

# User-controlled construction of parallel coordinate plots

Lucas Rizoli\*  
University of British Columbia

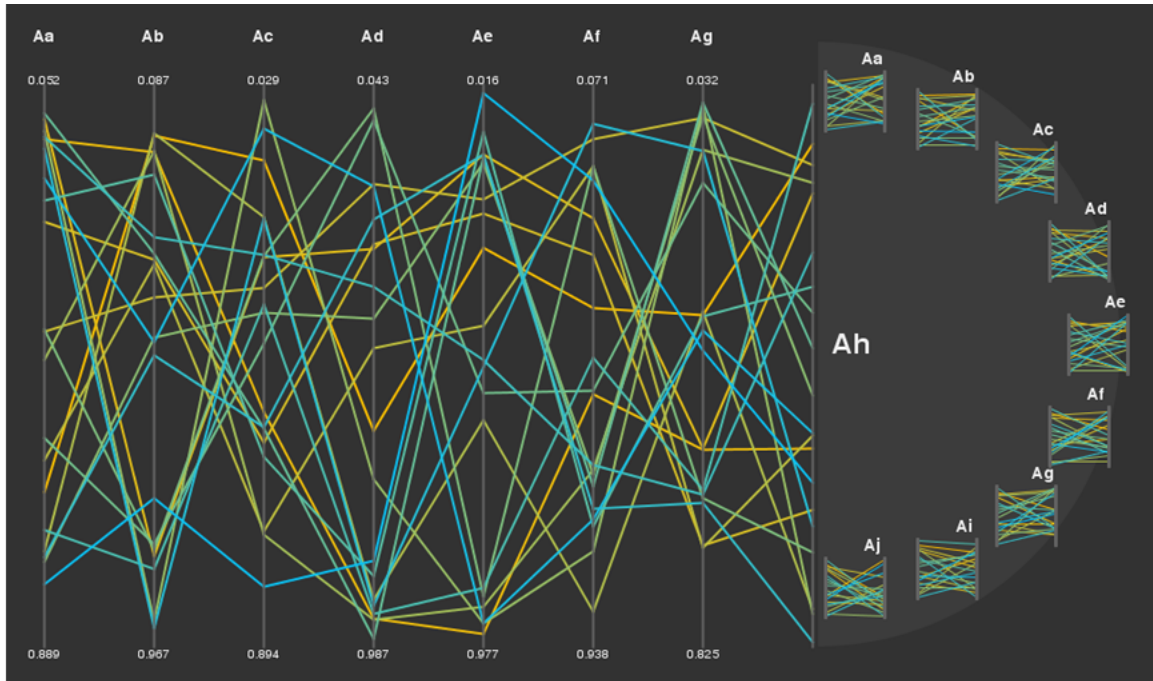


Figure 1: Screenshot of the prototype for parallel coordinate plot construction

**Keywords:** Parallel coordinates, dimension ordering, interactive techniques.

**Index Terms:** H.5.2 [Information interfaces and presentation]: User interfaces—Interaction styles; I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

## 1 INTRODUCTION

Parallel coordinate plots [7, 9] are useful for visualizing multidimensional data. There are many advantages to using parallel coordinate plots in data visualization: they can represent data across many dimensions, they are reasonably easy to interpret and do not require a great deal of expertise to use, they are not domain-specific and can be applied to almost any kind of data, and they can be plotted without requiring a great deal of computational resources.

Despite the many positive aspects of parallel coordinates, they do have certain weaknesses. Firstly, as with many other visualizations of data, parallel coordinate plots can become cluttered. The lines used to represent relationships between points can cross and become difficult to follow. When there are a large number of points to be plotted, any relationship between two dimensions that is not universal becomes difficult to discern.

\*Email: lrizoli [at] cs.ubc.ca

While parallel coordinate plots are capable of representing data in any number of dimensions, they become impractical when the dimensionality is high (typically when greater than ten or twelve). This is largely due to the fact that parallel coordinates cannot represent data in multiple dimensions simultaneously; the plots are sequences of pairwise comparisons between dimensions. As such, it becomes difficult to relate any more than two dimensions at once in the same area of the plot. Comparing relationships whose representations are visually distant is difficult: the user is required to store them in memory, which may quickly become crowded with the demands of considering the many pairwise relationships in high-dimensional data. The result is that plots representing more than twelve dimensions are taxing and less useful than they could be.

