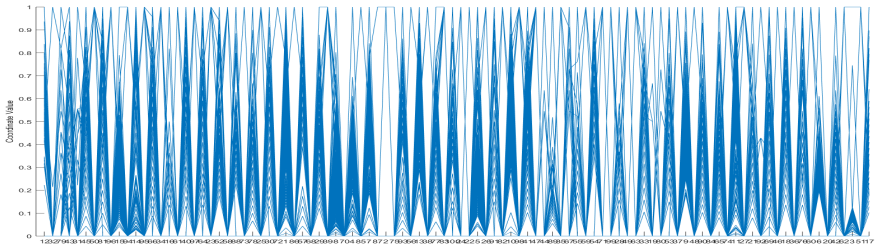
Fig. 3.18: Human Perception Results: Arrythmia Dataset, D=75



A



B



D

Fig. 3.19: Human Perception Results:Arrythmia Dataset, D=100

DN	15	26	50	75	100		
A	3	0	2	0	2	7	*1*
	6	2	4	5	1	18	*2*
	2	8	4	5	9	28	***3***
	1	9	2	2	0	14	*4*
B	1	2	5	6	1	15	*1*
	0	2	1	0	11	14	*2*
	2	1	1	0	0	4	*3*
	9	7	5	6	0	27	***4***
C	1	2	0	0		3	*1*
	1	7	5	5		18	*2*
	8	2	7	6		23	***3***
	2	1	0	1		4	*4*
D	7	7	5	6	9	34	***1***
	5	1	2	2	0	10	*2*
	0	2	1	1	3	7	*3*
	0	2	4	3	0	9	*4*

Table 3.5 Human Perception Results of 12 people

3.6 Summary

This chapter introduces a brief description of three methods used to enhance the PCP using the re-ordering axes technique. In the first method, NDS and VIKOR ranking are described to improve the PCP for lower dimensional-data visualization (D<=10). For (D>10) data, GA with a smart mutation operator, is explained in the second method. The evaluation metric is introduced to minimize the crossing lines and correlation between neighboring axes in the PCP. Finally, the Monte Carlo simulation for the smart mutation operator is described. A brief description of the experimental results is introduced using the three procedures of the methods, which are handled to enhance order in the PCP. In the first stage, the first proposed method for PVRPCP results demonstrated significant improvement of the PCP in terms of the two metrics given, minimizing the crossing lines and maximizing the correlation. Next, in the second stage, the GA results for a single objective with smart mutation operated also minimizing the crossing lines between adjacent axes in the PCP. Moreover, in the third stage, the experimental results of NSGA-II and NSGA-III showed compelling results, upgrading order coordinates in the PCP by multi-criteria and many criteria optimization. In addition, the subjective assessment experiment is conducted on NSGA-III results proved that the NSGA-III method improved the PCP in terms of the utilized criteria.

Chapter 4

Enhancing the RadVis

4.1 Introduction

This chapter proposes a new method to enhance the RadVis visualization to solve the overlapping data point problem using k-means clustering and internal validity measures by exhaustive search and the Genetic Algorithm for for high-dimensional datasets. The RadVis method is described in detail. The proposed method's concept is similar to the previous methods for the PCP; it rearranges the dimensions around the circle's circumference and can have $n!$ possible order axes. For instance, if the data with 7 is provided, the dimensions can have 7! possible permutations or positions around the RadVis technique's circumference. Thus, having different possible orders for the dimensions around the circle in RadVis can present a solution for overlapping data points in the RadVis. The method finds the dimensions' optimal order around the circumference in RadVis using k-means clustering quality, exhaustive search and the GA optimizer for high-dimensional datasets.

4.2 Visualization Method

The RadVis method has been utilized to visualize high-dimensional datasets by using a reduction technique. It reduces the M-dimensional dataset visualization space to 2D space using the circle technique. RadVis arranges the data points around the circumference of the unit circle, so each point includes one dimension, and holds the data points in the spring. All the dimensions springs are in a puck in 2D-space [14]. Therefore, it assigns each dimension to one angle and the data points around the circumference of the circle based on the following equations:

$$x_i = \frac{\left(\sum_{j=1..,d} a_{i,j} \cos\theta_j\right)}{\left(\sum_{j=1..,d} a_{i,j}\right)}, i = 1..., n \tag{4.1}$$

$$y_i = \frac{\left(\sum_{j=1..,d} a_{i,j} \sin \theta_j\right)}{\left(\sum_{j=1..,d} a_{i,j}\right)}, i = 1..., n \tag{4.2}$$

Where is d is the dimension number, n is the numerous samples, x_i is the data points of x axis, y_i is the data points of y axis, and θ_j= $(0 : 360)/d$: $(d-1) * (360/d)$. The θ_j is the angle for each dimension around the circumference in a circle and d is the dimensions, and $a_i,_j$ is the data point in each dimension .

Consequently, permuting the dimensions around the circumference in a circle may improve visualization in RadVis. For a better understanding, the dimensions with subsets are s_1.....s_j distributed on circumference by different degrees, and sets are fixed and attached to the springs until all springs reach the end of the puck. When the puck reaches the equilibrium position, for instance, the point $(v_i, u_j)^T$ is projected in 2D space (x_i, y_j). Optimizing the order of the angles' dimensions in RadVis can improve the visualization of it. For example, if an m-dimensional dataset is provided, m! possible permutations can be permuted for each dimension based on given criteria. Thus, each permutation can have different position projections for the data points. This chapter tries to improve the RadVis by optimizing the order of its axes based on clustering quality.

