Graph-Based Tools for Parallel Coordinates

Kai Lun Chung and Wei Zhuo

The Hong Kong University of Science and Technology
cspeter@cse.ust.hk|ee_zwxaa@stu.ust.hk

Abstract. Parallel coordinates is a fundamental visualization technique in multivariate data visualization. Visual clutter is one of the inherent weaknesses using parallel coordinates. In this paper, we present two graph-based tools, the Selection Graph and the Relation Graph, to reduce the visual clutter. The Selection Graph is a brushing tool which helps users highlight the regions of interest. The Relation Graph organizes clusters in a structural manner, providing an intuitive interface for users to explore relations among clusters. Both tools neither distort nor filter the underlying data in parallel coordinates. The experiments on several real datasets demonstrate the effectiveness of our tools.

1 Introduction

Parallel coordinates is a well-established visualization technique first proposed by Inselberg [1, 2]. It is a scalable framework in the sense that the increase of the dimensions corresponds to the addition of extra axes. However, the visual clutter caused by an excessive overlaying of polylines limits the effectiveness of parallel coordinates in visualizing a dense data set. This can produce misleading perceptions, and cause extra difficulties for users to locate and select the regions of interest.

Visual clustering, filtering, and axis reordering are three common methods to reduce clutters in parallel coordinates. Visual clustering first detects polylines in the same cluster and then edge distortion is applied to improve the space usage [3]. Some filtering techniques remove part of data in a pre-processing, however, interesting patterns such as outliers may also be filtered out. Axis reordering computes the best axis order with minimal clutters, but it is very likely that the best axis order still leads to a unsatisfactory and cluttered display.

In this paper, we introduce two graph-based tools as abstractions of the parallel coordinates—the Selection Graph and the Relation Graph. The Selection Graph is a brushing tool which helps users to select the regions of interest. It can effectively identify some desired patterns in a cluttered display. The Relation Graph is used to explore the inter-cluster relationship by organizing the clusters in a force-based layout. A dual domain interaction is provided between the tools and the parallel coordinates, where all data can be authentically presented without distortion and filtering.

This paper is organized as follows: We summarize some previous work on visual clustering, regions of interest selection, and analytic tools in Section 2.

Section 3 briefly introduces our preprocessing step for the data. Section 4 and Section 5 present the Selection Graph and the Relation Graph respectively. The experiment results are provided in Section 6 and the conclusion is drawn in Section 7.

2 Related work

Visual clustering Visual clustering has been proposed for parallel coordinates to reduce visual clutter. Hong et al. [3] proposed a novel framework, in which edge clutters are reduced by optimizing the arrangement of the curved edges. In their work, edges are not discarded from the display and the relative positions of edges are kept after visual clustering. McDonnell et al. [4] developed a suite of artistic rendering techniques to improve parallel coordinates. Their work uses shading to illustrate the local line density, and silhouettes to aid human eyes in distinguishing overlapping clusters.

Regions of interest selection Regions of interest selection is usually performed by brushing. Standard brushing [5] selects a region on an axis, then the selected polylines are highlighted. Wong and Bergeron [6] developed a wavelet brushing technique for high dimensional data sets. The brushed data is displayed in a higher resolution than the remaining data. The construction of the proposed brushing method is based on the idea of wavelet approximations.

Fua et al. [7] proposed a structure-based brushing to help users select regions of interest through navigating the hierarchies. Hauser et al. [8] proposed angular brushing, which is able to highlight features depending on two data dimensions. Another tool called smooth brushing is also proposed to let users specify non-binary degree of interest functions.

Analytic tools for parallel coordinates The most commonly used analytic technique is to embed scatterplots on top of the parallel coordinates. The structure-based brushing [7] can be regarded as an analytic tool for parallel coordinates. A hierarchical tree structure is built to select the regions of interest from the parallel coordinates in different levels of detail.

Ericson et al. [9] developed a tool named as Visual Data Mining Display (VDMD) to display statistical measures of data. Interactions with the parallel coordinates are reflected as the changes in the statistics display. The interaction provides a way to explore the data and the find structures hidden in the parallel coordinates. Qu et al. [10] proposed the weighted complete graph to help reorder axes in parallel coordinates.The weighted complete graph is a visualization tool which acts as a high-level guide map to display the relationship between dimensions. Based on the structure of the weighted complete graph, users are able to generate an optimized axis order for parallel coordinates interactively or automatically.