The effectiveness of parallel coordinates is also limited by the order of the dimensions plotted [10]. Since comparisons are pairwise, only a small subset of the relationships in the data can be plotted without repeating axes. For example, in a dataset representing local weather conditions and school closures, it would not be possible to plot the relationship between closures (C) and daytime temperature (T), precipitation (P), and date (D) without repeating the axis representing closures at least once. These relationships would be explicit given the order T, C, P, C, D. However, that order would fail to plot other, potentially important relationships between the other dimensions, such as temperature and precipitation. Although that relationship may be somewhat intuitive in the given example, when the relationships between the dimensions of the data to be plotted are not well known, the ability to discover interesting aspects of the data will hinge entirely on the order in which they are plotted.

## 1.1 Related work

Of these three weaknesses, visual clutter and difficulty with high-dimensional data appear to be the primary focus of previous research. There have been a number of suggestions as to how to reduce clutter and improve the representation of high-dimensional data. These include distortion [3], clustering [5], dimensional reduction [11], automated ordering as to reduce clutter [8], and a combination of techniques and data metrics [10].

There have been a few proposed remedies to the problem of the order of axes in parallel coordinates: from dimension reduction [11] to computer-imposed axis ordering [6]. Both systems are complex and computationally expensive. The effectiveness of each is reliant on the kind and quality of the data being plotted, and despite the power and soundness of the methods, they may not provide the best ordering under all interpretations.

## 2 USER-CONTROLLED ORDERING METHODS

Users of parallel coordinates generally find greater satisfaction with orders that they are able to manipulate for themselves [10]. Unfortunately, there has not been much research into what axis ordering methods are best. Currently, the most common method for a user to manipulate the order of axes is through what will be referred to as the *shuffle* method.

### 2.1 The shuffle method

The shuffle method is quite simple. Users begin with a complete plot where every axis is shown. Users select a single axis and then select where in the order this axis should be placed. The plot is then re-rendered, using the modified order. The method for selecting axes and positions in the order is typically done using a mouse. For example, in Figure 2, the user selects the axis second in the order (“Quail”), holds down the mouse button, and drags it so that it is between the fourth and fifth axes (between “Emu” and “Pigeon”). When the mouse is released, the data is replotted with the selected axis fourth in the order (“Quail” between “Emu” and “Pigeon”).

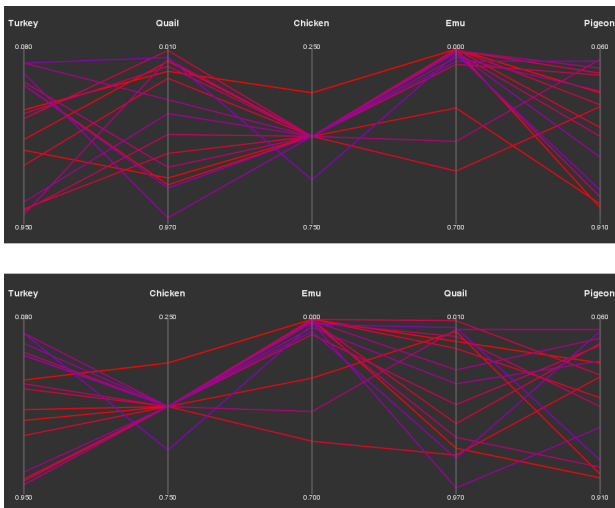


Figure 2: An example of the shuffle method: the “Quail” axis is shuffled from second in the order (top) to fourth (bottom).

#### 2.1.1 Problems with shuffling

The shuffle method, although simple, can become quite difficult to use. Even when manipulating the plot of a dataset of reasonable dimensionality (less than twelve), the number of possible changes available at any time is quite large. If users are not familiar with the domain of the data—that is, they do not know which relationships are

interesting to plot beforehand—they will have no way of reducing the number of possible actions. They are forced to consider the entire dataset before and after any shuffle action.

In the shuffle method, all axis manipulation is done in the same plot in that its results are displayed. There is no separation between what currently *is* and what *will be*; users can only see the result of a shuffle after it has been done. If users would like to predict what the result of a shuffle will be, it is necessary for them to imagine it.

Further, the shuffle method requires that users consider multiple relationships between dimensions for each axis shuffled. The fact that parallel coordinates show pairwise data means that every single axis manipulation results in many changes in the plot. In order to predict the consequences of their shuffling, users need to consider all these relationships. That is, they must consider what the relationships between the selected axis and the two axes it is shuffled between will be, as well as the new relationships between the axis which preceded the shuffled axis and the one which came after it. In the example shuffle shown in Figure 2, shuffling the “Quail” axis from second to fourth creates both the “Emu-Quail” and “Quail-Pigeon” relationships, but also creates “Turkey-Chicken”, and destroys “Turkey-Quail” and “Quail-Chicken”. It may seem reasonable to require the user to consider the first two new relationships as they are the direct result of the user’s drag-and-drop action. However the collateral creation and destruction of relationships in areas of the visualization not directly tied to the end of the user’s action are confusing. It is not easy to keep in mind all of the consequences of shuffling a single axis.