4.3 K-means Clustering

K-means clustering is a well-known approach to cluster the datasets into partitions [72] ,[73]. In the algorithm 6 of k-means clustering, the first step is to select k objects to be the cluster centroids as the following set: $C = C_1, C_2, ...C_k$. Then each object is assigned to the closest centroid by determining the distance between them.

$$\arg\min_{c_i \in C} dist(c_i, x)^2 \tag{4.3}$$

Where dist(.) measures the distance between data point x and the centroid cluster. Note there are several ways to calculate the distances such as Euclidean distance, Manhattan Distance, and Cosine Similarity. Thus, the well-known measure is applied, which is Euclidean Distance Measure in Eq. 4.4.

$$L(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \tag{4.4}$$

The third step is to find the average of all objects in the cluster in Eq.4.5.

$$C_i = \frac{1}{|S_i|} \sum_{x_i \in S_i} X_i \tag{4.5}$$

Where c_i is the cluster which has a set of all objects S_i.

Finally, the first and the second steps are repeated until all the assignments of the objects become stable in the same cluster in each iteration. Thus, there are no changes in assigning the objects to the clusters when the centroids of the clusters are removed. The pseudo-code of k-means clustering algorithm is demonstrated in algorithm 6.

Input : k is the number of clusters, with corresponding objects
Output: Set of k centroids
1-randomly assign k centroid to random objects
2-assign each object to the nearest centroid
3-move the centorid to the average of all objects in the cluster
4- **Repeat** 1 and 2 **until** none of the assignments change

Algorithm 6: Pseudo-code for k-means clustering

Consequently, there are several evaluation measures to achieve better visualization of k-means clustering such as maximizing and minimizing the distance between each cluster. This concept is utilized to solve the overlapping data points in RadVis for high-dimensional datasets. The internal validity measure for k-means clustering is applied. This is the Dunn measure for optimizing the order of the angles' dimensions in the datasets in order to get a better visualization in RadVis.

4.4 Evaluation metric: Internal Validity Measures (Dunn measure) as Clustering Quality

The internal validity measure aims to improve the clustering quality. Therefore, the datasets' visualization can be reflected by the quality of clusters, so several internal validity measures have been proposed that generated varied results such as Dunn and Sum-of-Squares [74]. Dunn is one of the commonly used internal validity measures is Dunn. It relies on the score used as fitness for the optimizer [75]. It determines the square root of the minimum space between two clusters divided through the maximum space between two points in the

same cluster. Note that the distance between two clusters is determined as the space between the two closest points from the two clusters in Eq. 4.6.

$$DI(score) = \frac{\sqrt{MinInterclusterDist}}{\sqrt{MaxInterclusterDist}} \tag{4.6}$$

Therefore, the Dunn index is defined in Eq. 4.7, and the user specifies the number of clusters.

$$DI(Ind) = \min_{i=1...k} \left(\min_{j=i+1...k} \left(\frac{d(C_i, C_j))}{\max_{z=1..k} (diam(C_z))} \right) \right) \tag{4.7}$$

Where $d(C_i, C_j)$ is the minimum distance between the Inter-clusters; C_i and C_j clusters in Eq.4.8.

$$d(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y) \tag{4.8}$$

Indeed, the diameter of a cluster is calculated based on the following Eq.4.9.

$$diam(C) = \max_{(x,y) \in C} d(x, y) \tag{4.9}$$

Note that the big score of the Dunn measure is indicated to have well-separated clusters. Thus, this chapter uses this metric as an objective function for the GA and exhaustive search methods to enhance the RadVis.

4.5 Proposed Methods

This section introduces two methods for improving the RadVis in terms of overlapping in data points. The first method uses exhaustive search based on internal validity measures for k-means clustering for data visualizations. Similarly, the second uses the optimizer GA instead of an exhaustive search for the high-dimensional data visualizations.

4.5.1 Enhancing RadVis Plot Using Exhaustive Search on Clustering Quality

In this section, an improvement in RadVis is achieved using exhaustive search for lower dimensional-datasets, k-means clustering, and the Dunn measure to enhance the clusters. Thus, the contribution of this section is to have non-overlapping clusters since it is the main

issue in the RadVis, especially in high-dimensional and large-scale datasets. The first procedure is to permute the angles of the dimensions x and y for all possible permutations after applying the RadVis. Thus, different order angles of x and y in j=1....d are provided. A matrix of x and y is introduced. Then, based on k-means clustering and the Dunn score (DI). The x_i and y_i vectors are selected based on the highest value of DI, and the x_i, and y_i vectors rely on the worst value of DI by checking the all possible permutation of the dimensions. Last, the visualization of the best and the worst values of DI are demonstrated using a RadVis plot to investigate the success of the proposed method.

Visualized Datasets: This section demonstrates that datasets in table 4.1are used in the experimental results to evaluate the proposed method; Iris, Ecoli, and Glass are used for an exhaustive search experiment to enhance the RadVis. Sonar, Clean2, Libras Movement, Driving performance evaluation (DirFace) , and a1-raw datasets are utilized for optimization algorithm experiments for improving the RadVis. They are from the UCI machine learning repository [13].

