Visualizing Clusters in Parallel Coordinates for Visual Knowledge Discovery

Yang Xiang[1], David Fuhry[2], Ruoming Jin[3], Ye Zhao[3], and Kun Huang[1]

[1] Department of Biomedical Informatics, The Ohio State University, Columbus, OH 43210, USA
{yang.xiang,kun.huang}@osumc.edu
[2] Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210, USA
fuhry@cse.ohio-state.edu
[3] Department of Computer Science, Kent State University, Kent, OH 44242, USA
{jin,zhao}@cs.kent.edu

Abstract. Parallel coordinates is frequently used to visualize multi-dimensional data. In this paper, we are interested in how to effectively visualize clusters of multi-dimensional data in parallel coordinates for the purpose of facilitating knowledge discovery. In particular, we would like to efficiently find a good order of coordinates for different emphases on visual knowledge discovery. To solve this problem, we link it to the metric-space Hamiltonian path problem by defining the cost between every pair of coordinates as the number of inter-cluster or intra-cluster crossings. This definition connects to various efficient solutions and leads to very fast algorithms. In addition, to better observe cluster interactions, we also propose to shape clusters smoothly by an energy reduction model which provides both macro and micro view of clusters.

Keywords: Multi-dimensional Data Visualization, Parallel Coordinates, Cluster, knowledge discovery, Graph Theory, Metric Space, Metric Hamiltonian path problem.

1 Introduction

Today the infusion of data from every facet of our society, through documenting, sensing, digitalizing and computing, challenges scientists, analysts and users with its typical massive size and high dimension. Data mining and visualization are two important areas in analyzing and understanding the data. The role of data mining is to discover hidden patterns of the data. In particular, various clustering techniques (see [19] for a review) have been proposed to reveal the structures of these data and support exploratory data analysis. By contrast, the role of visualization is to present the data in a clear and understandable manner for people. Many visualization techniques have been developed to facilitate exploratory analysis and analytical reasoning through the use of (interactive) visual interfaces.

P.-N. Tan et al. (Eds.): PAKDD 2012, Part I, LNAI 7301, pp. 505–516, 2012.

However, despite these efforts, current research is far from perfect in integrating these two endeavors in a close and uniform fashion. Given the discovered cluster structures from the data, how can we visualize them and provide users better insight of the data? How can visualization techniques help reveal and expose underlying structures of the data? Those research questions are clearly very critical for us to meet the challenges of the "data explosion". In this paper, we address those questions by developing a novel visualization model for visualizing discovered clusters in large and multivariate datasets. Our goal is to efficiently provide users different views of discovered clusters as well as preserve the details of these clusters to the maximal extent possible. Among many visualization techniques for multidimensional data [33], parallel coordinates is one of the most elegant yet simple tools, and we select it as the visualization platform for our proposed algorithms.

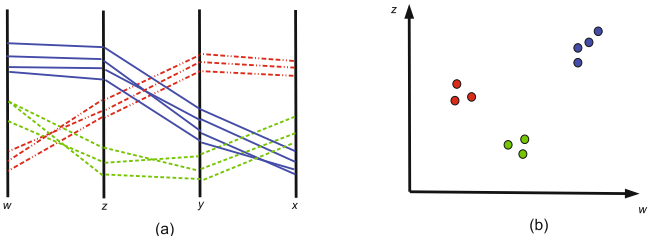


Fig. 1. (a) Data visualization with parallel coordinates w, x, y, z; (b) Data projection on wz plane

Parallel coordinates, which transforms multivariate data into 2D polylines (or 'lines' for short), has been widely used in many information visualization applications [18,28] as well as data mining [35]. Figure 1(a) is an example of visualizing 4-dimensional data by parallel coordinates. Wegman [30] shows Parallel Coordinates can be used to effectively reveal data correlation as well as cluster interaction. For instance, Figure 1(b) is the projection of three clusters in Figure 1(a) on the wz plane. Readers can easily observe what it implies for two clusters that are generally crossing over each other between two coordinates (e.g. wz coordinates in Figure 1), or generally parallel to each other. Such observation leads to important knowledge discovery. Using an example in [30] as an illustration, let us imagine we are comparing one group of heavy cars with another group of light cars, on weight, displacement, mileage, gear ratio, and price. A visualization of these two clusters on gear ratio and weight would show these two clusters generally cross each other, which implies that heavy cars would tend to have a large engine but a lower gear ratio, while light cars are just the reverse.