The concept of shuffling axes is not difficult, and the interaction itself is fast and easy. However, the shuffle method requires a lot of effort from users. They must rely on their own memory or test actions through tiresome trial-and-error shuffles. The extra time or effort required by shuffling discourages users from changing the order of axis, which in turn reduces exploration and the likelihood that interesting relationships between dimensions will be found.

### 2.2 Better user-controlled ordering methods

Users need not face the many difficulties of the shuffle method when manipulating the order of axes. There are a great number of ways in which shuffling may be improved upon. Methods that aim to do so should meet the following expectations.

- **Be simple and intuitive.** The shuffle method is not complex or difficult to understand. Any method improving upon it should not require significantly more expertise or effort.
- **Allow users to anticipate the results of their actions.** If users are uncertain—or worse, unaware—of the consequences of their changes, the method will be confusing or frustrating. Methods should behave consistently and not have unwanted, unpredictable effects.
- **Facilitate and encourage exploration of the data.** While domain experts may be able to determine the axis order that results in the most useful or interesting plot, those with little to no previous understanding of the data are likely to need to explore and experiment with the plot before finding a good order. This process should not be limited by the method used to change the order of axes. Indeed, the method should encourage it.
- **Keep the user in control.** Users’ control over the order should not be sacrificed for any of the points listed above.

## 3 SOLUTION: THE FAN MENU METHOD

We propose a user-controlled axis ordering method in which users build plots axis by axis. This method aims to facilitate the choice of axes by showing the user what potential plots would look like, as

well as providing an intuitive interface with which to build the plot. We call it the *fan menu method*.

### 3.1 Building a plot

Users begin with a blank area and a menu of all dimensions. Once a dimension has been selected, the axis representing it is rendered on the far left of the plot area; this is the current axis. A semicircular button with a large plus (the “add axis” button) appears next to it. Clicking on that button invokes the “add” menu.

This menu lists all the possible pairs of axes in which the current axis is included, less the current axis itself. That is, the menu lists the possible relationships between the current axis and all the other dimensions in the dataset; it does not include the relationship of the current axis with itself. Users can select one of these axis pairs and the relevant axis is added to the plot.

These pairs are rendered as small parallel coordinate plots called *thumbnails*. Each thumbnail is the plot of the points between the current axis and another axis: exactly what would be plotted were both axes next to each other in the larger plot, but in miniature. In a sense, a thumbnail is a preview of what will be added to the larger plot should that thumbnail be selected. The name of the axis that will be added appears above each thumbnail, right of centre. The name of the current axis is shown in the centre of the menu.

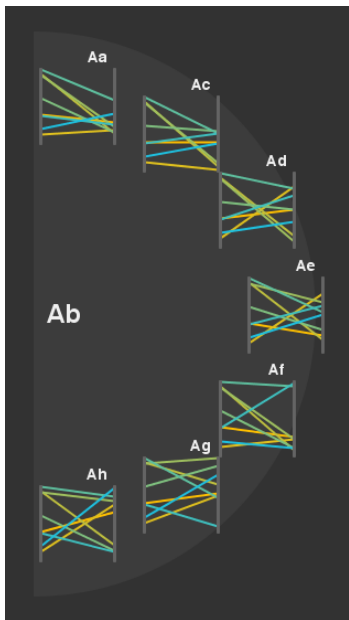


Figure 3: A fan menu listing thumbnails comparing “Ab” to all other dimensions in the dataset.

Thumbnails are listed as items in what will be referred to as a *fan menu* (Figure 3). A fan menu is a semicircular selection widget that lists a menu items along its curved edge. It is similar to an oriental fan in shape, and to a typical graphical user interface drop-down or pop-up menu in function.

Specifically, a fan menu is similar to what is known as a pie or marking menu [2]<sup>1</sup>. Both display items in such a way that allows for more efficient selection. When a marking or fan menu is displayed such that the mouse pointer is at the centre of the circle, each menu item is at an equal distance from the mouse pointer. This allows items at the end of the list items, which would be farther from the pointer in common, rectangular menus, to be selected with the same amount of mouse movement as any other. In order to ensure this,

<sup>1</sup>See also <http://www.piemenu.com/>

the button that invokes the fan menu is placed so that the mouse will be at the centre of the menu.