Dataset	Number of Dimensions	Number of Samples	number of classes
Iris	4	150	3
Ecoli	7	336	8
Glass	9	214	7
Sonar	60	208	2
Clean2	70	6598	2
Libras Movement	22	366	15
DirvFace	22	606	3
a1-raw	15	1747	5

Table 4.1 Details of Visualized Datasets

Experimental Results: The proposed method is tested on several datasets, which are Ecoli, Iris, and Glass datasets. For the Ecoli dataset result in 4.1, the clusters using k-means with the best Dunn score $(DI = 0.24)$ are separated well from each other and the samples are compared to the worst $(DI = 0.05)$ visualization. Similar to the Ecoli dataset result, in the Iris data visualization plots in 4.1, the best $(DI = 0.35)$ and k-means clustering visualization

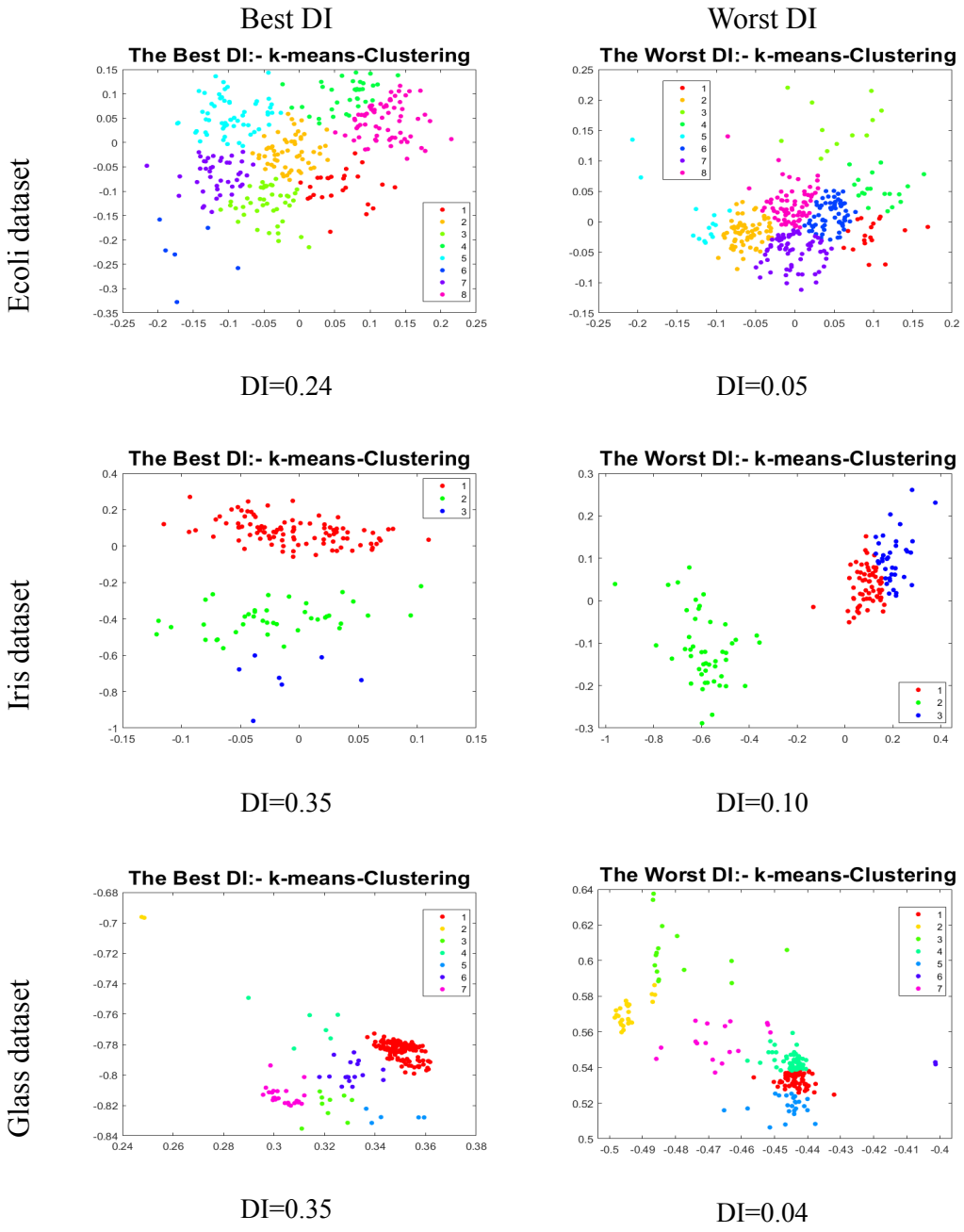


Fig. 4.1: Enhancing RadVis plot using exhaustive search on clustering quality: the best, and the worst clusters based on Dunn measure, which is indicated here as DI)

prove to be better than the worst $(DI = 0.10)$ in terms of clear clustering. Moreover, in the Glass dataset result in 4.1 where the Dunn score is high $(DI = 0.35)$, the clusters are distant in comparison to the worst $(DI = 0.04)$ plot.

Overall, the proposed method enhances the RadVis using the Dunn measure (DI) for k-means clustering, and any other internal validity measures can also be applied. It is up to the user's desired criteria.

4.5.2 Enhancing RadVis plot Using Optimization Algorithms Search on Clustering Quality

In this section, GA is used as an optimizer to improve the high- dimensional dataset visualizations using RadVis, k-means clustering and internal measures (Dunn measure).

Genetic Algorithm (GA): As introduced in chapter 3, GA is a suitable method to solve the combinatorial optimization problem as an NP-hard problem. Therefore, rearranging the dimensions around the circle in RadVis techniques is also a combinatorial optimization problem. Hence, reordering the dimensions in the RadVis can decrease the overlapping data points. This method utilizes the optimal order of the dimensions based on internal validity measures for k-means clustering. GA with random mutation scheme (scheme 1) is described in chapter 3, which is used to optimize the positions' dimensions in RadVis. The GA is applied after mapping the dimensions of two angle vectors (x_j, y_j), then the k-means clustering is utilized. Then Dunn's score is determined in this way. A combinatorial GA can be summarized with a set of individuals, population, which are a group of different orders of variables. For instance, in a 9-dimensional dataset, 9! possible orders of angles in RadVis are provided. A random number from 9! possible orders are chosen as a population.. The process maps the dimensions into 2 vectors, x_j and y_j, using the RadVis' reduction. Then for the whole population, the dataset is clustered using k-means, and the Dunn score is calculated. Secondly, based on the Dunn score (DI), which is given to the optimizer GA as fitness values, the based order of (x_j, y_j) angles is selected. Moreover, during the optimization process, the worst individual is selected from the initial population to be compared with the best-case scenario.