As there is a factorial number of ways to order the parallel coordinates for visualizing different aspects of the data, a major challenge arises: How to efficiently determine a good order of coordinates (i.e., columns or dimensions) for a specific knowledge discovery purpose? Such order is traditionally pre-determined,

and made flexible in some systems by allowing user adjustment. For data sets with many dimensions, this will impose unexpected challenge to end users, while they may have inadequate knowledge and experiences. Moreover, data clusters from aggregation and abstraction are even harder to be illustrated along multiple coordinates, together with many polylines.

To address these challenges, in this paper, we visualize clusters in parallel coordinates for visual knowledge discovery using a novel dimension ordering approach which is further refined by an energy reduction model.

2 Related Works

Parallel Coordinates for Clusters. Wegman [30] promotes parallel coordinates visualization in the aspects of geometry, statistics and graphics, which has been widely applied in information visualization [28,25]. For visualizing clustered data sets, many approaches have been conducted using parallel coordinates [20,23,3]. In particular, instead of visualizing each individual data item as a polyline, each cluster pattern is visualized as a fuzzy stripe [13,20,3]. Fua et al. [9] visualize clusters by variable-width opacity bands, faded from a dense middle to transparent edges. Such visualization focuses on the global pattern of clusters, but the general shape of a cluster might be adversely affected by a small number of outliers inside a cluster. In comparison, instead of displaying a shape profile of individual clusters, our method seeks to keep the line structures while highlighting clusters and their relationships, by seeking good orders of coordinates as well as shaping them smoothly by a quadratic energy reduction model which extends the linear system proposed in [36].

Dimension Ordering. The dimension ordering and permutation problem is naturally associated with parallel coordinate visualization. It is discussed in the early paper by Wegman [30] and the subsequent work by Hurley and Oldford [17,16]. In [30],Wegman points out the problem and gives a basic solution on how to enumerate the minimum number permutations such that every pair of coordinates can be visualized in at least one of the permutations. However, it is rather inefficient to display parallel coordinates corresponding to all these permutations. The grand tour animates a static display in order to examine the data from continuous different views [29,32,31]. The method is effective by seeking solution to temporal exploration for computational complex tasks such as manifesting outliers and clusters. Ankerst et al. [1] propose to rearrange dimensions such that dimensions showing a similar behavior are positioned next to each other. Peng et al. [27] try to find a dimension ordering that can minimize the "clutter measure", which is defined as the ratio of outliers to total data points. Since permutation related problems are mostly NP-hard, the existing work [1,27,34] primarily relies on heuristic algorithms to get a quick solution.

Ellis and Dix [8] use line crossings to reduce clutter. Dasgupta and Kosara [7] recently use number of crossings as an indication of clutter between two adjacent coordinates, and apply a simple Branch-and-Bound optimization for dimension

ordering. Hurley [15] uses crossings to study the correlation between two dimensions of a dataset as well as reduce clutters. Different from them, we define the crossing as an order change between a pair of inter-cluster items (or intra-cluster items, depending on the visualization focus) on two adjacent coordinates. Our definitions lead to an effective and efficient solution to study cluster interactions on parallel coordinates for visual knowledge discovery.

3 Dimension Ordering for Knowledge Discovery

Compared with the method of projecting data into a two dimensional plane for analysis (e.g. Figure 1(b)), an n-dimensional dataset visualized by n parallel coordinates is more efficient for data analysis as it displays data in $n-1$ pairs of dimensions at one time. It is obvious that different permutations of the n dimensions show different aspects of the dataset. For datasets with discovered clusters, different permutations give different views on the relations of those clusters. As shown in Figure 1, the overall crossing between red and green clusters on wz coordinates implies they are generally separable by a $z = w + c'$ line on the zw plane, while the overall non-crossing between blue cluster and red (or green) clusters on wz coordinates implies they are generally separable by a $z = -w + c''$ line on the zw plane. More importantly, cluster interactions often connect to important knowledge discovery, as the large car and small car example in Section 1 tells us. With today's data explosion in many applications people often have very limited time on viewing a dataset and would like to see the most informative aspect at the first look. To fit these applications we do not use coordinate permutation strategies in [30,17,16] (which generate many views of a dataset) to visualize cluster interactions. Instead, we ask the following question.