Fan menus also contribute to the visibility of thumbnails. The curved edge of a fan menu provides more space along which to arrange thumbnails than does a straight edge or column. Since parallel coordinate plots deal with high-dimensional data, the number of thumbnails generated in each “add” action may well exceed the number of thumbnails that can be drawn along a single column at a readable size. In Figure 3, it is possible to see that the thumbnail representing “Ab-Aa” would overlap the thumbnail for “Ab-Ac” were they drawn at the same size in a straight column. By arranging them on a curve, taking advantage of horizontal space as well as vertical, it is possible to fit more thumbnails, to allow for more padding between them, or to render them larger than would be possible in a straight arrangement.

Although a conventional columnar arrangement may be best when comparing thumbnails, the finer distinctions they allow are not likely to show any meaningful benefit over a curved arrangement, particularly when dealing with small thumbnails. Grid arrangements, where thumbnails appear in multiple columns, crowd the display, or put some thumbnails at a disadvantage because they are in a strange local context. The immediate neighbours in the grid alter their interpretation by the user. Some thumbnails in a grid may also be harder to select than others because they are farther from the mouse pointer’s initial location. For these reasons, the fan menu is preferable.

The new axis becomes the current axis. Again, the “add axis” button appears to the right of it and invokes a menu of all the possible combinations of the current axis and all other dimensions in the dataset—this includes the dimensions represented by axes that have already been added to the plot. The user selects a pairing and the axis necessary to complete the pair is added to the plot.

The “add” action is demonstrated in Figure 4. In the example, the user has already added four axes: those representing dimensions “Aa”, “Ab”, “Ac”, and “Ad”. When the user clicks on the “add axis” button next to the current axis, “Ad”, a menu appears which lists all the axis pairs involving “Ad” (but not “Ad-Ad”). Notice that this includes pairs between “Ad” and axis that are already in the plot, such as “Aa” and “Ac”. The user then selects one of these pairs—in this case, the pair “Ad-Ae”—and the relevant axis, “Ae”, is added to the plot.

Above each axis there is a small button with a minus on it, the “remove axis” button. When this is clicked, it removes the axis below it from the plot. Users add axes to the right of the plot in a serial manner and remove axes from the plot randomly. An example of this is shown in Figure 5.

Plots are thus built axis by axis, through individual “add” and “remove” actions; the order of axes is defined during the building process.

### 3.2 Advantages of the fan menu method

- **It is simple and easy to use.** The fan menu does not require users to learn any new method of interaction or interpret visual elements differently. The method is clear and easy to understand.
- **Users can look ahead.** The thumbnails in the fan menu provide a small preview of the potential relationships to be added to the plot. Users need not perform an “add” action in order to see its consequences.
- **Only one relationship is considered at a time.** Both “add” and “remove” actions result in a single changed relationship in the plot. When adding an axis, it is the relationship between the current axis and the axis added. When removing an axis, it is the relationship between the axes immediately before and

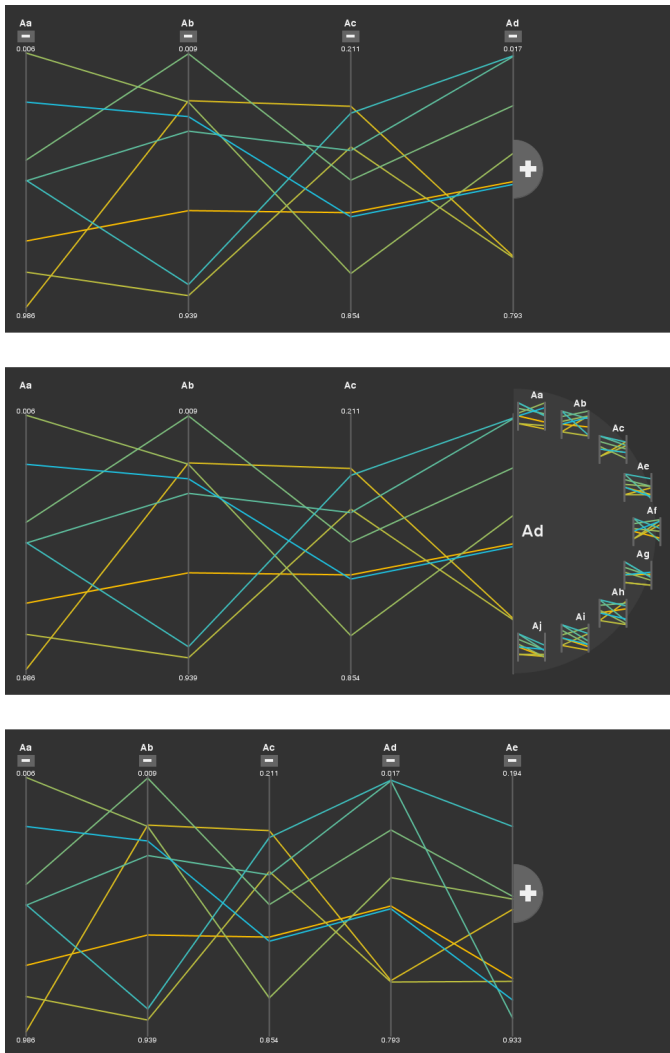


Figure 4: The process of adding an axis to a plot: the initial plot at the top, the selection step in the middle, the result at the bottom.

after the removed axis. Unlike the shuffle method, no other relationships are created or destroyed.