3 Pre-Processing

In the pre-processing step, we first detect clusters in the given data set. Numerous clustering algorithms can be applied to detect clusters. The clustering process provides an abstraction of data without filtering out potentially important information such as outliers. Throughout the experiments in this paper, the K-means clustering algorithm [11] is adopted. Although the algorithm is not guaranteed to return a global optimum, it is extremely fast and has been widely adopted in many research areas. The clustering algorithm is not the focus of this paper and any advanced clustering algorithms can be used together with our method. For example, extensions such as multiple-pass K-means can easily improve the results.

4 Selection Graph

The selection graph presented in this paper is in fact a brushing tool. In the preprocessing step, various clusters can be detected. The selection graph organizes those clusters detected between each pair of adjacent axes into a node-link diagram. The nodes and edges of the Selection Graph are used to encode useful information related to the clusters. We design an encoding scheme to abstract the overwhelming information related to clusters without filtering out any potentially interesting patterns such as outliers. Set operations are further defined on top of the graph.

4.1 Construction of Selection Graph

In the preprocessing step, clusters are detected between pairs of adjacent axes. In standard brushing techniques, users heavily rely on the information between every adjacent pair of axes. For example, standard brushing selects a subset of interesting polylines from an axis, while angular brushing selects regions of interest by specifying the range of angle between a pair of adjacent dimensions. Thus, local information between adjacent pairs of axes is more useful for the selection of regions of interest. For each cluster, the number of polylines, the average slopes, and the correlation among clusters are first quantified and recorded. The extracted information is used to construct the selection graph based on the encoding scheme presented in the next subsection.

4.2 Encoding Scheme of Selection Graph

The encoding scheme of the selection graph is summarized in Table 4.2.

Each node represents a local cluster detected by the K-means clustering algorithm. The node size represents the number of lines in this local cluster. A larger node can be selected to explore a larger cluster, while a smaller node can be selected to detect outliers. For each local cluster, an average slope is computed. A rainbow color scheme is used to represent the slope, with blue encoding the

Visual channel	**Meaning**
Number of nodes	Number of clusters between two adjacent axes
Node size	Number of lines in a cluster
Node color	Average slope of a cluster
Node position	Lying on a circle, sorted by the average slopes
Links	Correlation coefficient between two clusters

Table 1. The encoding scheme of the Selection Graph

minimum possible slope and red encoding the maximum possible slope. Nodes are sorted according to slopes, and placed in a circular order.

Inspired by the weighted complete graph [10], we compute the weight of every pairs of clusters using the correlation coefficient in the selection graph. The correlation between two dimensions i and j is defined as

$$Corr(X_i, Y_j) = \frac{(X_i - \overline{X}_i)(X_j - \overline{X}_j)^T}{((X_i - \overline{X}_i)(X_i - \overline{X}_i)^T)^{\frac{1}{2}}((X_j - \overline{X}_j)(X_j - \overline{X}_j)^T)^{\frac{1}{2}}} \quad (1)$$

where X_i and X_j are values in dimension i and j respectively. A slider is provided to set the threshold to filter out links with a relative weak correlation. The red links and the blue links encode the positive and the negative correlation respectively.

4.3 Set operations

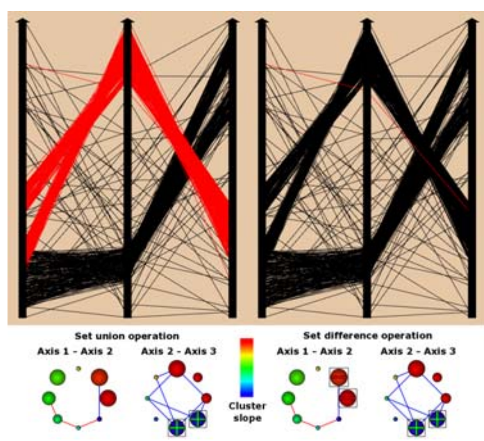


Fig. 1. The Selection Graph: (Left) An example of a set union operation. Lines from the second interval with negative slopes are selected. (Right) An example of a set difference operation. Lines from the first interval with positive slopes are subtracted.

We further provide two kinds of set operations, namely set union operation and set difference operation, for the selection graph.

In the node selection process, each node can be in three separate states: unselected, selected by set union, or selected by set difference. When a node is selected either by set union or set difference, all the polylines within the selected interval are traced. Figure 1 illustrates an example of applying a set union operation and a set difference operation separately in the selection graph.