> *Is it possible to quickly provide users a suggestive order of coordinates to view cluster relationships for some given preference?*

In statistics, people use data correlation (e.g. Pearson correlation) to describe the relation between two vectors. However, there is no widely adopted similar measurement for the relation between two clusters to our knowledge. Thus, analogous to data correlation, we use inter-cluster relation to describe the interaction between two clusters. We consider two clusters are clear positively-related if they are generally in parallel or have few crossings, and two clusters are clear negatively-related if they are generally crossing each other. The quantitative measurement of the interaction between two clusters is the number of inter-cluster crossings between them. This measurement links to knowledge discovery on cluster interactions.

Similarly, one can also define intra-cluster crossings, which reveals relations among data within a cluster. A coordinate order that minimizes intra-cluster crossings also has significant meanings in knowledge discovery. It reduces visual clutter caused by data interactions within a cluster, and thus is more likely to manifest inter-cluster relations to users. We provide the definitions in the following.

3.1 Inter-cluster and Intra-cluster Crossings

An *inter-cluster crossing* is defined as an order change between two items from two different clusters on two coordinates. For example, for two items $i \in Cluster_\alpha$ and $j \in Cluster_\beta$ on two coordinates x and y, if $x_i \prec x_j$ and $y_i \succ y_j$, then we say an inter-cluster crossing exists between item i and j on the xy-dimension. Similarly, one can define an *intra-cluster crossing* as an order change between two items from the same cluster on two coordinates.

Assume σ_x and σ_y are the order of data on the x-coordinate and the y-coordinate, respectively. Our definitions can be formalized as follows:

Definition 1. *The number of inter-cluster crossings between* $Cluster_\alpha$ *and* $Cluster_\beta$ *on the coordinates* x *and* y *is* $|C_{(\alpha,\beta)}|$ *where* $C_{(\alpha,\beta)} = \{(i,j)|\sigma_x(i) \prec \sigma_x(j)$ *and* $\sigma_y(j) \prec \sigma_y(i)$ *and* $i \in Cluster_\alpha, j \in Cluster_\beta\}$. *The number of intra-cluster crossings among* $Cluster_\alpha$ *on the coordinates* x *and* y *is* $|C_\alpha|$ *where* $C_\alpha = \{(i,j)|\sigma_x(i) \prec \sigma_x(j)$ *and* $\sigma_y(j) \prec \sigma_y(i)$ *and* $i,j \in Cluster_\alpha\}$.

Definition 2. *The number of total inter-cluster crossings on the coordinates* x *and* y *is* $|A|$ *where* $A = \{(i,j)|\sigma_x(i) \prec \sigma_x(j)$ *and* $\sigma_y(j) \prec \sigma_y(i)$ *and* i,j *belong to different clusters*$\}$. *The number of total intra-cluster crossings on the coordinates* x *and* y *is* $|B|$ *where*
$B = \{(i,j)|\sigma_x(i) \prec \sigma_x(j)$ *and* $\sigma_y(j) \prec \sigma_y(i)$ *and* i,j *belong to the same cluster*$\}$.

According to Definitions 1 and 2, we can calculate the four types of crossings on a pair of coordinates in $O(n^2)$ time, where n is the number of data items (i.e. lines in the parallel coordinates). It is interesting to observe that the definition of intra-cluster crossing among one given cluster on a pair of coordinates (Second part of Definition 1) corresponds to the Kendall's Tau coefficient [21,26] in statistics, and there is a $O(n \log n)$ algorithm [22] for calculating it. Although the inter-cluster crossings do not correspond to the Kendall's Tau coefficient, it is not difficult to design a $O(n \log n)$ algorithm to calculate each type of crossings defined in Definitions 1 and 2 (assuming the number of clusters is a constant). We omit further details due to the space limit.