Visualized Datasets: The used visualization datasets are introduced in the previous experiment in table 4.1.

Experimental Results: Different high-dimensional datasets are utilized to investigate the proposed methods' success in 4.1. For the clean2 dataset result in Fig. 4.2, there are two clusters with a high number of samples. The best $DI = 0.10$ k-means clustering visualization is good with space between them compared to the worst $DI = 0.02$ case. Moreover, for

the Libras dataset result in Fig. 4.2, the visualization in the best $DI = 0.20$ k-means clustering is better than the worst plot $DI = 0.05$, which indicates overlapping clusters. Similar to the Libras dataset result, the a1-raw in Fig. 4.3 is the worst case $DI = 0.02$ where the clusters are overlapping, but the best plot is not $DI = 0.08$. Consequently, in Figs. 4.3 and 4.2, the best DI k-means clustering visualizations have better and more well-separated clusters and points than the worst DI k-means clustering visualizations. Moreover, the best-case-clusters data points are also better spaced than in the worst overlapping case. These results are based on the Dunn measure for k-means clustering, which aims to have well-separated clusters. Other internal validity measures can aim to have different results based on the users' desired criteria.

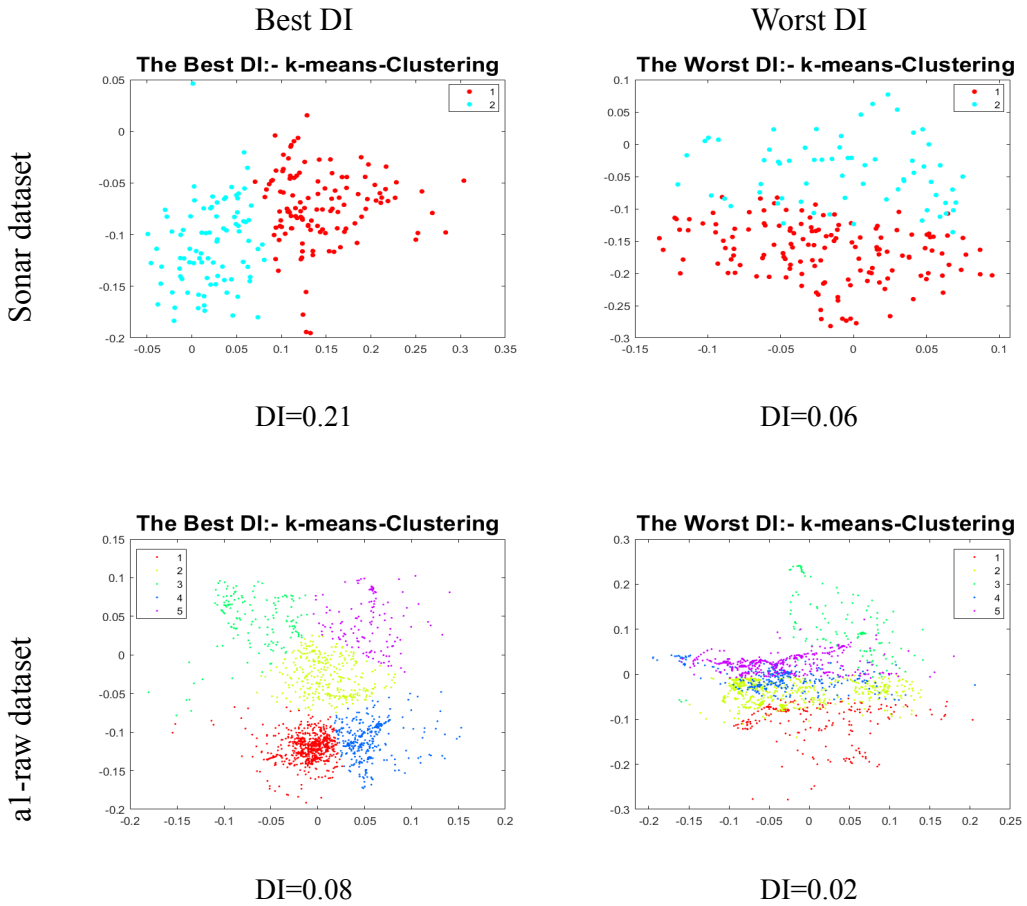


Fig. 4.3: Enhancing RadVis plot using GA on clustering quality: the best, and the worst clusters based on Dunn measure (DI).