The set operations are resolved in the following manner. The first step is to collect a set of polylines defined by the set union nodes. These polylines are first combined together using the union operators. After that, a sequence of the set difference operations defined by the set difference nodes is applied to remove polylines from the current selection. For example, in the right handed side of Figure 1, two set union nodes and set difference nodes are selected. No matter which order of selection, the polylines defined by the two set union nodes are first integrated, and then the polylines are removed from the set difference nodes.

The above resolving scheme avoids the ambiguity of different orders of selection. If the resolving scheme is dependent on the selection order, different results can be generated by the same set of selections of nodes.

5 Relation Graph

In parallel coordinates, edges from different clusters may occlude each other, which results in visual clutter. Visual clutter makes it hard for users to analyze relations among clusters. On the other hand, clustering algorithms usually detect clusters with different levels of detail. For example, structure-based brushing [7] provides users an intuitive interface to explore the levels of detail. However, it is difficult to visualize the relations among clusters in the same level of details.

The relation graph is proposed to visualize the inter-cluster relationship. The star glyph representation is used to encode the correlation information for each cluster. A force-based layout algorithm is used to reveal the relations among clusters in the same level of detail.

5.1 Construction of Relation Graph

In the preprocessing of the relation graph, the K-means clustering algorithm is adopted. Unlike the selection graph, clustering is done in any user chosen dimensions because the overall clustering results are much informative to explore than local clustering results. We pre-compute a sequence of clustering results. The number of clusters is used to control different levels of detail. Throughout the experiments, the range is selected from 2 to 32, and configurable through the user interface.

5.2 Encoding Scheme of Relation Graph

The encoding scheme of the relation graph is summarized in Table 5.2.

Visual channel	Meaning
Outer polygon	The number of edges is equal to the number of dimensions in the parallel coordinates. Each edge provides an attractive force for every node in the Relation Graph.
Node shape	Star glyph representation is used. Each dimension represents the correlation between two adjacent axes
Node size	Number of lines among all clusters
Node position	Computed by a force-based layout model

Table 2. The encoding scheme of Relation Graph

The star glyph pattern has been used to represent a polyline in parallel coordinates [5]. We further exploit star glyph to encode the correlation coefficient in each cluster. The definition of the correlation coefficient is defined in (1). The correlation coefficient is used to represent the shape of the glyph. The size of glyph is used to encode the relative size of a cluster. The maximum and the minimum radius of nodes can be configured by users. The largest node represents the cluster with the most data points.

5.3 Force-based layout

The complexity to compare a group of glyphs grows as the number of glyph increases. The straightforward solution is to put all glyphs in a sequential order, or layout the glyphs in a two dimensional grid. However, it is difficult to compare a pair of glyphs which are widely separated.

A comparison is usually carried out among glyphs having similar shapes. We thus introduce a force-based layout to glyphs sharing similar structures. Initially, all nodes are placed at the centroid of the outer polygon. For each node, an attractive force is computed between every edges of the outer polygon. The resulting force acting upon the node is used to compute the layout. Figure 2 illustrates the concept of the force-based layout for a glyph.

Position ambiguity may happen in the force-based layout. Two clusters having different correlation patterns may be positioned in the same position. In this case, the relative size and shape of clusters are used to resolve ambiguity. The cluster with a larger size is placed behind the cluster with a smaller size.

6 Experimental results

We have conducted experiments on a Dell computer with Intel Core 2 Duo CPUs and 1GB memory. The experiments are designed to demonstrate the effectiveness of the selection graph and the relation graph respectively.