3.2 Optimization with Hamiltonian Path

After we get the number of crossings between every pair of coordinates, we need to find an order of coordinates such that the number of crossings is minimized or maximized for different knowledge discovery purposes. This problem can be converted to the problem of finding a minimum (or maximum) weighted Hamiltonian path [10] in a complete graph, by turning each coordinate into a vertex, adding an edge between every two vertices, and setting the edge weight to be the number of crossings between the two corresponding coordinates. It is quite obvious that the minimum or maximum weighted Hamiltonian path problem for complete graphs is NP-hard, as it is easy to reduce the Hamiltonian path problem for an unweighted graph, which is NP-complete, to this problem.

Exact Solution. An exact solution for the minimum (or maximum) weighted Hamiltonian path problem exhaustively tries all the permutations of vertices. The complexity is $O(n!)$ and the method becomes intractable when n is slightly larger. However, in parallel coordinates visualization, it is not uncommon to see a dataset with 10 or less coordinates. For these applications, the exhaustive search algorithm is still one of the most simple and effective solutions. Ideas in various branch and bound approaches for the Traveling Salesman Problem (TSP for short) can be used to speed up the exhaustive search algorithm for the minimum (or maximum) weighted Hamiltonian path problem. Interested readers may refer to the TSP survey paper [24] for details.

Metric Space and Approximation Solutions. Since the exact solutions cannot easily handle high-dimensional data, we seek fast approximate solutions when the number of coordinates is large. As nice approximate algorithms for minimum or maximum metric-TSPs exist (see solutions in [5] for minimum metric-TSP, [12] for maximum Metric-TSP, [4] for minimum metric-TSP with a prescribed order of vertices), we are wondering if our problems are metric Hamiltonian path problems. If they are, can we have similar approximate algorithms? Fortunately, we have a positive answer as stated in Lemma 1 (proof omitted due to space limit) which extends the well-known fact that Kendall tau distance (corresponds to intra-cluster crossings) is a metric :

Lemma 1. *The graph G, constructed by converting each coordinate to a vertex and setting the weight of each edge between two vertices to be the number of inter-cluster crossings between the two corresponding coordinates, either within two specific clusters or among all clusters, forms a metric space, in which edge weights follow the triangle inequality.*

Thus, it is not difficult to show that, if a graph G, with n vertices forms a metric space (regardless whether there exists a prescribed order of some vertices), a k-approximation solution for the minimum (or maximum) traveling salesman problem implies $2k$-approximation solutions for minimizing (or maximizing) inter-cluster (or intra-cluster) crossings.

In some special cases, it is possible to achieve even better approximation ratio. For example, Hoogeveen [14] shows that Christofides' 1.5 Approximation algorithm [5] of minimum metric TSP can be modified for minimum metric Hamiltonian path problem with the same approximation ratio, but the time complexity of this algorithm or its modified version, though polynomial, is much larger than linear. To achieve an even faster running speed for minimizing inter-cluster (or intra-cluster) crossings, we implemented a linear 2-approximation minimum metric Hamiltonian algorithm modified from the well-known *linear* 2-approximation algorithm for the minimum metric-TSP [6].

3.3 Empirical Study on Real Datasets

In this subsection, we report our empirical results on data extracted from the UC Irvine Machine Learning Repository[1], which has been widely used as a primary