- **Actions and results are tied together.** An action has no effects beyond the area immediately relevant to its execution. Changes occur where actions are performed (near the buttons and menus) and only there. The result of an action unambiguous and clear.
- **Seeing is easier than thinking.** With each “add” action, all the pairs involving the current axis are rendered on-screen. It is possible to compare these relationships quickly, without having to generate and store them mentally.
- **Exploration is facilitated.** Even when users initiate and “add” with a specific pair in mind, they will see all pairs involving the current axis and may find interesting ones that they had not considered before. For users without such expectations, having thumbnails of these relationships may help them in understanding the data, in identifying which axis pairs go well together and which do not.
- **Plots are relevant.** Axes in the final plot were each specified

explicitly at some point of the plot’s construction: they are rendered because the user decided to render them. This can help in keeping plots lean, free of irrelevant axes. It also helps users understand and justify the plot: by building plots axis by axis, they are forced to consider each dimension as it relates to every other.

- **The user is in control.** At any point, users may modify the order by adding or removing axes. Only axes added by users are shown in the plot. The order of axes is entirely the result of users’ actions.

### 3.3 Disadvantages

There are disadvantages to the fan menu method. Building a plot axis by axis, while allowing the user a great amount of control, requires time. Removing an axis is fast and can be done regardless of its order in the plot. The “add” action and fan menu require that axes be added at the far left or right ends of the plot. Should a user decide to add an axis somewhere between existing axes, it would be necessary to remove all the axes between that location and the end of the the plot.

The fan menu takes up space. The entire screen can be dedicated to rendering the plot when only the shuffle method is available. The fan menu method requires that, at least when users are performing an “add” action, some of the space that could be used to render the plot be used instead to display the menu. Further, its semicircular shape requires more screen area than would a narrower, columnar menu, as it is both tall and wide.

As the number of data points or dimensions in a dataset increase, so do the difficulties of discerning important relationships in the plot. Unfortunately, the vulnerability to large or high-dimensional datasets that exist generally in any parallel coordinate system, also exist in the fan menu method: thumbnails can be difficult to differentiate when plots are cluttered, and a large number of dimensions cause thumbnails to be rendered at very small sizes in the fan menu. Possible improvements to the method are proposed in Section 6.

## 4 PROTOTYPE

A prototype of the fan menu method of building parallel coordinate plots was implemented in Processing [4], an open-source, Java-based environment. Processing was decided upon because it offers a lighter more informal syntax for rendering elements on-screen than does Java2D. The Interfascia user interface library [1] was eventually used to handle user-initiated events, after it became apparent that the event handling functions available in Processing are implemented at a level which makes complex interaction difficult to manage. The plots and interface were created for the prototype, as Processing does not include any parallel coordinate libraries or native interface widgets. Users were consulted during development, asked to comment on both early mock-ups and the semi-functional prototype. Their comments were generally positive and informed the look and arrangement of fan menu and interface elements. As of this writing, the prototype is not yet fully functional. All figures in the report are screenshots of the prototype.

## 5 EVALUATION

In order to evaluate the fan menu plot construction method, both in general and against the shuffle method, it would be necessary to conduct two kinds of user studies. The first, to determine the low-level performance differences between the fan menu and shuffle ordering methods, and the second to qualify the benefits of plot construction and the fan menu itself.

### 5.1 Comparing performance between methods

It would be possible to quantify the differences between the fan menu and shuffle methods through a series of performance trials.

Subjects would perform certain tasks sometimes using the fan menu method, sometimes the shuffle method.

The tasks would be simple re-ordering tasks performed on plots of randomly generated data. This would ensure that no structure in the data would confound the data collected, that the differences in performance could be attributable to the method of axis ordering used alone. The tasks would short and simple, such as placing an axis at the beginning or end of the plot, re-ordering axes in a plot or creating a plot where a certain sub-sequence is in order, or re-ordering plots to meet a certain total order.