/

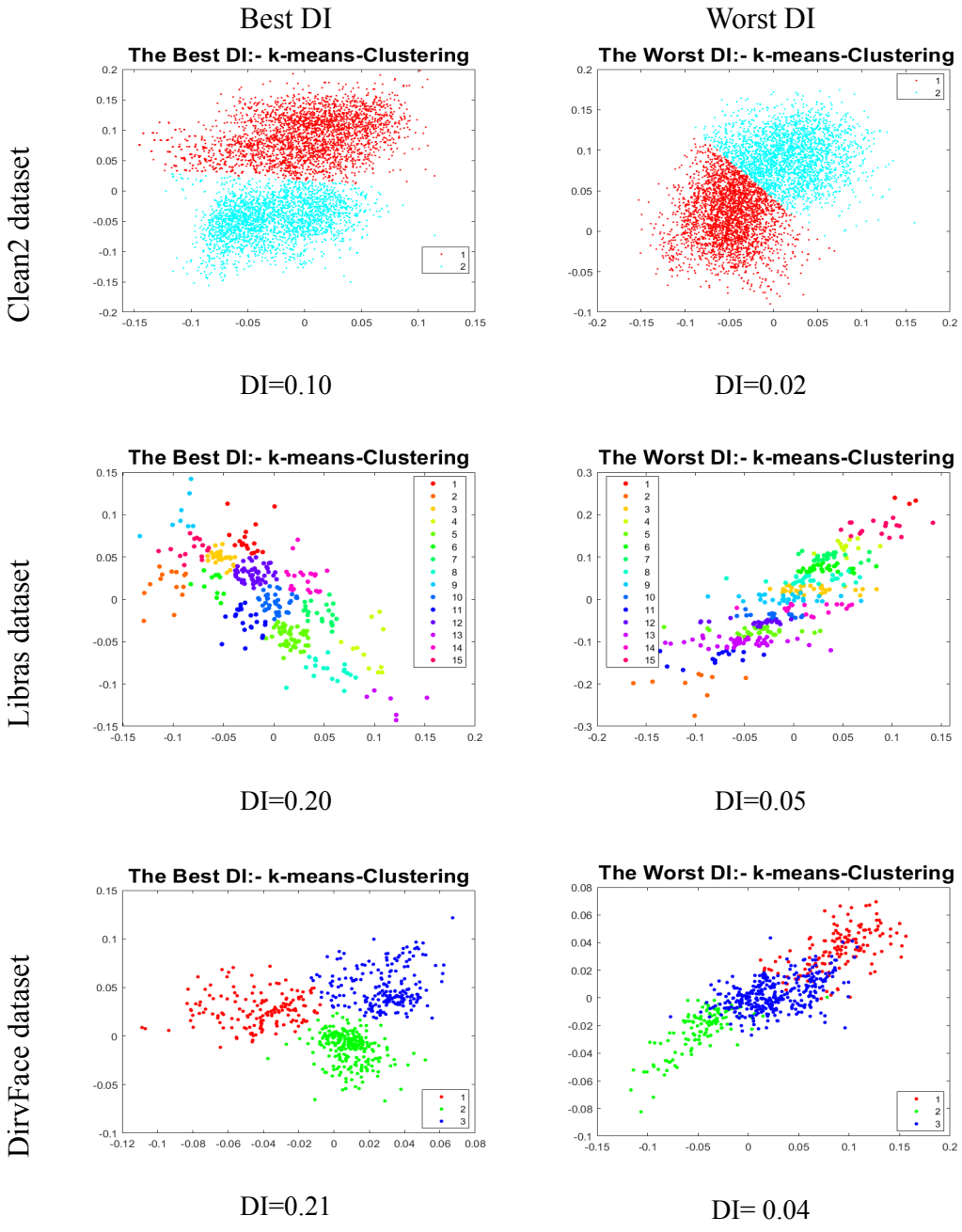


Fig. 4.2: Enhancing RadVis plot Using GA on Clustering quality: the Best, and the Worst Clusters Based on Dunn Measure (DI).

4.6 Summary

In this chapter, two methods are introduced to enhance the RadVis in terms of overlapped data points using K-means clustering and internal validity measures (Dunn). The first uses an exhaustive search, while the second utilizes the GA optimizer. The proposed methods show a significant improvement based on the given criteria (Dunn score) DI in exhaustive search

(D<10) datasets and the optimizer GA for (D>=10) data visualizations. The Dunn score aims not to create overlapped clusters. However, the proposed methods are flexible in choosing the criteria, which are Internal validity measures for unsupervised data, and it can be based on users' desired criteria.

Chapter 5

Summary and Conclusion Remarks

Improving visualization tools is a critical task for high-dimensional-dataset visualization because every tool has some issues. Therefore, this thesis aims to refine two main visualization tools. The first to enhance the PCP because the PCP has clutter issues and many crossing lines between adjacent axes. The second is to improve the RadVis in well-separated data points using internal validity measures for k-means clustering. The first part of this thesis introduces three stages for improving the PCP. The first stage presents using the PVRPCP method in multi-metric comparison to reorder the axes in the PCP to provide better visualization for (D<=10). The method ranks the permutation of PCP using a non-dominated sorting algorithm and VIKOR measure. It gives the best order of coordinates to give a more beneficial visualization of data. In the experiments, two metrics are utilized, minimizing the number of crossing lines and maximizing the correlation coefficient. Some datasets are designed based on these two metrics. The constructed benchmarks give more reliable results to indicate the benefit of the proposed method. The PVRPCP tested two other kinds of datasets: multi-objective function datasets and real-world datasets. This experiment shows an improvement in the PCP based on comparing the original to the first PCP rank. Note that the results of PVRPCP are dependent on the selected metrics since the ordering of axes can heavily affect the resulting visualization as a result. Moreover, the second stage introduces a new smart mutation for the GA to improve the visualization of PCP by finding the optimal order of PCP axes. In the defined decomposable objective function, i.e., minimizing the number of crossing lines, the new individuals can be generated based on optimizing the function's components. Accordingly, a smart mutation operator is proposed to select the genes intelligently which should be updated during the optimization process. In order to investigate the performance of the method, the Monte Carlo simulation and several experiments are conducted

on some benchmark datasets. The results indicate the significant improvement of the proposed smart mutation compared to other developed schemes based on the random selection of axes. Generally speaking, the proposed scheme is a framework that can be applied based on every user-desired metric, which is decomposable to enhance the PCP. In the third stage, a new method is proposed to optimize the coordinates' order in the PCP visualization for high-dimensional-measure datasets (D>10) using multi-and many-criteria evaluations by applying combinatorial optimization algorithms. Furthermore, the NSGA-II is to solve multi-objective optimization problems, and NSGA-III is to optimize many-objective optimization problems. The experimental results show how the proposed method can positively affect the optimization of the order coordinates in PCP. Furthermore, two criteria are utilized to minimize the crossing lines and maximize the correlation between adjacent axes. However, more than two criteria can be used. They depend on the reviewer's desired criteria.