[1] http://archive.ics.uci.edu/ml/

Table 1. Dataset characteristics and number of crossing changes

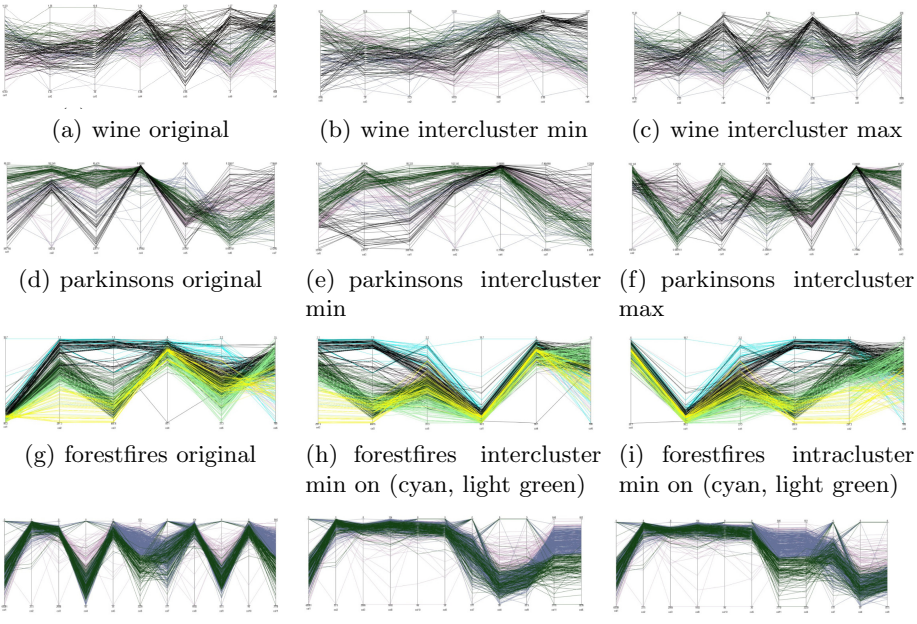
	dataset characteristics			number of crossing changes		
Dataset	Records	Columns	Clusters	inter min	inter max	intra min
eighthr	2533	12	2	-24.3%	+50.0%	-15.1%
forestfires	517	6	6	-29.6%	+24.7%	-21.9%
parkinsons	194	7	4	-42.0%	+40.8%	-26.3%
pima-indians	767	7	10	-15.2%	+20.6%	-15.4%
water-treatment	526	11	3	-41.9%	+13.6%	-37.0%
wdbc	568	5	4	-14.3%	+20.2%	-10.6%
wine	177	7	4	-46.8%	+13.4%	-11.0%

source of machine learning and data mining datasets. The basic characteristics of the datasets to be studied, are listed in Table 1. For our experiments, we chose the well-known K-means algorithm [11] to cluster the data items into exclusive clusters. We implemented the visualization program in JavaScript (web-based). For this study, we tested our visualization implementation in Firefox 3.6.12 on a mainstream desktop PC with an Intel Core i5 2.67GHz CPU and 8 GB of memory.

In our empirical study, we are primarily interested in observing the effects of proposed inter-cluster and intra-cluster ordering for visual knowledge discovery. Maximizing intra-cluster crossings does not clearly connect to the study of cluster interactions thus we omit it for the conciseness of the paper.

Table 1 reports the detailed changes of inter-cluster crossings after minimization and maximization, and intra-cluster crossings after minimization, for different datasets. Although crossing changes are substantial, it is more interesting to see what are the changes on the visualization results? A set of representative results are as shown in Figure 2.

Minimizing and Maximizing Total Inter-cluster Crossings:
Figure 2 (b) and (c) shows the visualization results for minimizing total inter-cluster crossings and maximizing total inter-cluster crossings, respectively, for dataset "wine". In the original order, i.e., Figure 2 (a), we can observe clusters are generally negatively related between col 3 and col 4, between col 4 and col 5, between col 5 and col 6, between col 6 and col 7. Quite impressively, clusters show much more positive relations in the adjacent coordinates in Figure 2 (b). In contrast, clusters show even more negative relations in Figure 2 (c). These results generally meet our expectation for the effects of minimizing and maximizing the total inter-cluster crossings. Interestingly, we can observe the last two columns (col 3 and col 7) in Figure 2 (c) contain a couple of strongly negatively-related cluster pairs which are not revealed by Figure 2 (a) on its original order. By checking the original data, we found col 3 corresponding to "alkalinity of ash", while col 7 corresponding to "proline". This helps explain the negative relations between clusters as alkalinity is the ability of a solution to neutralize acids, while the proline is an α-amino acid. A high in alkalinity is more likely to result in low α-amino acid.



(a) wine original (b) wine intercluster min (c) wine intercluster max

(d) parkinsons original (e) parkinsons intercluster min (f) parkinsons intercluster max

(g) forestfires original (h) forestfires intercluster min on (cyan, light green) (i) forestfires intracluster min on (cyan, light green)

(j) water-treatment original (k) water-treatment inter-cluster min (l) water-treatment inter-cluster min 2-appr

Fig. 2. Visualization results (colors shown in the web version of this paper)

Similarly, Figure 2 (e) and (f) shows the visualization results for minimizing total inter-cluster crossings and maximizing total inter-cluster crossings, respectively, for dataset "parkinsons", in which each col represents a measurement for "parkinsons". After our visualization, it is easy for health care providers to spot measurements that are strongly positively-related in Figure 2 (e), and measurements that are strongly negatively-related in Figure 2 (f).