These tasks would be repeated several times (to avoid learning effects), at least once for each combination of method and task. The time taken to complete each task would be collected, as would the number of errors made during the task. These times and error rates would be analyzed using statistical tests such as the ANOVA in order to determine if the methods used when they were collected (fan menu or shuffle) were related to any significant differences in error or speed.

The results of such a study would provide evidence that there is (or is not) a significant difference in the time and efficiency of each method for each task. This sort of hard, empirical evidence could be quite conclusive. However, discovering that one method is faster or less error prone may not be helpful. The speed at which users are able to re-order axes in a parallel coordinate plot may not be the most relevant way of judging whether or not a method is successful. While important, at least to some degree, speed and accuracy may be second to usefulness and satisfaction when deciding whether a method should be included in a visualization package.

## 5.2 Evaluating user-controlled construction

Judging the positive and negative effects of the fan menu on the use of parallel coordinate plots in general is more difficult than comparing performance measures. Aspects of interactive methods such as user satisfaction, sense of ease and efficiency, are more difficult to find and collect. Are users happier or more confident in their plots when they are built using the fan method? Just what would be sufficient support for the claim that the fan menu method facilitates data exploration?

A study whose goal is to answer such questions would have to rely on qualitative, observational, and anecdotal data. This would be collected through examining program logs, observing subjects using the method (perhaps through video), and having subjects complete questionnaires and participate in interviews.

The most difficult aspect of such a study would be designing the tasks through which subjects would experience the fan menu and shuffle methods. Tasks would have to be more longer, more involved, complex, and open ended than those used in the performance study. They certainly could not rely on randomly generated data. Datasets would have to represent something somewhat familiar and reasonably non-abstract so that subjects could participate without needing to know too much about what the data means. Things such as baseball statistics, college tuition costs and entrance averages, or the canonical example of automobile data<sup>2</sup> could be used.

Tasks would also be highly dependent on the datasets used. The goal of these tasks would be to have the subjects to perform some higher-level interaction with the plot by forcing them to perform an analysis of it. This would require data that, firstly, could be analyzed and could provide a useful result, and secondly, would not confound the study by making things too difficult or too reliant on the domains the data were collected from.

<sup>2</sup>These data sets were collected for the Data Analysis Exposition sponsored by the Statistical Graphics Section of the American Statistical Association in 1988, 1995, and 1983 respectively. They are available at the CMU StatLib Datasets Archive: <http://lib.stat.cmu.edu/datasets/>

The analysis tasks would likely be goal-based exploration of the datasets. For example, subjects could be asked to find the best baseball batters in a certain category such as team or league, or the a college with low tuition but high ratings of quality, or the relationship between the efficiency of a cars' engines, their top speed, and the country in which they were manufactured. All of these goals would have to be achievable through the re-ordering of axes, and so would have to require multiple dimensions of data to be considered at a time. The difficulty and complexity of the analysis necessary to compete the tasks would have to be roughly equivalent across all data and tasks. Were it not, any difference in the results collected could be due to differences in the data or task, not in the method used.

Subjects would be given a half-hour or so to complete each task, with a short unrelated task or break between analyses (It is likely that subjects would become fatigued if asked to perform more two analysis tasks back-to-back). Once the tasks are complete, they would fill out a questionnaire or take part in an interview.

The results of such a study would not allow any truly concrete judgements of either method to be made, but would instead serve to support conclusions about their effectiveness and their perception by users. The study would not prove that one method is better, but could give a sense of whether users perceived it to be, and whether they preferred one method to another. Observations and perhaps anecdotes collected during interviews could be used to determine whether users fully understood the methods, or if there were any found any novel uses of or insights into the methods while performing the tasks.

## 6 FUTURE WORK

Although the fan menu method has the potential to alleviate many of the difficulties related to finding a good axis ordering, it does have weaknesses and can be improved. We propose a few potential improvements below, avenues along which future work can proceed.

### 6.1 Combined fan menu and shuffle methods

As discussed in Section 3.3, the method can be time-consuming, particularly when users would like to add axes in the middle of the plot, rather than at the ends. One potential solution to this is to combine the fan menu and shuffle methods, so that users can add an axis using the fan menu, then drag it to the location in the order that they would like it to be. This would re-introduce some of the difficulties that come with the shuffle method (Section 2.1.1), but would reduce the number of actions necessary when adding axes randomly.

### 6.2 Filtering and sorting thumbnails

It is possible that the order of the thumbnails in the fan menu can effect the likelihood that they will be selected. There are many possible ways of sorting the thumbnails vertically in the fan menu, each with differing potential effects. Sorting thumbnails alphabetically by dimension label would allow users to quickly determine where a specific axes is in the menu display. This would not guarantee that the axes that result in the most dramatic descriptions of the data are easily found; sorting by correlation coefficient or by similarity could be more useful when exploring an unfamiliar data set. It may be best to provide control of the sorting method to the user through a combo box widget listing the different sorting criteria, or providing additional information alongside the thumbnails (labelling them with correlation coefficients).