The second part of the thesis defines two methods to improve the RadVis by defining the optimal order of the dimensions around the circumference using internal validity measures for k-means clustering. The first method applies exhaustive search for (D<10) to find the best order of dimensions around the circumference based on the internal validity measure for k-means clustering. The second approach applies the optimizer GA to have the best order in the dimensions of the angles in RadVis for high- dimensional dataset-visualizations for (D>10). The results show a significant improvement in terms of the given criteria (Dunn), which demonstrate well-separated clusters in both methods. Therefore, the proposed method can show different results depending on the type of internal validity measure.

5.1 Published Papers

1: In 2019 the conference paper " Proposing a Pareto-VIKOR Ranking Method for Enhancing Parallel Coordinates Visualization" 14th International Conference on Computer Science [15], pp. 895–902 .

2: A conference paper is recently published ; the title is "Enhancing Parallel Coordinates Visualization Using Genetic Algorithm with Smart Mutation" to IEEE SMC2020 [16], pp. 3746–3752.

5.2 Future Work

This research is a highly pertinent topic since it can be applied to enhance other visualization tools. Directions for future work could be:

1: Smart mutations can be utilized to achieve an improvement in the performance of other single objective algorithms to solve other combinatorial problems such as TSP and N-queen problem.

2: Indeed, smart mutations may aim to achieve an improvement in the performance of multi- and many-objective combinatorial optimization algorithms.

3: RadVis can be improved based on other internal validity measures for k-means clustering.

4: Utilizing the same combinatorics optimization to enhance other visualization techniques.

References

[1] A. Ibrahim, S. Rahnamayan, M. V. Martin, and K. Deb. “3D-RadVis Antenna: Visualization and performance measure for many-objective optimization”. In: *Swarm and evolutionary computation* 39 (2018), pp. 157–176.

[2] Q. Li. “Overview of Data Visualization”. In: *Embodying Data: Chinese Aesthetics, Interactive Visualization and Gaming Technologies*. Singapore: Springer Singapore, 2020, pp. 17–47. ISBN: 978-981-15-5069-0. DOI: 10.1007/978-981-15-5069-0_2. URL: https://doi.org/10.1007/978-981-15-5069-0_2.

[3] A. W. Crapo, L. B. Waisel, W. A. Wallace, and T. R. Willemain. “Visualization and the process of modeling: a cognitive-theoretic view”. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2000, pp. 218–226.

[4] H. Mei, H. Guan, C. Xin, X. Wen, and W. Chen. “Datav: Data visualization on large high-resolution displays”. In: *Visual Informatics* 4.3 (2020), pp. 12–23.

[5] Y. Hrabovskyi, N. Brynza, and O. Vilkhivska. “Development of Information Visualization Methods for Use in Multimedia Applications”. In: *EUREKA: Physics and Engineering* 1 (2020), pp. 3–17.

[6] W. S. Cleveland. *Visualizing data*. Hobart Press, 1993.

[7] J. J. van Wijk and R. van Liere. “Hyperslice”. In: *Proceedings Visualization’93*. IEEE. 1993, pp. 119–125.

[8] P. J. Huber. “Projection pursuit”. In: *The annals of Statistics* (1985), pp. 435–475.

[9] M. Petrakos, W. Dicarlo, and I. Kanellopoulos. “Projection pursuit and a VR environment for visualization of remotely sensed data”. In: *IEEE 1999 International Geoscience and Remote Sensing Symposium. IGARSS’99 (Cat. No. 99CH36293)*. Vol. 5. IEEE. 1999, pp. 2498–2500.

[10] G. W. Furnas and A. Buja. “Prosection views: Dimensional inference through sections and projections”. In: *Journal of Computational and Graphical Statistics* 3.4 (1994), pp. 323–353.

[11] A. Inselberg. “The plane with parallel coordinates”. In: *The visual computer* 1.2 (1985), pp. 69–91.

[12] J. Heinrich and D. Weiskopf. “State of the Art of Parallel Coordinates.” In: *Eurographics (STARs)*. 2013, pp. 95–116.

[13] *the UC Irvine Machine Learning Repository*. https://archive.ics.uci.edu/ml/index.php.

[14] C. Brunsdon, A. S. Fotheringham, and M. Charlton. “An investigation of methods for visualising highly multivariate datasets”. In: *Case Studies of Visualization in the Social Sciences* (1998), pp. 55–80.

[15] K. Aldwib, A. A. Bidgoli, S. Rahnamayan, and A. Ibrahim. “Proposing a Pareto-VIKOR Ranking Method for Enhancing Parallel Coordinates Visualization”. In: *2019 14th International Conference on Computer Science & Education (ICCSE)*. IEEE. 2019, pp. 895–902.

[16] K. Aldwib, S. Rahnamayan, and A. Ibrahim. “Enhancing Parallel Coordinates Visualization Using Genetic Algorithm with Smart Mutation”. In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2020, pp. 3746–3752.

[17] J. Schmidt. “Usage of Visualization Techniques in Data Science Workflows.” In: *VISIGRAPP (3: IVAPP)*. 2020, pp. 309–316.