Minimizing Inter-cluster and Intra-cluster Crossings on a Pair of Clusters:
We would like to see the difference between minimizing inter-cluster crossings and intra-cluster crossings. To ease our observation, we focus on only two clusters, cyan and light green, in Figure 2 (g), which shows the dataset "forestfires" in its original order. Figure 2 (h) and (i) show the visualization results corresponding to minimizing inter-cluster crossings and intra-cluster crossings, respectively. In Figure 2 (h) we can observe that the cyan cluster and the light green cluster are generally positively-related in all adjacent columns. This is understandable as the visualization goal is to minimize the inter-crossings between them. However, Figure 2 (i) shows a strongly negative-relation between them on the last two columns (col 2 and col 6). This is because the goal of minimizing intra-cluster crossings does not care about the relations between the cyan cluster and light green cluster. Rather, it tries to reduce crossing within the two clusters so as

to reduce visual clutter and provide a better chance to observe the relations, regardless of positive or negative, between the two clusters.

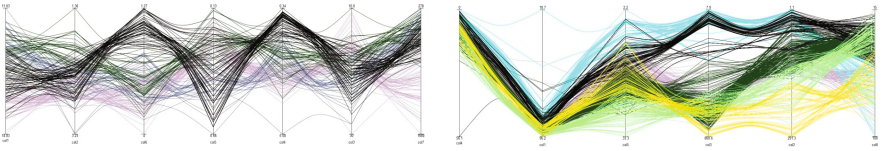
By checking the original data, we found col 2 corresponding to DMC and col 6 corresponding to RH. DMC is an indication (the larger the more likely) of the depth that fire will burn in moderate duff layers and medium size woody material, while RH is relative humidity. Thus, we understand the discovered result in Figure 2 (i) that clusters tend to be negatively-related between DMC and RH.

Minimizing Inter-cluster Crossings by the 2-Approximation Algorithm: In all the tested datasets, the exact algorithm finishes in no more than 100 milliseconds except for the datasets "eighthr" and "water-treatment". It takes about 2 minutes to exactly order "eighthr" (12 columns), and about 15 seconds to exactly order "water-treatment" (11 columns). This poses a concern on using exact algorithms for ordering datasets with more than 10 columns, and justify the importance of approximation algorithms for ordering large datasets. In the following we empirically study the effect of the popular 2-approximation algorithm (discussed at the end of Section 3.2) on our visualization scheme. In order to get a better ordering through the 2-approximation algorithm, we try DFS search from each vertex and find a lowest-cost result among all the 2-approximation results. Even with multi-DFS search, the ordering time is still lightning fast. For all datasets, including "eighthr" and "water-treatment", the multi-DFS search finishes within a couple of milliseconds. This makes our visualization schemes work for large datasets.

Figure 2 (l) shows the visualization result of minimizing inter-cluster crossings by the 2-approximation algorithm. Compared to the visualization result by the exact algorithm as in Figure 2 (k), it is hard to tell the actual difference between the two algorithms in revealing the positive relations among clusters. Detailed data may explain this: The numbers of inter-cluster crossings minimized by the 2-approximation algorithm are -23.0%,-29.6%,-42.0%,-8.4%,-41.6%,-14.3%,-46.8%, respectively, for the datasets in Table 1 (from top to bottom). Thus we can see there is very little performance degradation (in some datasets there is no difference) with the 2-approximation algorithm but very significant speed-up (linear vs factorial, in terms of complexity).

4 Shaping Clusters against Visual Clutters by an Energy Reduction Model

For some figures (e.g., Figure 2(c) and (i)) in the previous section, inter-cluster crossings are hard to discern even after ordering the coordinates for minimizing intra-cluster crossings. This is because a substantial amount of lines from a large-scale data set are typically entangled together in the limited space and resolution of display devices, confounding their belongings to different clusters. Consequently, the pattern and knowledge discovery of clustered data is hindered and the usage of parallel coordinates is limited. A handful of works [3,2,20] display the silhouette shape of individual clusters while the lines are intentionally brushed out. The shape of a cluster is sensitive to a few outliers, and as a result, the visualization of cluster relations is not satisfying. This scenario can be further deteriorated, when the lines of a given cluster are even more sparsely distributed.