### 6.3 Reducing clutter in thumbnails

The fan menu method is susceptible to the two other problems that plague most other parallel coordinate plots: clutter and high-dimensionality. In Figure 6, it is possible to see how thumbnails be-

come cluttered when the number of points in the data set increases: the many lines obscure the relationship between the dimensions.

Interactive distortion techniques can help reduce clutter in larger plots [3], but they may not be very useful in thumbnails. Firstly, thumbnails are usually quite small and distortion techniques may not be discernible in small plots. Secondly, interactive distortion would require users to manipulate the thumbnails individually. This would take time and would make it difficult to compare thumbnails, since each would be distorted differently or at different times. Even if the distortion were applied to all thumbnails at once, they could misrepresent the relationships and undermine the usefulness of thumbnails as predictors of the relationships added to the plot. Whether distortion techniques can be applied successfully to thumbnails remains to be determined.

It may be possible to filter the lines drawn in thumbnails such that lines with similar values along a dimension, or slopes between them, are represented in thumbnails with a single line. These lines could be rendered in a thicker stroke or brighter colour if they represent a greater number of lines. This is similar to clustering and may not be efficient enough to be calculated each time the fan menu is invoked.

#### 6.4 Improving selection in high-dimensional data

In a high-dimensional dataset, the number of thumbnails to be displayed in the fan menu can become so great that the only way to fit them all is to render them at a very small size. For example, in Figure 7, fifteen thumbnails are available in the fan menu. When the prototype is running with a window height of 600 pixels, each of these thumbnails is no bigger than 32 pixels. It is difficult to discern anything other than the gross differences between thumbnails when they are so small.

It may be possible to implement a fish-eye distortion system based on certain of the relationships represented by the thumbnails: those representing relationships with higher correlation coefficients would be rendered larger. This could be combined with the sorting criteria described above, such that users can choose why the sizes of thumbnails will be distorted.

#### 6.5 Hierarchical fan menus

Fan menu selection could also be improved through the use of hierarchical or nested menus. Users would not see all the thumbnails along the fan menu's edge after hitting the "add axis button", but would instead be given a smaller number of categories to choose from. After selecting one of these, the user would be shown the thumbnails belonging to that category, and would complete the "add" action by selecting one of these.

There are a number of potential hierarchies which could meaningfully separate the thumbnails. Grouping the thumbnails by the gross kind relationship or trend they represent, such as slightly increasing or decreasing average slopes, or the variance of the points along one dimension, or, again, by correlation coefficient, could work. It may also be possible to group thumbnails into clusters (similar to [5]) and have each second-level menu contain all the relationships within a certain cluster. This would require much more preprocessing, but would be a hierarchy tuned to the data being plotted. It would also be possible to organize the thumbnails using meta-data or domain knowledge which could be applied arbitrarily.

Hierarchical fan menus would also alleviate the thumbnail size problem in high-dimensional datasets. The criteria by which thumbnails are sorted into hierarchies could be tuned to create second-level menus with fewer than ten or twelve thumbnails.

## 7 CONCLUSION

Parallel coordinate plots are useful for visually representing data. Unfortunately, they can become difficult to read, especially when displaying a large number of data point or data in many dimensions.

Since parallel coordinates are sequences of pairwise relationships between dimensions, the order of the axes is also critical to the effectiveness of these plots. There have been a variety of proposed solutions to the number and order of axes in parallel coordinates. However, many of these solutions are computationally expensive and may not provide the best ordering in all cases. Users often find greater satisfaction with orders that they are able to manipulate.

The current method for users to manipulate the order of axes, shuffling, is deceptively simple. There are a number of drawbacks to the shuffle method. We have proposed a new method called the fan menu method in which the user builds a plot axis by axis. This method uses a combination of thumbnails of plots to allow users to see the consequences of their actions, to easily explore the dataset, to create relevant plots, and to easily alter the order of axes.

We discussed various approaches to evaluating such axis ordering methods, describing studies to determine both the immediate differences in performance and accuracy between them, as well as higher-level discovering any higher-level benefits they may have. Lastly, we discussed directions future work on the fan menu method may proceed.