[18] S. Wang, Y. Yang, J.-S. Chang, and F.-P. Lin. “Using penalized regression with parallel coordinates for visualization of significance in high dimensional data”. In: *Int. J. Adv. Comput. Sci. Appl* 4.10 (2013).

[19] M. Khan and S. S. Khan. “Data and information visualization methods, and interactive mechanisms: A survey”. In: *International Journal of Computer Applications* 34.1 (2011), pp. 1–14.

[20] G. E. Hinton and S. T. Roweis. “Stochastic neighbor embedding”. In: *Advances in neural information processing systems*. 2003, pp. 857–864.

[21] L. v. d. Maaten and G. Hinton. “Visualizing data using t-SNE”. In: *Journal of machine learning research* 9.Nov (2008), pp. 2579–2605.

[22] C. Shalizi. “Nonlinear Dimensionality Reduction I: Local Linear Embedding”. In: *Data Mining Class* (2009).

[23] *LLE Code Page*. https://cs.nyu.edu/ roweis/lle/code.html.

[24] D. De Ridder, O. Kouropteva, O. Okun, M. Pietikäinen, and R. P. Duin. “Supervised locally linear embedding”. In: *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003*. Springer, 2003, pp. 333–341.

[25] S.-q. Zhang. “Enhanced supervised locally linear embedding”. In: *Pattern Recognition Letters* 30.13 (2009), pp. 1208–1218.

[26] T. Kohonen. *Self-organizing maps*. Vol. 30. Springer Science & Business Media, 2012.

[27] A. Inselberg. *Parallel coordinates*. Springer, 2009.

[28] A. Dasgupta and R. Kosara. “Pargnostics: Screen-space metrics for parallel coordinates”. In: *IEEE Transactions on Visualization & Computer Graphics* 6 (2010), pp. 1017–1026.

[29] L. Zhen, M. Li, R. Cheng, D. Peng, and X. Yao. “Adjusting Parallel Coordinates for Investigating Multi-Objective Search”. In: *Asia-Pacific Conference on Simulated Evolution and Learning*. 2017, pp. 224–235.

[30] W. Peng, M. O. Ward, and E. A. Rundensteiner. “Clutter reduction in multi-dimensional data visualization using dimension reordering”. In: *Information Visualization, INFOVIS 2004. IEEE Symposium on*. IEEE. 2004, pp. 89–96.

[31] L. F. Lu. “Optimization of parallel coordinates for visual analytics”. PhD thesis. 2016.

[32] H. Makwana, S. Tanwani, and S. Jain. “Axes re-ordering in parallel coordinate for pattern optimization”. In: *International Journal of Computer Applications* 40.13 (2012), pp. 43–48.

[33] A. Nabil, K. M. Mohamed, and Y. M. Kamal. “Enhancing Visualization of Multidimensional Data by Ordering Parallel Coordinates Axes”. In: ().

[34] M. S. Akbar and B. Gabrys. “Data Analytics Enhanced Data Visualization and Interrogation with Parallel Coordinates Plots”. In: *2018 26th International Conference on Systems Engineering (ICSEng)*. IEEE. 2018, pp. 1–7.

[35] M. Blumenschein, X. Zhang, D. Pomerenke, D. A. Keim, and J. Fuchs. “Evaluating Reordering Strategies for Cluster Identification in Parallel Coordinates”. In: *Computer Graphics Forum*. Vol. 39. 3. Wiley Online Library. 2020, pp. 537–549.

[36] W. J. Raseman, J. Jacobson, and J. R. Kasprzyk. “An open source, interactive parallel coordinates library for multi-objective decision making”. In: *Environmental Modelling & Software* (2019).

[37] J. Yang, W. Peng, M. O. Ward, and E. A. Rundensteiner. “Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets”. In: *IEEE Symposium on Information Visualization 2003 (IEEE Cat. No. 03TH8714)*. IEEE. 2003, pp. 105–112.

[38] C. Tong, J. Zhang, A. Chowdhury, and S. G. Trost. “An interactive visualization tool for sensor-based physical activity data analysis”. In: *Proceedings of the Australasian Computer Science Week Multiconference*. 2019, pp. 1–4.

[39] R. C. Roberts, R. S. Laramee, G. A. Smith, P. Brookes, and T. D’Cruze. “Smart brushing for parallel coordinates”. In: *IEEE transactions on visualization and computer graphics* 25.3 (2018), pp. 1575–1590.

[40] H. Janetzko, M. Stein, D. Sacha, and T. Schreck. “Enhancing parallel coordinates: Statistical visualizations for analyzing soccer data”. In: *Electronic Imaging* 2016.1 (2016), pp. 1–8.

[41] R. G. Raidou, M. Eisemann, M. Breeuwer, E. Eisemann, and A. Vilanova. “Orientation-enhanced parallel coordinate plots”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2015), pp. 589–598.

[42] G. Palmas, M. Bachynskyi, A. Oulasvirta, H. P. Seidel, and T. Weinkauf. “An edge-bundling layout for interactive parallel coordinates”. In: *2014 IEEE Pacific visualization symposium*. IEEE. 2014, pp. 57–64.

[43] M. Graham and J. Kennedy. “Using curves to enhance parallel coordinate visualisations”. In: *Proceedings on Seventh International Conference on Information Visualization, 2003. IV 2003.* IEEE. 2003, pp. 10–16.

[44] H. Zhou, W. Cui, H. Qu, Y. Wu, X. Yuan, and W. Zhuo. “Splatting the lines in parallel coordinates”. In: *Computer Graphics Forum*. Vol. 28. 3. Wiley Online Library. 2009, pp. 759–766.