(a) enhancement for Figure 2(c) (b) enhancement for Figure 2(i)

Fig. 3. visualization enhancement (colors shown in the web version of this paper)

To tackle these visual clutters for better knowledge discovery, We innovate a quadratic energy reduction model to smoothly shape clusters against visual clutters while preserving essential details of each cluster, by associating each line i (with z_i being its center) between two adjacent dimension x and y with a "rubber band" effect with three potential energy:

Elastic Energy: $E_E(i) = (z_i - \frac{x_i+y_i}{2})^2$
Attraction Energy: $E_A(i, \hat{c}_p) = (z_i - \hat{c}_p)^2$
Repelling Energy: $E_R(i, \hat{c}_{p-1}, \hat{c}_{p+1}) = (z_i - \hat{c}_{p-1})^2 + (z_i - \hat{c}_{p+1})^2$

Here each cluster has an attracting center $\hat{c}_p$ which may serve as a repelling center for its adjacent clusters. We developed an efficient energy reduction model by properly initializing and manipulating $\hat{c}_p$ (omitted due to space limit).

The visualization effects are significantly enhanced by our energy reduction model. Figure 3(a) and Figure 3(b) are examples of enhanced visualization results for Figure 2(c) and (i), respectively, by our energy reduction models. Readers can easily observe more clusters and thus better understanding their relationships. It is easy to see the essential details of these clusters are not altered. More specifically, if two clusters are negatively-related or positively-related, the relationship not only remains after energy reduction, but gets further enhanced for human observation. For example, we can observe the blue cluster is negatively related to the pink cluster between the last two columns in Figure 3(a) while it is almost impossible to see this in Figure 2(c). As another example, the negative relation between cyan cluster and the light green cluster is more manifest in Figure 3(b) than in Figure 2(i). Finally, instead of affecting the observation of cluster interactions, outliers of each cluster can be easily identified as those few lines far away from the majority of lines.

In summary, given an order of coordinates, our energy reduction model efficiently provides better views of clusters for visual knowledge discovery at both the macro level (i.e., cluster interactions) and the micro level (i.e., individual lines with outliers clearly exposed).

5 Conclusion and Future Work

In this paper, we show a novel method to visualize discovered clusters in parallel coordinates. First, we provide good orders of coordinates for different knowledge discovery purposes. Second, we shape the clusters with a quadratic

energy reduction model, such that cluster interactions are much easier to observe without compromising their essential details. Our empirical study on visualizing real datasets confirms that our method is effective and efficient. Our visualization techniques can further be combined with other visualization tools for better results, e.g, applying various visual rendering algorithms to enhance our visualization effects.

Acknowledgement. This work was supported by the US National Science Foundation under Grant #1019343 to the Computing Research Association for the CIFellows Project.

References

1. Ankerst, M., Berchtold, S., Keim, D.A.: Similarity clustering of dimensions for an enhanced visualization of multidimensional data. In: IEEE Symposium on Information Visualization (INFOVIS), p. 52 (1998)
2. Artero, A.O., de Oliveira, M.C.F., Levkowitz, H.: Uncovering clusters in crowded parallel coordinates visualizations. In: IEEE Symposium on Information Visualization (INFOVIS), pp. 81–88 (2004)
3. Berthold, M.R., Hall, L.O.: Visualizing fuzzy points in parallel coordinates. IEEE Transactions on Fuzzy Systems 11(3), 369–374 (2003)
4. Böckenhauer, H.-J., Hromkovič, J., Kneis, J., Kupke, J.: On the Approximation Hardness of Some Generalizations of TSP. In: Arge, L., Freivalds, R. (eds.) SWAT 2006. LNCS, vol. 4059, pp. 184–195. Springer, Heidelberg (2006)
5. Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem. Graduate School of Industrial Administration, CMU, Report 388 (1976)
6. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. The MIT Press (2001)
7. Dasgupta, A., Kosara, R.: Pargnostics: Screen-Space Metrics for Parallel Coordinates. IEEE Transactions on Visualization and Computer Graphics 16(6), 1017–1026 (2010)
8. Ellis, G., Dix, A.: Enabling automatic clutter reduction in parallel coordinate plots. IEEE Transactions on Visualization and Computer Graphics 12, 717–724 (2006)
9. Fua, Y.-H., Ward, M.O., Rundensteiner, E.A.: Hierarchical parallel coordinates for exploration of large datasets. IEEE Visualization, 43–50 (1999)
10. Gross, J.L., Yellen, J.: Graph theory and its applications. CRC Press (2006)
11. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann (2000)
12. Hassin, R., Rubinstein, S.: A 7/8-approximation algorithm for metric max tsp. Inf. Process. Lett. 81(5), 247–251 (2002)
13. Holten, D., Van Wijk, J.J.: Evaluation of Cluster Identification Performance for Different PCP Variants. Computer Graphics Forum 29(3), 793–802 (2010)
14. Hoogeveen, J.A.: Analysis of christofides' heuristic: Some paths are more difficult than cycles. Operations Research Letters 10(5), 291–295 (1991)
15. Hurley, C.B.: Clustering visualizations of multidimensional data. Journal of Computational and Graphical Statistics 13(4), 788–806 (2004)
16. Hurley, C.B., Oldford, R.W.: Pairwise display of high-dimensional information via eulerian tours and hamiltonian decompositions. Journal of Computational and Graphical Statistics 19(4), 861–886 (2010)