## REFERENCES

- [1] B. Berg. Interfascia. version 003, Alpha <http://www.superstable.net/interfascia/>.
- [2] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman. An empirical comparison of pie vs. linear menus. In *CHI '88: Proceedings of the SIGCHI conference on human factors in computing systems*, pages 95–100, New York, NY, USA, 1988. ACM Press.
- [3] E. Fanea, S. Carpendale, and T. Isenberg. An interactive 3D integration of parallel coordinates and star glyphs. In *INFOVIS '05: Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, page 20, Washington, DC, USA, 2005. IEEE Computer Society.
- [4] B. Fry and C. Reas. Processing. version 123, Beta <http://www.processing.org/>.
- [5] Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. In *VIS '99: Proceedings of the conference on Visualization '99*, pages 43–50, Los Alamitos, CA, USA, 1999. IEEE Computer Society Press.
- [6] A. Inselberg and T. Avidan. The automated multidimensional detective. In *INFOVIS '99: Proceedings of the 1999 IEEE Symposium on Information Visualization*, page 112, Washington, DC, USA, 1999. IEEE Computer Society.
- [7] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *VIS '90: Proceedings of the 1st conference on Visualization*, pages 361–378, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.
- [8] W. Peng, M. O. Ward, and E. A. Rundensteiner. Clutter reduction in multi-dimensional data visualization using dimension reordering. In *INFOVIS '04: Proceedings of the 2004 IEEE Symposium on Information Visualization*, pages 89–96, Washington, DC, USA, 2004. IEEE Computer Society.
- [9] E. J. Wegman. Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85(411):664–675, September 1990.
- [10] J. Yang, W. Peng, M. O. Ward, and E. A. Rundensteiner. Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets. *INFOVIS '03: Proceedings of the 2003 IEEE Symposium on Information Visualization*, page 14, 2003.
- [11] J. Yang, M. O. Ward, E. A. Rundensteiner, and S. Huang. Visual hierarchical dimension reduction for exploration of high dimensional datasets. In *VISSYM '03: Proceedings of the symposium on Data visualization 2003*, pages 19–28, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

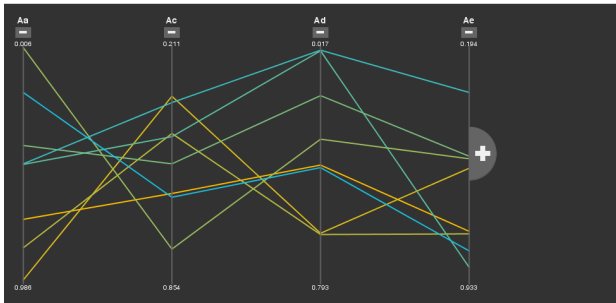
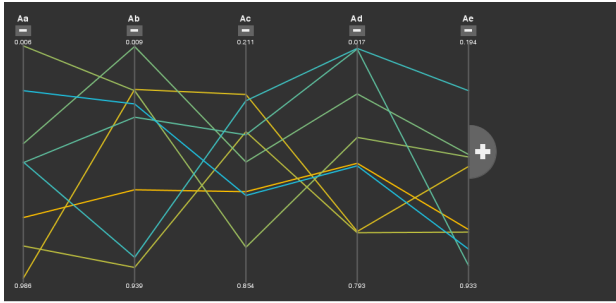


Figure 5: A demonstration of removable axes: the initial plot at the top; with axis “Ab” removed, bottom.

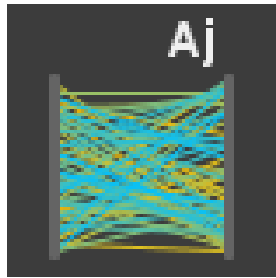
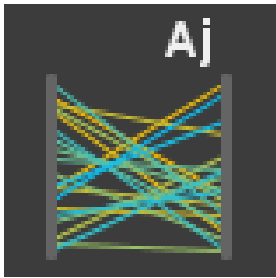


Figure 6: Thumbnails for the same dimensions and data, the one on the right with twice as many data points as the one on the left.

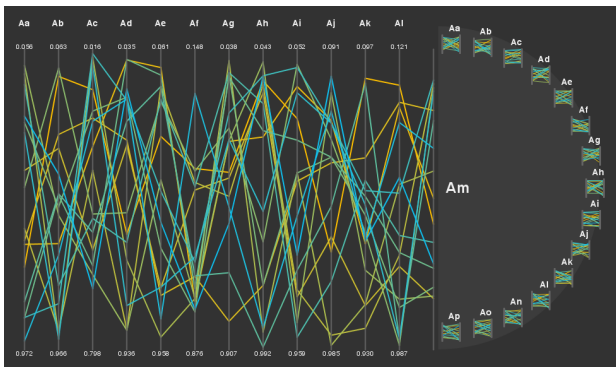


Figure 7: A plot and menu for a dataset with high dimensionality.