[45] L. Lu, W. Wang, and Z. Tan. “Double-Arc Parallel Coordinates and its Axes re-Ordering Methods”. In: *Mobile Networks and Applications* (2020), pp. 1–16.

[46] D. J. Walker, R. Everson, and J. E. Fieldsend. “Visualizing mutually nondominating solution sets in many-objective optimization”. In: *IEEE Transactions on Evolutionary Computation* 17.2 (2013), pp. 165–184.

[47] M. Angelini, G. Blasilli, S. Lenti, A. Palleschi, and G. Santucci. “Towards Enhancing RadViz Analysis and Interpretation”. In: *2019 IEEE Visualization Conference (VIS)*. IEEE. 2019, pp. 226–230.

[48] J. Yin, H. Shi, X. Zhou, L. Jin, S. Li, and Y. Zhang. “Multi-dimensional data visualisation method based on convex-corrected Radviz”. In: *International Journal of Computer Applications in Technology* 63.1-2 (2020), pp. 114–124.

[49] A. Ibrahim, S. Rahnamayan, M. V. Martin, and K. Deb. "3D-RadVis: Visualization of Pareto front in many-objective optimization". In: *Evolutionary Computation (CEC), 2016 IEEE Congress on*. IEEE, pp. 736–745.

[50] C. C. Coello. "Evolutionary multi-objective optimization: a historical view of the field". In: *IEEE computational intelligence magazine* 1.1 (2006), pp. 28–36.

[51] R. T. Marler and J. S. Arora. "Survey of multi-objective optimization methods for engineering". In: *Structural and multidisciplinary optimization* 26.6 (2004), pp. 369–395.

[52] M. K. Sayadi, M. Heydari, and K. Shahanaghi. "Extension of VIKOR method for decision making problem with interval numbers". In: *Applied Mathematical Modelling* 33.5 (2009), pp. 2257–2262.

[53] J. Lee Rodgers and W. A. Nicewander. "Thirteen ways to look at the correlation coefficient". In: *The American Statistician* 42.1 (1988), pp. 59–66.

[54] S. Karatsiolis and C. N. Schizas. "Region based Support Vector Machine algorithm for medical diagnosis on Pima Indian Diabetes dataset". In: *2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE)*. IEEE. 2012, pp. 139–144.

[55] D. Dheeru and E. Karra Taniskidou. *UCI Machine Learning Repository*. 2017.

[56] A. Abraham and L. Jain. "Evolutionary multiobjective optimization". In: *Evolutionary Multiobjective Optimization*. Springer, 2005, pp. 1–6.

[57] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao. "A benchmark test suite for evolutionary many-objective optimization". In: *Complex & Intelligent Systems* 3.1 (2017), pp. 67–81.

[58] J. H. Holland. "Genetic algorithms". In: *Scientific american* 267.1 (1992), pp. 66–73.

[59] A. Hussain, Y. S. Muhammad, M Nauman Sajid, I. Hussain, A. Mohamd Shoukry, and S. Gani. "Genetic algorithm for traveling salesman problem with modified cycle crossover operator". In: *Computational intelligence and neuroscience* 2017 (2017).

[60] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. "Modeling wine preferences by data mining from physicochemical properties". In: *Decision Support Systems* 47.4 (2009), pp. 547–553.

[61] *Mulan: A Java Library for Multi-Label Learning*. http://mulan.sourceforge.net/datasets-mlc.html.

[62] D. J. Sheskin. *Handbook of parametric and nonparametric statistical procedures*. crc Press, 2020.

[63] N. Srinivas and K. Deb. "Muiltiobjective optimization using nondominated sorting in genetic algorithms". In: *Evolutionary computation* 2.3 (1994), pp. 221–248.

[64] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Vol. 16. John Wiley & Sons, 2001.

[65] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE transactions on evolutionary computation* 6.2 (2002), pp. 182–197.

[66] Y. Yuan, H. Xu, and B. Wang. "An improved NSGA-III procedure for evolutionary many-objective optimization". In: *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. 2014, pp. 661–668.

[67] K. Deb and H. Jain. "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints". In: *IEEE transactions on evolutionary computation* 18.4 (2013), pp. 577–601.

[68] V. Chankong and Y. Y. Haimes. *Multiobjective decision making: theory and methodology*. Courier Dover Publications, 2008.

[69] I. Das and J. E. Dennis. “Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems”. In: *SIAM journal on optimization* 8.3 (1998), pp. 631–657.

[70] E. Zitzler and L. Thiele. “Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach”. In: *IEEE transactions on Evolutionary Computation* 3.4 (1999), pp. 257–271.

[71] M. Ward, G. Grinstein, and D. Keim. “Interactive Data Visualization: Foundations”. In: *Techniques, and Applications, AK Peters, Ltd., Natick, MA* (2010).

[72] J. Hartigan and M. Wong. “A K-means clustering algorithm. Applied Statistics”. In: *Applied Statistics* 28.100-108 (1979), p. 51.

[73] Z. Ansari, M. Azeem, W. Ahmed, and A. V. Babu. “Quantitative evaluation of performance and validity indices for clustering the web navigational sessions”. In: *arXiv preprint arXiv:1507.03340* (2015).

[74] J. Baarsch and M. E. Celebi. “Investigation of internal validity measures for K-means clustering”. In: *Proceedings of the international multiconference of engineers and computer scientists*. Vol. 1. sn. 2012, pp. 14–16.

[75] J. C. Dunn. “Well-separated clusters and optimal fuzzy partitions”. In: *Journal of cybernetics* 4.1 (1974), pp. 95–104.