6.1 Selection Graph

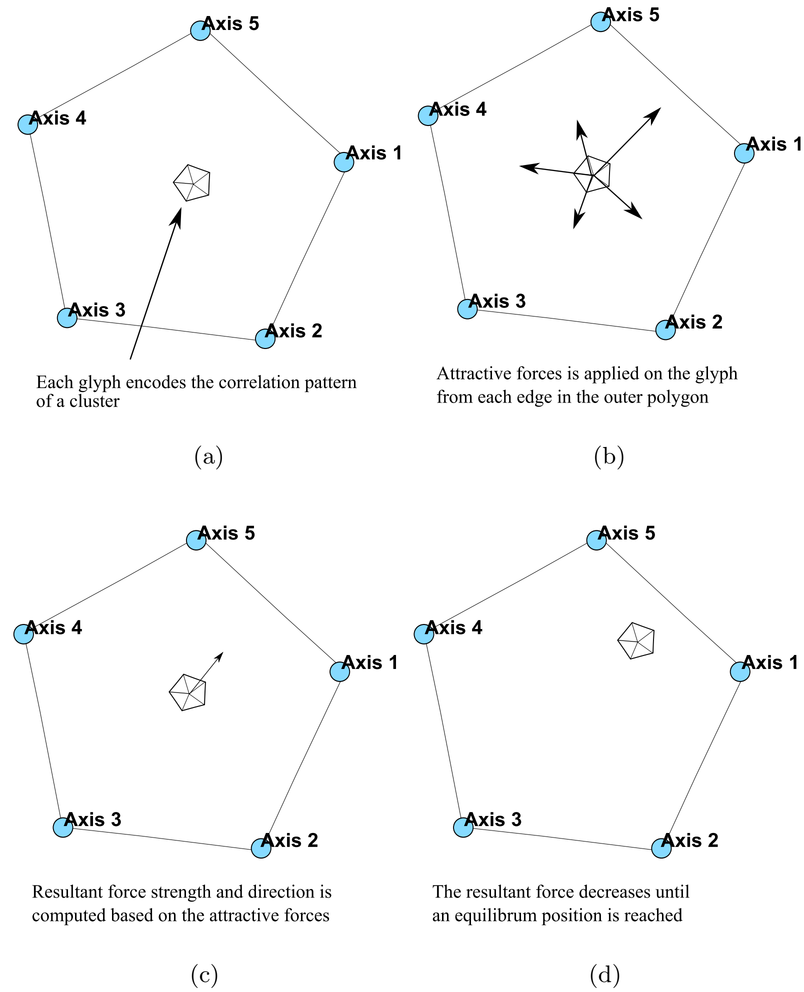
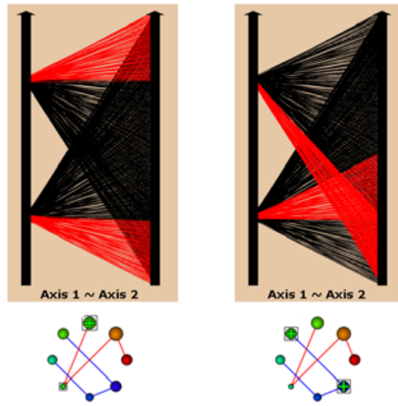


Fig. 2. An example to illustrate the force-based layout of a cluster in the Relation Graph: (a) Initial position of a glyph representing a cluster; (b) Attractive forces modeled by correlation strength ; (c) Resultant force computation; (d) Equilibrium position of the glyph



(a) (b)

Fig. 3. Usage of links on the Selection Graph: (a) Red links are used to select local clusters sharing similar patterns, but may be far apart; (b) Blue links are used to select local clusters crossing each other.

Usage of links: Links in the selection graph are helpful for users to select patterns, as demonstrated in Figure 3. In Figure 3(a), two local clusters which have similar patterns but are separated apart can be selected by choosing two nodes connected by a red link. In Figure 3(b), two local clusters which strongly crossed over each other can be selected by choosing two nodes connected by a blue link.

Usage of set operations: Figure 4 illustrates an example of applying a combination of set operations on top of the selection graph. Users can easily isolate the outlier from the cluttered parallel coordinates display, compared to standard brushing tools by which outliers are difficult to isolate from the cluttered parallel coordinates.

6.2 Relation Graph

Figure 5 demonstrates the usage of the relation graph with a car data set. The experiment uses the relation graph to explore the real data set with 7 variables and 392 data items. The number of clusters by the K-means clustering algorithm is chosen as six. After cluster detection, star glyph nodes are generated. The force-based layout algorithm is applied to layout the nodes.

Figure 5(a) shows the resulting layout. The clusters on the boundary indicate that there is a biased local correlation pattern. The relative size of the selected cluster is very large, indicating that many polylines exhibit such a correlation