17. Hurley, C.B., Oldford, R.W.: Eulerian tour algorithms for data visualization and the pairviz package. Computational Statistics 26(4), 613–633 (2011)
18. Inselberg, A.: The plane with parallel coordinates. The Visual Computer 1(2), 69–91 (1985)
19. Jain, A.K., Narasimha Murty, M., Flynn, P.J.: Data clustering: A review. ACM Comput. Surv. 31(3), 264–323 (1999)
20. Johansson, J., Ljung, P., Jern, M., Cooper, M.: Revealing structure within clustered parallel coordinates displays. In: IEEE Symposium on Information Visualization (INFOVIS), p. 17 (2005)
21. Kendall, M.G.: A new measure of rank correlation. Biometrika 30(1/2), 81–93 (1938)
22. Knight, W.R.: A computer method for calculating kendall's tau with ungrouped data. Journal of the American Statistical Association, 436–439 (1966)
23. Kosara, R., Bendix, F., Hauser, H.: Parallel sets: Interactive exploration and visual analysis of categorical data. IEEE Trans. Vis. Comput. Graph. 12(4), 558–568 (2006)
24. Laporte, G.: The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research 59(2), 231–247 (1992)
25. Moustafa, R., Wegman, E.: Multivariate Continuous Data - Parallel Coordinates. Springer, New York (2006)
26. Nelson, R.B.: Kendall tau metric. Encyclopaedia of Mathematics 3, 226–227 (2001)
27. Peng, W., Ward, M.O., Rundensteiner, E.A.: Clutter reduction in multidimensional data visualization using dimension reordering. In: IEEE Symposium on Information Visualization (INFOVIS), pp. 89–96 (2004)
28. Siirtola, H., Räihä, K.J.: Interacting with parallel coordinates. Interacting with Computers 18(6), 1278–1309 (2006)
29. Wegman, E.J.: The grand tour in k-dimensions. In: Computing Science and Statistics: Proceedings of the 22nd Symposium on the Interface, pp. 127–136 (1991)
30. Wegman, E.J.: Hyperdimensional data analysis using parallel coordinates. Journal of the American Statistical Association 85(411), 664–675 (1990)
31. Wegman, E.J.: Visual data mining. Statistics in Medicine 22, 1383–1397 (2003)
32. Wilhelm, A.F.X., Wegman, E.J., Symanzik, J.: Visual clustering and classification: The oronsay particle size data set revisited. Computational Statistics 14, 109–146 (1999)
33. Wong, P.C., Bergeron, R.D.: 30 years of multidimensional multivariate visualization. Scientific Visualization, 3–33 (1994)
34. Yang, J., Peng, W., Ward, M.O., Rundensteiner, E.A.: Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets. In: INFOVIS (2003)
35. Zhao, K., Liu, B., Tirpak, T.M., Schaller, A.: Detecting patterns of change using enhanced parallel coordinates visualization. In: ICDM, p. 747 (2003)
36. Zhou, H., Yuan, X., Qu, H., Cui, W., Chen, B.: Visual clustering in parallel coordinates. Comput. Graph. Forum 27(3), 1047–1054 (2008)