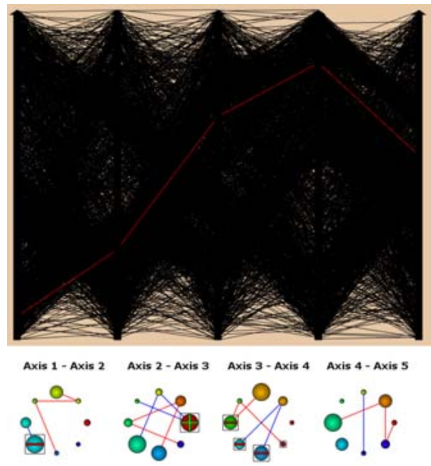


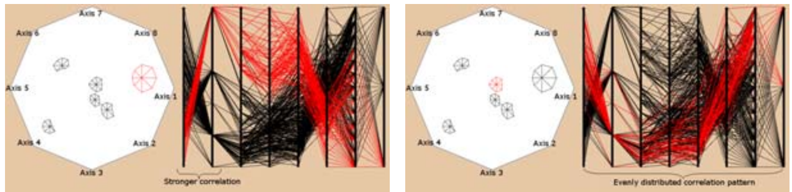
Fig. 4. A synthesized data set with more that 1500 data points and four clusters plus some random noises. After a sequence of set union and set difference operations, an outlier is identified from the cluttered display.

pattern. Figure 5(b) shows a cluster located at the center with an evenly distributed correlation pattern. There are two clusters nearby with different shapes, indicating that these two clusters also have evenly distributed correlation pattern, but their detailed correlation patterns are different.

7 Conclusion and future work

Parallel coordinates have been widely used as important tools for multivariate data visualization. However, visual clutter in parallel coordinates significantly limits its usage for visualizing large data sets. Inspired by some previous graph-based visual analytic tools, we proposed the selection graph and the relation graph to tackle the clutter problem in parallel coordinates. Our approaches do not alter the data in the parallel coordinates. Instead, they are developed on top of the parallel coordinates to enhance its usefulness in a cluttered display.

There are several avenues for the future work. The visualization techniques used in our work are graph-based and static. However, we notice that animations can be exploited to enhance the patterns and to encode more information in the underlying multivariate data set. Animated transitions can be used to visualize the changes in the levels of detail in the relation graph. Based on the relation graph and selection graph, effective animation sequences can be generated to visualize the changes among clusters in different levels of detail. Clutter reduction with animation is worth further study.



(a) (b)

Fig. 5. The experiment uses the Relation Graph to explore the real data set. The data is from http://lib.stat.cmu.edu/datasets/cars.data : (a) Clusters strongly biased towards a local correlation pattern; (b) Clusters with an evenly distributed correlation pattern.

References

1. Inselberg, A.: The plane with parallel coordinates. Journal The Visual Computer **1** (1985) 69–91
2. Inselberg, A., Dimsdale, B.: Parallel coordinates: a tool for visualizing multi-dimensional geometry. In: VIS '90: Proceedings of the 1st conference on Visualization '90, Los Alamitos, CA, USA, IEEE Computer Society Press (1990) 361–378
3. Zhou, H., Yuan, X., Qu, H., Cui, W., Chen, B.: Visual clustering in parallel coordinates. Computer Graphics Forum **27** (2008)
4. McDonnell, K., Mueller, K.: Illustrative parallel coordinates. Computer Graphics Forum (Special Issue Eurovis 2008) **27** (2008) 1031–1027
5. Ward, M.O.: Xmdvtool: integrating multiple methods for visualizing multivariate data. In: VIS '94: Proceedings of the conference on Visualization '94, Los Alamitos, CA, USA, IEEE Computer Society Press (1994) 326–333
6. Wong, P.C., Bergeron, R.D.: Multiresolution multidimensional wavelet brushing. In: VIS '96: Proceedings of the 7th conference on Visualization '96, Los Alamitos, CA, USA, IEEE Computer Society Press (1996) 141–ff.
7. Fua, Y.H., Ward, M.O., Rundensteiner, E.A.: Structure-based brushes: A mechanism for navigating hierarchically organized data and information spaces. IEEE Transactions on Visualization and Computer Graphics **6** (2000) 150–159
8. Hauser, H., Ledermann, F., Doleisch, H.: Angular brushing of extended parallel coordinates. In: INFOVIS '02: Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02), Washington, DC, USA, IEEE Computer Society (2002) 127
9. Ericson, D., Johansson, J., Cooper, M.: Visual data analysis using tracked statistical measures within parallel coordinate representations. In: CMV '05: Proceedings of the Coordinated and Multiple Views in Exploratory Visualization, Washington, DC, USA, IEEE Computer Society (2005) 42–53
10. Qu, H., Chan, W.Y., Xu, A., Chung, K.L., Guo, P., Lau, K.H.: Visual analysis of the air pollution problem in hong kong. IEEE Transactions on Visualization and Computer Graphics **13** (2007) 1408–1415
11. Hartigan, J.A., Wong, M.A.: A K-means clustering algorithm. Applied Statistics **28** (1979) 100–108